IPv6 Routing (deep dive)

2011 Rocky Mountain IPv6 Summit
26 April, 2011
Denver, CO

Chris Grundemann, CableLabs
Aaron Hughes, 6connect
Ken Sexton, tw telecom
Topics

- Routing Protocols for IPv6
  - OSPFv3
  - EIGRP for IPv6
  - BGP4+
  - Integrated IS-IS for IPv6

- Configuration & CLI examples from:
  - Cisco: IOS
  - Juniper: JunOS

  (NX-OS and IOS-XR not covered)
Goals and Assumptions

**Goals**
- Describe the operational differences between each of these protocols
- Review configuration differences and things to look out for when implementing for IPv6 vs. IPv4
- Provide a starting point for your own trials

**Assumptions**
- Have a basic understanding of IPv6
- Have an existing understanding of how to use these protocols with IPv4
Basic IPv6 Architecture (Typical Network)

- **Allocations & Assignments**
  - /36 per region
    - Notable that more than 15 regions requires larger than a /32
  - Top or bottom /48 reserved for infrastructure in each region
  - Top /64 of reserved /48 reserved for that region’s loopbacks
    - If the region becomes an island we still want it to aggregate and meet min /48

- **Protocols**
  - If OSPFv3
    - Loopbacks /128s and connected /126s and in OSPF
  - If IS-IS
    - Multi vs. Single topology pros and cons (runs next to IP)
  - iBGP
    - Inject all non-loopback connected into iBGP with admin distance set by OSPFv3
  - eBGP
    - Redistributed by community tagged aggregates
Experience is the best teacher
OSPF Agenda

- Review OSPFv2/OSPFv3 Similarities
- Review what’s new in OSPFv3
- Review OSPFv3 Configuration & Verification Commands

Does OSPF Version 3 replace Version 2?

<table>
<thead>
<tr>
<th>OSPFv2</th>
<th>OSPFv3</th>
<th>OSPFv3</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv4</td>
<td>IPv6</td>
<td>IPv4 &amp; IPv6</td>
</tr>
<tr>
<td>RFC 2328</td>
<td>RFC 5340</td>
<td>RFC 5838</td>
</tr>
</tbody>
</table>
**OSPFv3 Fundamentals**

- **Modified version of OSPF to include support for IPv6**
- **Based on OSPFv2, with enhancements**
  - Same basic packet types - Hello, DBD, LSA, LSU and LSR
  - Same neighbor discovery & adjacency formation

<table>
<thead>
<tr>
<th></th>
<th>OSPFv2</th>
<th>OSPFv3</th>
</tr>
</thead>
<tbody>
<tr>
<td>All OSPF Routers</td>
<td>224.0.0.5</td>
<td>FF02::5</td>
</tr>
<tr>
<td>All DR Routers</td>
<td>224.0.0.6</td>
<td>FF02::6</td>
</tr>
</tbody>
</table>

- Same interface types - Broadcast, NBMA, P2P, P2MP
- Same timers (hello, dead, LSA refresh/aging)
- Same metric (interface cost)
- Same protocol number: 89 (0x59)

IPv6 Header

Next Header = 89

OSPF Packet Header

OSPF Payload
OSPFv3 Fundamentals

- RFC 5340 - OSPF for IPv6 Unicast Routing
  - Describes the differences from OSPF for IPv4
  - Changes to the original OSPFv3 RFC 2740

- OSPFv3 runs independently from OSPFv2
  - Separate Routing Process
  - Same link – separate control protocol
  - Separate CLI commands and output
  - Separate routing table
What’s New in OSPFv3?

- Use of Link-Local FE80 Address -

- Source address for all OSPF control packets
  - Hellos, OSPF DBD, LS Ack, LS Update, LS Request
- Next Hop address for packet forwarding

```
R1# show ipv6 ospf interface
Vlan 17 is up, line protocol is up
  Link Local Address FE80::21F:CAFF:FE6B:E095, Interface ID 14
  Area 0, Process ID 1, Instance ID 0, Router ID 10.10.10.1
  Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID)10.10.10.1, local address FE80::21F:CAFF:FE6B:E095
  Backup Designated router (ID)10.10.10.3,local address FE80::21F:FF:F15B:4FDC
```
What’s New in OSPFv3?

- Protocol Processing Per Link -

- OSPFv3 is running per Link instead of Per IP Subnet
  - Removed terms “network” & “subnet”- replaced with “link”
  - An interface now connects to a link

- Regardless of assigned IPv6 unicast prefixes, two devices communicate using Link Local address; nodes can become adjacent even if they don’t share a common subnet

- Multiple IPv6 prefixes can be assigned to the same link
What’s New in OSPFv3?

- Removal of Addressing Semantics -

- IP prefix information is no longer present in OSPF packet Headers – carried as payload information

- OSPF RID, AID and LSID remain as 32-bit fields
  - Can not use an IPv6 128-bit address
  - IPv4 dotted decimal format

- Neighbors and DR/BDR are always identified by Router ID

10.40.64.27
OSPFv3

PACKET / LSA FORMAT CHANGES
### OSPFv3 Packet Header Format

#### OSPFv2

<table>
<thead>
<tr>
<th>Version</th>
<th>Type</th>
<th>Packet Length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Router ID
- Area ID
- Checksum
- Authentication
- Authentication

#### OSPFv3

<table>
<thead>
<tr>
<th>Version</th>
<th>Type</th>
<th>Packet Length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Router ID
- Area ID
- Checksum
- Instance ID

- Default: 0, range 0-255

- **Instance ID** – new field
- **Authentication fields** – moved to IPv6 Extension Header

- Size of Header is reduced from 24B to 16B
- All OSPF Packet types begin with a standard 16BHdr
- Version: 3
- RID and AID are still 32-bit numbers
### OSPFv3 Hello Packet Format

<table>
<thead>
<tr>
<th>OSPFv2</th>
<th>OSPFv3</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong> = 1</td>
<td><strong>Type</strong> = 1</td>
</tr>
<tr>
<td>OSPF Packet Header (24B)</td>
<td>OSPF Packet Header (16B)</td>
</tr>
<tr>
<td>Network Mask</td>
<td>Interface ID</td>
</tr>
<tr>
<td>Hello Interval</td>
<td>Options</td>
</tr>
<tr>
<td>Route Dead Interval</td>
<td></td>
</tr>
<tr>
<td>Designated Router</td>
<td></td>
</tr>
<tr>
<td>Backup Designated Router</td>
<td></td>
</tr>
<tr>
<td>Neighbor</td>
<td>Neighbor</td>
</tr>
</tbody>
</table>

- **Network Mask** - removed – not needed for IPv6
- **Interface ID** - new field
- **Options field** – 8 bits to 24 bits (added v6 and R bit)
- **Router Dead Interval** – 32 bits to 16 bits
### OSPFv3 DD Packet Format

#### OSPFv2

- **Type = 2**
- OSPF Packet Header (24B)
- Interface MTU
- Options
- I | M | MS
- DD Sequence Number
- LSA Header(s)

#### OSPFv3

- **Type = 2**
- OSPF Packet Header (24B)
- Interface MTU
- Options
- I | M | MS
- DD Sequence Number
- LSA Header(s)

- **Options field** – 8 bits to 24 bits (added v6 and R bit)
- All other fields the same
## OSPFv3 LSA Packet Format

- Link State Request,
- Link State Update
- LS Acknowledgement

- Fields the same
- No changes

### OSPFv2/v3 - LSR

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>OSPF Packet Header (24B)</td>
</tr>
<tr>
<td></td>
<td>Link State Type</td>
</tr>
<tr>
<td></td>
<td>Link State ID</td>
</tr>
<tr>
<td></td>
<td>Advertising Router</td>
</tr>
</tbody>
</table>

### OSPFv2/v3 - LSU

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>OSPF Packet Header (24B)</td>
</tr>
<tr>
<td></td>
<td># of Advertisements</td>
</tr>
<tr>
<td></td>
<td>Link State Advertisements</td>
</tr>
<tr>
<td></td>
<td>(LSA Hdr + LSA information)</td>
</tr>
</tbody>
</table>

### OSPFv2/v3 - LSA

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>OSPF Packet Header (24B)</td>
</tr>
<tr>
<td></td>
<td>Link State Advertisement</td>
</tr>
<tr>
<td></td>
<td>Header</td>
</tr>
</tbody>
</table>
**OSPFv3 LSA Header Format**

### OSPFv2

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>4</td>
</tr>
<tr>
<td>OSPF Packet Header</td>
<td>24B</td>
</tr>
<tr>
<td>LSU (# of Advertisements)</td>
<td></td>
</tr>
<tr>
<td>LSA Age</td>
<td></td>
</tr>
<tr>
<td>Options</td>
<td></td>
</tr>
<tr>
<td>LS Type</td>
<td></td>
</tr>
<tr>
<td>Link State ID</td>
<td></td>
</tr>
<tr>
<td>Advertising Router</td>
<td></td>
</tr>
<tr>
<td>LS Sequence Number</td>
<td></td>
</tr>
<tr>
<td>LS Checksum</td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td></td>
</tr>
</tbody>
</table>

### OSPFv3

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>4</td>
</tr>
<tr>
<td>OSPF Packet Header</td>
<td>24B</td>
</tr>
<tr>
<td>LSU (# of Advertisements)</td>
<td></td>
</tr>
<tr>
<td>LSA Age</td>
<td></td>
</tr>
<tr>
<td>LS Type</td>
<td></td>
</tr>
<tr>
<td>Link State ID</td>
<td></td>
</tr>
<tr>
<td>Advertising Router</td>
<td></td>
</tr>
<tr>
<td>LS Sequence Number</td>
<td></td>
</tr>
<tr>
<td>LS Checksum</td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td></td>
</tr>
</tbody>
</table>

- **Options field** – moved to Packet Header
- **LS type field** – 8 bits to 16 bits (See next slide)
- All other fields the same
# OSPFv3 LSA Type Review

## LSA Header

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSA Age</td>
<td></td>
</tr>
<tr>
<td>LS Type</td>
<td></td>
</tr>
<tr>
<td>Link State ID</td>
<td></td>
</tr>
<tr>
<td>Advertising Router</td>
<td></td>
</tr>
<tr>
<td>LS Sequence Number</td>
<td></td>
</tr>
<tr>
<td>LS Checksum</td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td></td>
</tr>
</tbody>
</table>

## LSA Type Field (2B)

<table>
<thead>
<tr>
<th>U</th>
<th>S2</th>
<th>S1</th>
<th>LSA Function Code</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

## Flooding Scope

<table>
<thead>
<tr>
<th>S2</th>
<th>S1</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Link-Local</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Area</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>AS (Routing Domain)</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

## OSPFv3 LSAs

<table>
<thead>
<tr>
<th>LSA FC</th>
<th>LSA Name</th>
<th>LS Type</th>
<th>Flooding Scope</th>
<th>OSPFv2 LSAs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Router LSA</td>
<td>0x2001</td>
<td>Area Scope</td>
<td>Router LSA</td>
</tr>
<tr>
<td>2</td>
<td>Network LSA</td>
<td>0x2002</td>
<td>Area Scope</td>
<td>Network LSA</td>
</tr>
<tr>
<td>3</td>
<td>Inter-Area Prefix LSA</td>
<td>0x2003</td>
<td>Area Scope</td>
<td>Network Summary LSA</td>
</tr>
<tr>
<td>4</td>
<td>Inter-Area Router LSA</td>
<td>0x2004</td>
<td>Area Scope</td>
<td>ASBR Summary LSA</td>
</tr>
<tr>
<td>5</td>
<td>AS-External LSA</td>
<td>0x4005</td>
<td>AS Scope</td>
<td>AS-External LSA</td>
</tr>
<tr>
<td>6</td>
<td>Group-Membership LSA</td>
<td>0x2006</td>
<td>Area Scope</td>
<td>Group-Membership LSA</td>
</tr>
<tr>
<td>7</td>
<td>Type-7 LSA</td>
<td>0x2007</td>
<td>Area Scope</td>
<td>NSSA External LSA</td>
</tr>
<tr>
<td>8</td>
<td>Link LSA</td>
<td>0x0008</td>
<td>Link-Local Scope</td>
<td>N/A</td>
</tr>
<tr>
<td>9</td>
<td>Intra-Area Prefix LSA</td>
<td>0x2009</td>
<td>Area Scope</td>
<td>N/A</td>
</tr>
</tbody>
</table>
The OSPF LSA Header contains **LS Type** Field used to identify type of LSA being advertised.

In some cases, OSPFv3 LSA formats are different than OSPFv2.

OSPFv3 introduces two new LSA types.

### OSPFv3 LSAs vs OSPFv2 LSAs

<table>
<thead>
<tr>
<th>OSPFv3 LSAs</th>
<th>OSPFv2 LSAs</th>
</tr>
</thead>
<tbody>
<tr>
<td>LS Type</td>
<td>LSA Name</td>
</tr>
<tr>
<td>0x2001</td>
<td>Router LSA</td>
</tr>
<tr>
<td>0x2002</td>
<td>Network LSA</td>
</tr>
<tr>
<td>0x2003</td>
<td>Inter-Area Prefix LSA</td>
</tr>
<tr>
<td>0x2004</td>
<td>Inter-Area Router LSA</td>
</tr>
<tr>
<td>0x4005</td>
<td>AS-External LSA</td>
</tr>
<tr>
<td>0x2006</td>
<td>Group-Membership LSA</td>
</tr>
<tr>
<td>0x2007</td>
<td>Type-7 LSA</td>
</tr>
<tr>
<td>0x0008</td>
<td>Link LSA</td>
</tr>
<tr>
<td>0x2009</td>
<td>Intra-Area Prefix LSA</td>
</tr>
</tbody>
</table>
Router LSA

- **OSPFv2 LSA Type 1**
  - Router LSAs send link topology information & IPv4 Link network and mask information

  Link connected to: a Stub Network
  - (Link ID) Network/subnet number: 10.1.199.0
  - (Link Data) Network Mask: 255.255.255.128
  - Number of TOS metrics: 0
  - TOS 0 Metrics: 1

- **OSPFv3 LSA Type 0x2001**
  - Router LSAs only send topology information

  Link connected to: a Transit Network
  - Link Metric: 1
  - Local Interface ID: 14
  - Neighbor (DR) Interface ID: 13
  - Neighbor (DR) Router ID: 10.10.10.3
Network LSA (originated by DR)

- **OSPFv2 LSA Type 2**
  - Network Mask & Attached Routers on Link (RID)
  
  Options: (No TOS-capability, DC)
  
  Link State ID: 10.20.30.1 (address of Designated Router)
  
  Advertising Router: 10.10.10.1

  Network Mask: /24
  
  Attached Router: 10.10.10.1
  
  Attached Router: 10.10.10.2

- **OSPFv3 LSA Type 0x2002**
  
  Attached Routers on Link (RID)

  Options: (V6-Bit, E-Bit, R-bit, DC-Bit)
  
  Link State ID: 13 (Interface ID of Designated Router)
  
  Advertising Router: 10.10.10.1

  Attached Router: 10.10.10.2
  
  Attached Router: 10.10.10.2
OSPFv3 Link LSA (0x0008) - New

- Carries IPv6 link-local address used for NH calculation
- Advertise IPv6 unicast addresses to other routers on the link
- A link LSA per link
- Link local scope flooding on the link with which they are associated
OSPFv3 Intra-Area Prefix LSA (0x2009) - New

- Intra-Area-Prefix-LSA will advertise prefix information:
  - The node’s local interfaces (loopback)
  - Any IPv6 prefix address/length information within Area
OSPFv3

CONFIGURATION EXAMPLES (JUNOS & IOS)

* Does not cover Cisco IOS-XR, NX-OS
OSPFv3 Example - Juniper

Loopbacks & Router IDs
R1 = 2001:db8::1/128 & 10.10.10.1
R2 = 2001:db8::2/128 & 10.10.10.2
R3 = 2001:db8::3/128 & 10.10.10.3
R4 = 2001:db8::4/128 & 10.10.10.4
R5 = 2001:db8::5/128 & 10.10.10.5

AS 64109
2001:db8:8000::/48
JunOS - Command Comparison

- **OSPFv3**

```bash
set interfaces lo0 unit 0 family inet6 address 2001:DB8::1/128
set interfaces ge-1/0/2 unit 0 family inet6 address 2001:DB8:0:4::1/126

set routing-options router-id 10.10.10.1

set protocols ospf3 area 0.0.0.0 interface lo0.0 passive
set protocols ospf3 area 0.0.0.0 interface ge-0/0/0.0
```

- The above command lines provide the minimum configuration to enable OSPF
- Assumes the link-local addresses are built dynamically
JunOS - Command Comparison

- **OSPFv2**
  
  ```
  set interfaces lo0 unit 0 family inet address 10.10.10.1/32
  set interfaces ge-1/0/2 unit 0 family inet address 10.10.4.1/30
  set routing-options router-id 10.10.10.1
  set protocols ospf area 0.0.0.0 interface lo0.0 passive
  set protocols ospf area 0.0.0.0 interface ge-0/0/0.0
  ```

- **OSPFv3**
  
  ```
  set interfaces lo0 unit 0 family inet6 address 2001:DB8::1/128
  set interfaces ge-1/0/2 unit 0 family inet6 address 2001:DB8:0:4::1/126
  set routing-options router-id 10.10.10.1
  set protocols ospf3 area 0.0.0.0 interface lo0.0 passive
  set protocols ospf3 area 0.0.0.0 interface ge-0/0/0.0
  ```

- The above command lines provide the minimum configuration to enable OSPF
- Assumes the link-local addresses are built using M-EUI-64
OSPFv3 Example - Cisco

Loopbacks & Router IDs
- R1 = 2001:db8::1/128 & 10.10.10.1
- R2 = 2001:db8::2/128 & 10.10.10.2
- R3 = 2001:db8::3/128 & 10.10.10.3
- R4 = 2001:db8::4/128 & 10.10.10.4
- R5 = 2001:db8::5/128 & 10.10.10.5

Cisco 871
Version 12.4(24)T4
The above command lines provide the minimum configuration to enable OSPF

Assumes the link-local addresses are built dynamically
**OSPFv3 - Cisco Best Practices**

**Do you have the Wrong RID?**

- **Router ID** = Identifies Router in OSPFv3
  - 32-bit number like OSPFv2
  - IPv6 Native – RID must be configured explicitly
  - Dual Stack - OSPFv3 searches for an IPv4 address to get the “Router ID” for the IPv6 routing process

```plaintext
interface Vlan8
  ipv6 ospf 1 area 0
!
ipv6 router ospf 1
  router-id 10.10.10.1
```

!!!! Best Common Practice !!!!

If trying to control RID #, configure IPv6 OSPF Process first, with RID
Then configure interface statements
OSPFv3 - Cisco Best Practices
Other things to watch out for….

- Last “ipv6 ospf area” statement removed from an interface will remove the OSPFv3 router process

```bash
interface Vlan8
  ipv6 address 2001:DB8:0:8::1/126
  no ipv6 ospf 1 area 0  # Last “ipv6 ospf area” removed

ipv6 router ospf 1   # All statements automatically removed
  router-id 10.10.10.1
  redistribute bgp 65342 metric 10 remote-map ROUTES_IN
  log-adjacency-changes
```

- Remove process – all interface OSPF statements are removed
Other things to watch out for...

- Wrong “redistribution” statement can start an unwanted router process (EIGRP, OSPF, RIP, etc)

```
ipv6 router ospf 1
  router-id 10.10.10.1
  redistribute eigrp 10 metric 40  ← Entered ASN 10 instead of 1
!
ipv6 router eigrp 1
  no shutdown
  router-id 10.10.10.1
  passive-interface loopback 0
!
ipv6 router eigrp 10  ← Another Process automatically started
  shutdown
```
Other things to watch out for....

- Adding an interface with the wrong OSPF Process ID will start an unwanted router process
- Look for wrong RID on neighbor

```
interface Vlan8
  ipv6 ospf 10 area 0 ➡️ Entered PID 10 instead of 1
!
ipv6 router ospf 1
  router-id 10.10.10.1
  log-adjacency-changes
!
ipv6 router ospf 10 ➡️ Another Process automatically started
  log-adjacency-changes
```
Cisco - Command Comparison

Optional Commands

- **OSPFv2**
  
  **Interface Commands:**
  
  interface Vlan8
  - `ip ospf cost 30`
  - `ip ospf hello-interval 15`
  - `ip ospf network point-to-point`

  **OSPF Router Commands:**
  
  router ospf 1
  - default-information originate
  - default-metric 10
  - redistribute bgp 64109 metric 20 metric-type 1
  - area 0 range 10.10.0.0 255.255.0.0
  - summary-address 192.168.24.0 255.255.255.0
  - passive-interface loopback 0

- **OSPFv3**
  
  **Interface Commands:**
  
  interface Vlan8
  - `ipv6 ospf cost 30`
  - `ipv6 ospf hello-interval 15`
  - `ipv6 ospf network point-to-point`

  **OSPF Router Commands:**
  
  `ipv6 router ospf 1`
  - default-information originate
  - default-metric 10
  - redistribute bgp 64109 metric 20 metric-type 1
  - area 0 range 2001:DB8:2:D50::/60
  - summary-prefix 2001:DB8:0:F20::/60
  - passive-interface loopback 0

- Very similar – in some cases, it’s “ipv6” versus “ip”
JunOS Configuration for R1

```
routing-options {
  router-id 10.10.10.1;
}

protocols {
  ospf3 {
    area 0.0.0.0 {
      interface lo0.0 {
        passive;
      }
      interface ge-1/0/2.0;
      interface ge-1/0/3.0;
      interface ge-1/0/4.0;
    }
  }
}
```

Loopbacks & Router IDs
- R1 = 2001:db8::1/128 & 10.10.10.1
- R2 = 2001:db8::2/128 & 10.10.10.2
- R3 = 2001:db8::3/128 & 10.10.10.3
- R4 = 2001:db8::4/128 & 10.10.10.4
- R5 = 2001:db8::5/128 & 10.10.10.5
ipv6 unicast-routing
!
interface Loopback0
ipv6 address 2001:DB8::1/128
ipv6 ospf 100 area 0
!
interface Vlan2 – Vlan3 – Vlan9
ipv6 address 2001:DB8:0:x::1/126
ipv6 ospf 100 area 0
!
ipv6 router ospf 100
log-adjacency-changes
router-id 10.10.10.1
passive-interface loopback 0
JunOS Configuration for R4

```
route-options {
    router-id 10.10.10.4;
}
protocols {
    ospf3 {
        area 0.0.0.0 {
            interface lo0.0 {
                passive;
            }
            interface ge-1/0/1.0;
            interface ge-1/0/2.0;
            interface ge-1/0/3.34;
        }
        area 0.0.0.1 {
            interface ge-1/0/3.45;
        }
    }
}
```

Loopbacks & Router IDs
R1 = 2001:db8::1/128 & 10.10.10.1
R2 = 2001:db8::2/128 & 10.10.10.2
R3 = 2001:db8::3/128 & 10.10.10.3
R4 = 2001:db8::4/128 & 10.10.10.4
R5 = 2001:db8::5/128 & 10.10.10.5

AS 64109
2001:db8:8000::/48
168.215.48.2/30
interface Loopback0
  ipv6 address 2001:DB8::4/128
  ipv6 ospf 100 area 0

interface Vlan1 – Vlan2 – Vlan5
  ipv6 address 2001:DB8:0:x::2/126
  ipv6 ospf 100 area 0

interface Vlan8
  description "To R5"
  ipv6 address 2001:DB8:0:8::2/126
  ipv6 ospf 100 area 1

ipv6 router ospf 100
  router-id 10.10.10.4
  log-adjacency-changes
  passive-interface Loopback 0
  }

Loopbacks & Router IDs
R1 = 2001:db8::1/128 & 10.10.10.1
R2 = 2001:db8::2/128 & 10.10.10.2
R3 = 2001:db8::3/128 & 10.10.10.3
R4 = 2001:db8::4/128 & 10.10.10.4
R5 = 2001:db8::5/128 & 10.10.10.5
Other OSPF Commands

- **OSPFv3 Authentication & Encryption**
  - OSPFv3 Packet Header **does not** support Authentication
  - OSPFv3 uses the Layer 3 IPv6 “IPSec Extension Headers”
  - IPSec Protocols supported:
    - Authentication Header (**AH**) – supports authentication only
    - Encapsulation Security Payload (**ESP**) - supports both authentication and encryption

---

IPv6 Header
Next Header = 51

AH Header
Next Header = 89

OSPF Packet Header

OSPF Payload
OSPFv3 Authentication & Encryption
Juniper - Configuration Examples

- Define IPSec Security Association (Protocols AH, ESP supported)
- **Apply Per – interface**

**Define IPSec SA (example of AH, SHA-1 Hash)**
- `edit security ipsec security-association OSPF_IPSEC direction bidirectional`
  - `set protocol ah`
  - `set spi 1000`
  - `set authentication algorithm [hmac-md5-96 | hmac-sha1-96]`
  - `set authentication key ascii-text <key>`

**Apply SA to OSPF per-interface**
- `set protocols ospf3 area 0 interface <interface> ipsec-sa OSPF_IPSEC`
OSPFv3 Authentication & Encryption
Cisco - Configuration Examples

- Crypto image (K9) is required – uses IPSec Secure Socket API

**Option 1 – Per-Interface**

**Interface**
- `ipv6 ospf authentication ipsec spi 34567 md5 0 <Hex key>` - auth only
- `ipv6 ospf encryption ipsec spi 2641 esp null 0 <Hex key>` - auth only
- `ipv6 ospf encryption ipsec spi 789 esp 3des 0 <Hex key>` - auth/encryption

**Option 2 – Per-Area**

- Policy is applied to all of the interfaces in that area, except for the interfaces that have IPsec configured directly

**Area – ipv6 router ospf <pid>**
- `area 0 authentication ipsec spi 34567 md5 0 <Hex key>` - auth only
- `area 0 encryption ipsec spi 2641 esp null 0 <Hex key>` - auth only
- `area 0 encryption ipsec spi 789 esp 3des 0 <Hex key>` - auth/encryption
Other OSPF Commands

- **OSPFv3 ASBR Redistribution - Juniper**
  - You can redistribute IPv6 routes learned from other routing protocols into an OSPFv3 autonomous system
  - Routing Protocols include:
    - RIPng, OSPFv3, BGP, EIGRP, Static, IS-IS (v6 <-> v6)

Example: OSPF3 routes into BGP

```bash
set policy-options policy-statement ROUTES term 1 from protocol ospf
set policy-options policy-statement ROUTES term 1 from route-filter 2001:ACDC:1::/48
set policy-options policy-statement ROUTES term 1 from route-filter 2001:8D1:800D:0::/48 prefix-length-range /48-/64
set policy-options policy-statement ROUTES term 1 then accept
set policy-options policy-statement ROUTES term 2 then reject

set protocols bgp group CUSTA neighbor 2001:DB1:682:0::2 family inet6 unicast
set protocols bgp group CUSTA neighbor 2001:DB1:682:0::2 export ROUTES
set protocols bgp group CUSTA neighbor 2001:DB1:682:0::2 peer-as 65000
```
Other OSPF Commands

- **OSPFv3 ASBR Redistribution - Cisco**
  - You can redistribute IPv6 routes learned from other routing protocols into an OSPFv3 autonomous system
  - Routing Protocols include:
    - RIPng, OSPFv3, BGP, EIGRP, Static, IS-IS (v6 ↔ v6)

Example: BGP 64109 learn routes into OSPFv3 PID 100

```conf
ipv6 prefix-list EBGP-IPv6-FROM-P1 seq 5 permit 2001:506:8::/48
!
route-map BGP_INTO_OSPF deny 10
  match ipv6 address prefix-list EBGP-IPv6-FROM-P1  ➔ Denies BGP 2001:506:8::/48
route-map BGP_INTO_OSPF permit 20  ➔ Permits everything else into OSPF
!
ipv6 router ospf 100
  redistribute bgp 64109 metric 50 route-map BGP_INTO_OSPF
```
ipv6 router ospf <pid>
default-metric <cost>
default-information originate [always | metric <cost> metric-type [1|2] route-map <WORD>]
redistribute rip <name> [metric <cost> metric-type [1|2] route-map <WORD> include-connected]
redistribute eigrp <asn> [metric <cost> metric-type [1|2] route-map <WORD> include-connected]
redistribute connected [metric <cost> metric-type [1|2] route-map <WORD>]
redistribute static [metric <cost> metric-type [1|2] route-map <WORD>]
redistribute bgp <asn> [metric <cost> metric-type [1|2] route-map <WORD>]
redistribute ospf <pid> match [internal | external | nssa-external]
[metric <cost> metric-type [1|2] route-map <WORD> include-connected]

• “include-connected” versus “redistribute connected”
• “subnets” keyword NOT used
Best Common Practice !!!

For now – Use separate Routing Process for each family

“Ships-in-the-night” approach

OSPFv3  - RFC 5838

- A single OSPFv3 routing process will now support multiple Address Families:
  - Unicast IPv4 and IPv6  &  Multicast IPv4 and IPv6
  - Each family has its own neighbor adjacencies, link state database, protocol data structures, and shortest path first (SPF) computation

- Too new – limited vendor support ............

!!!! Best Common Practice !!!!
Multiple Address Families (JunOS)

protocols {
opf3 {
  realm ipv4-unicast {
    area 0.0.0.0 {
      interface ge-1/0/2.0;
      interface ge-1/0/3.0;
      interface ge-1/0/4.0;
    }
  }
  realm ipv6-unicast {
    area 0.0.0.0 {
      interface ge-1/0/2.0;
      interface ge-1/0/3.0;
      interface ge-1/0/4.0;
    }
  }
}

Loopbacks & Router IDs
R1 = 2001:db8::1/128 & 10.10.10.1
R2 = 2001:db8::2/128 & 10.10.10.2
R3 = 2001:db8::3/128 & 10.10.10.3
R4 = 2001:db8::4/128 & 10.10.10.4
R5 = 2001:db8::5/128 & 10.10.10.5

AS 64109
2001:db8:8000::48
OSPFv3
VERIFICATION COMMANDS (JUNOS & IOS)

* Does not cover Cisco IOS-XR, NX-OS
Verifying OSPFv3

- Verify OSPFv3 adjacencies
- Confirm that all expected OSPF routes are present and active
- Further verify the exchange of routes by reviewing the OSPFv3 link state database
- Conduct ping and traceroute testing
## Juniper / Cisco Verification Command Summary

<table>
<thead>
<tr>
<th>Juniper Command</th>
<th>Cisco IOS Command</th>
<th>Co-Ordinating Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ipv6 neighbors</td>
<td>show ipv6 neighbors</td>
<td>Show IPv6 neighbor cache information</td>
</tr>
<tr>
<td>show interface</td>
<td>show ipv6 interface</td>
<td>Show IPv6 interface information</td>
</tr>
<tr>
<td>show ospf3 overview</td>
<td>show ipv6 ospf</td>
<td>Show overview of OSPF information</td>
</tr>
<tr>
<td>show ospf3 neighbor</td>
<td>show ipv6 ospf neighbor</td>
<td>Show OSPFv3 neighbor status</td>
</tr>
<tr>
<td>show ospf3 interface</td>
<td>show ipv6 ospf interface</td>
<td>Shows Link-local address, Link type, PID, Cost, Hello/Dead timers, RID, Instance ID</td>
</tr>
<tr>
<td>show ospf3 database</td>
<td>show ipv6 ospf database</td>
<td>Show OSPF LSDB related information</td>
</tr>
<tr>
<td>show ospf3 route</td>
<td>show ipv6 route ospf</td>
<td>Show the current state of the routing table</td>
</tr>
<tr>
<td>show route table inet6</td>
<td>show ipv6 route summary</td>
<td></td>
</tr>
<tr>
<td>show pfe route inet6 [table IPv6]</td>
<td>show ipv6 route summary</td>
<td></td>
</tr>
<tr>
<td>show ospf3 log</td>
<td></td>
<td>Show OSPFv3 SPF log</td>
</tr>
<tr>
<td>show ospf3 statistics</td>
<td>show ipv6 ospf traffic</td>
<td>Show OSPFv3 traffic statistics</td>
</tr>
<tr>
<td>show ospf3 io-statistics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>clear ospf3 database</td>
<td>clear ipv6 ospf</td>
<td>Clear OSPF process, counters, etc.</td>
</tr>
<tr>
<td>Use traceoptions with flags</td>
<td>debug ipv6 ospf</td>
<td>undebug</td>
</tr>
<tr>
<td>ping [ipv6]</td>
<td>ping [ipv6]</td>
<td>Send echo messages</td>
</tr>
<tr>
<td>traceroute [ipv6]</td>
<td>traceroute [ipv6]</td>
<td>Trace route to destination</td>
</tr>
</tbody>
</table>

- Commands may include additional extensions
Verify OSPFv3 Adjacencies - JunOS

IPv6@r3> show ospf3 interface

<table>
<thead>
<tr>
<th>Interface</th>
<th>State</th>
<th>Area</th>
<th>DR-ID</th>
<th>BDR-ID</th>
<th>Nbrs</th>
</tr>
</thead>
<tbody>
<tr>
<td>ge-1/0/1.0</td>
<td>BDR</td>
<td>0.0.0.0</td>
<td>10.10.10.2</td>
<td>10.10.10.3</td>
<td>1</td>
</tr>
<tr>
<td>ge-1/0/2.0</td>
<td>BDR</td>
<td>0.0.0.0</td>
<td>10.10.10.1</td>
<td>10.10.10.3</td>
<td>1</td>
</tr>
<tr>
<td>ge-1/0/4.34</td>
<td>DR</td>
<td>0.0.0.0</td>
<td>10.10.10.3</td>
<td>10.10.10.4</td>
<td>1</td>
</tr>
<tr>
<td>lo0.0</td>
<td>DRouter</td>
<td>0.0.0.0</td>
<td>0.0.0.0</td>
<td>0.0.0.0</td>
<td>0</td>
</tr>
<tr>
<td>ge-1/0/4.35</td>
<td>BDR</td>
<td>0.0.0.1</td>
<td>10.10.10.5</td>
<td>10.10.10.3</td>
<td>1</td>
</tr>
</tbody>
</table>

IPv6@r3> show ospf3 neighbor

<table>
<thead>
<tr>
<th>ID</th>
<th>Interface</th>
<th>State</th>
<th>Pri</th>
<th>Dead</th>
<th>Neighbor-address</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.2</td>
<td>ge-1/0/1.0</td>
<td>Full</td>
<td>128</td>
<td>31</td>
<td>fe80::202:b3ff:fece:94b0</td>
</tr>
<tr>
<td>10.10.10.1</td>
<td>ge-1/0/2.0</td>
<td>Full</td>
<td>128</td>
<td>33</td>
<td>fe80::202:b3ff:fe0a:c826</td>
</tr>
<tr>
<td>10.10.10.4</td>
<td>ge-1/0/4.34</td>
<td>Full</td>
<td>128</td>
<td>34</td>
<td>fe80::202:b300:2214:a3e</td>
</tr>
<tr>
<td>10.10.10.5</td>
<td>ge-1/0/4.35</td>
<td>Full</td>
<td>128</td>
<td>33</td>
<td>fe80::290:2700:233f:1cbe</td>
</tr>
</tbody>
</table>
Verify OSPFv3 Adjacencies - IOS

```
R3# show ipv6 ospf interface brief
Interface  PID   Area     Intf ID  Cost  State Nbrs F/C
Lo0        100   0        12      1    LOOP  0/0
V19        100   0        15      1    DR    1/1
V14        100   0        13      1    BDR   1/1
V11        100   0        11      1    BDR   1/1
V17        100   1        14      1    BDR   1/1
```

```
R3# show ipv6 ospf neighbor
Neighbor ID Pri State   Dead Time Interface ID Interface
10.10.10.1 1 FULL/BDR 00:00:32 15      Vlan9
10.10.10.2 1 FULL/DR  00:00:32 16      Vlan4
10.10.10.4 1 FULL/DR  00:00:39 11      Vlan1
10.10.10.5 1 FULL/DR  00:00:35 14      Vlan7
```
Confirming OSPFv3 Routes - JunOS

IPv6@r1> show route protocol ospf3

inet6.0: 25 destinations, 28 routes (25 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

    > to fe80::202:b3ff:fe0a:a170 via ge-1/0/4.0

<snip>

    to fe80::290:27ff:fe46:8ac6 via ge-1/0/2.0
    > to fe80::2a0:c9ff:feca:9cc2 via ge-1/0/3.0

    > to fe80::290:27ff:fe46:8ac6 via ge-1/0/2.0

    > to fe80::2a0:c9ff:feca:9cc2 via ge-1/0/3.0

ff02::5/128  *[OSPF3/10] 20:21:22, metric 1
    MultiRecv
Confirming OSPFv3 Routes - IOS

R1#show ipv6 route ospf
IPv6 Routing Table - Default - 23 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
B - BGP, M - MIPv6, R - RIP, D - EIGRP  EX - EIGRP external
O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
OE2 ::/0 [110/1], tag 100
  via FE80::21F:CAFF:FE6B:DF42, Vlan9
  via FE80::21F:CAFF:FE6B:E26C, Vlan2
  via FE80::21F:CAFF:FE6B:DF42, Vlan9
  via FE80::21F:CAFF:FE6B:E26C, Vlan2
<brnip>
O  2001:DB8::3/128 [110/1]
  via FE80::21F:CAFF:FE6B:DF42, Vlan9
OI  2001:DB8:0:8::/64 [110/2]
  via FE80::21F:CAFF:FE6B:E26C, Vlan2
### The OSPFv3 LSDB – JunOS

IPv6@r5> **show ospf3 database**

**OSPF3 link state database, Area 0.0.0.1**

<table>
<thead>
<tr>
<th>Type</th>
<th>ID</th>
<th>Adv Rtr</th>
<th>Seq</th>
<th>Age</th>
<th>Cksum</th>
<th>Len</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Router</strong></td>
<td>0.0.0.0</td>
<td>10.10.10.3</td>
<td>0x80000024</td>
<td>1643</td>
<td>0xf7b1</td>
<td>40</td>
</tr>
<tr>
<td><strong>Network</strong></td>
<td>*0.0.0.1</td>
<td>10.10.10.5</td>
<td>0x80000020</td>
<td>873</td>
<td>0x852b</td>
<td>32</td>
</tr>
<tr>
<td><strong>InterArPfx</strong></td>
<td>0.0.0.1</td>
<td>10.10.10.3</td>
<td>0x80000023</td>
<td>908</td>
<td>0x2a63</td>
<td>44</td>
</tr>
<tr>
<td><strong>IntraArPfx</strong></td>
<td>*0.0.0.2</td>
<td>10.10.10.5</td>
<td>0x80000020</td>
<td>573</td>
<td>0x3d4f</td>
<td>44</td>
</tr>
</tbody>
</table>

**OSPF3 Link-Local link state database, interface ge-1/0/1.45**

<table>
<thead>
<tr>
<th>Type</th>
<th>ID</th>
<th>Adv Rtr</th>
<th>Seq</th>
<th>Age</th>
<th>Cksum</th>
<th>Len</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Link</strong></td>
<td>0.0.0.4</td>
<td>10.10.10.4</td>
<td>0x80000024</td>
<td>130</td>
<td>0xd889</td>
<td>56</td>
</tr>
<tr>
<td>Link</td>
<td>*0.0.0.1</td>
<td>10.10.10.5</td>
<td>0x80000021</td>
<td>1473</td>
<td>0x4895</td>
<td>56</td>
</tr>
</tbody>
</table>
IPv6@r5> `show ospf3 database detail inter-area-prefix advertising-router 10.10.10.3`

OSPF3 link state database, Area 0.0.0.1

<table>
<thead>
<tr>
<th>Type</th>
<th>ID</th>
<th>Adv Rtr</th>
<th>Seq</th>
<th>Age</th>
<th>Cksum</th>
<th>Len</th>
</tr>
</thead>
<tbody>
<tr>
<td>InterArPfx</td>
<td>0.0.0.1</td>
<td>10.10.10.3</td>
<td>0x80000024</td>
<td>276</td>
<td>0x2864</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>Prefix 2001:db8::1/128</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Prefix-options 0x0, Metric 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>InterArPfx</td>
<td>0.0.0.2</td>
<td>10.10.10.3</td>
<td>0x80000024</td>
<td>246</td>
<td>0x3852</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>Prefix 2001:db8::2/128</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Prefix-options 0x0, Metric 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>InterArPfx</td>
<td>0.0.0.3</td>
<td>10.10.10.3</td>
<td>0x80000024</td>
<td>111</td>
<td>0xc86</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>Prefix 2001:db8::3/128</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Prefix-options 0x0, Metric 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;snip&gt;</code></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>InterArPfx</td>
<td>0.0.0.11</td>
<td>10.10.10.3</td>
<td>0x80000025</td>
<td>1476</td>
<td>0x106e</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>Prefix 2001:db8::4/128</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Prefix-options 0x0, Metric 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### OSPFv3 Router with ID (10.10.10.5) (Process ID 100)

#### Router Link States (Area 1)

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Fragment ID</th>
<th>Link count</th>
<th>Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.3</td>
<td>627</td>
<td>0x80000038</td>
<td>0</td>
<td>1</td>
<td>B</td>
</tr>
<tr>
<td>10.10.10.4</td>
<td>1051</td>
<td>0x8000003B</td>
<td>0</td>
<td>1</td>
<td>B</td>
</tr>
<tr>
<td>10.10.10.5</td>
<td>1244</td>
<td>0x80000341</td>
<td>0</td>
<td>2</td>
<td>E</td>
</tr>
</tbody>
</table>

#### Net Link States (Area 1)

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Link ID</th>
<th>Rtr count</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.5</td>
<td>732</td>
<td>0x80000036</td>
<td>14</td>
<td>2</td>
</tr>
</tbody>
</table>

#### Inter Area Prefix Link States (Area 1)

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.3</td>
<td>627</td>
<td>0x80000036</td>
<td>2001:DB8::3/128</td>
</tr>
</tbody>
</table>

#### Link (Type-8) Link States (Area 1)

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Link ID</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.4</td>
<td>1052</td>
<td>0x80000311</td>
<td>15</td>
<td>Vl8</td>
</tr>
</tbody>
</table>

#### Intra Area Prefix Link States (Area 1)

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Link ID</th>
<th>Ref-lstype</th>
<th>Ref-LSID</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.5</td>
<td>733</td>
<td>0x80000278</td>
<td>0</td>
<td>0x2001</td>
<td>0</td>
</tr>
</tbody>
</table>

#### Type-5 AS External Link States

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.5</td>
<td>1504</td>
<td>0x80000007</td>
<td>2001:506:8::/48</td>
</tr>
</tbody>
</table>
R5# `show ipv6 ospf database inter-area prefix adv-router 10.10.10.3`

OSPFv3 Router with ID (10.10.10.5) (Process ID 100)

Inter Area Prefix Link States (Area 1)

Routing Bit Set on this LSA
LS age: 1180
LS Type: Inter Area Prefix Links
Link State ID: 0
Advertising Router: 10.10.10.3
LS Seq Number: 80000036
Checksum: 0x67D
Length: 44
Metric: 0
Prefix Address: 2001:DB8::3
Prefix Length: 128, Options: None

<snip>
IPv6@r1> **ping** 2001:db8::5 count 3 rapid
PING6(56=40+8+8 bytes) 2001:db8:0:2::1 --> 2001:db8::5
!!!
--- 2001:db8::4 ping6 statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/std-dev = 0.399/0.458/0.562/0.074 ms

IPv6@r5> **traceroute** 2001:db8::1
traceroute6 to 2001:db8::1 (2001:db8::1) from
2001:db8:0:8:2a0:c900:2dca:9cc6, 30 hops max, 12 byte packets
1  2001:db8:0:8:202:b300:2d14:a3e (2001:db8:0:8:202:b300:2d14:a3e) 596.314 ms 0.613 ms 0.508 ms
2  2001:db8::1 (2001:db8::1)  1.101 ms 0.989 ms 0.951 ms
R5#traceroute 2001:db8::1
Type escape sequence to abort.
Tracing the route to 2001:db8::1

1 2001:db8:0:7::3 0 msec 0 msec 0 msec
2 2001:db8:0:4::1 4 msec 0 msec 4 msec

R1#ping 2001:db8::5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:db8::5, timeout is 2 seconds: !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/3/8 ms

Cisco 871
Version 12.4(24)T4
IPv4/IPv6 EIGRP Similarities

- Uses RTP for reliable delivery of packets
- Uses DUAL to perform route computation
- Same basic packet types
  - Hello, Acks, Updates, Query, Replays
- Same mechanisms for neighbor discovery and recovery
- Same timers, composite metric attributes
- Builds neighbor, topology, routing tables (although separate from IPv4)
- Same protocol number: 88 (0x58)
What’s New for IPv6 in EIGRP?

- Enhancements to support IPv6 prefixes
- New Protocol Dependent Module (PDM) to route IPv6
- New TLV’s to carry IPv6 internal/external prefixes
- Runs directly over IPv6
  - EIGRP control packets sourced from link-local address and destined to FF02:A (All EIGRP Routers) or neighbor’s link-local
- “no auto-summary” is disabled by default
- EIGRP runs independently from both IPv4 and IPv6
  - Separate and distinct processes, tables (routing, topology, neighbor), CLI output, etc.
- Available in IOS 12.4(6)T and later
EIGRP for IPv6

PACKET FORMAT CHANGES
EIGRP Packet Header Format

All EIGRP Packet types begin with a standard 20BHdr
All fields are the same for IPv4 and IPv6
Three new IPv6 Specific TLV Types added:

- 0x0401 (1025) = IPv6_REQUEST_TYPE
- 0x0402 (1026) = IPv6_METRIC_TYPE (Internal Routes)
- 0x0403 (1027) = IPv6_EXTERNAL_TYPE (External Routes)
EIGRP for IPv6

CONFIGURATION EXAMPLES

* Does not cover Cisco IOS-XR, NX-OS
EIGRP Example - Cisco

Cisco 871
Version 12.4(24)T4

Loopbacks & Router IDs
R1 = 2001:db8::1/128 & 10.10.10.1
R2 = 2001:db8::2/128 & 10.10.10.2
R3 = 2001:db8::3/128 & 10.10.10.3
R4 = 2001:db8::4/128 & 10.10.10.4
R5 = 2001:db8::5/128 & 10.10.10.5

AS 64109
2001:db8:8000::/48

AS 64109
2001:db8:8000::/48
The following command lines provide the minimum set of requirements to enable EIGRP:

- **EIGRP for IPv4**
  ```
  ip cef
  
  interface Vlan8
  ip address 10.10.10.34 255.255.255.0
  no shut
  
  router eigrp 1
  network 10.10.10.0 0.0.0.255
  no auto-summary
  ```

- **EIGRP for IPv6**
  ```
  ipv6 unicast-routing
  ipv6 cef
  
  interface Vlan8
  ipv6 address 2001:DB8:0:8::5/126
  ipv6 eigrp 1
  
  ipv6 router eigrp 1
  eigrp router-id 10.10.10.1
  no shut
  ```

Assumes the link-local addresses are built using M-EUI-64.
### Command Comparison

**Optional Commands**

#### EIGRP for IPv4

**Interface Commands:**
- `interface Vlan8`
  - `ip summary-address eigrp 10 1.1.0 255.255.255.0 5`
  - `ip hello-interval eigrp 1 30`
  - `ip hold-time eigrp 1 90`
  - `ip bandwidth-percent eigrp 1 75`

**EIGRP Router Commands:**
- `router eigrp 1`
  - `default-information originate`
  - `default-metric 10000 100 255 1 1500`
  - `redistribute bgp 64109 metric 100000 100 255 1 1500`
  - `eigrp log-neighbor-changes`
  - `eigrp log-neighbors warnings`
  - `eigrp event-logging`
  - `no auto-summary`
  - `passive-interface loopback 0`

#### EIGRP for IPv6

**Interface Commands:**
- `interface Vlan8`
  - `ipv6 summary-address eigrp 10 2001:1:1::/56 5`
  - `ipv6 hello-interval eigrp 1 30`
  - `ipv6 hold-time eigrp 1 90`
  - `ipv6 bandwidth-percent eigrp 1 75`

**EIGRP Router Commands:**
- `ipv6 router eigrp 1`
  - `default-metric 10000 100 255 1 1500`
  - `redistribute bgp 64109 metric 100000 100 255 1 1500`
  - `eigrp log-neighbor-changes`
  - `eigrp log-neighbors warnings`
  - `eigrp event-logging`
  - `passive-interface loopback 0`
EIGRP Router ID

- **Router ID** = Identifies who sent the EIGRP packet
  - 32-bit number like OSPFv3
  - EIGRP searches for an IPv4 address to get the “Router ID” for the IPv6 routing process

- **IPv6-only Environment**
  - No IPv4 addresses defined - Must manually define Router ID “router-id x.x.x.x” in order to establish a neighbor

- **Dual Stack (IPv4/6) Environment** – The EIGRP router-id will be, in order:
  - The one configured by the “router-id” command, if none
  - The highest IPv4 loopback interface when EIGRP process starts, if none
  - The highest IPv4 on a non-loopback interface

- **Best Practices**
  - Configure IPv6 EIGRP Process first, with manually configured RID
  - Then configure interface statements
ipv6 unicast-routing

interface Loopback0
ipv6 address 2001:DB8::1/128

interface Vlan2
description "To R4"
ipv6 address 2001:DB8:0:2::1/126

ipv6 eigrp 1

interface Vlan3
description "To R2"
ipv6 address 2001:DB8:0:1::1/126

ipv6 eigrp 1

interface Vlan9
description "To R3"
ipv6 address 2001:DB8:0:4::1/126

ipv6 eigrp 1

ipv6 router eigrp 1
no shutdown
router-id 10.10.10.1
passive-interface loopback 0
**IOS Configuration for R4**

```
!ipv6 unicast-routing
!
interface Loopback0
ipv6 address 2001:DB8::4/128
!
interface Vlan1 – Vlan2 – Vlan 5
!
ipv6 eigrp 1
!
interface Vlan8
description "To R5"
ipv6 address 2001:DB8:0:8::4/126
!
ipv6 eigrp 2
!
ipv6 router eigrp 1
router-id 10.10.10.4
no shut
passive-interface loopback 0
!
ipv6 router eigrp 2
router-id 10.10.10.4
no shut
passive-interface loopback 0
```
interface Loopback0
ipv6 address 2001:DB8::5/128
!
interface Vlan7 – Vlan8
  ipv6 eigrp 2
!
interface Vlan11
description "To P1"
ipv6 address 2001:db8:8000:4200::2/126
!
routing bgp 64109
bgp router-id 10.10.10.5
no bgp default ipv4-unicast
neighbor 22001:db8:8000:4200::1 remote-as 4323
  address-family ipv6
  neighbor 2001:db8:8000:4200::1 activate
  no synchronization
  exit-address-family
!
ipv6 router eigrp 2
router-id 10.10.10.5
no shutdown
passive-interface Loopback0
passive-interface vlan 11
redistribute bgp 64109 metric 100000 100 255 1 1500
Other EIGRP Commands

- **EIGRP MD5 Authentication**
  - Same MD5 authentication mechanism as EIGRP for IPv4
  - Authentication ensures that neighboring router only accept packets from other routers that know the same pre-shard key
    - The MD5 value generated is appended to the EIGRP packet and is then sent to a neighbor.
    - The receiving neighbor then compares the result with the MD5 fingerprint in the packet
  - Shared secrets are manually configured on the routers
    - No mechanism for automatic key generation or key distribution exits
    - Shared secrets are stored in the router configuration in plain-text
Global Example

- key chain chain1
  - key 1
  - key-string testkey
  - accept-lifetime 14:30:00 Apr 27 2011 duration 7200
  - send-lifetime 15:00:00 Apr 27 2011 duration 3600

Apply to Interface

- ipv6 authentication mode eigrp 1 md5
- ipv6 authentication key-chain eigrp 1 chain1

Verification

- show ipv6 eigrp interfaces detail
- show key chain
- debug ip eigrp packets verbose -
  - EIGRP: ignored packet from x.x.x.x opcode = 5 (invalid authentication)
Other EIGRP Commands

- **EIGRP Distribute List (DL)**
  - To filter EIGRP routing updates, use a distribute list (w/ prefix-list)
  - A DL can be used to filter inbound or outbound routing updates
  - Unlike OSPFv3, EIGRP can use a DL to limit which prefixes assigned to an interface are advertised to a neighbor

Example: Deny 2001:DB5::/32 routes with prefix length between /126 and /128
ipv6 prefix-list FILTER_IPV6 seq 5 deny 2001:DB5::/32 ge 64 → Deny
ipv6 prefix-list FILTER_IPV6 seq 10 permit ::/0 le 128 → Permit all other routes

!ipv6 router eigrp 1
eigrp router-id 10.10.10.2
no shutdown
distribute-list prefix-list FILTER_IPV6 out Vlan3
Other EIGRP Commands

- **EIGRP Redistribution**
  - When redistributing IPv6 routes from one protocol into EIGRP, use the **redistribute** command
  - Routing Protocols include:
    - RIPng, OSPFv3, BGP, another EIGRP AS, Static, IS-IS

Example: BGP 64109 learn routes into EIGRP ASN 2

```plaintext
ipv6 prefix-list EBGP-IPv6-FROM-P1 seq 5 permit 2001:506:8::/48
!
routemap BGP_INTO_EIGRP deny 10
  match ipv6 address prefix-list EBGP-IPv6-FROM-P1 ➔ Denies 2001:506:8::/48
routemap BGP_INTO_EIGRP permit 20 ➔ Permits everything else
!
ipv6 router eigrp 2
  redistribute bgp 64109 metric 100000 100 255 1 1500 route-map BGP_INTO_EIGRP
```
EIGRP Redistribution Configuration Stanza

- **Cisco**

```
ipv6 router eigrp <asn>

default-metric <BW> <DLY> <REL> <LOAD><MTU>
redistribute eigrp <asn> [<BW> <DLY> <REL> <LOAD><MTU>] [route-map] [include-connected]
redistribute connected [<BW> <DLY> <REL> <LOAD><MTU>] [route-map]
redistribute static [<BW> <DLY> <REL> <LOAD><MTU>] [route-map]
redistribute rip <name> [<BW> <DLY> <REL> <LOAD><MTU>] [route-map] [include-connected]
redistribute bgp <asn> [<BW> <DLY> <REL> <LOAD><MTU>] [route-map]
redistribute ospf <pid> match [internal | external | nssa-external] [<BW> <DLY> <REL> <LOAD><MTU>] [route-map] [include-connected]
```

- “include-connected” versus “redistribute connected”
- “subnets” keyword not used
EIGRP for IPv6

VERIFICATION COMMANDS

* Does not cover Cisco IOS-XR, NX-OS
Verifying EIGRP

- Verify EIGRP neighbor adjacencies
- Confirm that all expected EIGRP routes are present and in a Passive (P) state
- Further verify the exchange of routes by reviewing the EIGRP topology database
- Conduct ping and traceroute testing
# EIGRP for IPv6 Verification Command Summary

<table>
<thead>
<tr>
<th>Cisco IOS Command</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ipv6 route eigrp [updated]</td>
<td>Show IPv6 EIGRP routes (last updated)</td>
</tr>
<tr>
<td>show ipv6 eigrp interface</td>
<td>Show IPv6 EIGRP interfaces</td>
</tr>
<tr>
<td>show ipv6 eigrp neighbors</td>
<td>Show IPv6 EIGRP Neighbor table</td>
</tr>
<tr>
<td>show ipv6 eigrp topology</td>
<td>Show IPv6 EIGRP Topology Table</td>
</tr>
<tr>
<td>show ipv6 eigrp traffic</td>
<td>Show IPv6 EIGRP traffic statistics</td>
</tr>
<tr>
<td>show ipv6 protocols</td>
<td>Show IPv6 EIGRP process information</td>
</tr>
<tr>
<td>clear ipv6 eigrp neighbors</td>
<td>Clear OSPF process, counters, etc.</td>
</tr>
<tr>
<td>debug ipv6 eigrp</td>
<td>Debugging functions</td>
</tr>
<tr>
<td>ping [ipv6]</td>
<td>Send echo messages</td>
</tr>
<tr>
<td>traceroute [ipv6]</td>
<td>Trace route to destination</td>
</tr>
</tbody>
</table>

- Commands may include additional extensions
R1#show ipv6 eigrp neighbors
IPv6-EIGRP neighbors for process 1

<table>
<thead>
<tr>
<th>H</th>
<th>Address</th>
<th>Interface</th>
<th>Hold Uptime (sec)</th>
<th>SRTT (ms)</th>
<th>RTO</th>
<th>Q</th>
<th>Seq Num</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Link-local address:</td>
<td>Vl2</td>
<td>11 2d20h</td>
<td>1</td>
<td>200</td>
<td>0</td>
<td>155</td>
</tr>
<tr>
<td></td>
<td>FE80::21F:CAFF:FE6B:E26C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Link-local address:</td>
<td>Vl9</td>
<td>13 07:38:53</td>
<td>1</td>
<td>200</td>
<td>0</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td>FE80::21F:CAFF:FE6B:DF42</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>Link-local address:</td>
<td>Vl3</td>
<td>12 07:38:57</td>
<td>1</td>
<td>200</td>
<td>0</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td>FE80::21F:CAFF:FE6B:E576</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
IPv6 Routing Table - Default - 25 entries

Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
B - BGP, M - MIPv6, R - RIP, D - EIGRP, EX - EIGRP external
O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2

EX ::/0 [170/53760]
  via FE80::21F:CAFF:FE6B:DF42, Vlan9
D 2001:DB8::/64 [90/156160]
  via FE80::21F:CAFF:FE6B:E576, Vlan3
<br>snip>
EX 2001:DB8:3:ABCD::/64 [170/53760]
  via FE80::21F:CAFF:FE6B:DF42, Vlan9
Confirming EIGRP Topology Table

R1#show ipv6 eigrp topology
IPv6-EIGRP Topology Table for AS(1)/ID(10.10.10.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P ::/0, 1 successors, FD is 53760
   via FE80::21F:CAFF:FE6B:DF42 (53760/51200), Vlan9

<snip>
P 2001:DB8:0:7::/64, 1 successors, FD is 53760
   via FE80::21F:CAFF:FE6B:DF42 (53760/51200), Vlan9
P 2001:DB8:0:6::/64, 1 successors, FD is 30720
   via FE80::21F:CAFF:FE6B:DF42 (30720/28160), Vlan9
P FEC0::/64, 1 successors, FD is 28160
   via Connected, Vlan9
R1#show ipv6 eigrp topology 2001:DB8:0:6::/64
IPv6-EIGRP (AS 1): Topology entry for 2001:DB8:0:6::/64
State is Passive, Query origin flag is 1, 1 Successor(s), FD is 30720
Routing Descriptor Blocks:
FE80::21F:CAFF:FE6B:DF42 (Vlan9), from FE80::21F:CAFF:FE6B:DF42, Sendflag is 0x0
Composite metric is (30720/28160), Route is Internal
Vector metric:
Minimum bandwidth is 100000 Kbit
Total delay is 200 microseconds
Reliability is 255/255
Load is 1/255
Minimum MTU is 1500
Hop count is 1
IPv6-EIGRP (AS 1): Topology entry for 2001:DB8:0:8::/64
State is Passive, Query origin flag is 1, 1 Successor(s), FD is 53760
Routing Descriptor Blocks:
FE80::21F:CAFF:FE6B:DF42 (Vlan9), from FE80::21F:CAFF:FE6B:DF42, Send flag is 0x0
Composite metric is (53760/51200), Route is External
Vector metric:
Minimum bandwidth is 100000 Kbit
Total delay is 1100 microseconds
Reliability is 255/255
Load is 1/255
Minimum MTU is 1500
Hop count is 1
External data:
Originating router is 10.10.10.3
AS number of route is 2
External protocol is EIGRP, external metric is 30720
Administrator tag is 0 (0x00000000)
BGP for IPv6

Border Gateway Protocol
IPv6 BGP Basics

- Supports both IPv4 and IPv6
- Supports the exchange of IPv6 routes in two ways:
  
  **Native IPv6 Peering:** This method does not require any IPv4 addresses (other than a single 32-bit router ID) and supports IPv6 Network Layer Reachability Information (NLRI) only.

  **Advertising IPv6 NLRI over IPv4 Peering:** Leveraging multiprotocol BGP extensions, this method supports both IPv6 and IPv4 NLRI, in addition to any other needed NLRI
BGP Example - Juniper

AS 65000
2001:db8::/48

R1
ge-1/0/2
2001:db8:0:4::/126
10.10.4.0/30

R3
ge-1/0/1
2001:db8:0:7::/126
10.10.7.0/30

R4
ge-1/0/4
2001:db8:0:4::/126
10.10.4.0/30

R5
ge-1/0/3.45
2001:db8:0:8::/126
10.10.8.0/30

P1
2001:db8:8000:4200::2/126
168.215.48.2/30

AS 64109
2001:db8:8000::/48
MD5 = X-L@RG3

Loopbacks & Router IDs
R1 = 2001:db8:1/128 & 10.10.10.1
R2 = 2001:db8:2/128 & 10.10.10.2
R3 = 2001:db8:3/128 & 10.10.10.3
R4 = 2001:db8:4/128 & 10.10.10.4
R5 = 2001:db8:5/128 & 10.10.10.5
BGP Example - Cisco

AS 65000
2001:db8::/48

Cisco 871
Version 12.4(24)T4

Loopbacks & Router IDs
R1 = 2001:db8::1/128 & 10.10.10.1
R2 = 2001:db8::2/128 & 10.10.10.2
R3 = 2001:db8::3/128 & 10.10.10.3
R4 = 2001:db8::4/128 & 10.10.10.4
R5 = 2001:db8::5/128 & 10.10.10.5

AS 64109
2001:db8:8000::/48
MD5 = X-L@RG3
Native IPV6 BGP Peering

CONFIGURING DEDICATED IPV6 PEERING

VERIFYING IPV6 PEERING SESSIONS
JunOS Configuration for R5

policy-options {
    policy-statement peer-export {
        term agg {
            from {
                protocol aggregate;
                route-filter 2001:db8::/48 exact;
            }
            then accept;
        }
        term kil {
            then reject;
        }
    }
}

protocols bgp group PEERS {
    export peer-export;
    type external;
    neighbor 2001:db8:8000:4200::1 {
        authentication-key "$9$XDMxw2q.Pz3/9Av87-sY5Qz"; ## SECRET-DATA
        peer-as 64109;
    }
}

[routing-options {
    router-id 10.10.10.5
    autonomous-system 65000;
    rib inet6.0 {
        aggregate {
            route 2001:db8::/48;
        }
    }
}]

protcols bgp group PEERS {
    export peer-export;
    type external;
    neighbor 2001:db8:8000:4200::1 {
        authentication-key "$9$XDMxw2q.Pz3/9Av87-sY5Qz"; ## SECRET-DATA
        peer-as 64109;
    }
}
Enable MP-BGP with IPv6 address family to exchange IPv6 unicast prefixes

R1#
router bgp 65000
  bgp router-id 10.10.10.1       → Manually configure, or use highest IP, with preference to LBPK
  no bgp default ipv4-unicast
  bgp log-neighbor-changes       → Enabled by default
neighbor 2001:DB8:0:4::2 remote-as 65000
!
  address-family ipv6 unicast    → Place router in address-family IPv6 configuration submode
    neighbor 2001:DB8:0:4::2 activate → Activate peer session
    neighbor 2001:DB8:0:4::2 soft-reconfiguration inbound
    neighbor 2001:DB8:0:4::2 route-map SETNH out
  network 2001:DB8:FFFF:123::/64 → Specify an IPv6 prefix to announce via BGP4+
exit-address-family
router bgp 65000
  bgp router-id 10.10.10.5
  no bgp default ipv4-unicast
  bgp log-neighbor-changes
  neighbor 2001:DB8::1 remote-as 65000
  neighbor 2001:DB8::1 description "To R1"  \ Full mesh IBGP
<snip>
  neighbor 2001:DB8::4 remote-as 65000
  neighbor 2001:DB8::4 description "To R4"
  neighbor 2001:DB8:8000:4200::1 remote-as 64109  \ eBGP
  neighbor 2001:DB8:8000:4200::1 description "To P1"
!
  address-family ipv6
  neighbor 2001:DB8::1 activate
  neighbor 2001:DB8::1 next-hop-self
<snip>
  neighbor 2001:DB8::4 activate
  neighbor 2001:DB8::4 next-hop-self
  neighbor 2001:DB8:8000:4200::1 activate
  neighbor 2001:DB8:8000:4200::1 soft-reconfiguration inbound
  no synchronization
  exit-address-family
router bgp 65000
   bgp router-id 10.10.10.5
   no bgp default ipv4-unicast
neighbor 168.215.48.1 remote-as 64109  ➔ IPv4 EBGP Peer
neighbor 2001:db8:8000:4200::1 remote-as 64109  ➔ IPv6 EBGP Peer

address-family ipv4
   neighbor 168.215.48.1 activate
   neighbor 168.215.48.1 prefix-list V4_IN in
   neighbor 168.215.48.1 prefix-list V4_OUT out
   no auto-summary
   no synchronization
   exit-address-family

ip prefix-list V4_IN seq 5 permit 0.0.0.0/0 le 32
ip prefix-list V4_OUT seq 5 permit 172.20.10.0/24 le 32

ipv6 prefix-list V6_IN seq 5 permit 2001::/32 le 64
ipv6 prefix-list V6_OUT seq 5 permit 2001:DB8::/32 le 48

address-family ipv6
   neighbor 2001:db8:8000:4200::1 activate
   neighbor 2001:db8:8000:4200::1 prefix-list V6_IN in
   neighbor 2001:db8:8000:4200::1 prefix-list V6_OUT out
   network 2001:db8:FF21:34::/64
   exit-address-family
Verifying Native IPv6 peering

- Verify BGP neighbors
- Confirm BGP routes are being exchanged as expected
- Conduct ping and traceroute testing
Verifying BGP Neighbors - JunOS

```
show bgp summary
Groups: 1 Peers: 1 Down peers: 0
Table      Tot Paths  Act Paths Suppressed  History Damp State   Pending
inet6.0    1        1        0        0          0          0

Peer                  OutPkt  OutQ  Flaps  Last Up/Dwn  State|Active/Received/Damped
2001:db8:8000:4200::1  64109 373    371      2:45:56      Establ

inet6.0: 1/1/0
```

Diagram showing network topology with routers R2 and R4 connected to a common network, with AS 65000 and AS 64109 highlighted.
Verifying BGP neighbors – JunOS (cont.)

ipv6@r5> show bgp neighbor 2001:db8:8000:4200::1
Type: External    State: Established    Flags: <Sync>
Last State: OpenConfirm    Last Event: RecvKeepAlive
Last Error: None
Export: [ peer-export ]
Options: <Preference AuthKey PeerAS Refresh>
Authentication key is configured
Holdtime: 90 Preference: 170
Number of flaps: 0
Peer ID: 10.0.1.8    Local ID: 10.10.10.5    Active Holdtime: 90
Keepalive Interval: 30    Peer index: 0
BFD: disabled, down
Local Interface: ge-1/0/3.84
NLRI advertised by peer: inet6-unicast
NLRI for this session: inet6-unicast
Peer supports Refresh capability (2)
Table inet6.0 Bit: 10000
    RIB State: BGP restart is complete
    Send state: in sync
    Active prefixes: 1
    Received prefixes: 1
    Suppressed due to damping: 0
    Advertised prefixes: 1
Last traffic (seconds): Received 3    Sent 22    Checked 18
Input messages: Total 417    Updates 1    Refreshes 0    Octets 7979
Output messages: Total 415    Updates 1    Refreshes 0    Octets 7967
Output Queue[0]: 0
Confirm BGP Routes - JunOS

ipv6@r4> show route receive-protocol bgp 2001:db8:8000:4200::1

inet.0: 0 destinations, 0 routes (0 active, 0 holddown, 0 hidden)

inet6.0: 29 destinations, 33 routes (29 active, 0 holddown, 0 hidden)

Prefix            Nexthop            MED   Lclpref   AS path
* 2001:db8:8000::/48  2001:db8:8000:4200::1  64109   I

__juniper_private1__.inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)

ipv6@r4> show route advertising-protocol bgp 2001:db8:8000:4200::1

inet6.0: 29 destinations, 33 routes (29 active, 0 holddown, 0 hidden)

Prefix            Nexthop            MED   Lclpref   AS path
* 2001:db8::/48    Self               I
Exchanging IPV6 over IPv4 Peering

ADDING THE IPV6 NLRI TO EXISTING PEERING
Configuration Changes for R5 (JunOS)

ipv6@r5# show | compare
[edit interfaces lo0 unit 0 family inet6 address 2001:db8::2/128]
+    primary;
+    preferred;
[edit interfaces lo0 unit 0 family inet6]
    address 2001:db8::2/128 { ... }
+    address ::fff:10.10.10.5/128;
[edit protocols bgp group IBGP]
+    family inet {
+        unicast;
+    }
+    family inet6 {
+        unicast;
+    }
+    export nhs;
[edit policy-options]
+    policy-statement nhs {
+        term P1 {
+            from {
+                protocol bgp;
+                neighbor 168.215.48.1;
+            }
+            then {
+                next-hop self;
+            }
+        }
+    }

Loopbacks & Router IDs
R1 = 2001:db8::1/128 & 10.10.10.1
R2 = 2001:db8::2/128 & 10.10.10.2
R3 = 2001:db8::3/128 & 10.10.10.3
R4 = 2001:db8::4/128 & 10.10.10.4
R5 = 2001:db8::5/128 & 10.10.10.5
Carrying IPv6 inside IPv4 peering
Cisco IOS - Configuration Examples

- IPv6 prefixes can be exchanged between IPv4 BGP Peers
  - Note that we need to “fix” the next-hop

```
router bgp 65002
  neighbor 10.10.8.5 remote-as 65000

! address-family ipv6 v6 unicast
  neighbor 10.10.8.5 activate allows neighbors to exchange v6 prefixes
  neighbor 10.10.8.5 soft-reconfiguration inbound
  neighbor 10.10.8.5 route-map Set-IPv6-NH in change Inbound BGP NH
  exit-address-family

! route-map Set-IPv6-NH permit 10
  set ipv6 next-hop 2001:db8:0:8::5 IPv6 Next Hop address of remote peer
```
Carrying IPv4 inside IPv6 peering
Cisco IOS - Configuration Examples

- IPv4 prefixes can be exchanged between IPv6 BGP Peers
  - Note that we need to “fix” the next-hop

```
router bgp 65002
  neighbor 2001:db8:0:8::5 remote-as 65000
  !
  address-family ipv4
    neighbor 2001:db8:0:8::5 activate → allows neighbors to exchange v4 prefixes
    neighbor 2001:db8:0:8::5 soft-reconfiguration inbound
    neighbor 2001:db8:0:8::5 route-map Set-IPv4-NH in → Inbound
    exit-address-family
  !
  route-map Set-IPv4-NH permit 10
    set ip next-hop 10.10.8.4 → IPv4 Next Hop address of remote peer
```
Verify IPv6 support over IPv4-based BGP peering

- Confirm that all IBGP sessions are established
- Look for hidden routes
- Examine BGP neighbor details
- Verify that all BGP routers are receiving the proper BGP routes
- Test connectivity with Ping and Traceroute
Verifying IBGP Neighbors - JunOS

ipv6@r1> show bgp summary
Groups: 1 Peers: 4 Down peers: 0

Table | Tot Paths | Act Paths | Suppressed | History | Damp State | Pending |
--- | --- | --- | --- | --- | --- | --- |
Inet.0 | 0 | 0 | 0 | 0 | 0 | 0 |
inet6.0 | 1 | 1 | 0 | 0 | 0 | 0 |

Peer | AS | InPkt | OutPkt | OutQ | Flaps | Last Up/Dwn St | State | #Active/Received/Damped...
--- | --- | --- | --- | --- | --- | --- | --- | ---
10.10.10.2 | 65000 | 70 | 72 | 0 | 0 | 31:57 | Establ |
inet.0: 0/0/0 inet6.0: 0/0/0
10.10.10.3 | 65000 | 69 | 71 | 0 | 0 | 31:53 | Establ |
inet.0: 0/0/0 inet6.0: 0/0/0
10.10.10.4 | 65000 | 72 | 71 | 0 | 0 | 31:57 | Establ |
inet.0: 0/0/0 inet6.0: 1/1/0
10.10.10.5 | 65000 | 71 | 73 | 0 | 0 | 32:00 | Establ |
inet.0: 0/0/0 inet6.0: 0/0/0
Verifying BGP Neighbor - JunOS

ipv6@r1> show bgp neighbor 10.10.10.3

Peer: 10.10.10.3+179 AS 65000 Local: 10.10.10.1+1138 AS 65000
Type: Internal State: Established Flags: <Sync>
Last State: OpenConfirm Last Event: RecvKeepAlive
Last Error: None
Options: <Preference LocalAddress AuthKey AddressFamily Refresh>
Authentication key is configured

Address families configured: inet-unicast inet6-unicast
Local Address: 10.10.10.1 Holdtime: 90 Preference: 170
Number of flaps: 0
Peer ID: 10.10.10.3 Local ID: 10.10.10.1 Active Holdtime: 90
Keepalive Interval: 30 Peer index: 3
BFD: disabled, down

NLRI advertised by peer: inet-unicast inet6-unicast
NLRI for this session: inet-unicast inet6-unicast
Peer supports Refresh capability (2)

Table inet.0 Bit: 10000
RIB State: BGP restart is complete
Send state: in sync
Active prefixes: 0
Received prefixes: 0
Suppressed due to damping: 0
Advertised prefixes: 0

Table inet6.0 Bit: 20000
RIB State: BGP restart is complete
Send state: in sync
Active prefixes: 0
Received prefixes: 0
Suppressed due to damping: 0
Advertised prefixes: 0

Last traffic (seconds): Received 10 Sent 5 Checked 55
Input messages: Total 80 Updates 0 Refreshes 0 Octets 1520
Output messages: Total 82 Updates 0 Refreshes 0 Octets 1592
Output Queue[0]: 0
Output Queue[1]: 0
ipv6@r1> **show route 2001:db8:8000::/48 detail**

inet6.0: 27 destinations, 30 routes (27 active, 0 holddown, 0 hidden)
2001:db8:8000::/48 (1 entry, 1 announced)
  *BGP Preference: 170/-101
  Next-hop reference count: 3
  Source: 10.10.10.4
  Next hop: fe80::2a0:c9ff:fe6c:9cc2 via ge-1/0/3.0, selected
  **Protocol next hop: ::ffff:10.10.10.4**
  Indirect next hop: 87839a0 131070
  State: <Active Int Ext>
  Local AS: 65000 Peer AS: 65000
  Age: 56:12 Metric2: 1
  Task: BGP_65000.10.10.10.4+2563
  Announcement bits (2): 0-KRT 4-Resolve tree 2
  AS path: 64109 I Aggregator: 64109 10.0.1.8
  Localpref: 100
  Router ID: 10.10.10.4
ipv6@r1> **show route hidden**
inet.0: 19 destinations, 19 routes (19 active, 0 holddown, 0 hidden)
inetc.0: 27 destinations, 30 routes (27 active, 0 holddown, 0 hidden)
__juniper_private1__.inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
R3# `show bgp ipv6 unicast summary`
BGP router identifier 10.10.10.3, local AS number 65000
BGP table version is 10, main routing table version 10
3 network entries using 468 bytes of memory
3 path entries using 228 bytes of memory
5/3 BGP path/bestpath attribute entries using 840 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 1560 total bytes of memory
BGP activity 16/6 prefixes, 18/8 paths, scan interval 60 secs

<table>
<thead>
<tr>
<th>Neighbor</th>
<th>V</th>
<th>AS</th>
<th>MsgRcvd</th>
<th>MsgSent</th>
<th>TblVer</th>
<th>InQ</th>
<th>OutQ</th>
<th>Up/Down</th>
<th>State/PfxRcd</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001:DB8:0:4::1</td>
<td>4</td>
<td>65000</td>
<td>1670</td>
<td>1670</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>1d02h</td>
<td>1</td>
</tr>
<tr>
<td>2001:DB8:0:6::4</td>
<td>4</td>
<td>65000</td>
<td>308</td>
<td>311</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>01:04:46</td>
<td>0</td>
</tr>
<tr>
<td>2001:DB8:0:7::5</td>
<td>4</td>
<td>65000</td>
<td>1679</td>
<td>1664</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>1d03h</td>
<td>2</td>
</tr>
</tbody>
</table>
Verifying BGP Neighbor - Cisco

```
R3#show bgp ipv6 unicast neighbors 2001:DB8:0:7::5
BGP neighbor is 2001:DB8:0:7::5, remote AS 65000, internal link
  Description: "To R5"
  BGP version 4, remote router ID 10.10.10.5
  BGP state = Established, up for 1d03h
  Last read 00:00:03, last write 00:00:49, hold time is 180, keepalive<...>
  Neighbor capabilities:
    Route refresh: advertised and received (new)
    New ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
    Address family IPv6 Unicast: advertised and received
  Message statistics:
    InQ depth is 0
    OutQ depth is 0
```
Confirm BGP Routes - Cisco

show bgp ipv6 unicast 2001:4870:FF:FF::/64
BGP routing table entry for 2001:4870:FF:FF::/64, version 10
Paths: (1 available, best #1, table Default)
   Not advertised to any peer
   Local, (received & used)
      2001:DB8:0:4::1 from 2001:DB8:0:4::1 (10.10.10.1)
      Origin IGP, metric 0, localpref 100, valid, internal, best
## BGP for IPv6 Verification Command Summary

<table>
<thead>
<tr>
<th>Juniper Command</th>
<th>Cisco IOS Command</th>
<th>Co-Ordinating Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ipv6 neighbors</td>
<td>show ipv6 neighbors</td>
<td>Show IPv6 neighbor cache information</td>
</tr>
<tr>
<td></td>
<td>show ipv6 protocols</td>
<td>Show IPv6 Routing Protocols</td>
</tr>
<tr>
<td>show interface</td>
<td>show ipv6 interface</td>
<td>Show IPv6 interface information</td>
</tr>
<tr>
<td>show route protocol bgp table inet6.0</td>
<td>show bgp all</td>
<td>Show IPv6 unicast routes [Network, Next Hop, Metric, LocPrf, Weight, AS Path, etc.]</td>
</tr>
<tr>
<td></td>
<td>show bgp ipv6 unicast</td>
<td></td>
</tr>
<tr>
<td>show bgp summary</td>
<td>show bgp ipv6 unicast summary</td>
<td>Show IPv6 unicast peers [AS, Up/Down, # of learned prefixes, etc.]</td>
</tr>
<tr>
<td>show bgp neighbor</td>
<td>show bgp ipv6 unicast neighbors</td>
<td>Show detailed information about each peer</td>
</tr>
<tr>
<td>show route advertising-protocol bgp</td>
<td>show bgp ipv6 unicast neighbors</td>
<td>Show IPv6 prefixes advertised to a peer</td>
</tr>
<tr>
<td>&lt;ipv6 addr&gt;</td>
<td>show bgp ipv6 unicast neighbors</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[&lt;ipv6 addr&gt; advertised-routes]</td>
<td></td>
</tr>
<tr>
<td>show route receive-protocol bgp &lt;ipv6</td>
<td>show bgp ipv6 unicast neighbors</td>
<td>Show IPv6 prefixes received from a peer—requires “neighbor soft-reconfiguration”</td>
</tr>
<tr>
<td>addr&gt; table inet6.0</td>
<td>[&lt;ipv6 addr&gt; received-routes]</td>
<td></td>
</tr>
<tr>
<td>show route &lt;prefix&gt;/&lt;length&gt;</td>
<td>show bgp ipv6 &lt;prefix&gt;/&lt;length&gt;</td>
<td>Show information about a specific IPv6 prefix</td>
</tr>
<tr>
<td>show policy &lt;policy-name&gt;</td>
<td>show ipv6 prefix-list [summary</td>
<td>detail]</td>
</tr>
<tr>
<td></td>
<td>show ipv6 prefix-list &lt;name&gt;</td>
<td></td>
</tr>
<tr>
<td>Use traceoptions with flags</td>
<td>debug bgp ipv6 unicast</td>
<td>Debug BGP IPv6 packets</td>
</tr>
<tr>
<td>clear bgp neighbor &lt;neighbor&gt;</td>
<td>clear bgp ipv6 unicast *</td>
<td>Clear the BGP Session to all peers</td>
</tr>
<tr>
<td>ping [ipv6]</td>
<td>ping [ipv6]</td>
<td>Send echo messages</td>
</tr>
<tr>
<td>traceroute [ipv6]</td>
<td>traceroute [ipv6]</td>
<td>Trace route to destination</td>
</tr>
</tbody>
</table>
IS-IS for IPv6

IPv4 -> IPv6 Differences
Basic differences IS-IS v4 -> v6

- TLV Type changes
  - Reachability Type: v4 (128), v6 (236)
  - Interface Address Type: v4 (132), v6 (232)
  - Type 129 same but stores protocol
    - CLNP (0x81), IPv4 (0xCC), IPv6 (0x8E)
Basic differences IS-IS v4 -> v6
Deeper into TLV additions

- IPv6 Reachability TLV 236
  - Defines both IPv6 Internal and External reachability information
  - Metric is still 32 bits
  - U: Up/Down
  - X: External origin bit
  - S: Sub-TLV present
  - Prefix length: length of prefix 8 bits
  - Prefix: Number of biglets is calculated based on length
Basic differences IS-IS v4 -> v6
Deeper into TLV additions

- IPv6 Address TLV 232
  - Modified to carry IPv6 Address
  - For hello, PDU interface address must use link-local IPv6 address assigned to the interface
  - For LSP, non-link-local address must be used
IS-IS for IPv6

Single-Topology
Single Topology Overview

- This is the default implementation
- Same topology for IPv4 and IPv6
- Same SPF (slow start v6 does not work)
- Interface metrics apply to both IPv4 and IPv6
Single Topology Cisco Example

- IPv6 only

```
Router1#
  interface ethernet-1
    ipv6 address 2001:0001::45c/64
    ipv6 router isis
    isis circuit-type level-2-only

  interface ethernet-2
    ipv6 address 2001:0002::45a/64
    ipv6 router isis

  router isis
    address-family ipv6
    redistribute static
    exit-address-family
    net 42.0001.0000.0000.072c.00
```
Single Topology Cisco Example

- IPv6 & IPv4
- Must match

```
Router1#  
interface ethernet-1  
ip address 10.1.1.1 255.255.255.0  
ipv6 address 2001:0001::45c/64  
ip router isis  
ipv6 router isis  

interface ethernet-2  
ip address 10.2.1.1 255.255.255.0  
ipv6 address 2001:0002::45a/64  
ip router isis  
ipv6 router isis  

router isis  
address-family ipv6  
redistribute static  
extit-address-family  
net 42.0001.0000.0000.072c.00  
redistribute static
```
Single Topology Juniper Example

- IPv6 & IPv4
- Match interfaces

```plaintext
protocols {
    isis {
        interface all;
    }
}

isis {
    interface interface-name {
        level level-number {
            ipv6-unicast metric number;
        };
    }
}
isis {
    interface interface-name {
        no-ipv6-unicast;
    }
}
```
IS-IS for IPv6

Multi-Topology
Multi Topology Overview (RFC5120)

- Independent IPv4 and IPv6 topologies
- Independent interface metrics
- Multi instance may be beneficial here
  - Process isolation
  - Process priority
  - Flooding priority
- New TLV Multi-Topology ID
  - 0 for IPv4/CLNS, 2 for IPv6
- Enable on routers with IPv6 additions
  - Not requires on IPv4 only routers
Multi Topology Overview (RFC5120)

- Interfaces with both IPv4 and IPv6 must run same level (1, 2, 1 & 2)
- Metric style wide is required
Multi Topology Cisco Example (RFC5120)

Router1#
  interface ethernet-1
    ip address 10.1.1.1 255.255.255.0
    ipv6 address 2001:0001::45c/64
    ip router isis
    ipv6 router isis
    isis ipv6 metric 20

  interface ethernet-2
    ip address 10.2.1.1 255.255.255.0
    ipv6 address 2001:0002::45a/64
    ip router isis
    ipv6 router isis
    isis ipv6 metric 20

  router isis
  net 49.0000.0100.0000.0000.00500
  metric-style wide
  !
  address-family ipv6
  multi-topology
  exit-address-family
To enable an alternate IPv6 unicast topology for IS-IS, include the `ipv6-unicast` statement:

```plaintext
isis {
    topologies {
        ipv6-unicast;
    }
}
```

To configure a metric for an alternate IPv6 unicast topology, include the `ipv6-unicast-metric` statement:

```plaintext
isis {
    interface interface-name {
        level level-number {
            ipv6-unicast-metric number;
        }
    }
}
```

To exclude an interface from the IPv6 unicast topologies for IS-IS, include the `no-ipv6-unicast` statement:

```plaintext
isis {
    interface interface-name {
        no-ipv6-unicast;
    }
}
```
Multi Topology Juniper Example (RFC5120)

```conf
lo0 {
    unit 0 {
        family iso {
            address 49.0001.1921.6800.0001.00;
        }
    }
}
interface ge-0/0/0.0 {
    level 1 disable;
}
interface ge-0/0/0.0 {
    level 2 disable;
}
ge-0/0/0 {
    unit 0 {
        family inet {
            address 192.168.1.1/26;
        }
        family iso;
        family inet6 {
            address 2001:db8:192:168:1::1/112;
        }
    }
}[edit protocols isis]
level 1
    wide-metrics-only;
}
level 2 {
    wide-metrics-only;
}
interface ge-0/0/0.0 {
    level 1 {
        metric 400;
        ipv6-unicast-metric 400;
    }
    level 2 {
        metric 400;
        ipv6-unicast-metric 400;
    }
}
Discussion / Questions?

Chris Grundemann
c.grundemann@cablelabs.com

Aaron Hughes
aaron@6connect.net

Ken Sexton
ken.sexton@twtelecom.com