

# APNIC Training

The logo for the South Asian Network Operators Group (SANOG), featuring the word "SANOG" in white, bold, sans-serif capital letters on a black rectangular background.

## IPv6 Deployment Tutorial

July 20 - 21, 2010 – Paro, Bhutan

**16 South Asian Network Operators Group (SANOG) Conference**

In conjunction with Bhutan Telecom Ltd.



# Introduction

- Presenters
  - Nurul Islam Roman
    - Technical Training Officer
    - [nurul@apnic.net](mailto:nurul@apnic.net)

# Acknowledgements

- APNIC acknowledge with thanks to the following organizations and individuals for their support of developing this training materials:
  - Cisco System
  - Juniper Networks
  - Philip Smith of Cisco,
  - Geoff Huston of APNIC

# Overview

## IPv6 Deployment Tutorial

- IPv6 Protocol Architecture Overview
- IPV6 Addressing and Sub-netting
- IPv6 Host Configuration
- Training ISP Network Topology Overview
- Deployment of IPV6 in Interior Gateway
- IPv4 to IPv6 Transition technologies
- Planning & Implementation of IPv6 on Exterior Gateway (BGP)
- Connecting ISP network to an IXP



# Overview

## IPv6 Deployment Tutorial

- **IPv6 Protocol Architecture Overview**
- IPV6 Addressing and Sub-netting
- IPv6 Host Configuration
- Training ISP Network Topology Overview
- Deployment of IPV6 in Interior Gateway
- IPv4 to IPv6 Transition technologies
- Planning & Implementation of IPv6 on Exterior Gateway (BGP)
- Connecting ISP network to an IXP

# What Is IPv6?

- IP stands for **I**nternet **P**rotocol which is one of the main pillars that supports the Internet today
- Current version of IP protocol is IPv4
- The new version of IP protocol is IPv6
- There is a version of IPv5 but it was assigned for experimental use [RFC1190]
- IPv6 was also called IPng in the early days of IPv6 protocol development stage

# Background Of IPv6 Protocol

- During the late 1980s (88-89) Internet has started to grow exponentially
- The ability to scale Internet for future demands requires a limitless supply of IP addresses and improved mobility
- In 1991 IETF decided that the current version of IP (IPv4) had outlived its design and need to develop a new protocol for Internet
- In 1994 IETF gave a clear direction of IPng or IPv6 after a long process of discussion

# Background Of IPv6 Protocol

- August 1990
  - First wakeup call by Solensky in IETF on IPv4 address exhaustion
- December 1994
  - IPng area were formed within IETF to manage IPng effort [RFC1719]
- December 1994
  - List of technical criteria was defined to choose IPng [RFC1726]
- January 1995
  - IPng director recommendation to use 128 bit address [RFC1752]
- December 1995
  - First version of IPv6 address specification [RFC1883]
- December 1998
  - Updated version changing header format from 1<sup>st</sup> version [RFC2460]

# Motivation Behind IPv6 Protocol

- New generation Internet need:
  - Plenty of address space (PDA, Mobile Phones, Tablet PC, Car, TV etc etc ☺ )
  - Solution of very complex hierarchical addressing need, which IPv4 is unable provide
  - End to end communication without the need of NAT for some real time application i.e online transaction
  - Ensure security, reliability of data and faster processing of protocol overhead
  - Stable service for mobile network i.e Internet in airline

# New Functional Improvement In IPv6

- Address Space
  - Increase from 32-bit to 128-bit address space
- Management
  - Stateless autoconfiguration means no more need to configure IP addresses for end systems, even via DHCP
- Performance
  - Fixed header sizes (40 byte) and 64-bit header alignment mean better performance from routers and bridges/switches
- No hop-by-hop segmentation
  - Path MTU discovery

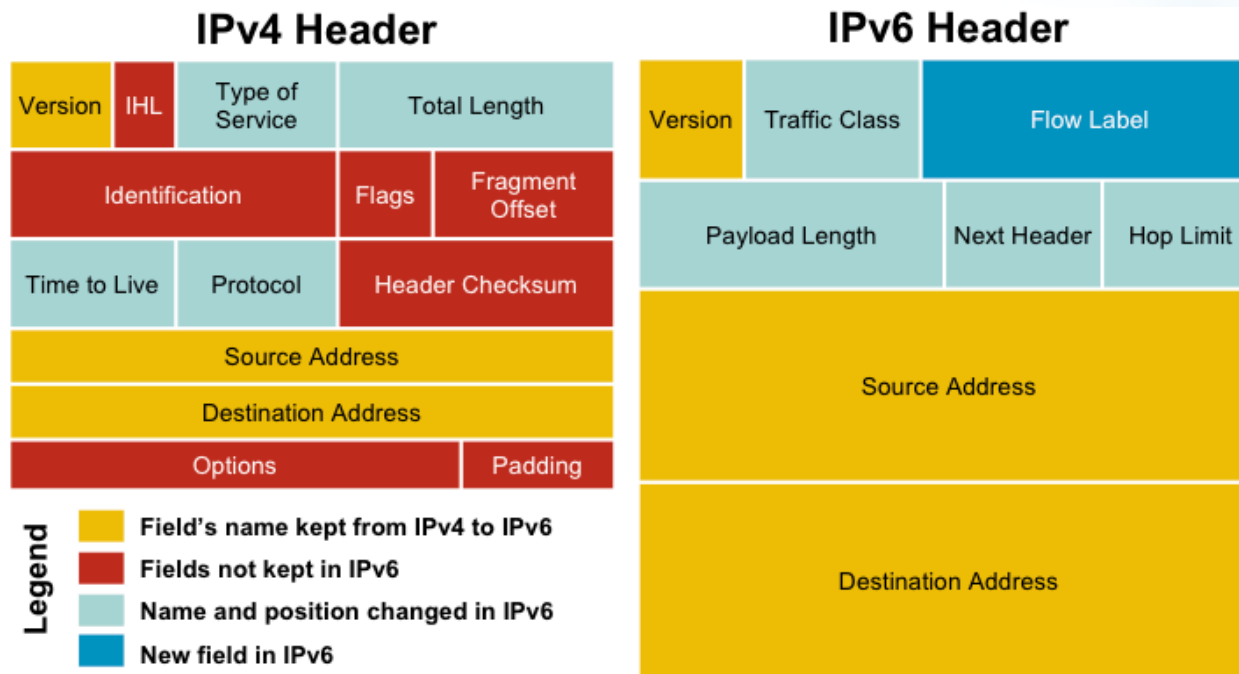




# New Functional Improvement In IPv6

- Multicast/Multimedia
  - Built-in features for multicast groups, management, and new "anycast" groups
- Mobile IP
  - Eliminate triangular routing and simplify deployment of mobile IP-based systems
- Virtual Private Networks
  - Built-in support for ESP/AH encrypted/authenticated virtual private network protocols; built-in support for QoS tagging
- No more broadcast

# Protocol Header Comparison



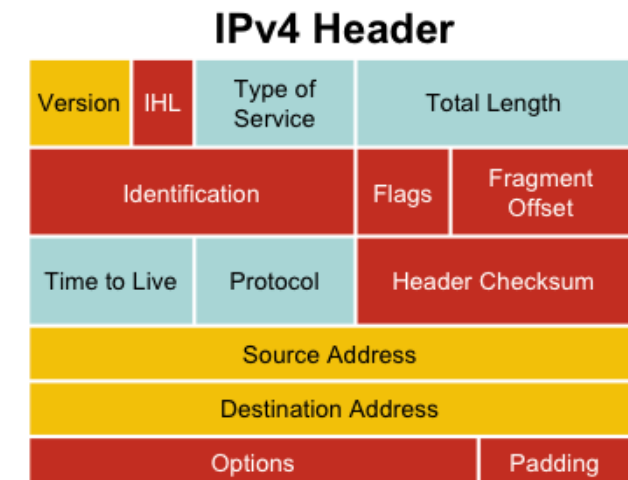
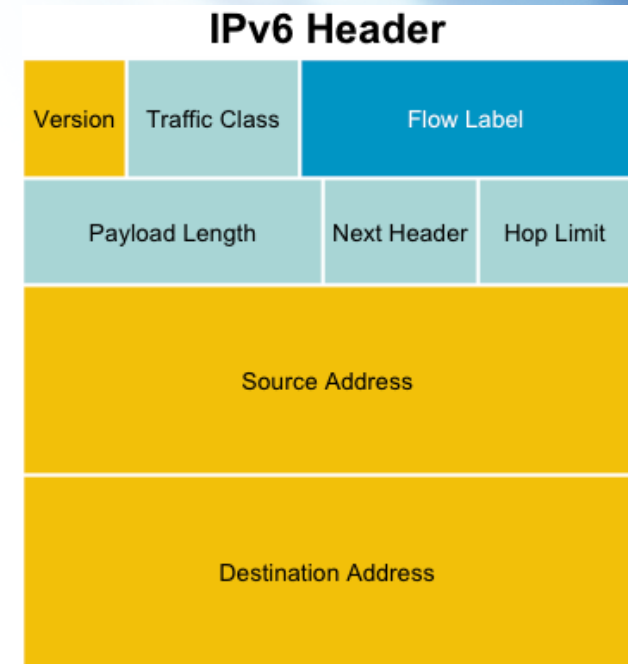
- IPv4 contain 10 basic header field
- IPv6 contain 6 basic header field
- IPv6 header has 40 octets in contrast to the 20 octets in IPv4
- So a smaller number of header fields and the header is 64-bit aligned to enable fast processing by current processors



# IPv6 Protocol Header Format

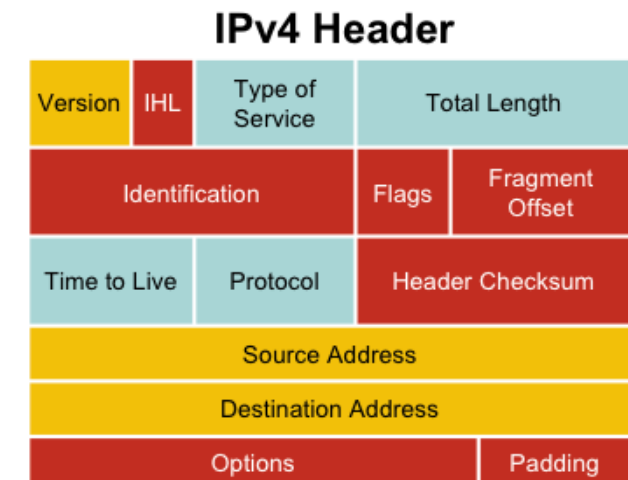
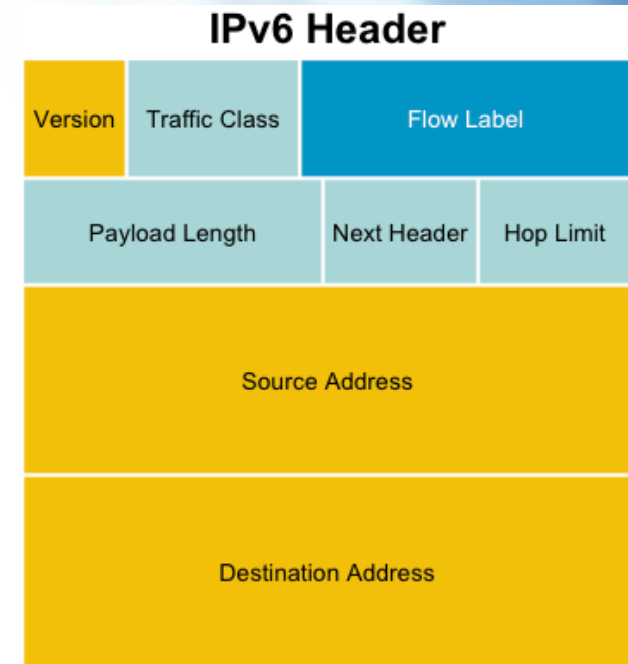
The IPv6 header fields:

- **Version:**
  - A 4-bit field, same as in IPv4. It contains the number 6 instead of the number 4 for IPv4
- **Traffic class:**
  - A 8-bit field similar to the type of service (ToS) field in IPv4. It tags packet with a traffic class that it uses in differentiated services (DiffServ). These functionalities are the same for IPv6 and IPv4.
- **Flow label:**
  - A completely new 20-bit field. It tags a flow for the IP packets. It can be used for multilayer switching techniques and faster packet-switching performance



# IPv6 Protocol Header Format

- **Payload length:**
  - This 16-bit field is similar to the IPv4 Total Length Field, except that with IPv6 the Payload Length field is the length of the data carried after the header, whereas with IPv4 the Total Length Field included the header.  $2^{16} = 65536$  Octets.
- **Next header:**
  - The 8-bit value of this field determines the type of information that follows the basic IPv6 header. It can be a transport-layer packet, such as TCP or UDP, or it can be an extension header. The next header field is similar to the protocol field of IPv4.
- **Hop limit:**
  - This 8-bit field defines by a number which count the maximum hops that a packet can remain in the network before it is destroyed. With the IPv4 TLV field this was expressed in seconds and was typically a theoretical value and not very easy to estimate.



# IPv6 Extension Header

- Adding an optional Extension Header in IPv6 makes it simple to add new features in IP protocol in future without a major re-engineering of IP routers everywhere
- The number of extension headers are not fixed, so the total length of the extension header chain is variable
- The extension header will be placed in-between main header and payload in IPv6 packet

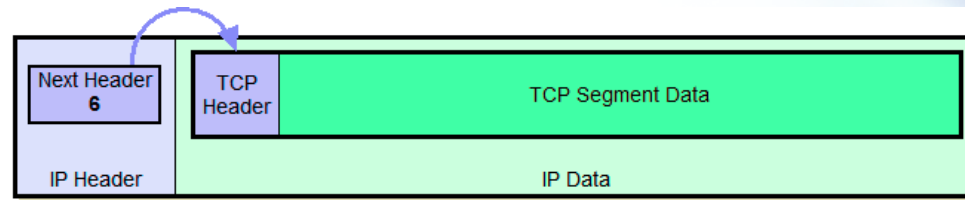
# IPv6 Extension Header

- If the Next Header field value (code) is 6 it determine that there is no extension header and the next header field is pointing to TCP header which is the payload of this IPv6 packet
- Code values of Next Header field:
  - 0 Hop-by-hope option
  - 2 ICMP
  - 6 TCP
  - 17 UDP
  - 43 Source routing
  - 44 Fragmentation
  - 50 Encrypted security payload
  - 51 Authentication
  - 59 Null (No next header)
  - 60 Destination option

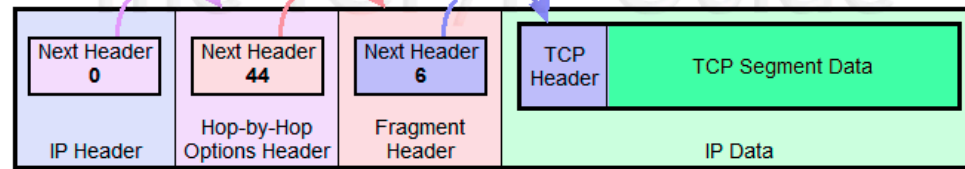
# Extension Header Type

- Six type of extension header defined:
  1. Hop-by-hop option i.e Router Alert, Jumbogram
  2. Routing Header ie. Source Routing
  3. Fragment header
  4. Authentication header
  5. Encrypted security payload
  6. Destination option header

# Link listed Extension Header



IPv6 Datagram With No Extension Headers Carrying TCP Segment



IPv6 Datagram With Two Extension Headers Carrying TCP Segment

- Link listed extension header can be used by simply using next header code value
- Above example use multiple extension header creating link list by using next header code value i.e 0 44 6
- The link list will end when the next header point to transport header i.e next header code 6



# Order Of Extension Header

- Source node follow the order:
  - 1. Hop-by-hop
  - 2. Routing
  - 3. Fragment
  - 4. Authentication
  - 5. Encapsulating security payload
  - 6. Destination option
  - 7. Upper-layer
- Order is important because:
  - Only hop-by-hop has to be processed by every intermediate nodes
  - Routing header need to be processed by intermediate routers
  - At the destination fragmentation has to be processed before others
  - This is how it is easy to implement using hardware and make faster processing engine

# Fragmentation Handling In IPv6

- Routers handle fragmentation in IPv4 which cause variety of processing performance issues
- IPv6 routers no longer perform fragmentation. IPv6 host use a discovery process [Path MTU Discovery] to determine most optimum MTU size before creating end to end session
- In this discovery process, the source IPv6 device attempts to send a packet at the size specified by the upper IP layers [i.e TCP/Application].
- If the device receives an “ICMP packet too big” message, it informs the upper layer to discard the packet and to use the new MTU.
- The “ICMP packet too big” message contains the proper MTU size for the pathway.
- Each source device needs to track the MTU size for each session.



# MTU Size Guideline

- MTU for IPv4 and IPv6
  - MTU is the largest size datagram that a given link layer technology can support [i.e HDLC]
  - Minimum MTU 68 Octet [IPv4] 1280 Octet [IPv6]
  - Most efficient MTU 576 [IPv4] 1500 [IPv6]
- Important things to remember:
  - Minimum MTU for IPv6 is 1280
  - Most efficient MTU is 1500
  - Maximum datagram size 64k

# Size of The IPv6 Datagram

- The maximum size of IPv6 datagram will be determined by two factor:
  - Maximum Transmission Unit (MTU) of intermediate nodes [L2 link technology can support i.e HDLC]
  - Payload length of IPv6 header which is 16 bit so normal payload can not be larger then 64k octets.
  - Jumbogram can increase IPv6 datagram size larger then 64k octets. But they need special processing on each hop since they are oversize.
  - One of two uses of hop-by-hop option header is Jumbogram

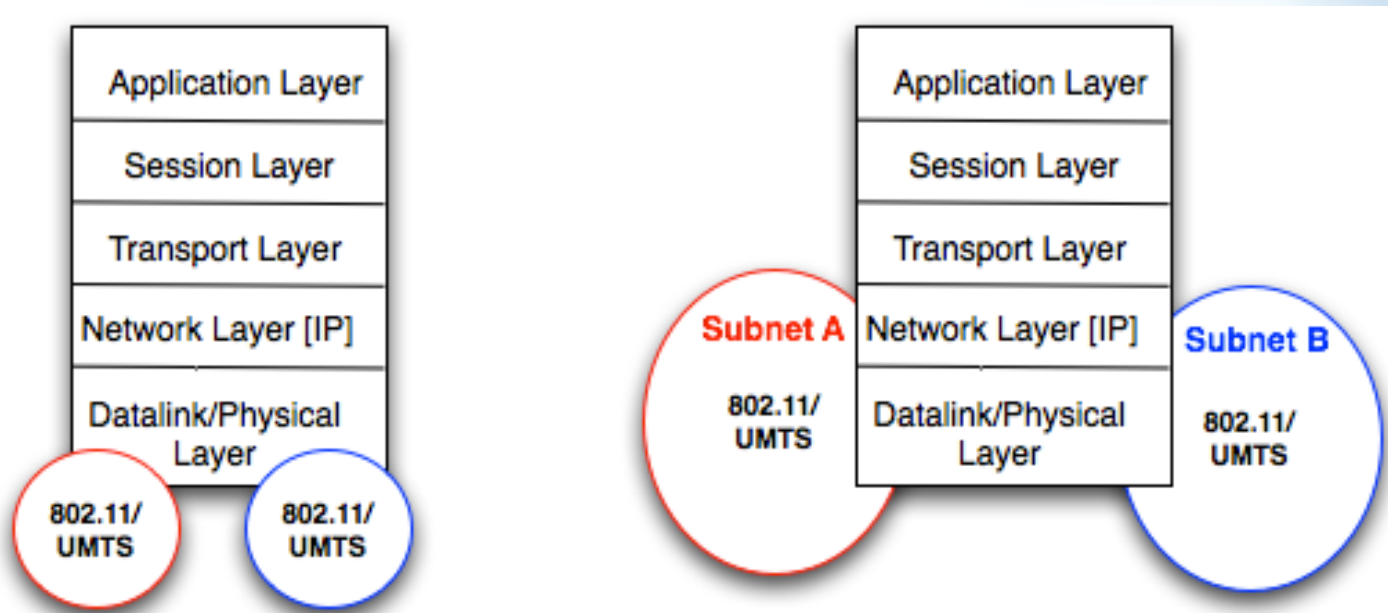
# IPv6 Header Compression

- IPv6 header size is double then IPv4
- Some time it becomes an issue on limited bandwidth link i.e Radio
- Robust Header Compression [RoHC] standard can be used to minimize IPv6 overhead transmission in limited bandwidth link
- RoHC is IETF standard for IPv6 header compression

# IP Address Mobility

- IP address mobility is a mechanism that will sustain the IP connection even when the IP address change if the device move from one location to other location (subnet)
- IP address mobility is achieved by using Mobile IP
- Mobile IP is designed to work with both IPv4 [RFC3344] and IPv6 [RFC3775]
- Mobile IP operation is optimized for IPv6

# Layer2 Vs Layer3 Mobility



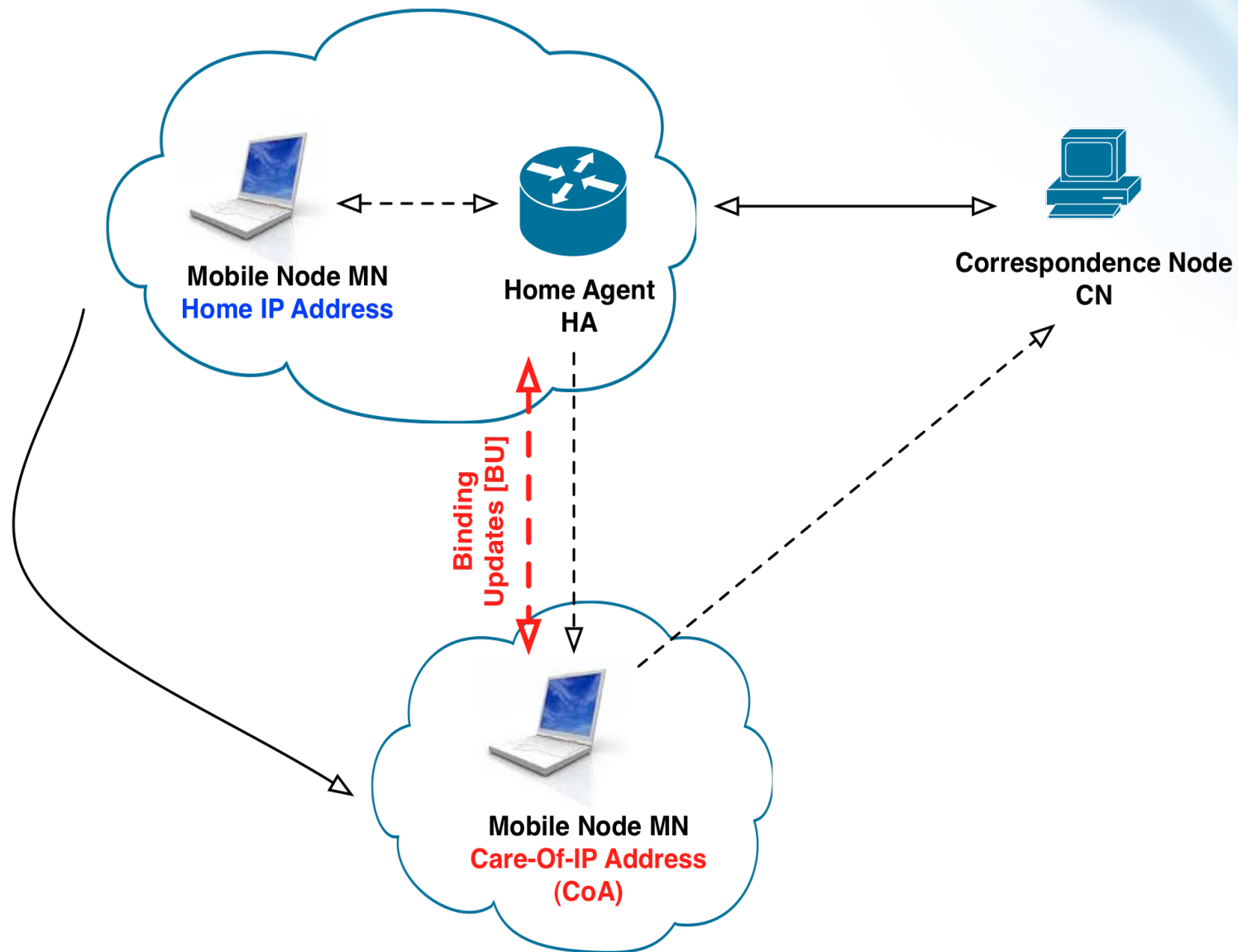
- If device move from one location to other location but belongs to the same IP subnet this will not consider as IP mobility and do not need mobile IP

# IP Address Mobility Terminology

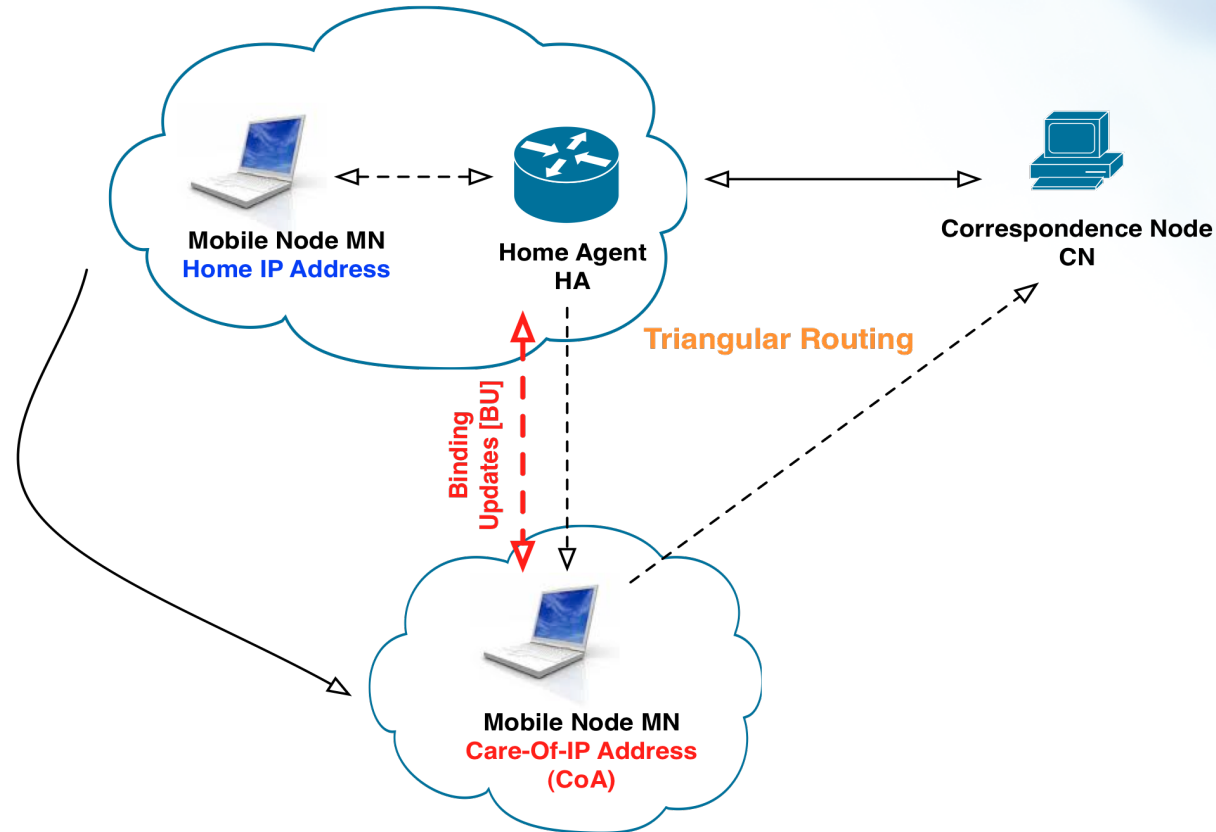
- Mobile Node [MN]
  - Is the mobile user
- Correspondent Node [CN]
  - Fixed [or may be mobile] user
- Home Agent [HA]
  - Usually a router in home representing MN
- Home IP Address
  - Primary (fixed) IP address of MN
- Care-Of-Address [CoA]
  - Secondary (variable) IP address of MN
- Binding Update [BA]
  - Process to register new IP address to HA [some time CN]



# Basic Mobile IP Operation



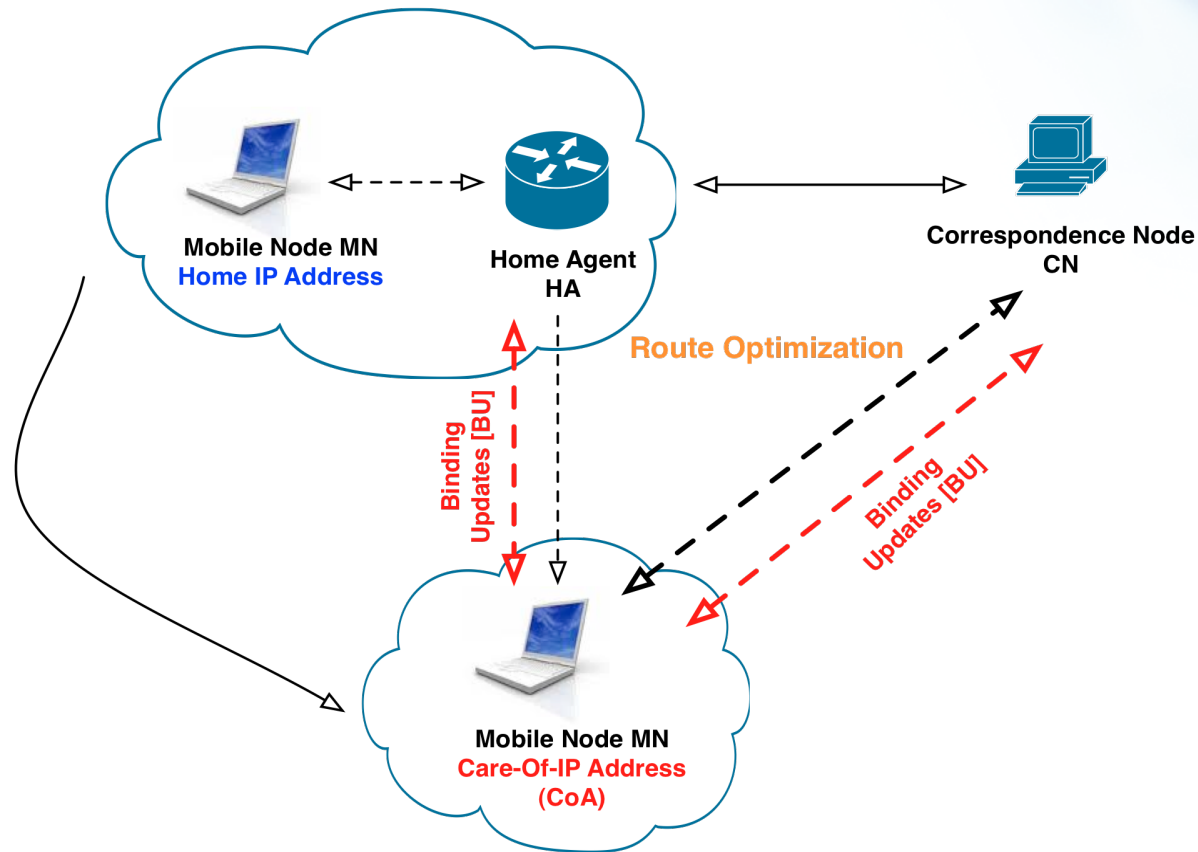
# Triangular Routing Issue



- Triangular routing creates delay which affects real time application i.e VoIP, streaming etc



# Route Optimization



- MN send BU to CN informing its CoA
- Direct data communication will start between MN and CN

# Mobile IP Operation in IPv6

- Applications are transparent to IP mobility
  - Application software on both source and destination are not aware about mobile IP
- Mobile node [MN] is in home [HA]
  - HA are mobile IP enabled router sending Router Advertisement RA to MN advertising HA prefix
  - MN received RA compare with existing HA prefix. If match MN is at home
  - MN behaves as normal IP node

# Mobile IP Operation in IPv6

- Mobile node [MN] is away from home [HA]
  - New network will receive new RA/DHCPv6 and will be different then existing prefix
  - Mobile node [MN] is away from home and will register it new CoA with HA by using BU
  - If there is an existing/new data session MN will also send BU to CN to eliminate/optimize triangular routing
  - To ensure authentic and secure BU IPsec ESP is used between MN and HA

# Issue with Mobile IP Operation in IPv6

- In optimized routing environment source & destination address of IP packet between MN and CN will be CoA and CN IPv6 address
- This will create inconsistency in application software and firewall policy in both side [MN and CN]
- IPv6 use extension header [Destination option header] to carry home IP address within the same IPv6 packet which will be replaced at CN
- Router will still look at the header for faster processing

# Questions?



# Overview

## IPv6 Deployment Tutorial

- IPv6 Protocol Architecture Overview
- **IPV6 Addressing and Sub-netting**
- IPv6 Host Configuration
- Training ISP Network Topology Overview
- Deployment of IPV6 in Interior Gateway
- IPv4 to IPv6 Transition technologies
- Planning & Implementation of IPv6 on Exterior Gateway (BGP)
- Connecting ISP network to an IXP

## Size of the IPv6 address space

- An IPv6 address is 16 octets (128 bits)
- This would allow every person on the planet to have their own internet as large as the current Internet
- It is difficult to foresee running out of IPv6 addresses

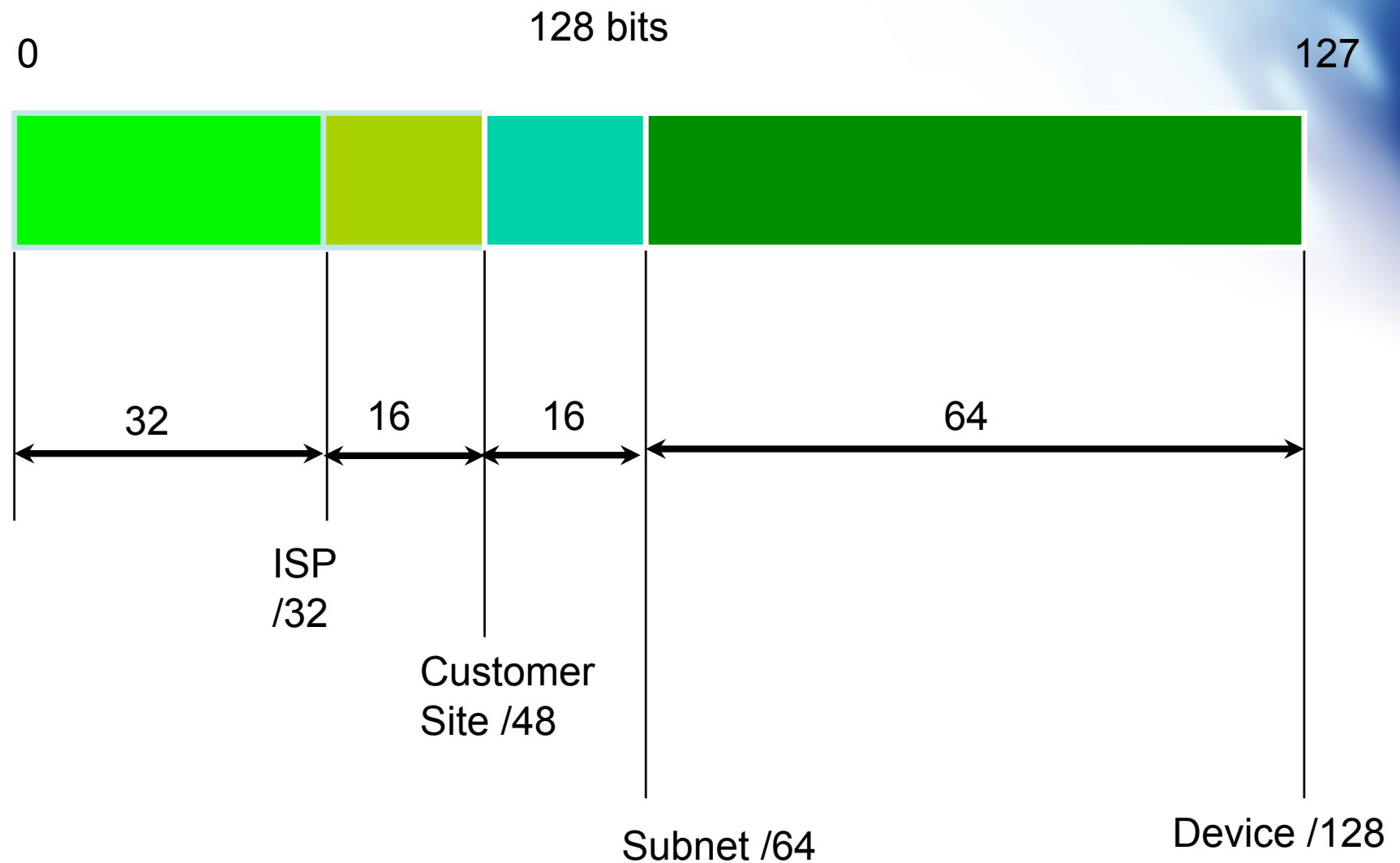


# IPv6 addressing

- 128 bits of address space
- Hexadecimal values of eight 16 bit fields
  - X:X:X:X:X:X:X:X (X=16 bit number, ex: A2FE)
  - 16 bit number is converted to a 4 digit hexadecimal number
- Example:
  - FE38:DCE3:124C:C1A2:BA03:6735:EF1C:683D
  - Abbreviated form of address
    - 4EED:0023:0000:0000:0000:036E:1250:2B00
    - →4EED:23:0:0:0:36E:1250:2B00
    - →4EED:23::36E:1250:2B00
    - (Null value can be used only once)



# IPv6 addressing structure



# IPv6 address prefix

- When you do IPv6 sub-netting, you need to think in binary values not in hexadecimal value
- 2001:1::/32  
=2001:0001::/32  
Hex 2001 = Binary 0010 0000 0000 0001  
Hex 0001 = Binary 0000 0000 0000 0001
- 2001:2:3::/48  
=2001:0002:0003::/48  
Hex 2001 = Binary 0010 0000 0000 0001  
Hex 0002 = Binary 0000 0000 0000 0010  
Hex 0003 = Binary 0000 0000 0000 0011

# IPv6 address prefix

- /64s in 2001:2:3::/48 are
  - 2001:0002:0003:0001::/64
  - 2001:0002:0003:0002::/64
  - 2001:0002:0003:0003::/64
  - Etc.
  - 16 bits of address space
    - You can have 65536 /64s in one /48 IPv6 address
    - Note:: indicates the remaining 64 bits are all zeros and can then be used to identify hosts::

# IPv6 address prefix

- Another example:
- 2001:1::/32  
=2001:0001::/32  
Hex 2001 = Binary 0010 0000 0000 0001  
Hex 0001 = Binary 0000 0000 0000 0001
- How about /47s in 2001:1::/32?

# IPv6 address prefix

- How about /47s in 2001:1::/32?

Hex 2001 = Binary 0010 0000 0000 0001 = 16 bits

Hex 0001 = Binary 0000 0000 0000 0001 = 32

Hex 0000 = Binary 0000 0000 0000 0000 = 47 (32 bits in prefix  
– “fixed”, 15 bits in subnet)

So the 15 subnet bits (red), are used to identify the /47s: Subnets  
numbered using these bits

Binary 0000 0000 0000 0000 = Hex 0000

The first /47 is 2001:0001:0000::/47

-----  
Binary 0000 0000 0000 0010 = Hex 0002

So the second /47 is 2001:0001:0002::/47

-----  
Binary 0000 0000 0000 0100 = Hex 0004

So the third /47 is 2001:0001:0004::/47

-----  
Binary 0000 0000 0000 0110 = Hex 0006

So the fourth /47 is 2001:0001:0006::/47

-----  
Binary 0000 0000 0000 1000 = Hex 0008

So the fifth /47 is 2001:0001:0008 ::/47

# Exercise 1

## IPv6 Sub-netting

## Exercise 1.1: IPv6 subnetting

1. Identify the first four /64 address blocks out of 2001:AA:2000::/48

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_
4. \_\_\_\_\_



## Exercise 1.2: IPv6 subnetting

1. Identify the first four /36 address blocks out of 2001:ABC::/32

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_
4. \_\_\_\_\_

## Exercise 1.3: IPv6 subnetting

3. Identify the first six /37 address blocks out of 2001:AA::/32

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_
4. \_\_\_\_\_
5. \_\_\_\_\_
6. \_\_\_\_\_

# IPv6 addressing model

- **IPv6 Address type**



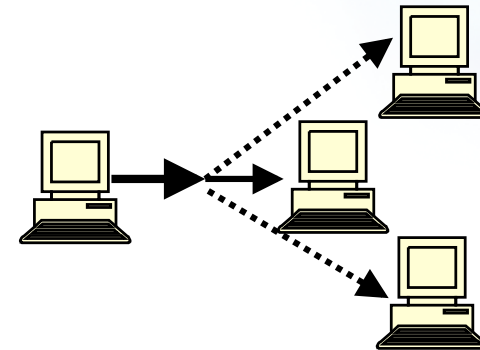
- Unicast

- An identifier for a single interface



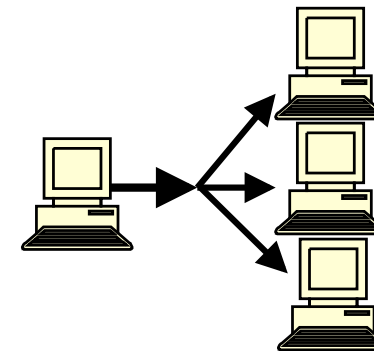
- Anycast

- An identifier for a set of interfaces



- Multicast

- An identifier for a group of nodes

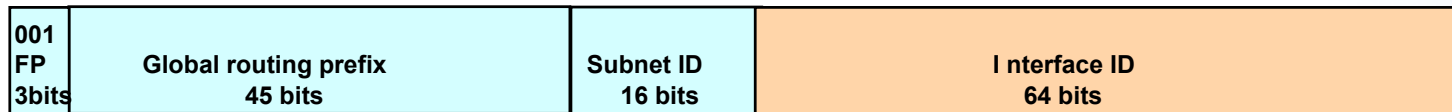


# IPv6 Address Range

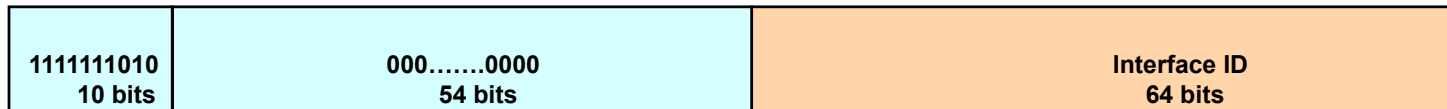
- Unspecified Address `::/128`
- Loopback `::1/128`
- Global Unicast 0010 `2000::/3`
- Link Local 1111 1110 10 `FE80::/10`
- Multicast Address 1111 1111 `FF00::/8`

# Unicast address

- Address given to interface for communication between host and router
  - Global unicast address currently delegated by IANA



- Local use unicast address
  - Link-local address (starting with FE80::)



# Special addresses

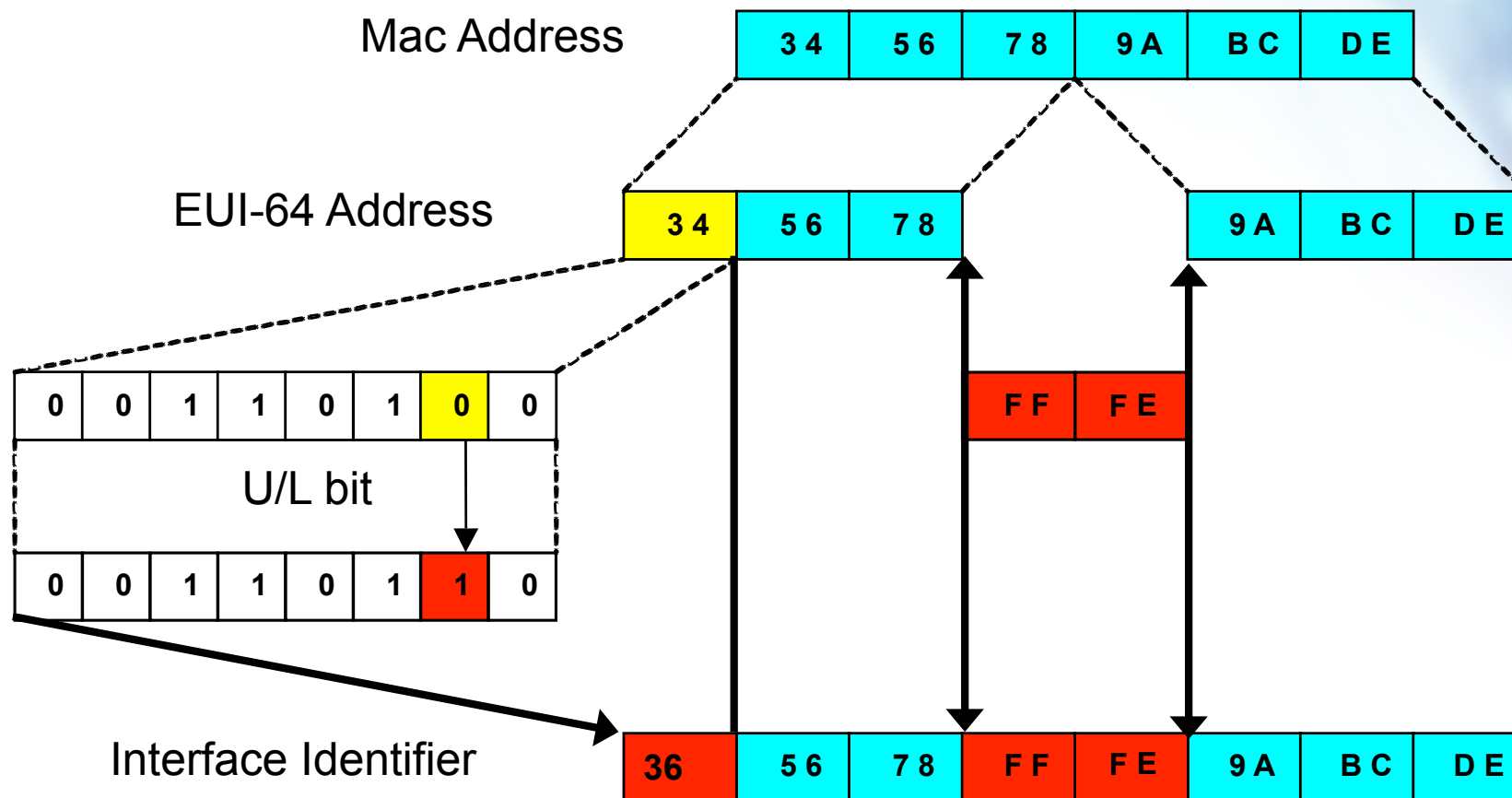
- The unspecified address
  - A value of 0:0:0:0:0:0:0:0 (::)
  - It is comparable to 0.0.0.0 in IPv4
- The loopback address
  - It is represented as 0:0:0:0:0:0:0:1 (:::1)
  - Similar to 127.0.0.1 in IPv4

## Interface ID

- The lowest-order 64-bit field addresses may be assigned in several different ways:
  - auto-configured from a 48-bit MAC address expanded into a 64-bit EUI-64
  - assigned via DHCP
  - manually configured
  - auto-generated pseudo-random number
  - possibly other methods in the future



# EUI-64



# Zone IDs for local-use addresses

- In Windows XP for example:
- Host A:
  - fe80::2abc:d0ff:fee9:4121%4
- Host B:
  - fe80::3123:e0ff:fe12:3001%3
- Ping from Host A to Host B
  - ping fe80::3123:e0ff:fe12:3001%4 (not %3)
    - identifies the interface zone ID on the host which is connected to that segment.

# IPv6 autoconfiguration

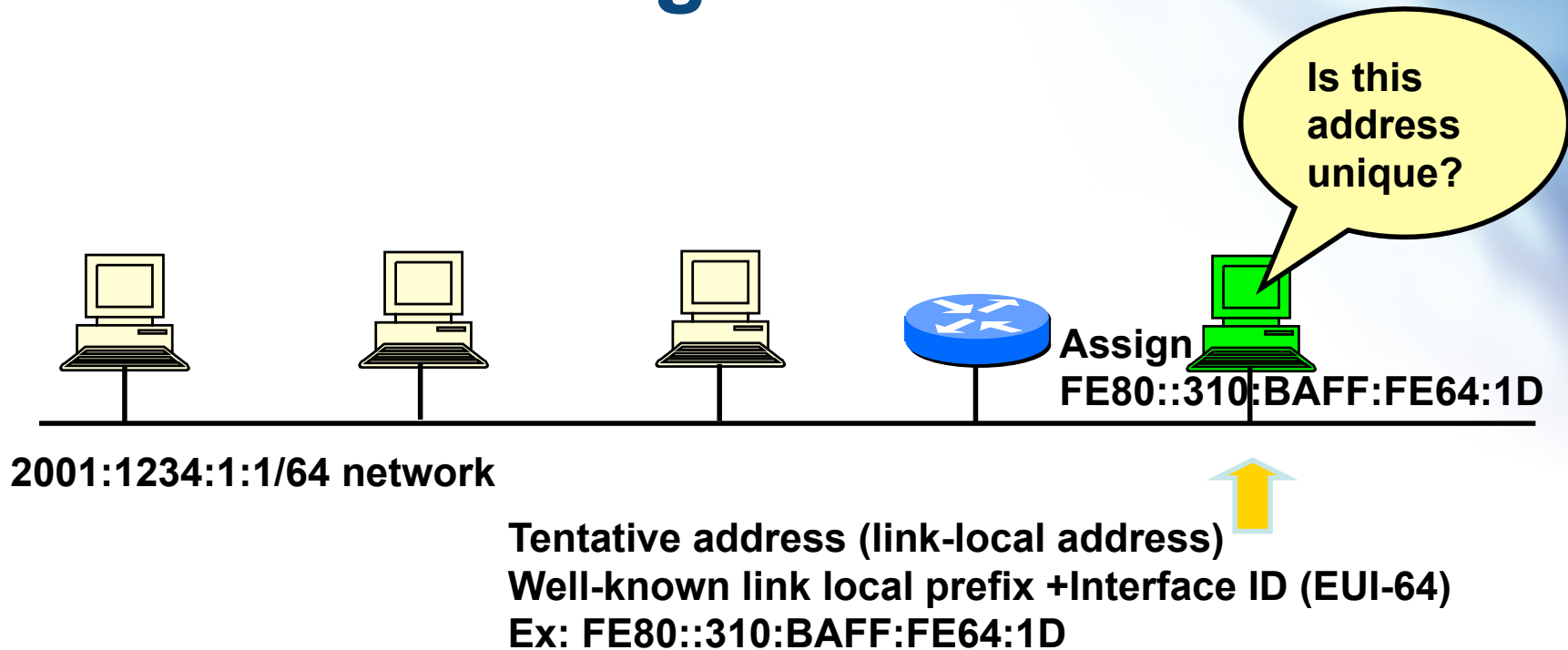


- Stateless mechanism
  - For a site not concerned with the exact addresses
  - No manual configuration required
  - Minimal configuration of routers
  - No additional servers
- Stateful mechanism
  - For a site that requires tighter control over exact address assignments
  - Needs a DHCP server
    - DHCPv6

# Plug and Play

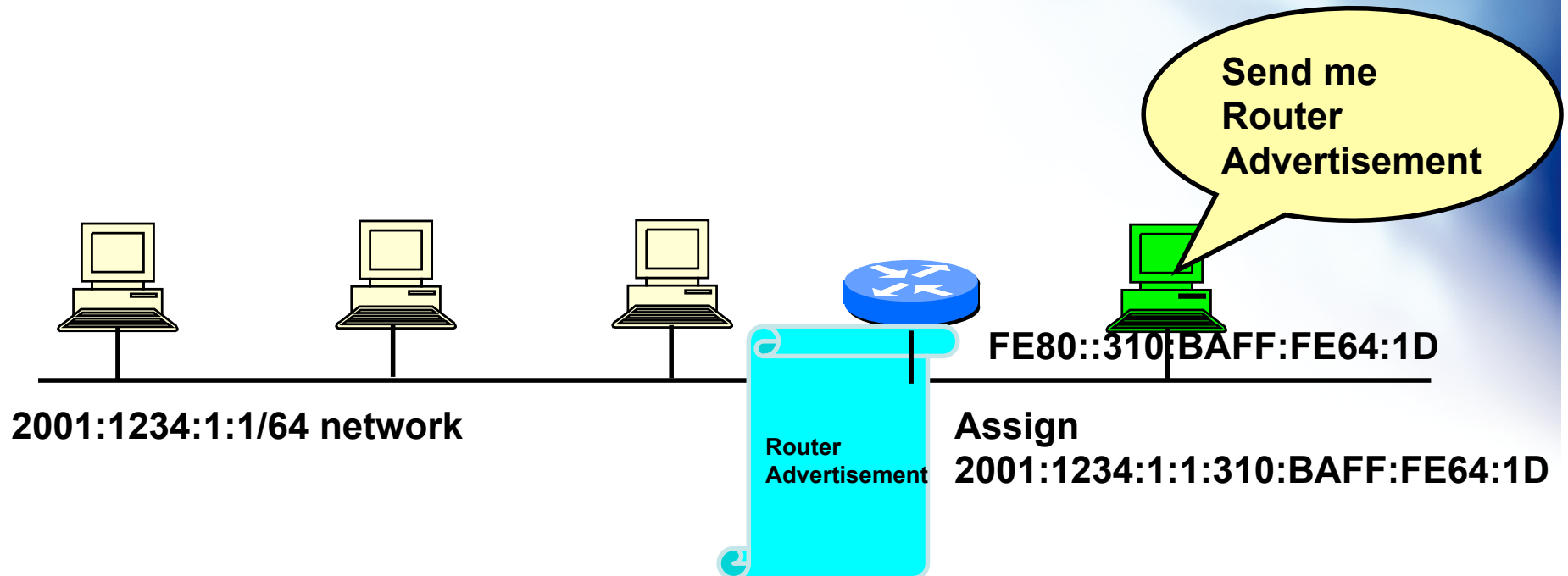
- IPv6 link local address
  - Even if no servers/routers exist to assign an IP address to a device, the device can still auto-generate an IP address
    - Allows interfaces on the same link to communicate with each other
- Stateless
  - No control over information belongs to the interface with an assigned IP address
    - Possible security issues
- Stateful
  - Remember information about interfaces that are assigned IP addresses

# IPv6 autoconfiguration



1. A new host is turned on.
2. Tentative address will be assigned to the new host.
3. Duplicate Address Detection (DAD) is performed. First the host transmit a Neighbor Solicitation (NS) message to the solicited node multicast address (FF02::1:FF64:1D) corresponding to its to be used address
5. If no Neighbor Advertisement (NA) message comes back then the address is unique.
6. FE80::310:BAFF:FE64:1D will be assigned to the new host.

# IPv6 autoconfiguration



1. The new host will send Router Solicitation (RS) request to the all-routers multicast group (FF02::2).
2. The router will reply Routing Advertisement (RA).
3. The new host will learn the network prefix. E.g, 2001:1234:1:1/64
4. The new host will assigned a new address Network prefix+Interface ID  
E.g, 2001:1234:1:1:310:BAFF:FE64:1D

# Questions?





# Overview

## IPv6 Deployment Tutorial

- IPv6 Protocol Architecture Overview
- IPV6 Addressing and Sub-netting
- **IPv6 Host Configuration**
- Training ISP Network Topology Overview
- Deployment of IPV6 in Interior Gateway
- IPv4 to IPv6 Transition technologies
- Planning & Implementation of IPv6 on Exterior Gateway (BGP)
- Connecting ISP network to an IXP

# Configuration of IPv6 Node Address

- There are 3 ways to configure IPv6 address on an IPv6 node:
  - Static address configuration
  - DHCPv6 assigned node address
  - Auto-configuration [New feature in IPv6]

# Configuration of IPv6 Node Address

Quantity	Address	Requirement	Context
One	Link-local	Must define	Each Interface
One	Loopback	Must define	Each node
Zero to many	Unicast	Optional define	Each interface
Any	Unique-local	Optional define	Each interface
One	All-nodes multicast ff01::1	Must join	Each interface
One	Solicited-node multicast ff02:0:0:0:0:1:ff/104	Must join	Each unicast and anycast define
Any	Multicast Group	Optional join	Each interface

ULA are unicast address globally unique but used locally within sites. Any sites can have /48 for private use. Each /48 is globally unique so no Collision of identical address in future when they connect together. But this Idea is deprecated now

# Exercise 1: IPv6 Host Configuration

- Windows XP SP2
- *netsh interface ipv6 install*
- Windows XP
- *ipv6 install*

# Exercise 1: IPv6 Host Configuration

- Configuring an interface
  - *netsh interface ipv6 add address "Local Area Connection" 2406:6400::1*
- Prefix length is not specified with address which will force a /64 on the interface

# Exercise 1: IPv6 Host Configuration

Verify your Configuration

- `c:\>ipconfig`

# Exercise 1: IPv6 Host Configuration

Testing your configuration

- *`ping fe80::260:97ff:fe02:6ea5%4`*

Note: the Zone id is YOUR interface index



# Exercise 1: IPv6 Host Configuration

- Enabling IPv6 on Linux
  - Set the NETWORKING\_IPV6 variable to yes in /etc/sysconfig/network

```
# vi /etc/sysconfig/network  
NETWORKING_IPV6=yes  
# service network restart
```
- Adding IPv6 address on an interface

```
# ifconfig eth0 add inet6 2406:6400::1/64
```

# Exercise 1: IPv6 Host Configuration

- Configuring RA on Linux
  - Set IPv6 address forwarding on  
`# echo "1" /proc/sys/net/ipv6/conf/all/forward`
  - Need radvd-0.7.1-3.i386.rpm installed
  - On the demon conf file /etc/radvd.conf  
`# vi /etc/radvd.conf`  
Interface eth1 {  
  advSendAdvert on;  
  prefix 2406:6400::/64 {  
    AdvOnLink on;  
  };  
};

# Exercise 1: IPv6 Host Configuration

- Enabling IPv6 on FreeBSD
  - Set the `ipv6_enable` variable to `yes` in the `/etc/rc.conf`  
`# vi /etc/rc.conf`  
`ipv6_enable=yes`
- Adding IPv6 address on an interface  
`# ifconfig fxp0 inet6 2406:6400::1/64`

# Questions?



# Overview

## IPv6 Deployment Tutorial

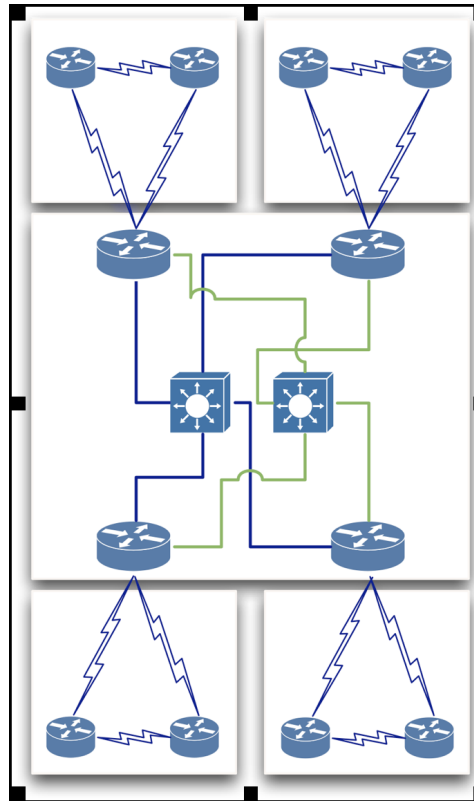
- IPv6 Protocol Architecture Overview
- IPV6 Addressing and Sub-netting
- IPv6 Host Configuration
- **Training ISP Network Topology Overview**
- Deployment of IPV6 in Interior Gateway
- IPv4 to IPv6 Transition technologies
- Planning & Implementation of IPv6 on Exterior Gateway (BGP)
- Connecting ISP network to an IXP

# Training ISP Network Topology

## Scenario:

- Training ISP has 4 main operating area or region
- Each region has 2 small POP
- Each region will have one datacenter to host content
- Regional network are inter-connected with multiple link

# Training ISP Network Topology



Training ISP Topology Diagram



# Training ISP Network Topology

## Regional Network:

- Each regional network will have 3 routers
- 1 Core & 2 Edge Routers
- 2 Point of Presence (POP) for every region
- POP will use a router to terminate customer network i.e Edge Router
- Each POP is an aggregation point of ISP customer

# Training ISP Network Topology

## Access Network:

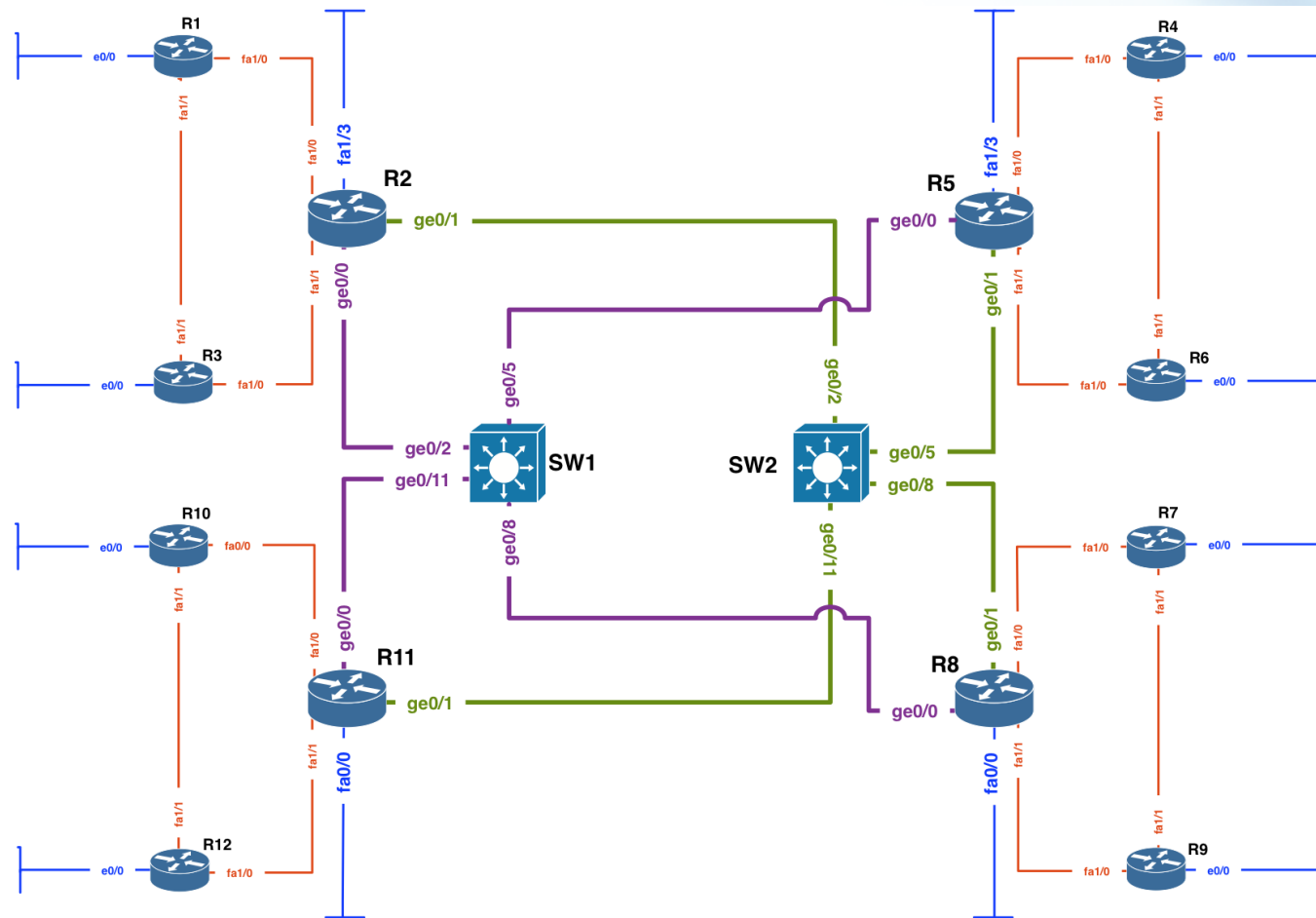
- Connection between customer network & Edge router
- Usually 10 to 100 MBPS link
- Separate routing policy from most of ISP
- Training ISP will connect them on edge router with separate customer IP prefix

# Training ISP Network Topology

## Transport Link:

- Inter-connection between regional core router
- Higher data transmission capacity than access link
- Training ISP has 2 transport link for link redundancy
- 2 Transport link i.e Purple link & Green link are connected to two L3 switch

# Training ISP Topology Diagram



# Training ISP Network Topology

## Design Consideration:

- Each regional network should have address summarization capability for customer block.
- Prefix planning should have scalability option for next couple of years for both customer block and infrastructure
- No Summarization require for WAN and loopback address

# Training ISP Network Topology

## Design Consideration:

- Conservation will get high preference for IPv4 address planning and aggregation will get high preference for IPv6 address planning.

# Training ISP Network Topology

## Design Consideration:

- OSPF is running in ISP network to carry infrastructure IP prefix
- Each region is a separate OSPF area
- Transport core is in OSPF area 0
- Customer will connect on either static or eBGP (Not OSPF)
- iBGP will carry external prefix within ISP network



# Training ISP Network Topology

## Design Consideration:

- Training ISP is already in production with IPv4 protocol
- Need to implement IPv6 within the same infrastructure
- Down time need to minimize as less as possible
- There has to be a smooth migration plan from IPv4 to IPv6

# Training ISP IPV4 Addressing Plan

## Current IPv4 Addressing Plan:

Summary parent block IPV4

Block#	Prefix	Size	Description
1	172.16.0.0	/19	Parent block
2	172.16.0.0	/20	Infrastructure
3	172.16.16.0	/20	Customer network

# Training ISP IPV4 Addressing Plan

## Current IPv4 Addressing Plan:

### Detail DC infrastructure block IPV4

Block#	Prefix	Size	Description	SOR	Register
2	172.16.0.0	/20	Infrastructure		
4	172.16.0.0	/23	Router2 DC summary net		
5	172.16.0.0	/24	Router2 DC	No	Recommended
6	172.16.2.0	/23	Router5 DC summary net		
7	172.16.2.0	/24	Router5 DC	No	Recommended
8	172.16.4.0	/23	Router8 DC summary net		
9	172.16.4.0	/24	Router8 DC	No	Recommended
10	172.16.6.0	/23	Router11 DC summary net		
11	172.16.6.0	/24	Router11 DC	No	Recommended

# Training ISP IPV4 Addressing Plan

## Current IPv4 Addressing Plan:

### Detail infrastructure WAN block IPV4

12	172.16.10.0	/24	WAN prefix		Optional
13	172.16.10.0	/30	Router2-1 WAN	No	
14	172.16.10.4	/30	Router2-3 WAN	No	
15	172.16.10.8	/30	Router1-3 WAN	No	
16	172.16.10.24	/30	Router5-4 WAN	No	
17	172.16.10.28	/30	Router5-6 WAN	No	
18	172.16.10.32	/30	Router4-6 WAN	No	
19	172.16.10.48	/30	Router8-7 WAN	No	
20	172.16.10.52	/30	Router8-9 WAN	No	
21	172.16.10.56	/30	Router7-9 WAN	No	
22	172.16.10.72	/30	Router11-10 WAN	No	
23	172.16.10.76	/30	Router11-12 WAN	No	
24	172.16.10.80	/30	Router10-12 WAN	No	

# Training ISP IPV4 Addressing Plan

## Current IPv4 Addressing Plan:

### Detail infrastructure block Transport & Loopback IPV4

25	172.16.12.0	/24	Transit link BLUE	No	
26	172.16.13.0	/24	Transit link GREEN	No	
27	172.16.15.0	/24	Loopback	No	

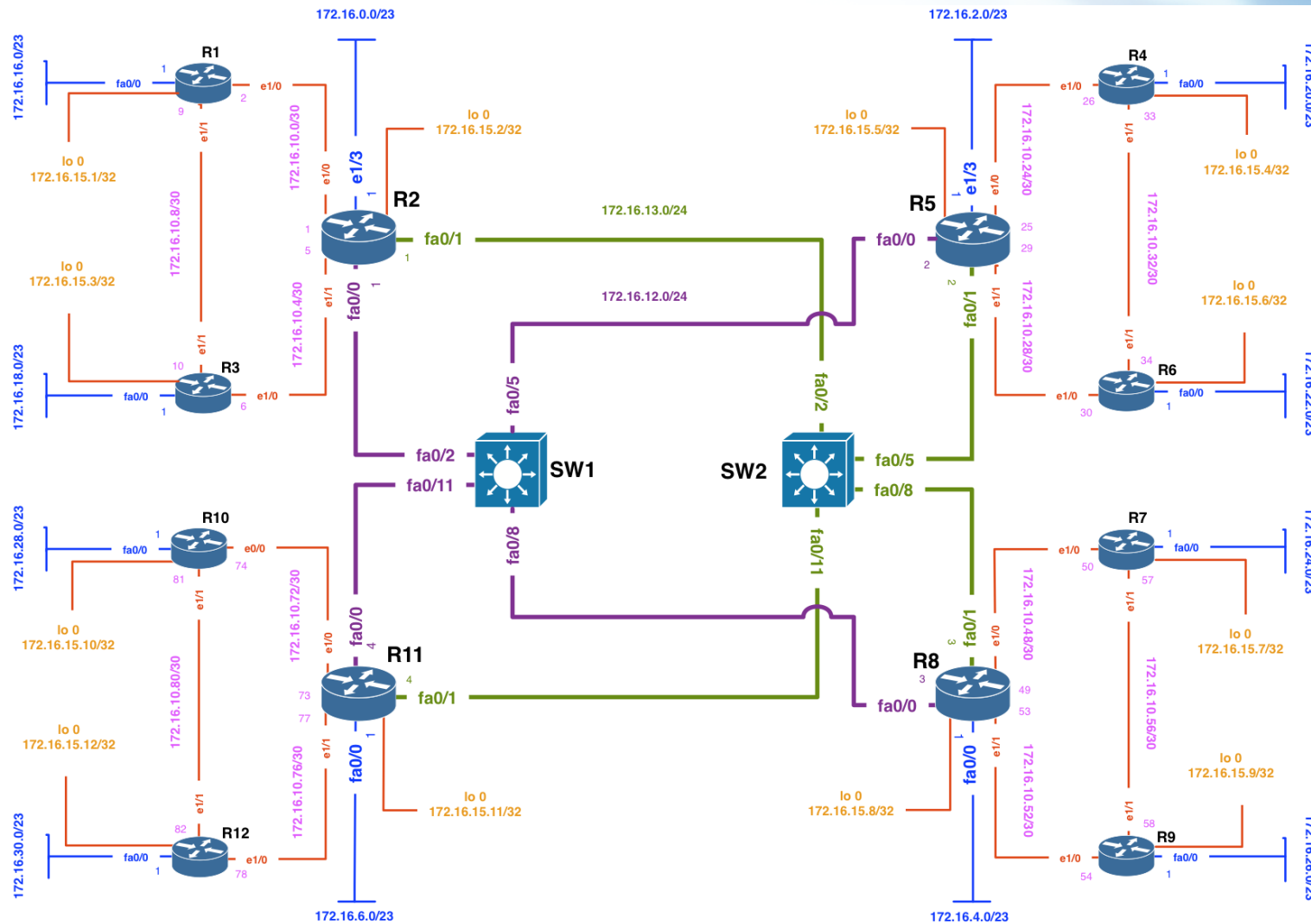
# Training ISP IPV4 Addressing Plan

## Current IPv4 Addressing Plan:

### Detail customer block IPV4

Block#	Prefix	Size	Description	SOR	Register
28	172.16.6.0	/20	Customer network		
29	172.16.16.0	/22	Router2 summary net		
30	172.16.16.0	/23	Router1 CS network	Yes	Must
31	172.16.18.0	/23	Router3 CS network	Yes	Must
32	172.16.20.0	/22	Router5 summary net		
33	172.16.20.0	/23	Router4 CS network	Yes	Must
34	172.16.22.0	/23	Router6 CS network	Yes	Must
35	172.16.24.0	/22	Router8 summary net		
36	172.16.24.0	/23	Router7 CS network	Yes	Must
37	172.16.26.0	/23	Router9 CS network	Yes	Must
38	172.16.28.0	/22	Router11 summary net		
39	172.16.28.0	/23	Router10 CS network	Yes	Must
40	172.16.30.0	/23	Router12 CS network	Yes	Must

# Training ISP IPv4 Addressing Plan



Training ISP IPv4 Address Plan



# Training ISP IPV6 Addressing Pan

IPv6 address plan consideration:

- Big IPv6 address space can cause very very large routing table size
- Most transit service provider apply IPv6 prefix filter on anything other than /32 & /48 prefix size
- Prefix announcement need to send to Internet should be either /32 or /48 bit boundary

# Training ISP IPV6 Addressing Pan

IPv6 address plan consideration (RFC3177):

- WAN link can be used on /64 bit boundary
- End site/Customer sub allocation can be made on /56 bit boundary
- Utilization/HD ratio will be calculated based on /56 end site assignment/sub-allocation

# Training ISP IPV6 Addressing Plan

## IPv6 Address Plan:

Summary Parent Block & Regional Network (256x/40)

Block#	Prefix	Description
	<b>2406:6400::/32</b>	<b><i>Parent Block</i></b>
	2406:6400:0000:0000::/40	Infrastructure
	2406:6400:0100:0000::/40	Customer network Region 1
	2406:6400:0200:0000::/40	Customer network Region 2
	2406:6400:0300:0000::/40	Customer network Region 3
	2406:6400:0400:0000::/40	Customer network Region 4
	2406:6400:0500:0000::/40	
	2406:6400:0600:0000::/40	
	2406:6400:0700:0000::/40	
	2406:6400:0800:0000::/40	
	2406:6400:0900:0000::/40	
	2406:6400:0A00:0000::/40	
	2406:6400:0B00:0000::/40	
	2406:6400:0C00:0000::/40	
	2406:6400:0D00:0000::/40	
	2406:6400:0E00:0000::/40	
	2406:6400:0F00:0000::/40	

# Training ISP IPV6 Addressing Plan

## IPv6 Address Plan:

Summary Infrastructure Prefix (256x/48)

Block#	Prefix	Description
	<b>2406:6400:0000:0000::/40</b>	<b>Infrastructure</b>
	2406:6400:0000:0000::/48	Loopback
	2406:6400:0001:0000::/48	R2 DC Summary
	2406:6400:0002:0000::/48	R5 DC Summary
	2406:6400:0003:0000::/48	R8 DC Summary
	2406:6400:0004:0000::/48	R11 DC Summary
	2406:6400:0005:0000::/48	
	2406:6400:0006:0000::/48	
	2406:6400:0007:0000::/48	
	2406:6400:0008:0000::/48	
	2406:6400:0009:0000::/48	
	2406:6400:000A:0000::/48	
	2406:6400:000B:0000::/48	
	2406:6400:000C:0000::/48	
	2406:6400:000D:0000::/48	Purple Transport
	2406:6400:000E:0000::/48	Green Transport
	2406:6400:000F:0000::/48	WAN Prefix

# Training ISP IPV6 Addressing Plan

## IPv6 Address Plan WAN Prefix:

**2406:6400:000F:0000::/48**
**WAN Prefix (65535x/64)**

2406:6400:000F:0000::/64	R2-R1
2406:6400:000F:0001::/64	R2-R3
2406:6400:000F:0002::/64	R1-R3
2406:6400:000F:0003::/64	
2406:6400:000F:0004::/64	
2406:6400:000F:0005::/64	
2406:6400:000F:0006::/64	
2406:6400:000F:0007::/64	
2406:6400:000F:0008::/64	
2406:6400:000F:0009::/64	
2406:6400:000F:000A::/64	
2406:6400:000F:000B::/64	
2406:6400:000F:000C::/64	
2406:6400:000F:000D::/64	
2406:6400:000F:000E::/64	
2406:6400:000F:000F::/64	

2406:6400:000F:0010::/64	R5-R4
2406:6400:000F:0011::/64	R5-R6
2406:6400:000F:0012::/64	R4-R6
2406:6400:000F:0013::/64	
2406:6400:000F:0014::/64	
2406:6400:000F:0015::/64	
2406:6400:000F:0016::/64	
2406:6400:000F:0017::/64	
2406:6400:000F:0018::/64	
2406:6400:000F:0019::/64	
2406:6400:000F:001A::/64	
2406:6400:000F:001B::/64	
2406:6400:000F:001C::/64	
2406:6400:000F:001D::/64	
2406:6400:000F:001E::/64	
2406:6400:000F:001F::/64	

# Training ISP IPV6 Addressing Plan

## IPv6 Address Plan:

**2406:6400:000F:0000::/48**

**WAN Prefix (65535x/64)**

2406:6400:000F:0030::/64	R11-R10
2406:6400:000F:0031::/64	R11-R12
2406:6400:000F:0032::/64	R10-R12
2406:6400:000F:0033::/64	
2406:6400:000F:0034::/64	
2406:6400:000F:0035::/64	
2406:6400:000F:0036::/64	
2406:6400:000F:0037::/64	
2406:6400:000F:0038::/64	
2406:6400:000F:0039::/64	
2406:6400:000F:003A::/64	
2406:6400:000F:003B::/64	
2406:6400:000F:003C::/64	
2406:6400:000F:003D::/64	
2406:6400:000F:003E::/64	
2406:6400:000F:003F::/64	

2406:6400:000F:0020::/64	R8-R7
2406:6400:000F:0021::/64	R8-R9
2406:6400:000F:0022::/64	R7-R9
2406:6400:000F:0023::/64	
2406:6400:000F:0024::/64	
2406:6400:000F:0025::/64	
2406:6400:000F:0026::/64	
2406:6400:000F:0027::/64	
2406:6400:000F:0028::/64	
2406:6400:000F:0029::/64	
2406:6400:000F:002A::/64	
2406:6400:000F:002B::/64	
2406:6400:000F:002C::/64	
2406:6400:000F:002D::/64	
2406:6400:000F:002E::/64	
2406:6400:000F:002F::/64	

# Training ISP IPV6 Addressing Plan

## IPv6 Address Plan:

Summary Customer net Region 1 (256x/48)

Block#	Prefix	Description
	2406:6400:0100:0000::/40	Customer network Region 1
	<b>2406:6400:0100:0000::/48</b>	<b>R1 Cust Net</b>
	2406:6400:0101:0000::/48	
	2406:6400:0102:0000::/48	
	2406:6400:0103:0000::/48	
	2406:6400:0104:0000::/48	
	2406:6400:0105:0000::/48	
	2406:6400:0106:0000::/48	
	2406:6400:0107:0000::/48	
	<b>2406:6400:0108:0000::/48</b>	<b>R3 Cust Net</b>
	2406:6400:0109:0000::/48	
	2406:6400:010A:0000::/48	
	2406:6400:010B:0000::/48	
	2406:6400:010C:0000::/48	
	2406:6400:010D:0000::/48	
	2406:6400:010E:0000::/48	
	2406:6400:010F:0000::/48	



# Training ISP IPV6 Addressing Plan

## Future IPv6 Address Plan:

Summary Customer net Region 2 (256x/48)

Block#	Prefix	Description
	2406:6400:0200:0000::/40	Customer network Region 2
	<b>2406:6400:0200:0000::/48</b>	<b>R4 Cust Net</b>
	2406:6400:0201:0000::/48	
	2406:6400:0202:0000::/48	
	2406:6400:0203:0000::/48	
	2406:6400:0204:0000::/48	
	2406:6400:0205:0000::/48	
	2406:6400:0206:0000::/48	
	2406:6400:0207:0000::/48	
	<b>2406:6400:0208:0000::/48</b>	<b>R6 Cust Net</b>
	2406:6400:0209:0000::/48	
	2406:6400:020A:0000::/48	
	2406:6400:020B:0000::/48	
	2406:6400:020C:0000::/48	
	2406:6400:020D:0000::/48	
	2406:6400:020E:0000::/48	
	2406:6400:020F:0000::/48	

# Training ISP IPV6 Addressing Plan

## IPv6 Address Plan:

Summary Customer net Region 3 (256x/48)

Block#	Prefix	Description
	2406:6400:0300:0000::/40	Customer network Region 3
	<b>2406:6400:0300:0000::/48</b>	<b>R7 Cust Net</b>
	2406:6400:0301:0000::/48	
	2406:6400:0302:0000::/48	
	2406:6400:0303:0000::/48	
	2406:6400:0304:0000::/48	
	2406:6400:0305:0000::/48	
	2406:6400:0306:0000::/48	
	2406:6400:0307:0000::/48	
	<b>2406:6400:0308:0000::/48</b>	<b>R9 Cust Net</b>
	2406:6400:0309:0000::/48	
	2406:6400:030A:0000::/48	
	2406:6400:030B:0000::/48	
	2406:6400:030C:0000::/48	
	2406:6400:030D:0000::/48	
	2406:6400:030E:0000::/48	
	2406:6400:030F:0000::/48	

# Training ISP IPV6 Addressing Plan

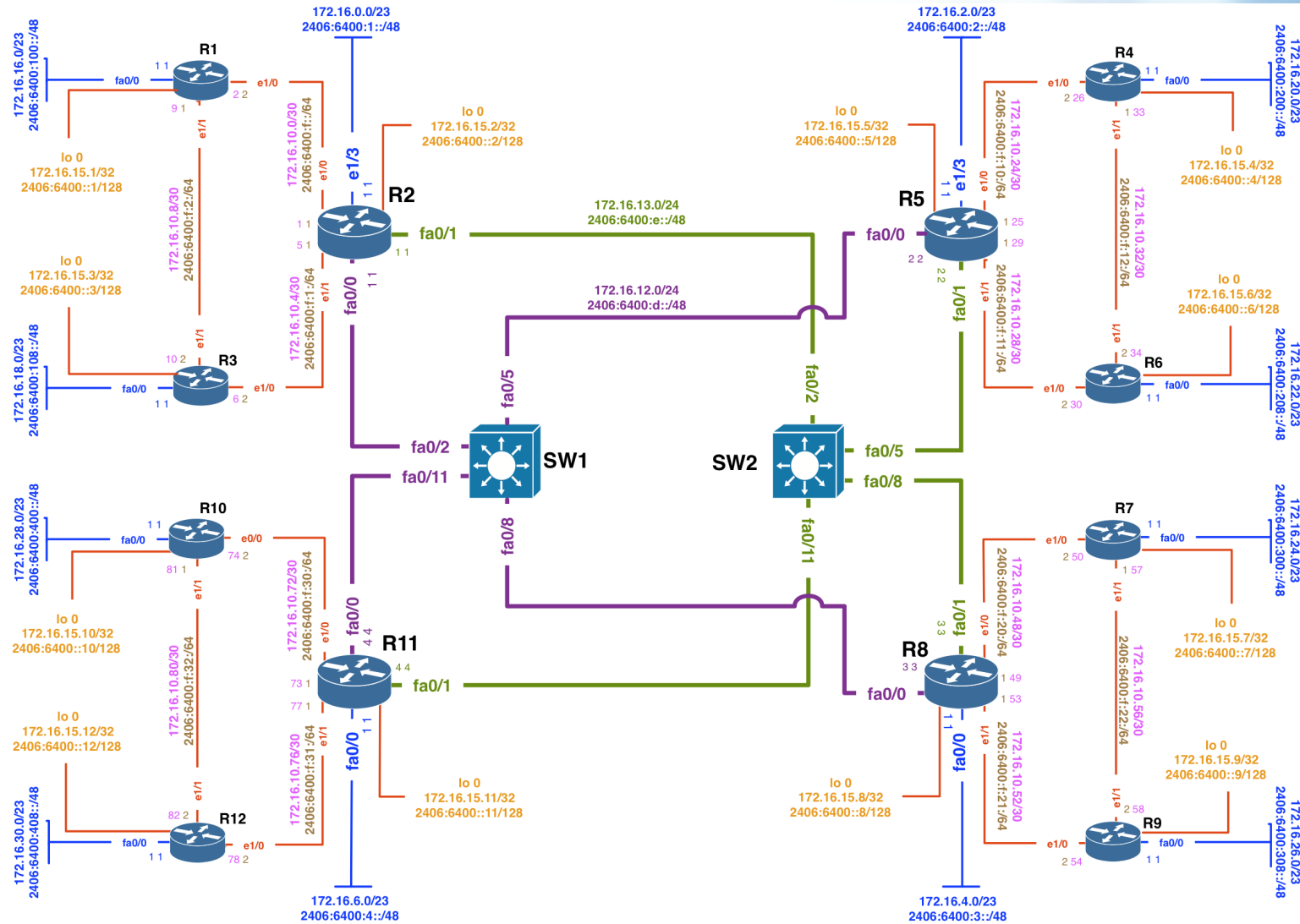
## IPv6 Address Plan:

Summary Customer net Region 4 (256x/48)

Block#	Prefix	Description
	2406:6400:0400:0000::/40	Customer network Region 4
	<b>2406:6400:0400:0000::/48</b>	<b>R10 Cust Net</b>
	2406:6400:0401:0000::/48	
	2406:6400:0402:0000::/48	
	2406:6400:0403:0000::/48	
	2406:6400:0404:0000::/48	
	2406:6400:0405:0000::/48	
	2406:6400:0406:0000::/48	
	2406:6400:0407:0000::/48	
	<b>2406:6400:0408:0000::/48</b>	<b>R12 Cust Net</b>
	2406:6400:0409:0000::/48	
	2406:6400:040A:0000::/48	
	2406:6400:040B:0000::/48	
	2406:6400:040C:0000::/48	
	2406:6400:040D:0000::/48	
	2406:6400:040E:0000::/48	
	2406:6400:040F:0000::/48	



# Training ISP IPV6 Addressing Plan



Training ISP IPv6 Address Plan

# Questions?



# Overview

## IPv6 Deployment Tutorial

- IPv6 Protocol Architecture Overview
- IPV6 Addressing and Sub-netting
- IPv6 Host Configuration
- Training ISP Network Topology Overview
- **Deployment of IPV6 in Interior Gateway**
- IPv4 to IPv6 Transition technologies
- Planning & Implementation of IPv6 on Exterior Gateway (BGP)
- Connecting ISP network to an IXP

# Configuration of OSPF as IGP

## Scenario:

- Training ISP need to configure OSPF as IGP for both IPv4 and IPv6
- Dual stack mechanism will be used to ensure both IPv4 and IPv6 operation
- OSPFv3 supports IPv6 routed protocol
- IGP is used to carry next hop only for BGP



# Configuration of OSPF as IGP

Minimum Router OS require for OSPF3:

Cisco IOS

- 12.2(15)T or later (For OSPFv3)
- 12.2(2)T or later (For IPv6 support)

Jun OS

- JUNOS 8.4 or later

# Configuration of OSPF as IGP

Before enabling OSPF3 on an Interface following steps must be done on a Router:

- Enable IPv6 unicast routing
- Enable IPv6 CEF

```
config t  
ipv6 unicast-routing  
ipv6 cef (distributed cef)
```

# Configuration of OSPF as IGP

Configure interface for both IPv4 and IPv6:

```
interface e1/0  
description WAN R1-R2  
no ip redirects  
no ip directed-broadcast  
no ip unreachable  
ip address 172.16.10.2 255.255.255.252  
no shutdown
```

```
interface e1/0  
ipv6 address 2406:6400:000F:0000::2/64  
ipv6 enable
```

# Configuration of OSPF as IGP

Verify Interface configuration:

```
sh ip interface e0/0  
ping 172.16.10.1
```

```
sh ipv6 interface e0/0  
ping 2406:6400:000F:0000::2
```

# Configuration of OSPF as IGP

## IPv4 Interface configuration for Router1:

```
interface loopback 0
description Router1 Loopback
no ip redirects
no ip directed-broadcast
no ip unreachable
ip address 172.16.15.1 255.255.255.255
no shutdown
interface e1/0
description WAN R1-R2
no ip redirects
no ip directed-broadcast
no ip unreachable
ip address 172.16.10.2 255.255.255.252
no shutdown
interface e1/1
description WAN R1-R3
no ip redirects
no ip directed-broadcast
no ip unreachable
ip address 172.16.10.9 255.255.255.252
no shutdown
interface fa0/0
description Router1 customer network
no ip redirects
no ip directed-broadcast
no ip unreachable
no cdp enable
ip address 172.16.16.1 255.255.255.0
no shutdown
```

# Configuration of OSPF as IGP

## IPv6 Interface configuration for Router1:

```
interface loopback 0
ipv6 address 2406:6400:0000:0000::1/128
ipv6 enable
interface e1/0
ipv6 address 2406:6400:000F:0000::2/64
ipv6 enable
interface e1/1
ipv6 address 2406:6400:000F:0002::1/64
ipv6 enable
interface fa0/0
ipv6 address 2406:6400:0100:0000::1/48
ipv6 enable
```

# Configuration of OSPF as IGP

## OSPF Configuration for IPv4:

- OSPF for IPv4 can be configured from global configuration mode
- Interface mode configuration will also activate OSPF process on your running config



# Configuration of OSPF as IGP

## OSPF Configuration for IPv6:

- OSPF for IPv6 need to configure from Interface configuration mode
- Interface mode configuration will automatically activate OSPF process on your running config

# Configuration of OSPF as IGP

## OSPF for IPv6 Configuration Command:

```
router ospf 17821
log-adjacency-changes
passive-interface default
network 172.16.15.1 0.0.0.0 area 1
no passive-interface e1/0
network 172.16.10.0 0.0.0.3 area 1
no passive-interface e1/1
network 172.16.10.8 0.0.0.3 area 1
```

# Configuration of OSPF as IGP

## OSPF for IPv6 Configuration Command:

```
interface loopback 0
ipv6 ospf 17821 area 0
interface e1/0
ipv6 ospf 17821 area 1
```

# Configuration of OSPF as IGP

## Verify OSPF configuration:

```
sh runn
```

```
!
```

```
interface Ethernet1/0
```

```
description WAN R1-R2
```

```
ip address 172.16.10.2 255.255.255.252
```

```
no ip redirects
```

```
no ip unreachable
```

```
half-duplex
```

```
ipv6 address 2406:6400:F::2/64
```

```
ipv6 enable
```

```
ipv6 ospf 17821 area 1
```

# Configuration of OSPF as IGP

## Example OSPF configuration for Router1:

```
router ospf 17821
log-adjacency-changes
passive-interface default
network 172.16.15.1 0.0.0.0 area 1
no passive-interface e1/0
network 172.16.10.0 0.0.0.3 area 1
no passive-interface e1/1
network 172.16.10.8 0.0.0.3 area 1
```

```
interface loopback 0
ipv6 ospf 17821 area 1
interface e1/0
ipv6 ospf 17821 area 1
interface e1/1
ipv6 ospf 17821 area 1
```

# OSPF Packet Type

## Five OSPF Packet Type:

t: Specifies the OSPF packet type:

- 1: hello [every 10 sec]
- 2: DBD [Database Descriptor Packet]
- 3: LSR [Link State Request Packet]
- 4: LSU [Link State Update Packet]
- 5: LSAck [Link State Ack Packet]

```
debug ip ospf packet
```

```
debug ipv6 ospf packet
```

# Deployment IPV6 in IGP

## OSPFv3 or OSPF for IPv6 Overview:

- OSPFv3 is described in RFC 2740
- Most of OSPF3 functions are same as OSPFv2
- In OSPFv3 routing process does not need to be explicitly created. Simply enabling OSPF on an interface will create routing process on a router



# Deployment IPV6 in IGP

## OSPFv3 or OSPF for IPv6 Overview:

- Multiple instances of OSPFv3 can be run on a link which is unlike in OSPFv2
- OSPFv3 still use 32 bit address as router ID. If no IPv4 address is configured on any interface need to use router-id command to set 32 bit router-id.

# Deployment IPV6 in IGP

## OSPFv3 or OSPF for IPv6 Overview:

- OSPFv3 require IPsec to enable authentication. Crypto images are required to use adjacency authentication
- To use IPsec AH you must use *IPv6 OSPF authentication* command
- To use IPsec ESP you must enable *IPv6 OSPF encryption* command

# Deployment IPV6 in IGP

## OSPFv3 or OSPF for IPv6 Overview:

- LSA types and functions in OSPF3 are same as OSPF2
- OSPFv3 use IPv6 address  
FF02::5 for AllSPF router multicast  
and IPv6 address FF02::6 for AllD  
router multicast
- DR/BDR concepts for Broadcast  
Multi-access network are same in  
OSPFv3 as OSPFv2

# Deployment IPV6 in IGP

- OSPFv3 or OSPF for IPv6 Overview:
  - The Hello packet now contains no address information at all, and includes an Interface ID which the originating router has assigned to uniquely identify (among its own interfaces) its interface to the link.
  - This Interface ID becomes the Network-LSA's Link State ID, obviously when the router becomes Designated-Router on the link.

# Questions?



# Overview

## IPv6 Deployment Tutorial

- IPv6 Protocol Architecture Overview
- IPV6 Addressing and Sub-netting
- IPv6 Host Configuration
- Training ISP Network Topology Overview
- Deployment of IPV6 in Interior Gateway
- **IPv4 to IPv6 Transition technologies**
- Planning & Implementation of IPv6 on Exterior Gateway (BGP)
- Connecting ISP network to an IXP



# Transition overview

- How to get connectivity from an IPv6 host to the global IPv6 Internet?
  - Via an native connectivity
  - Via IPv6-in-IPv4 tunnelling techniques
- IPv6-only deployments are rare
- Practical reality
  - Sites deploying IPv6 will not transit to IPv6-only, but transit to a state where they support both IPv4 and IPv6 (dual-stack)



# IPv4 to IPv6 transition

- Implementation rather than transition
  - No fixed day to convert
- The key to successful IPv6 transition
  - Maintaining compatibility with IPv4 hosts and routers while deploying IPv6
    - Millions of IPv4 nodes already exist
    - Upgrading every IPv4 nodes to IPv6 is not feasible
      - No need to convert all at once
    - Transition process will be gradual

# Transition overview

- Three basic ways of transition
  - Dual stack
  - Deploying IPv6 and then implementing IPv6-in-IPv4 tunnelling
  - IPv6 only networking
- Different demands of hosts and networks to be connected to IPv6 networks will determine the best way of transition

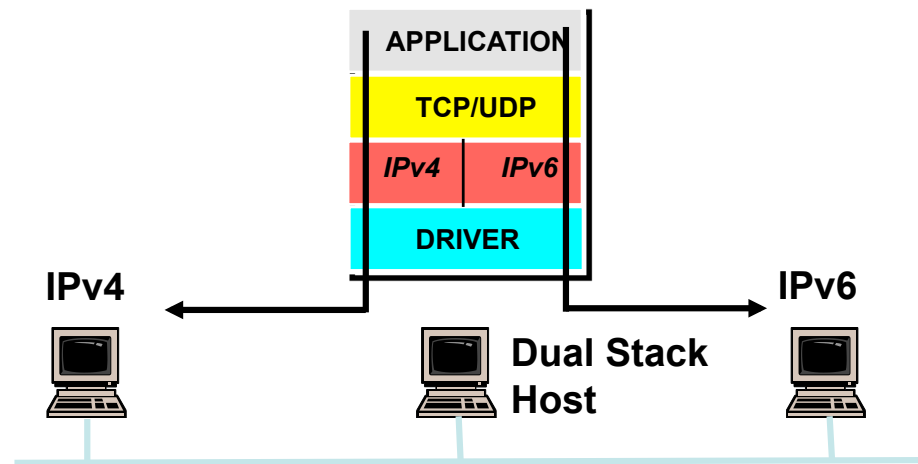
# Transition overview

- Dual stack
  - Allow IPv4 and IPv6 to coexist in the same devices and networks
- Tunnelling
  - Allow the transport of IPv6 traffic over the existing IPv4 infrastructure
- Translation
  - Allow IPv6 only nodes to communicate with IPv4 only nodes

# Dual stack transition



- Dual stack = TCP/IP protocol stack running both IPv4 and IPv6 protocol stacks simultaneously
  - Application can talk to both
- Useful at the early phase of transition



# Dual stack

- A host or a router runs both IPv4 and IPv6 in the protocol TCP/IP stack.
- Each dual stack node is configured with both IPv4 and IPv6 addresses
- Therefore it can both send and receive datagrams belonging to both protocols
- The simplest and the most desirable way for IPv4 and IPv6 to coexist

# Dual stack

- Challenges

- Compatible software

- Eg. If you use OSPFv2 for your IPv4 network you need to run OSPFv3 in addition to OPSFv2

- Transparent availability of services

- Deployment of servers and services
    - Content provision
    - Business processes
    - Traffic monitoring
    - End user deployment

# Dual stack and DNS

- DNS is used with both protocol versions to resolve names and IP addresses
  - An dual stack node needs a DNS resolver that is capable of resolving both types of DNS address records
    - DNS A record to resolve IPv4 addresses
    - DNS AAAA record to resolve IPv6 addresses
- Dual stack network
  - Is an infrastructure in which both IPv4 and Ipv6 forwarding is enabled on routers

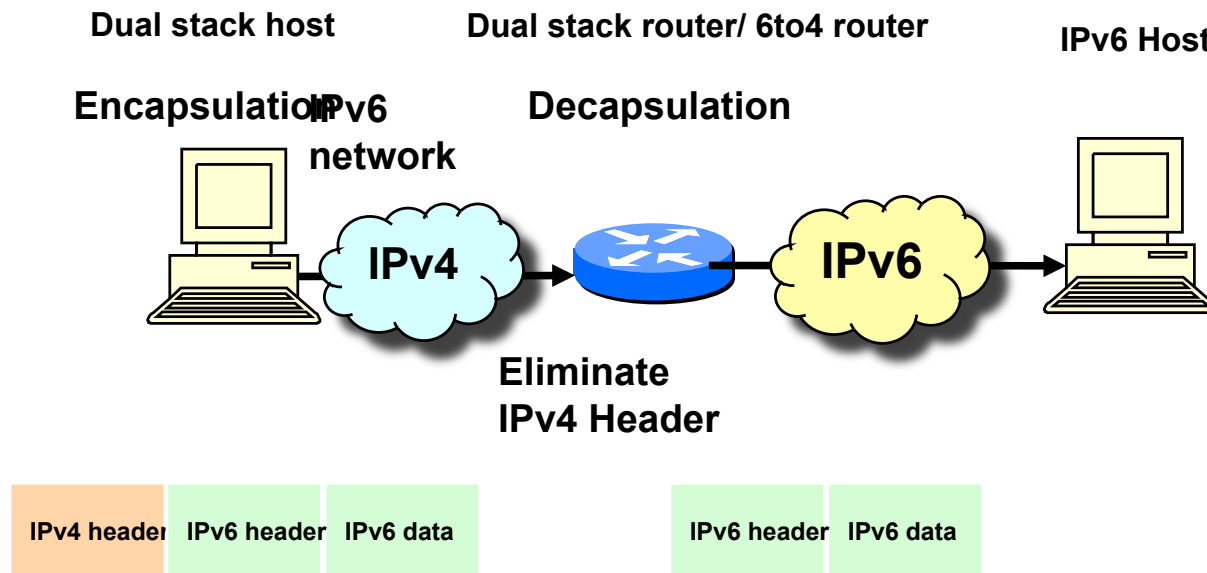


# Tunnels

- Part of a network is IPv6 enabled
  - Tunnelling techniques are used on top of an existing IPv4 infrastructure and uses IPv4 to route the IPv6 packets between IPv6 networks by transporting these encapsulated in IPv4
  - Tunnelling is used by networks not yet capable of offering native IPv6 functionality
  - It is the main mechanism currently being deployed to create global IPv6 connectivity
- Manual, automatic, semi-automatic configured tunnels are available

# Tunneling – general concept

- Tunneling can be used by routers and hosts
  - IPv6-over-IPv4 tunneling



# Tunnelling – general concept

- A tunnel can be configured in four different ways:
  - Router to router
    - Spans one hop of the end-to-end path between two hosts. Probably the most common method
  - Host to router
    - Spans the first hop of the end-to-end path between two hosts. Found in the tunnel broker model
  - Host to host
    - Spans the entire end-to-end path between two hosts
  - Router to host
    - Spans the last hop of the end-to-end path between two hosts

# Tunnel encapsulation

- The steps for the encapsulation of the IPv6 packet
  - The entry point of the tunnel decrements the IPv6 hop limit by one
  - Encapsulates the packet in an IPv4 header
  - Transmits the encapsulated packet through the tunnel
  - The exit point of tunnel receives the encapsulated packet
    - If necessary, the IPv4 packet is fragmented

# Tunnel encapsulation (Cont)

- It checks whether the source of the packet (tunnel entry point) is an acceptable source (according to its configuration)
  - If the packet is fragmented, the exit point reassembles it
- The exit point removes the IPv4 header
- Then it forwards the IPv6 packet to its original destination

# Encapsulated IPv6 packets in IPv4

capture-v4-v6-tunnel.pcap - Wireshark

File Edit View Go Capture Analyze Statistics Help

Filter: Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
28	14.019048	Cisco_b2:5d:0d	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x801a
29	14.019876	Cisco_b2:5d:0e	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x801b
30	14.020676	Cisco_b2:5d:18	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x8026
31	16.021756	Cisco_b2:5d:0d	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x801a
32	16.022587	Cisco_b2:5d:0e	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x801b
33	16.023389	Cisco_b2:5d:18	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x8026
34	16.328525	2001:1234:1:0:203:	2001:1234:2:0:203:	ICMPv6 Echo request	
35	16.330377	2001:1234:2:0:203:	2001:1234:1:0:203:	ICMPv6 Echo reply	
36	17.338335	2001:1234:1:0:203:	2001:1234:2:0:203:	ICMPv6 Echo request	
37	17.340143	2001:1234:2:0:203:	2001:1234:1:0:203:	ICMPv6 Echo reply	
38	18.024485	Cisco_b2:5d:0d	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x801a
39	18.025312	Cisco_b2:5d:0e	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x801b
40	18.026126	Cisco_b2:5d:18	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x8026
41	18.338399	2001:1234:1:0:203:	2001:1234:2:0:203:	ICMPv6 Echo request	
42	18.340163	2001:1234:2:0:203:	2001:1234:1:0:203:	ICMPv6 Echo reply	
43	19.338453	2001:1234:1:0:203:	2001:1234:2:0:203:	ICMPv6 Echo request	
44	19.340256	2001:1234:2:0:203:	2001:1234:1:0:203:	ICMPv6 Echo reply	
45	20.027200	Cisco_b2:5d:0d	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x801a
46	20.028032	Cisco_b2:5d:0e	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x801b
47	20.028827	Cisco_b2:5d:18	Spanning-tree-(for STP	Conf. Root = 32768/00:d0:58:b2:5d:27	Cost = 0 Port = 0x8026
48	20.338525	2001:1234:1:0:203:	2001:1234:2:0:203:	ICMPv6 Echo request	
49	20.340256	2001:1234:2:0:203:	2001:1234:1:0:203:	ICMPv6 Echo reply	

Type: IP (0x0800)

Internet Protocol, Src: 192.168.3.1 (192.168.3.1), Dst: 192.168.3.2 (192.168.3.2)

Version: 4

Header length: 20 bytes

Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)

0000 00.. = Differentiated Services Codepoint: Default (0x00)

.... ..0. = ECN-Capable Transport (ECT): 0

.... ..0 = ECN-CE: 0

Total Length: 124

Identification: 0x17fb (6139)

Flags: 0x00

Fragment offset: 0

Time to live: 255

Protocol: IPv6 (0x29)

Header checksum: 0x1c0a [correct]

Source: 192.168.3.1 (192.168.3.1)

Destination: 192.168.3.2 (192.168.3.2)

Internet Protocol Version 6

Version: 6

Traffic class: 0x00

Flowlabel: 0x000000

Payload length: 64

Next header: ICMPv6 (0x3a)

Hop limit: 63

Source address: 2001:1234:1:0:203:47ff:fea0:2543

Destination address: 2001:1234:2:0:203:47ff:fea0:2549

0010 00 7c 17 fb 00 00 ff 29 1c 0a c0 a8 03 01 c0 a8 .[.]. . . . .

0020 03 02 60 00 00 00 00 40 3a 3f 20 01 12 34 00 01 . . . . .

0030 00 00 02 03 47 ff fe a0 25 43 20 01 12 34 00 02 . . . . .

0040 00 00 02 03 47 ff fe a0 25 49 80 00 f5 77 4e 31 . . . . .

0050 00 00 3b b9 4a 45 80 9a 0a 00 08 09 0a 0b 0c 0d . . . . .

0060 00 00 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d . . . . .

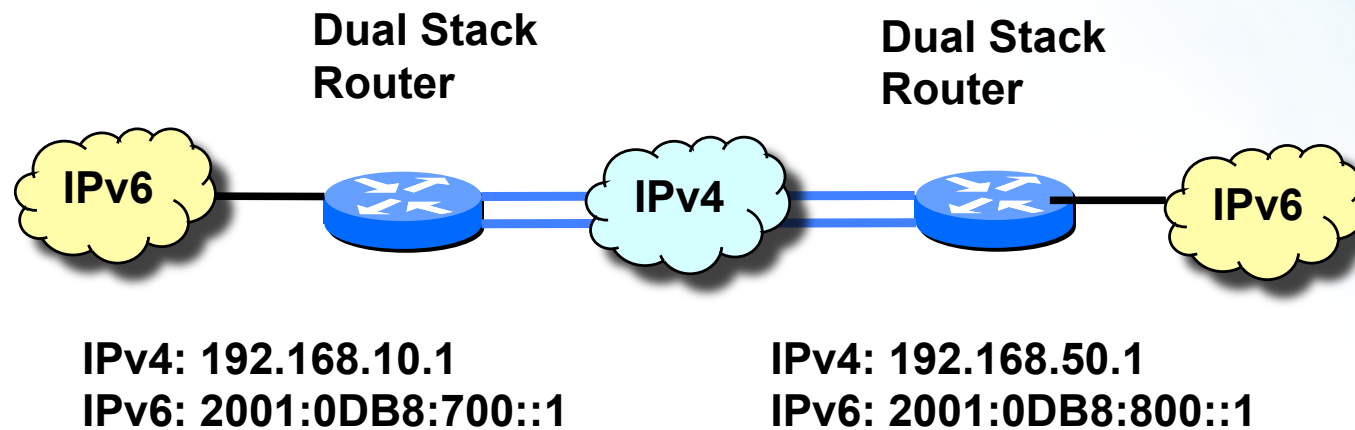
Flags (ip.flags), 1 byte

IP: 123 D: 123 M: 0

start 2 Mozilla T... Cisco IOS I... IPv6-traine... 2 Window... 2 Notepad Mozilla Firefox capture-v4-... EN 1:03 PM



# Manual configuration

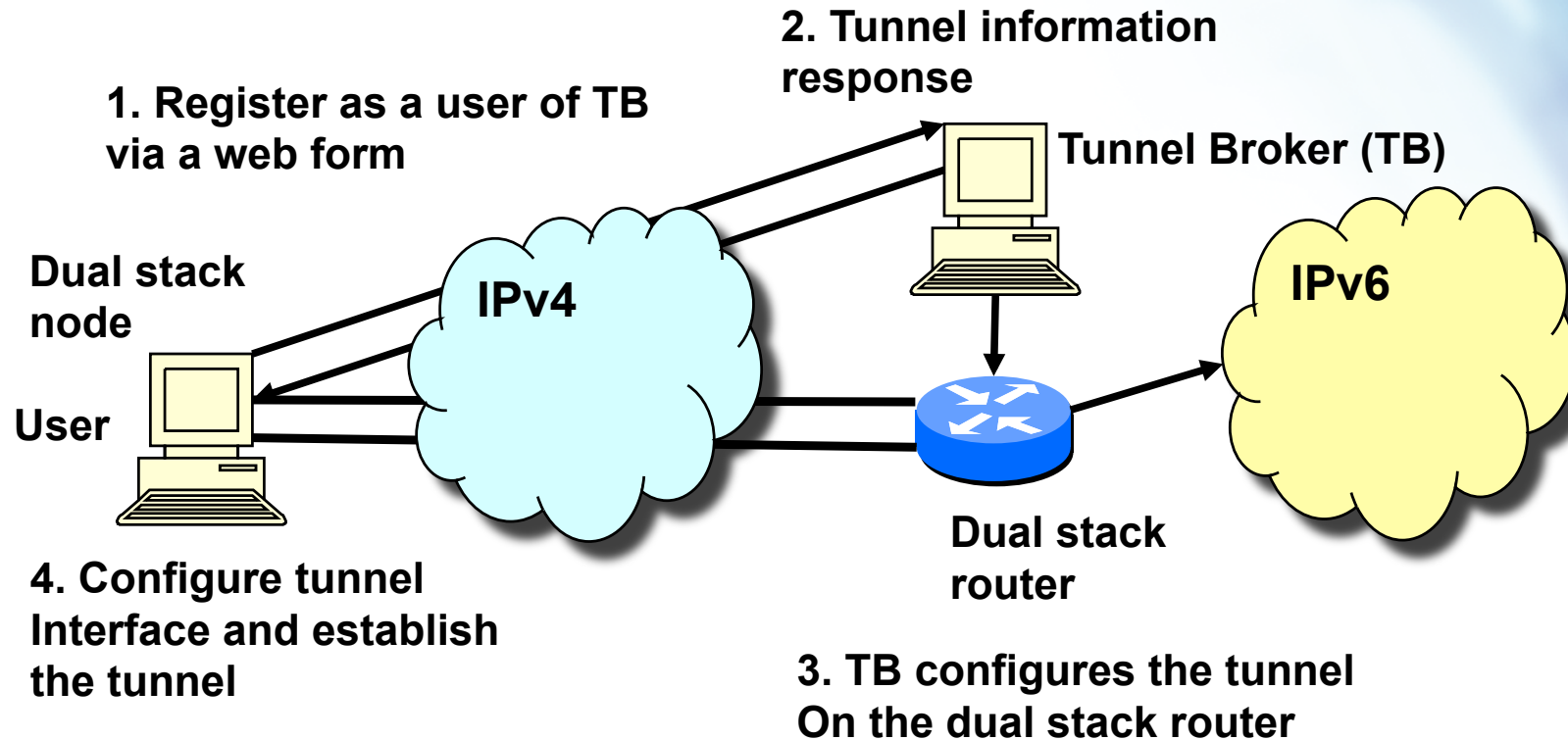


Manually configured tunnels require:

- Dual stack end points
- Explicit configuration with both IPv4 and IPv6 addresses at each end



# Tunnel broker



# Questions?



# Overview

## IPv6 Deployment Tutorial

- IPv6 Protocol Architecture Overview
- IPV6 Addressing and Sub-netting
- IPv6 Host Configuration
- Training ISP Network Topology Overview
- Deployment of IPV6 in Interior Gateway
- IPv4 to IPv6 Transition technologies
- **Planning & Implementation of IPv6 on Exterior Gateway (BGP)**
- Connecting ISP network to an IXP

# Case study- Deployment IPv6 in EGP

## Scenario:

- BGP4 is used in Training ISP network
- iBGP is used between internal routers in Training ISP to carry external prefixes (i.e Customer & Global Internet Prefixes)
- Route Reflector is used to resolve iBGP full mesh scalability issue.

# Case study- Deployment IPv6 in EGP

## Scenario:

- Transit service with upstream ASes is configured with eBGP
- Customer network from downstream can also be configured with eBGP/Static
- Training ISP is having one native IPv6 transit and one tunnel IPv6 transit with AS45192 & AS131107 (2.35 as dot)

# Case study- Deployment IPv6 in EGP

## Basic BGP Configuration:

```
router bgp 17821  
address-family ipv6  
no synchronization
```

# Case study- Deployment IPv6 in EGP

## Adding iBGP Neighbor:

```
router bgp 17821  
address-family ipv6  
!
```

```
neighbor 2406:6400:0000:0000::2 remote-as 17821  
neighbor 2406:6400:0000:0000::2 update-source loopback 0  
neighbor 2406:6400:0000:0000::2 activate
```

iBGP neighbor is always recommended with loopback interface



# Case study- Deployment IPv6 in EGP

## Announcing IPv6 Prefix:

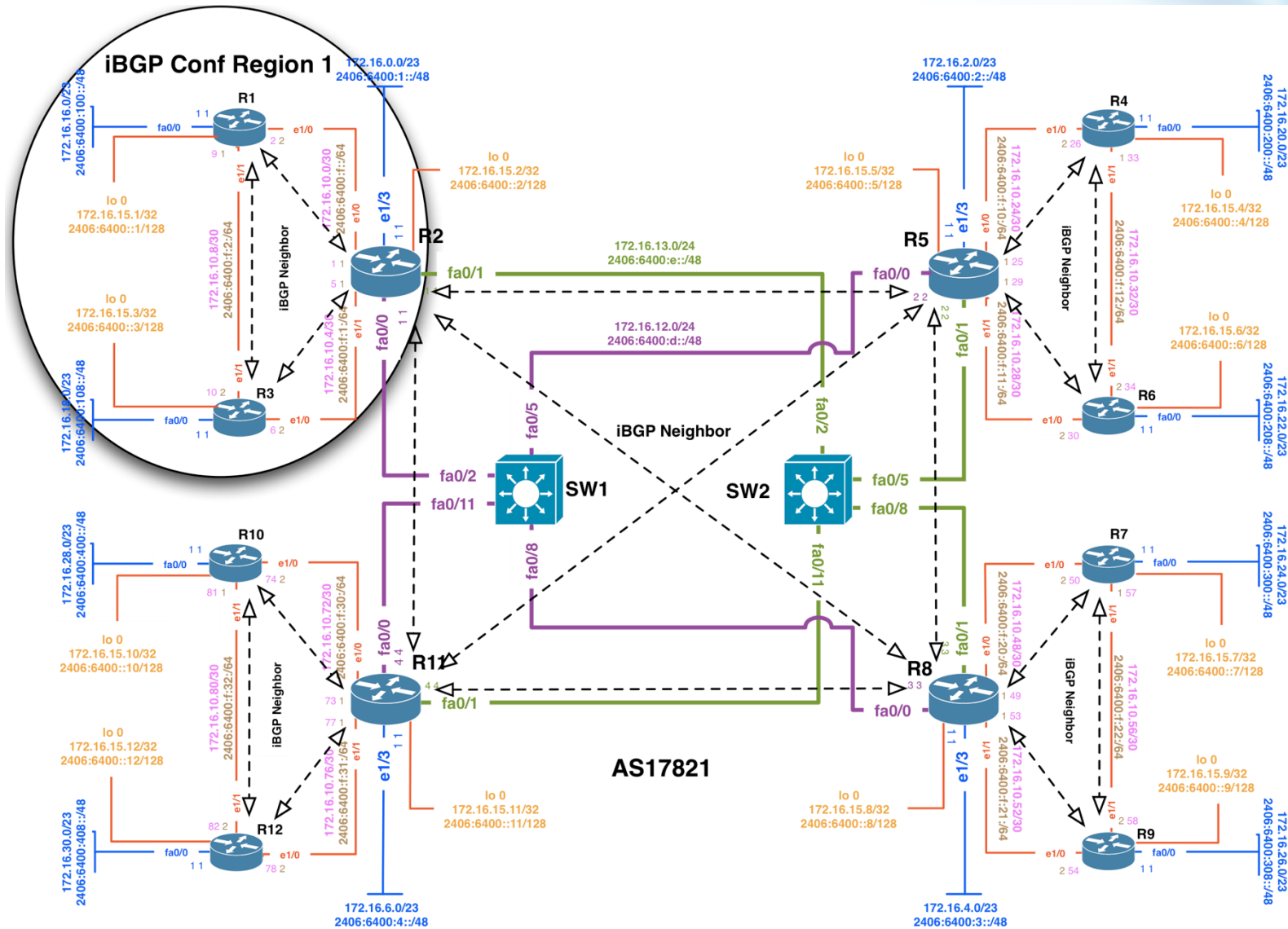
```
router bgp 17821
address-family ipv6
!
neighbor 2406:6400:0000:0000::2 remote-as 17821
neighbor 2406:6400:0000:0000::2 update-source loopback 0
neighbor 2406:6400:0000:0000::2 activate
!
network 2406:6400:0100:0000::/48
```

# Case study- Deployment IPv6 in EGP

Add Pull-up route if needed:

```
router bgp 17821
address-family ipv6
!
neighbor 2406:6400:0000:0000::2 remote-as 17821
neighbor 2406:6400:0000:0000::2 update-source loopback 0
neighbor 2406:6400:0000:0000::2 activate
!
network 2406:6400:0100:0000::/48
exit
exit
ipv6 route 2406:6400:0100:0000::/48 null 0
```

# iBGP Peering For Region 1



# IPv4 iBGP Conf POP Router

- Router1

```
config t
router bgp 17821
address-family ipv4
no auto-summary
no synchronization
neighbor 172.16.15.2 remote-as 17821
neighbor 172.16.15.2 update-source loopback 0
neighbor 172.16.15.2 activate
neighbor 172.16.15.3 remote-as 17821
neighbor 172.16.15.3 update-source loopback 0
neighbor 172.16.15.3 activate
network 172.16.16.0 mask 255.255.254.0
exit
exit
ip route 172.16.16.0 255.255.254.0 null 0 permanent
exit
wr
```

# IPv4 iBGP Configuration Verification

- POP Router

```
sh bgp ipv4 unicast summary
sh bgp ipv4 unicast
sh ip route bgp
sh bgp ipv4 unicast neighbors [router 1.....router12
    loopback] advertised-routes
sh bgp ipv4 unicast neighbors [router 1.....router12
    loopback] received-routes
sh ip route [R2, R5, R8, R11 datacenter prefix]
```

# IPv6 iBGP Conf POP Router

- Router1

```
config t
router bgp 17821
address-family ipv6
no synchronization
neighbor 2406:6400:0000:0000::2 remote-as 17821
neighbor 2406:6400:0000:0000::2 update-source loopback 0
neighbor 2406:6400:0000:0000::2 activate
neighbor 2406:6400:0000:0000::3 remote-as 17821
neighbor 2406:6400:0000:0000::3 update-source loopback 0
neighbor 2406:6400:0000:0000::3 activate
network 2406:6400:0100:0000::/45
exit
exit
ipv6 route 2406:6400:0100:0000::/45 null 0
exit
wr
```

# IPv6 iBGP Configuration Verification

- POP Router

```
sh bgp ipv6 unicast summary
sh bgp ipv6 unicast
sh ipv6 route bgp
sh bgp ipv6 unicast neighbors [router 1.....router12
    loopback] advertised-routes
sh bgp ipv6 unicast neighbors [router 1.....router12
    loopback] received-routes
sh ipv6 route [R2, R5, R8, R11 datacenter prefix]
```



# IPv4 iBGP Conf Core Router

## Router2 Configuration

```
config t
router bgp 17821
address-family ipv4
no auto-summary
no synchronization
neighbor 172.16.15.1 remote-as 17821
neighbor 172.16.15.1 update-source loopback 0
neighbor 172.16.15.1 activate
neighbor 172.16.15.3 remote-as 17821
neighbor 172.16.15.3 update-source loopback 0
neighbor 172.16.15.3 activate
neighbor 172.16.15.5 remote-as 17821
neighbor 172.16.15.5 update-source loopback 0
neighbor 172.16.15.5 activate
neighbor 172.16.15.8 remote-as 17821
neighbor 172.16.15.8 update-source loopback 0
neighbor 172.16.15.8 activate
neighbor 172.16.15.11 remote-as 17821
neighbor 172.16.15.11 update-source loopback 0
neighbor 172.16.15.11 activate
network 172.16.0.0 mask 255.255.254.0
exit
exit
ip route 172.16.0.0 255.255.254.0 null 0 permanent
exit
Wr
```

# IPv4 iBGP Configuration Verification

- Core Router

```
sh bgp ipv4 unicast summary
sh bgp ipv4 unicast
sh ip route bgp
sh bgp ipv4 unicast neighbors [router 1.....router12
    loopback] advertised-routes
sh bgp ipv4 unicast neighbors [router 1.....router12
    loopback] received-routes
sh ip route [R2, R5, R8, R11 datacenter prefix]
```

# IPv6 iBGP Conf Core Router

## Router2 Configuration

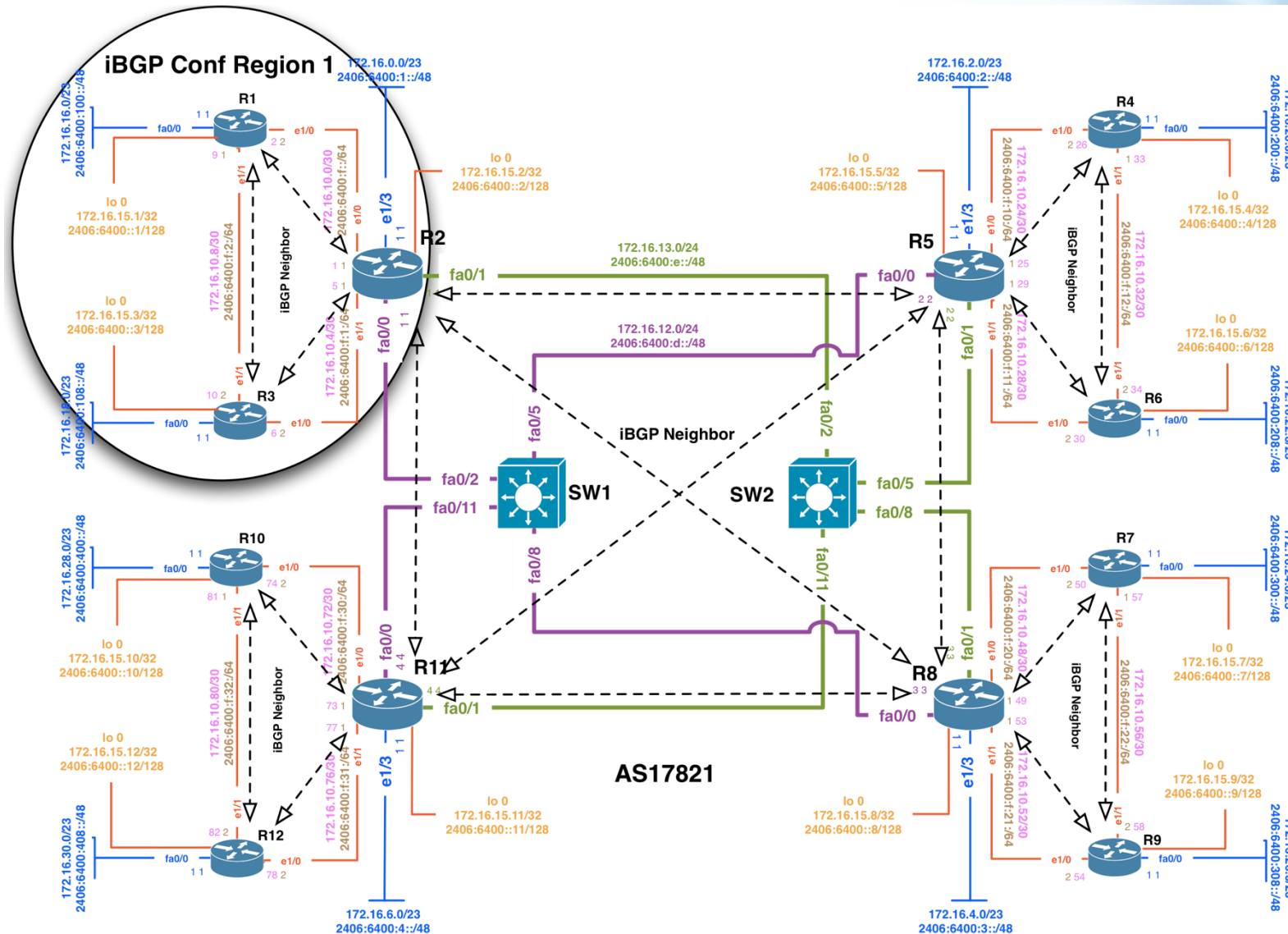
```
config t
router bgp 17821
address-family ipv6
no synchronization
neighbor 2406:6400:0000:0000::1 remote-as 17821
neighbor 2406:6400:0000:0000::1 update-source loopback 0
neighbor 2406:6400:0000:0000::1 activate
neighbor 2406:6400:0000:0000::3 remote-as 17821
neighbor 2406:6400:0000:0000::3 update-source loopback 0
neighbor 2406:6400:0000:0000::3 activate
neighbor 2406:6400:0000:0000::5 remote-as 17821
neighbor 2406:6400:0000:0000::5 update-source loopback 0
neighbor 2406:6400:0000:0000::5 activate
neighbor 2406:6400:0000:0000::8 remote-as 17821
neighbor 2406:6400:0000:0000::8 update-source loopback 0
neighbor 2406:6400:0000:0000::8 activate
neighbor 2406:6400:0000:0000::11 remote-as 17821
neighbor 2406:6400:0000:0000::11 update-source loopback 0
neighbor 2406:6400:0000:0000::11 activate
network 2406:6400:0001:0000::/48
exit
exit
ipv6 route 2406:6400:0001:0000::/48 null 0
exit
wr
```

# IPv6 iBGP Configuration Verification

- Core Router

```
sh bgp ipv6 unicast summary
sh bgp ipv6 unicast
sh ipv6 route bgp
sh bgp ipv6 unicast neighbors [router 1.....router12
    loopback] advertised-routes
sh bgp ipv6 unicast neighbors [router 1.....router12
    loopback] received-routes
sh ipv6 route [R2, R5, R8, R11 datacenter prefix]
```

# iBGP Full Mesh Issue



# iBGP Full Mesh Issue

## Route reflector configuration:

```
router bgp 17821
address-family ipv6
!
neighbor 2406:6400:0000:0000::1 remote-as 17821
neighbor 2406:6400:0000:0000::1 update-source loopback 0
neighbor 2406:6400:0000:0000::1 activate
!
neighbor 2406:6400:0000:0000::1 route-reflector-client
```



# Controlling IPV6 Route Aggregation

## IPv6 prefix filter configuration Customer:

```
config t
```

```
ipv6 prefix-list IPV6-CUST-OUT seq 5 permit ::/0 ge 32 le 32
```

```
ipv6 prefix-list IPV6-CUST-OUT seq 10 permit ::/0 ge 48 le 48
```

```
ipv6 prefix-list IPV6-CUST-IN seq 5 permit cust::/32 ge 32 le 32
```

```
ipv6 prefix-list IPV6-CUST-IN seq 10 permit cust::/32 ge 48 le 48
```

```
router bgp 17821
```

```
address-family ipv6
```

```
neighbor cust::2 prefix-list IPV6PREFIX out
```

```
exit
```

```
exit
```

```
exit
```

```
clear bgp ipv6 unicast cust::2 soft out
```



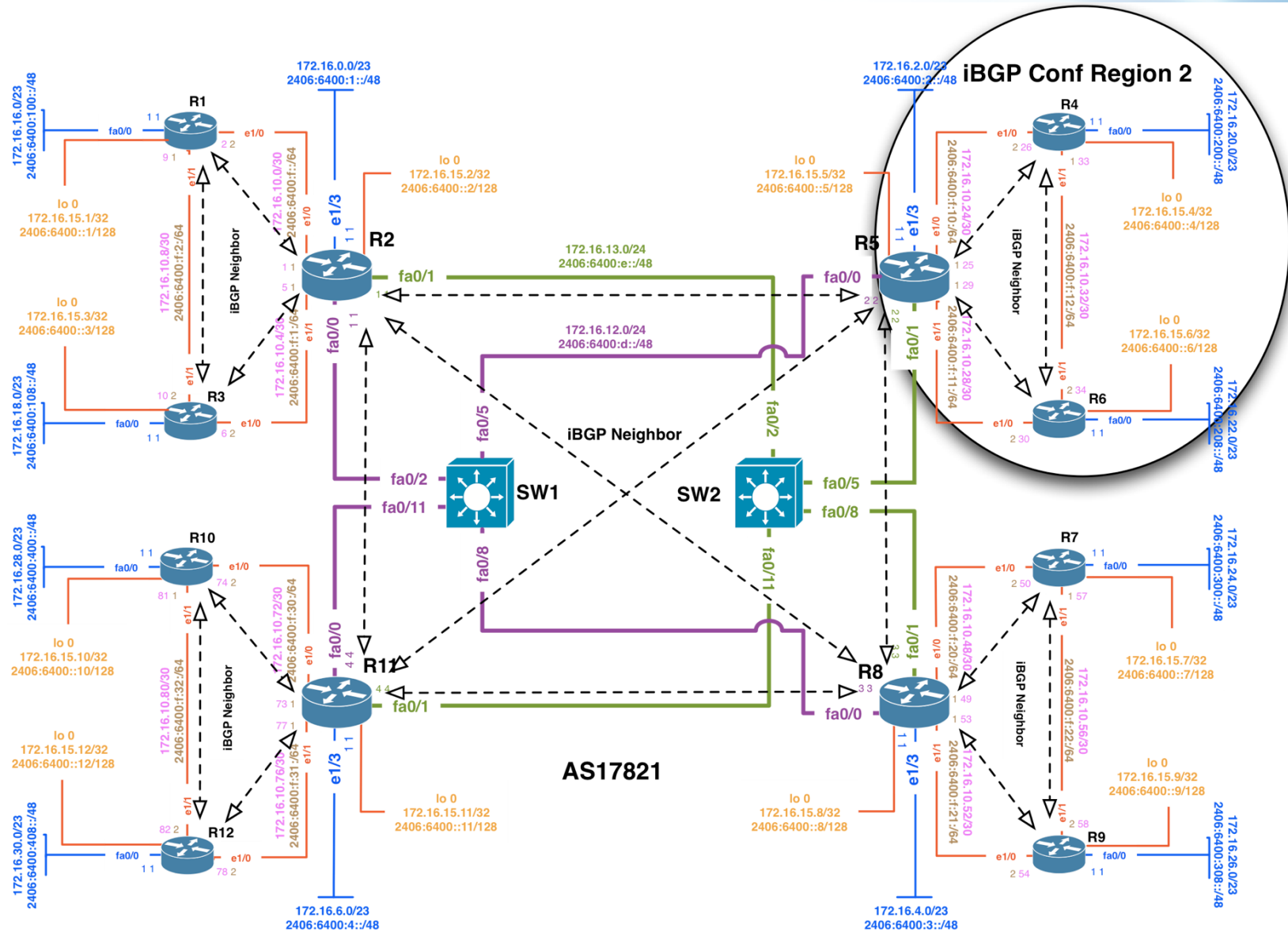
# Case study- Deployment IPv6 in EGP

## IPv6 address summarization:

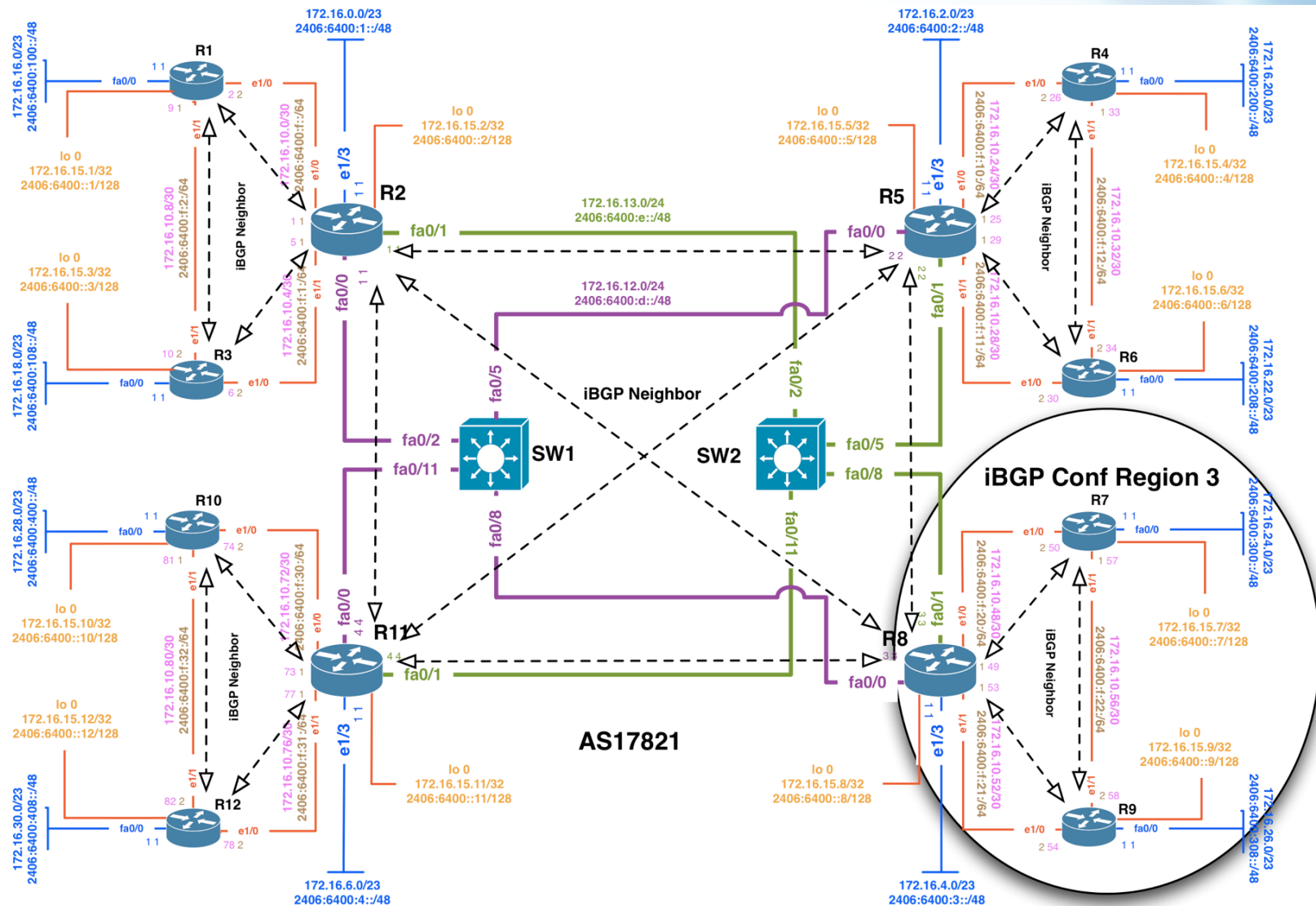
```
router bgp 17821
address-family ipv6
!
aggregate-address 2406:6400::/32
```

Need to be very careful when you summarize address

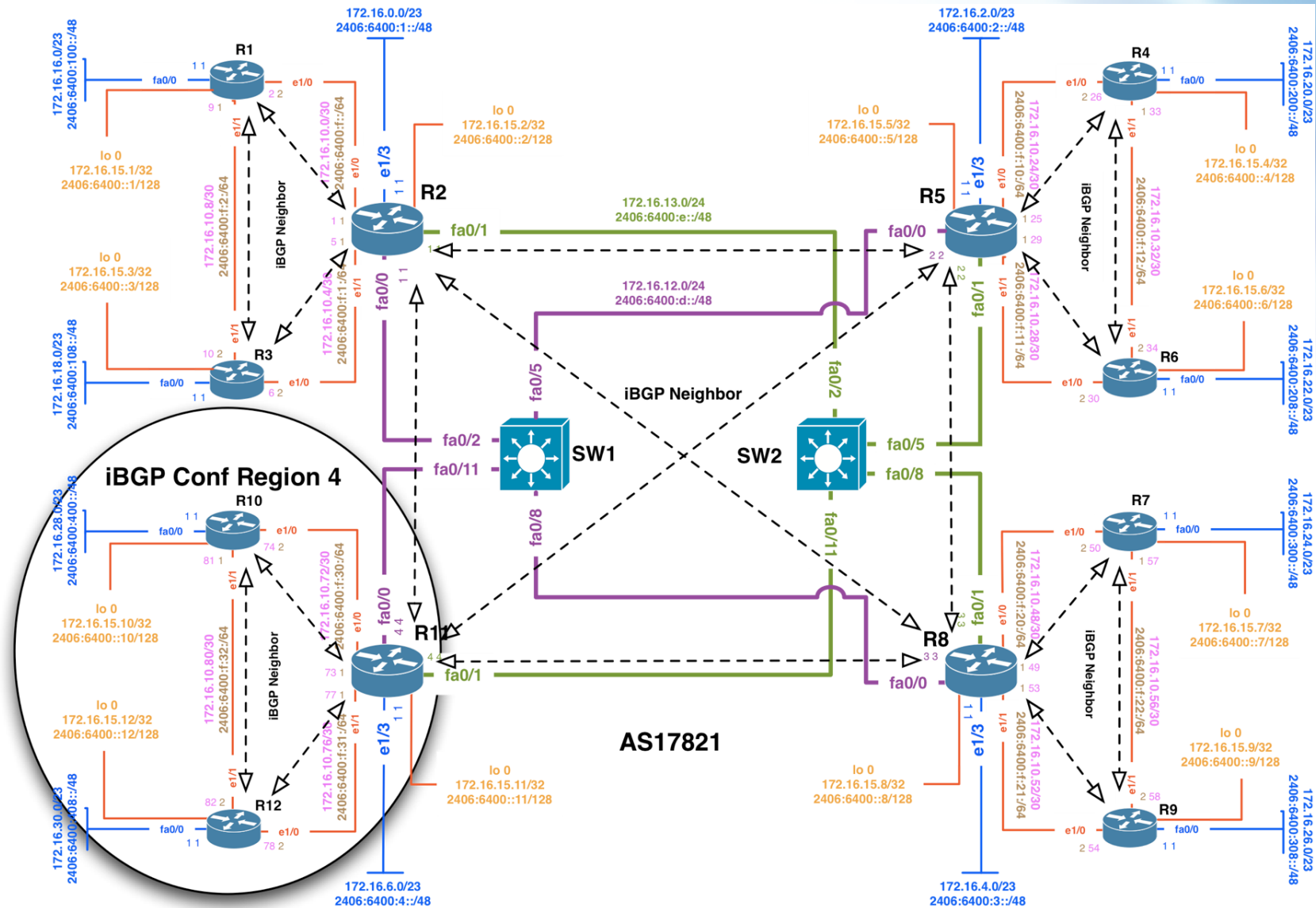
# iBGP Peering For Region 2



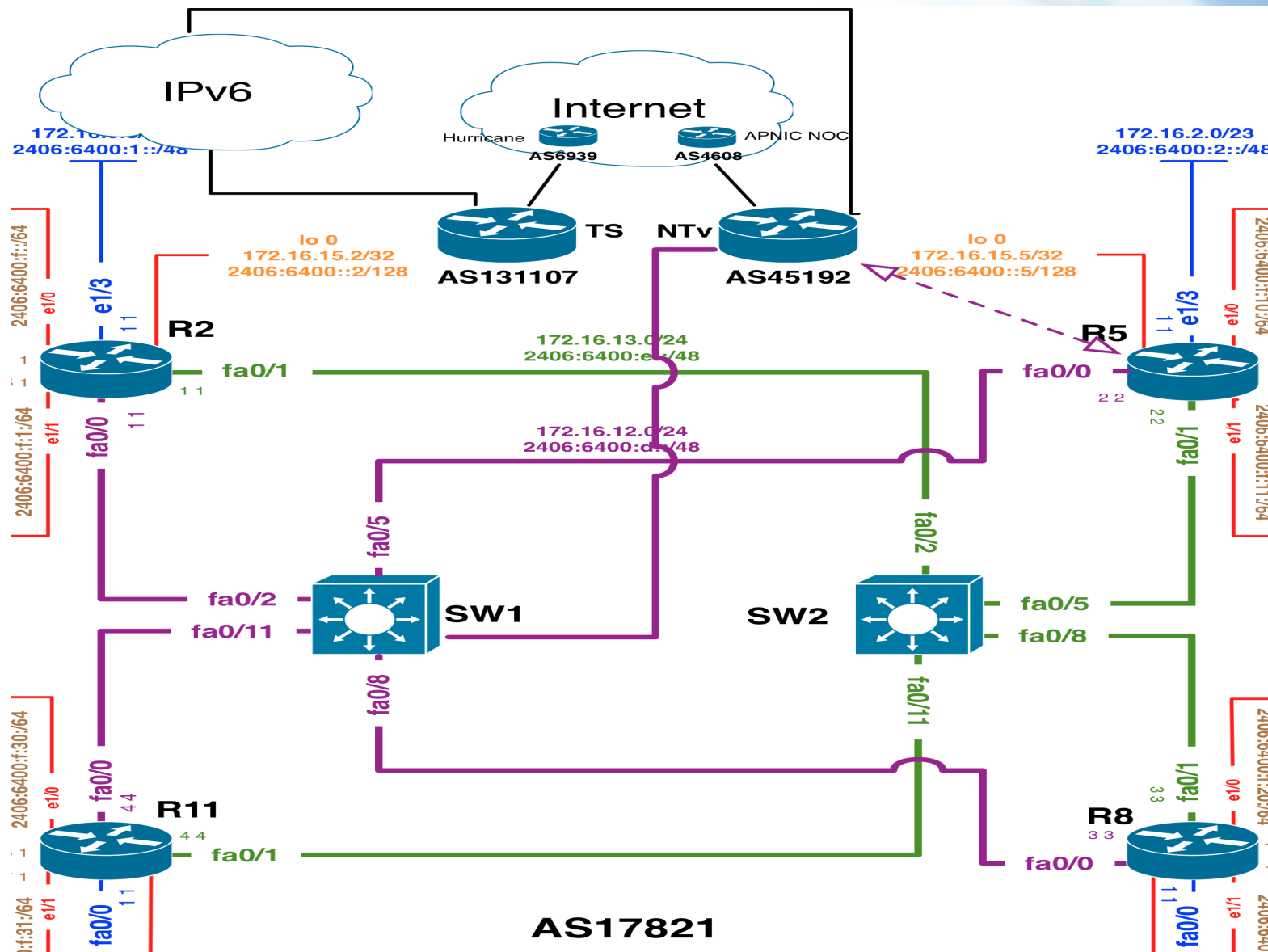
# iBGP Peering For Region 3



# iBGP Peering For Region 4



# IPv6 Native Transit Conf Plan





# IPv6 IOS Command For eBGP

## Adding eBGP Neighbor:

```
router bgp 17821
address-family ipv6
!
neighbor 2406:6400:000D:0000::5 remote-as 45192
neighbor 2406:6400:000D:0000::5 activate
```

eBGP neighbor is always recommended with directly connected interface

# IPv6 Native Transit Configuration

- Router5

```
config t
router bgp 17821
address-family ipv6
neighbor 2406:6400:000D:0000::5 remote-as 45192
neighbor 2406:6400:000D:0000::5 activate
neighbor 2406:6400:000E:0000::5 remote-as 45192
neighbor 2406:6400:000E:0000::5 activate
exit
exit
exit
Wr
```



# Controlling IPV6 Route Aggregation

## IPv6 prefix filter configuration Native Transit:

```
config t
```

```
ipv6 prefix-list IPV6-GLOBAL-IN seq 5 permit ::/0 ge 32 le 32
```

```
ipv6 prefix-list IPV6-GLOBAL-IN seq 10 permit ::/0 ge 48 le 48
```

```
!
```

```
ipv6 prefix-list IPV6-GLOBAL-OUT seq 5 permit ::/0 ge 32 le 32
```

```
ipv6 prefix-list IPV6-GLOBAL-OUT seq 10 permit ::/0 ge 48 le 48
```

```
router bgp 17821
```

```
address-family ipv6
```

```
neighbor 2406:6400:000D:0000::5 prefix-list IPV6-GLOBAL-IN in
```

```
neighbor 2406:6400:000D:0000::5 prefix-list IPV6-GLOBAL-OUT out
```

```
exit
```

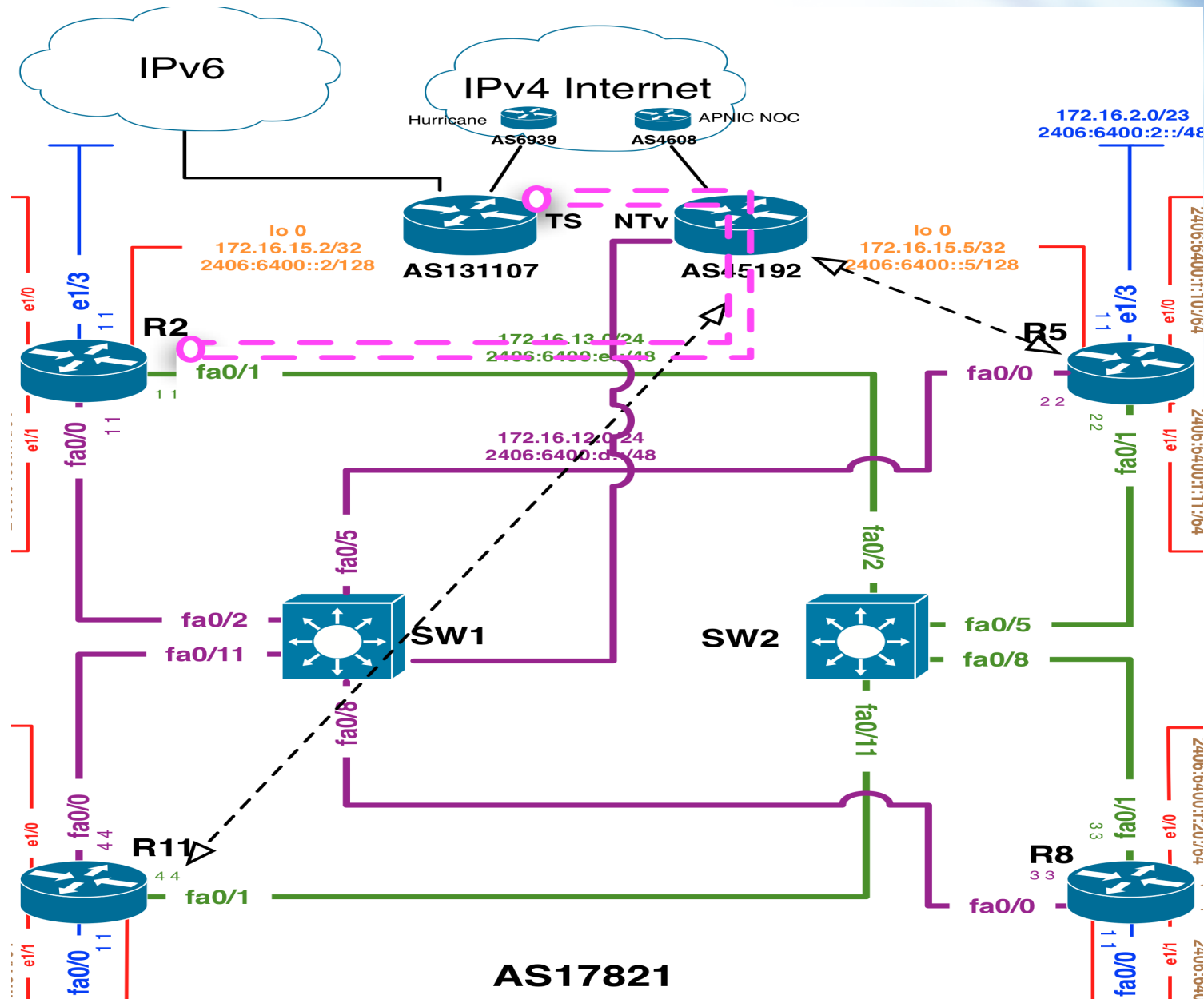
```
exit
```

```
exit
```

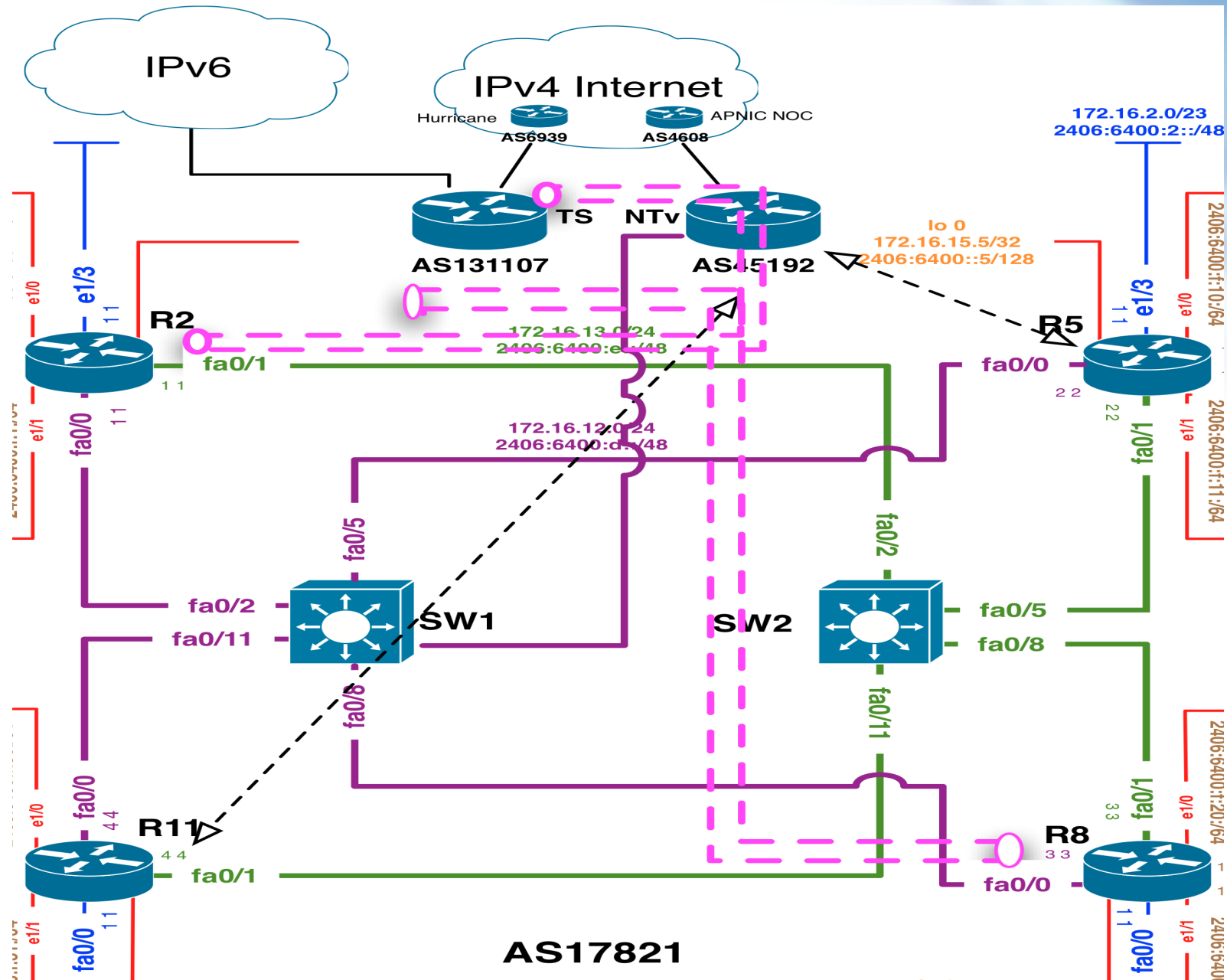
```
clear bgp ipv6 unicast 2406:6400:000D:0000::5 soft in
```

```
clear bgp ipv6 unicast 2406:6400:000D:0000::5 soft out
```

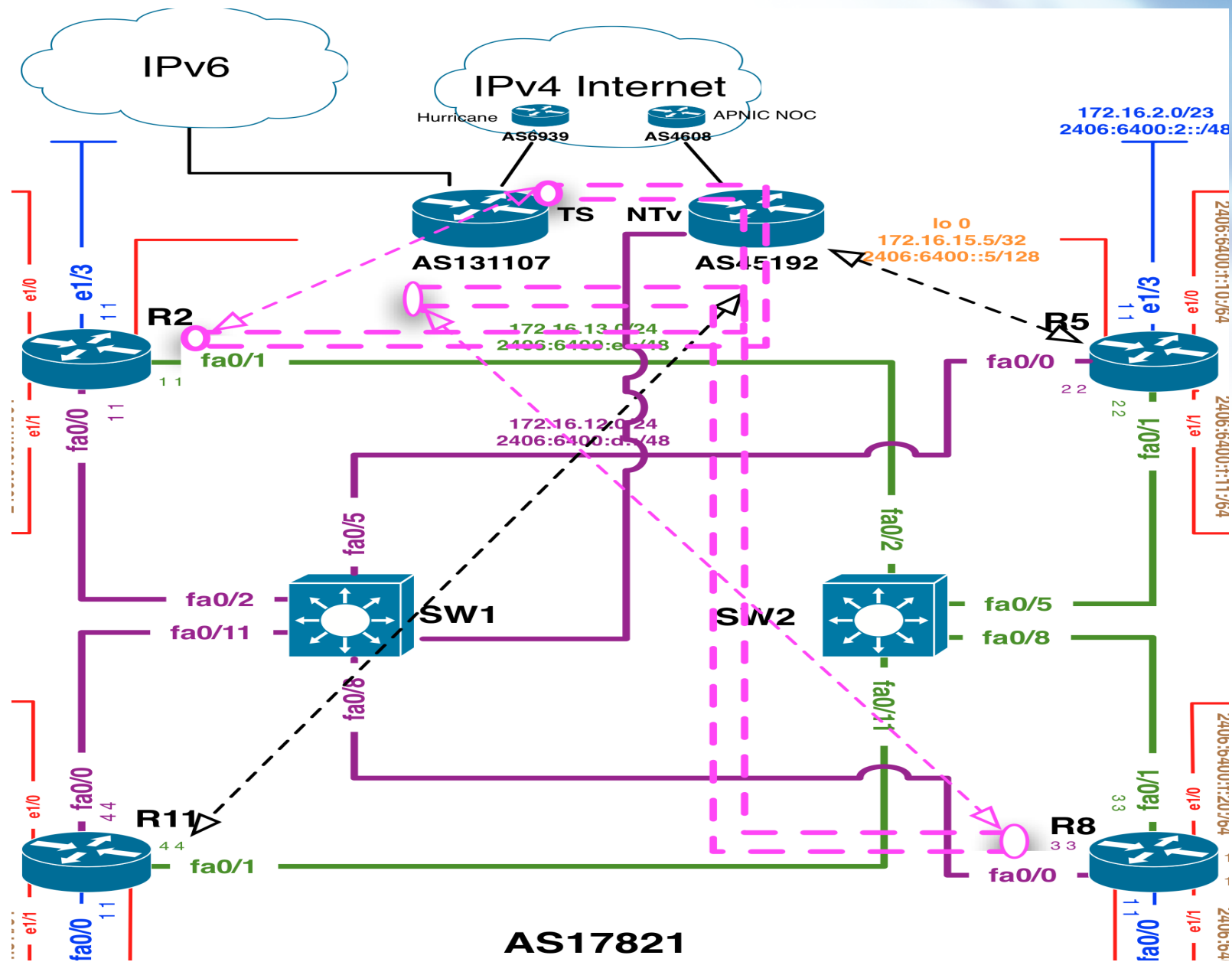
# IPv6 Tunnel Transit Conf Plan



# IPv6 Tunnel Transit Configuration



# IPv6 Tunnel Transit Configuration



# 6 to 4 Tunnel Configuration

## IOS Command for Tunnel Interface:

Router2

```
config t
```

```
interface Tunnel0
```

```
tunnel source 172.16.12.1
```

```
tunnel destination 192.168.1.1
```

```
tunnel mode ipv6ip
```

```
ipv6 address 2406:6400:F:40::2/64
```

```
ipv6 enable
```

# 6 to 4 Tunnel Configuration

## IOS Command for Tunnel Peering:

```
router bgp 17821  
address-family ipv6  
neighbor 2406:6400:F:40::1 remote-as 23456  
neighbor 2406:6400:F:40::1 activate
```



# Controlling IPv6 Route Aggregation

## IPv6 prefix filter configuration Tunnel Transit:

```
config t
```

```
ipv6 prefix-list IPV6-GLOBAL-IN seq 5 permit ::/0 ge 32 le 32
```

```
ipv6 prefix-list IPV6-GLOBAL-IN seq 10 permit ::/0 ge 48 le 48
```

```
!
```

```
ipv6 prefix-list IPV6-GLOBAL-OUT seq 5 permit ::/0 ge 32 le 32
```

```
ipv6 prefix-list IPV6-GLOBAL-OUT seq 10 permit ::/0 ge 48 le 48
```

```
router bgp 17821
```

```
address-family ipv6
```

```
neighbor 2406:6400:F:40::1 prefix-list IPV6-GLOBAL-IN in
```

```
neighbor 2406:6400:F:40::1 prefix-list IPV6-GLOBAL-OUT out
```

```
exit
```

```
exit
```

```
exit
```

```
clear bgp ipv6 unicast 2406:6400:F:40::1 soft in
```

```
clear bgp ipv6 unicast 2406:6400:F:40::1 soft out
```



# AS Numbers

- Two Ranges:
  - [0 – 65535] are the original 16 bit
  - [65536 – 4294967295] are the new 32 bit
- Usages
  - 0 and 65535 Reserved
  - 1 to 64495 Public Internet
  - 64496 to 64511 Documentation –RFC5398
  - 64512 to 65534 Private use
  - 23456 represent 32 Bit range in 16 bit world
  - 65536 to 65551 Documentation – RFC 5398
  - 65552 to 4294967295 Public Internet

# 32 bit AS number representation

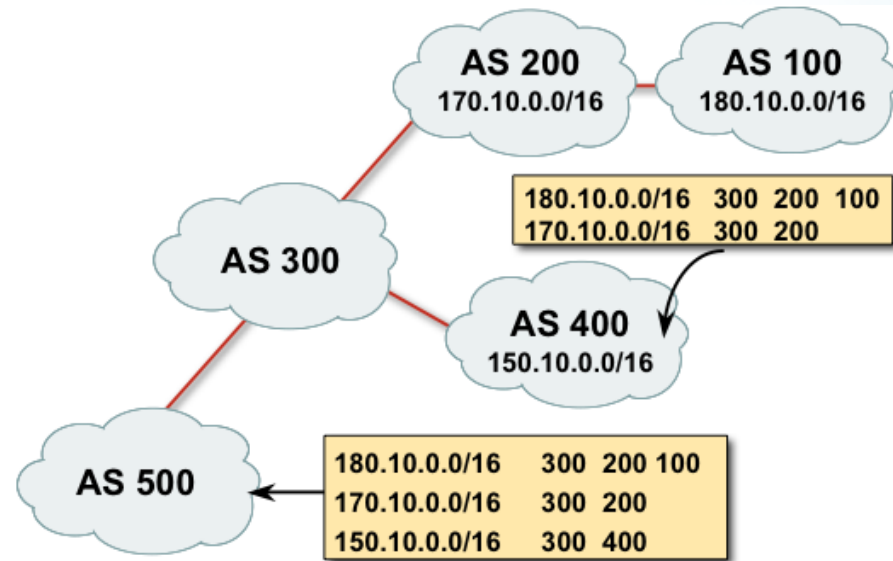
- AS DOT
  - Based upon 2-Byte AS representation
  - <Higher2bytes in decimal> . <Lower2bytes in decimal>
  - For example: AS 65546 is represented as 1.10
  - Easy to read, however hard for regular expressions
  - There is a meta character “.” in regular expression
  - i.e For example, a.c matches "abc", etc., but [a.c] matches only "a", ".", or "c".
- AS PLAIN
  - ASPLAIN IETF preferred notation
  - Continuation on how a 2-Byte AS number has been represented historically
  - Notation: The 32 bit binary AS number is translated into a Single decimal value Example: AS 65546
  - Total AS Plain range (0 – 65535 - 65,536 - 4,294,967,295)

## 4 Byte AS number

- 32 Bit range representation specified in RFC5396
- APNIC resource range:
  - In AS DOT: 2.0 ~ 2.1023
  - In AS PLAIN: 131072 ~ 132095
- AS number converter

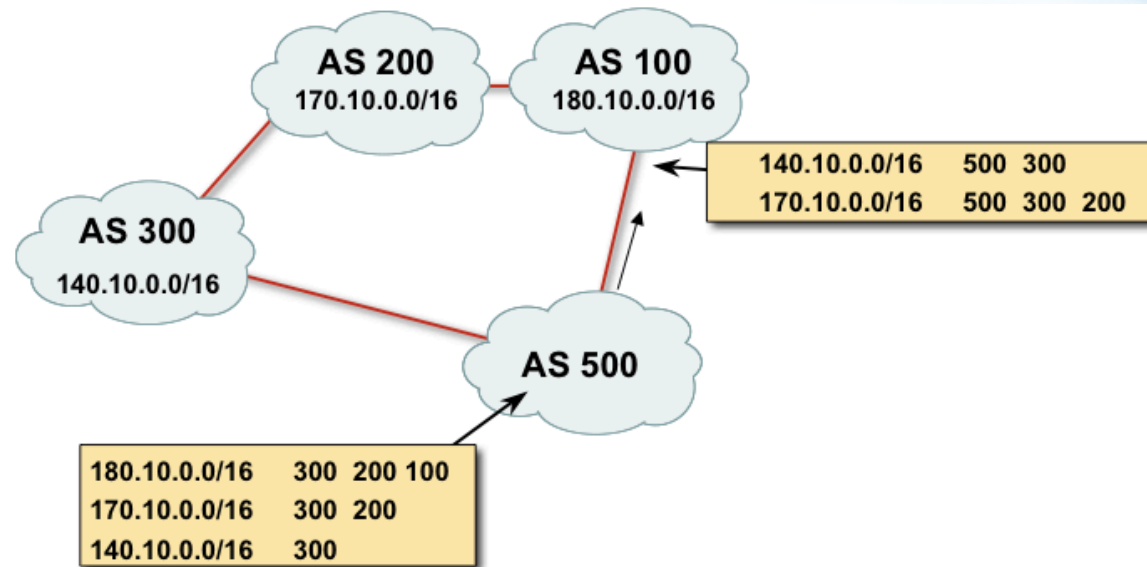
<http://submit.apnic.net/cgi-bin/convert-asn.pl>

# AS Path Attribute



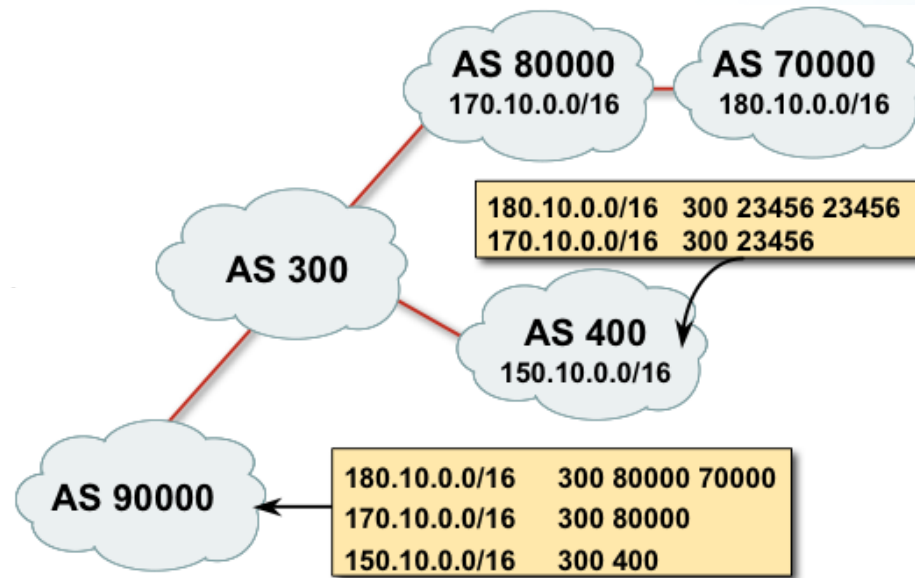
- Sequence of ASes a route has traversed
- Used for
  - Loop detection
  - Path metrics where the length of the AS Path is used as in path selection

# AS Path Loop Detection



- 180.10.0.0/16 is not accepted by AS100 as the prefix has AS100 in its AS-PATH
- This is loop detection in action

# AS Path Attribute (2 byte and 4 byte)

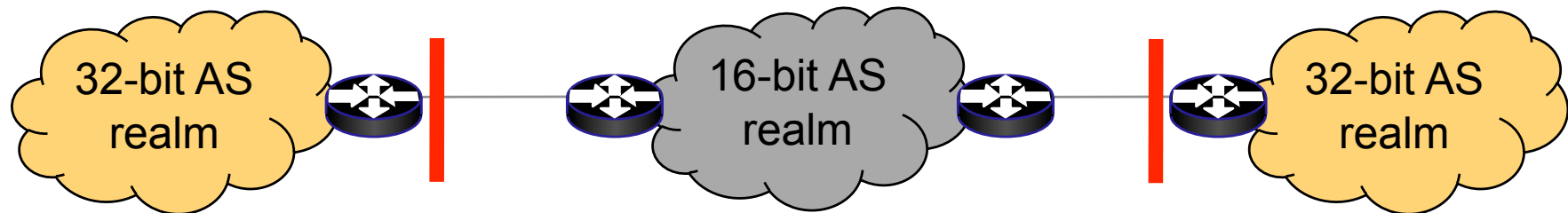


- Internet with 16-bit and 32-bit ASNs
  - 32-bit ASNs are 65536 and above
  - AS-PATH length maintained



# 32-bit AS Transition

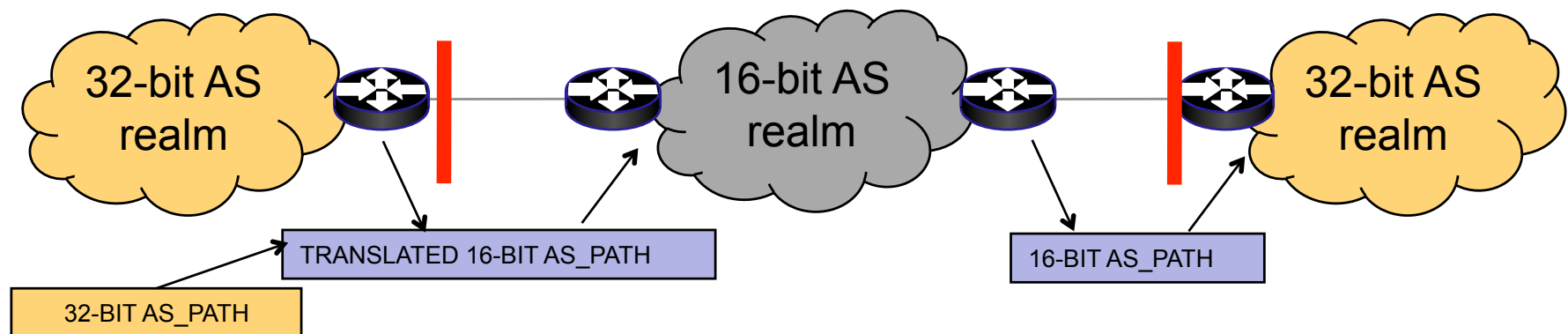
- Think about this space as a set of NEW / OLD boundaries
- Define the NEW / OLD and the OLD / NEW transitions
- Preserve all BGP information at the transition interfaces
  - **Translate** 32-bit AS Path information into a 16-bit representation
  - **Tunnel** 32-bit AS Path information through 16-bit AS domain as an update attribute





# 32-bit AS Transition

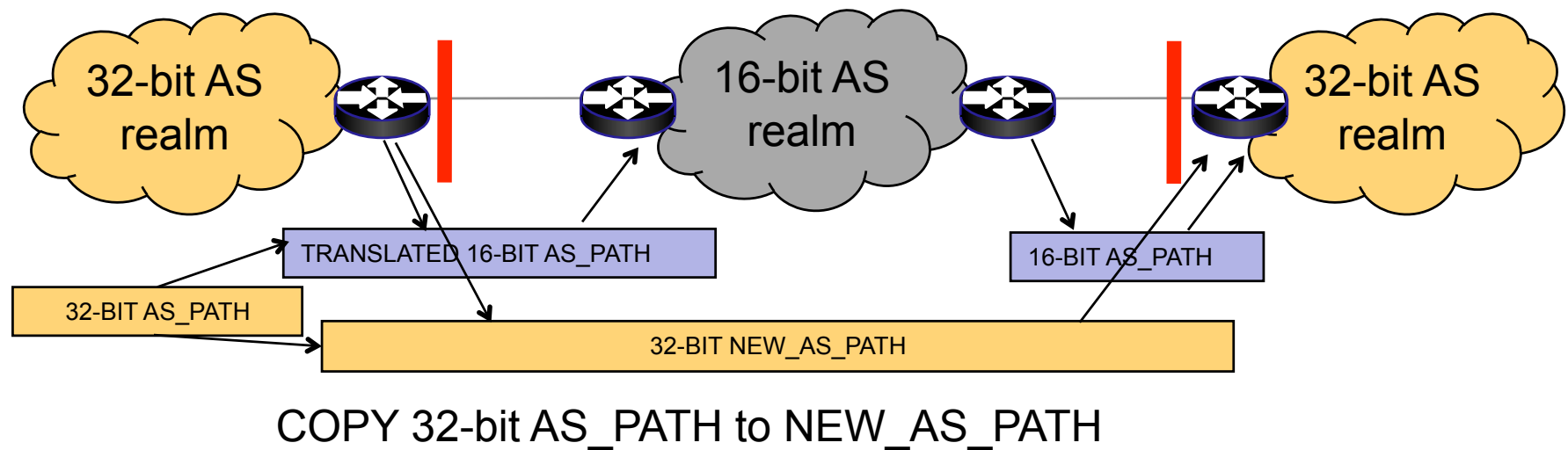
- Think about this space as a set of NEW / OLD boundaries
- Define the NEW / OLD and the OLD / NEW transitions
- Preserve all BGP information at the transition interfaces
  - **Translate** 32-bit AS Path information into a 16-bit representation
  - **Tunnel** 32-bit AS Path information through 16-bit AS domain as an update attribute



TRANSLATE all 32-bit-only AS numbers to **AS23456**

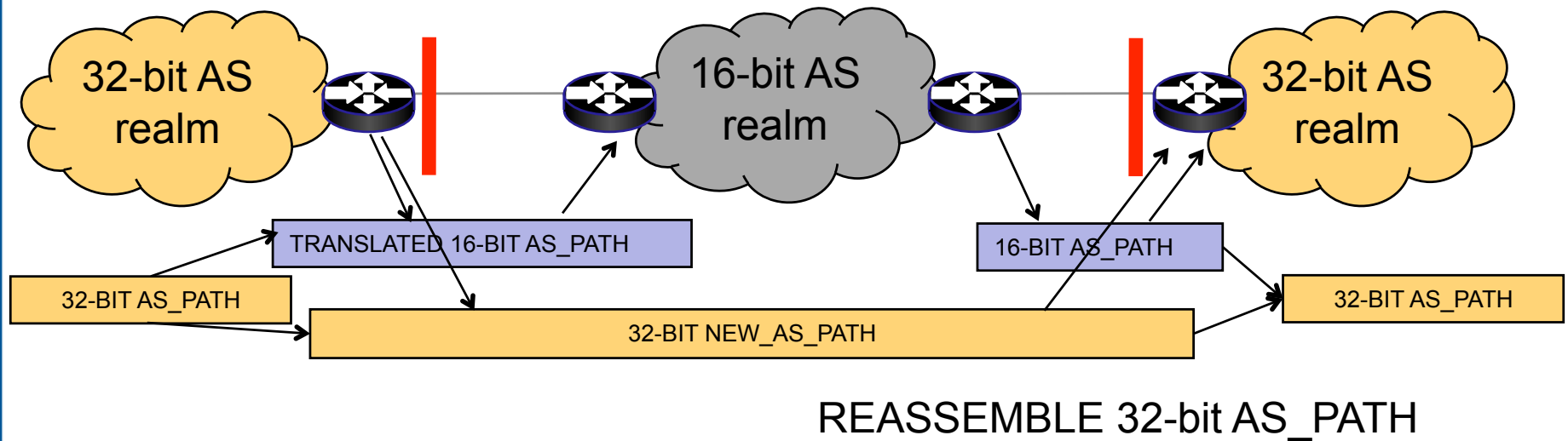
# 32-bit AS Transition

- Think about this space as a set of NEW / OLD boundaries
- Define the NEW / OLD and the OLD / NEW transitions
- Preserve all BGP information at the transition interfaces
  - **Translate** 32-bit AS Path information into a 16-bit representation
  - **Tunnel** 32-bit AS Path information through 16-bit AS domain as an update attribute

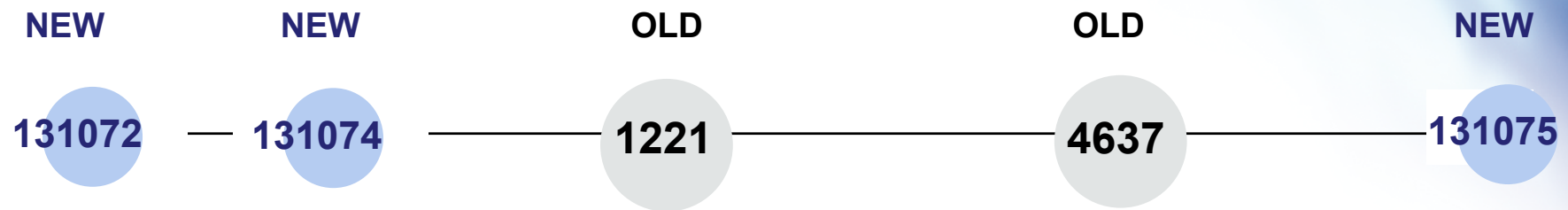


# 32-bit AS Transition

- Think about this space as a set of NEW / OLD boundaries
- Define the NEW / OLD and the OLD / NEW transitions
- Preserve all BGP information at the transition interfaces
  - **Translate** 32-bit AS Path information into a 16-bit representation
  - **Tunnel** 32-bit AS Path information through 16-bit AS domain as an update attribute



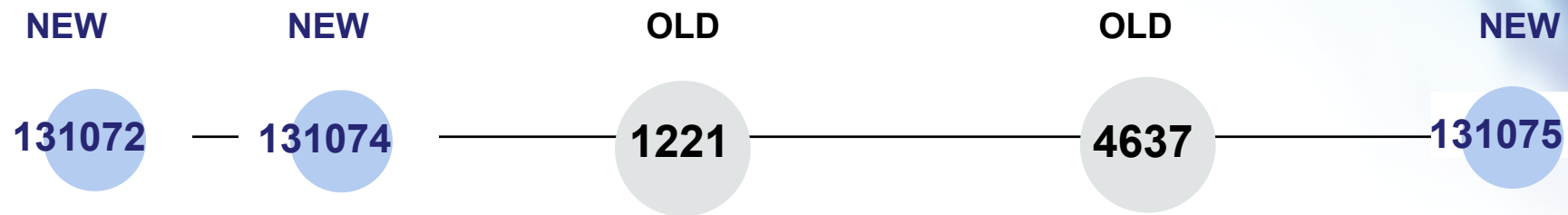
# 32-bit / 16-bit BGP Example



**AS Path** in the RIB



# 32-bit / 16-bit BGP Example



**AS Path** in the RIB

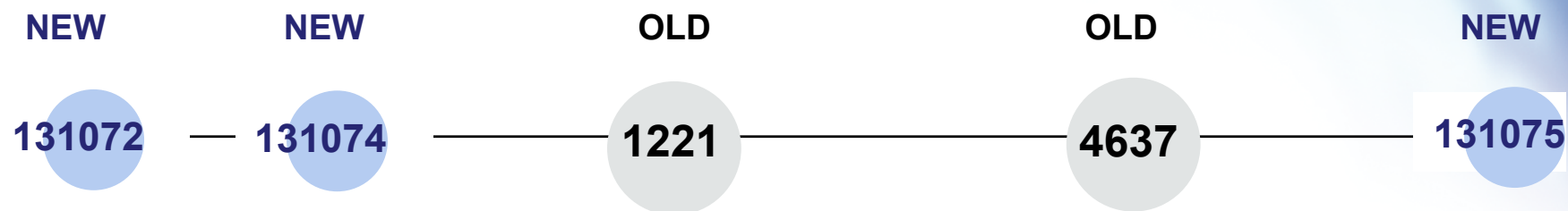
i



**AS Path Attribute** in the UPDATE Message

131072

# 32-bit / 16-bit BGP Example



**AS Path** in the RIB

i

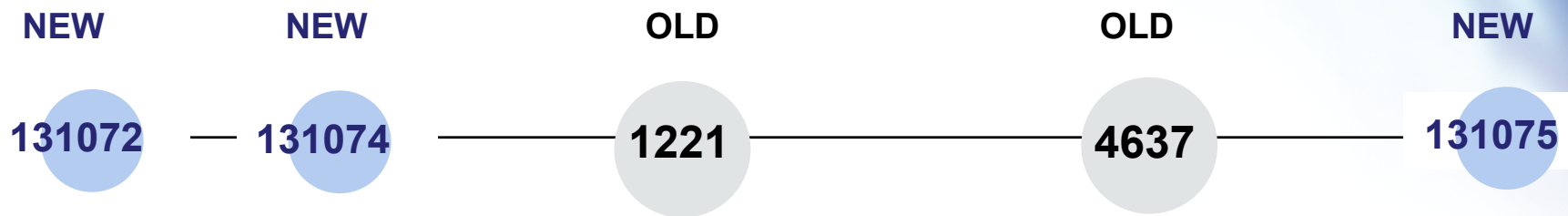
131072



**AS Path Attribute** in the UPDATE Message

131072

# 32-bit / 16-bit BGP Example



**AS Path** in the RIB

i

131072



**AS Path Attribute** in the UPDATE Message

131072

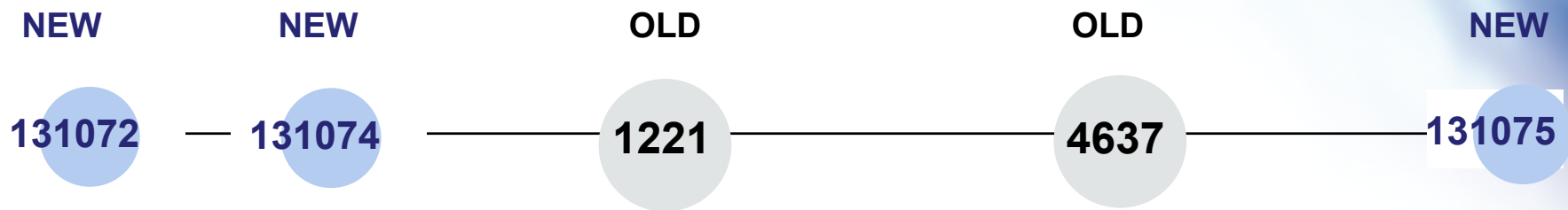
23456 23456

**NEW\_AS\_PATH Attribute** in the UPDATE Message

131074 131072



# 32-bit / 16-bit BGP Example



**AS Path** in the RIB

i

131072

23456 23456

**AS Path Attribute** in the UPDATE Message

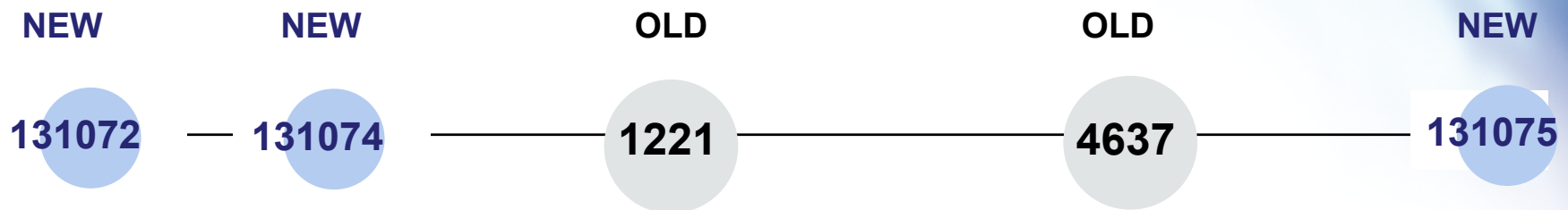
131072

23456 23456

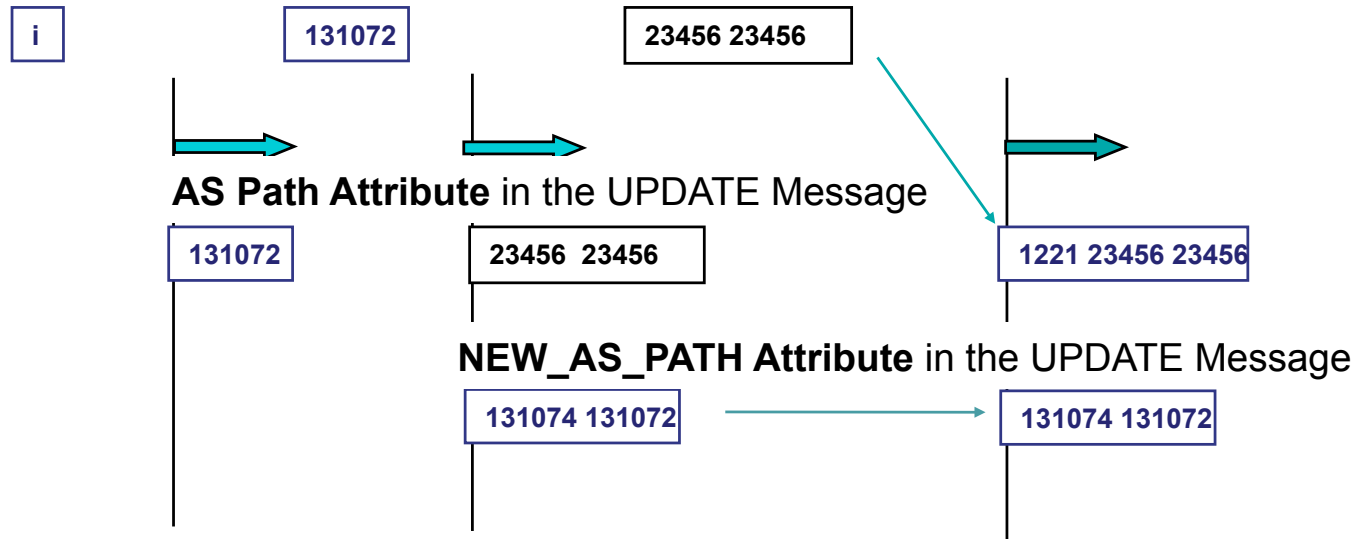
**NEW\_AS\_PATH** Attribute in the UPDATE Message

131074 131072

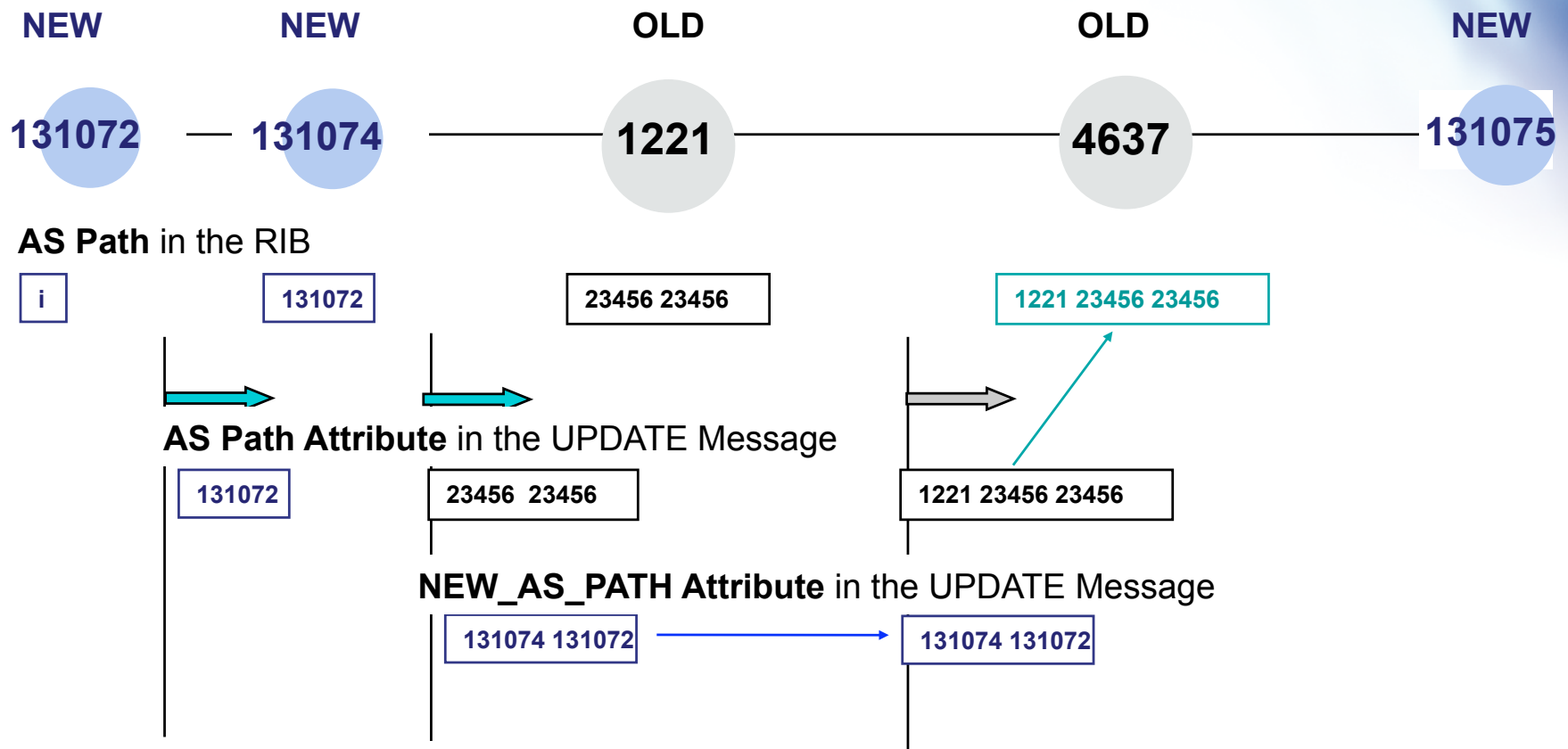
# 32-bit / 16-bit BGP Example



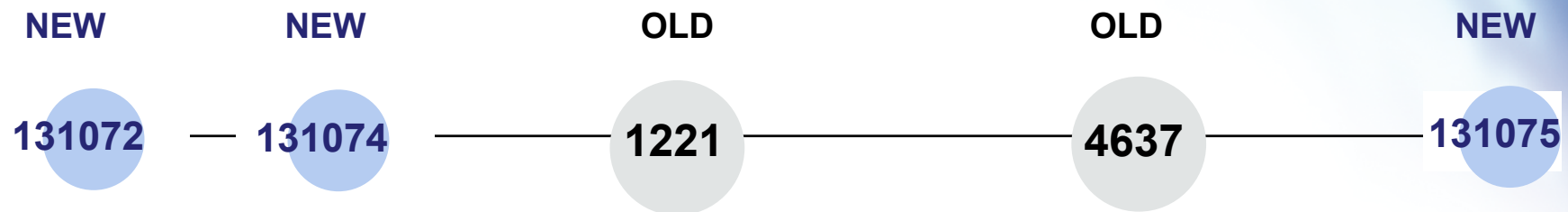
**AS Path** in the RIB



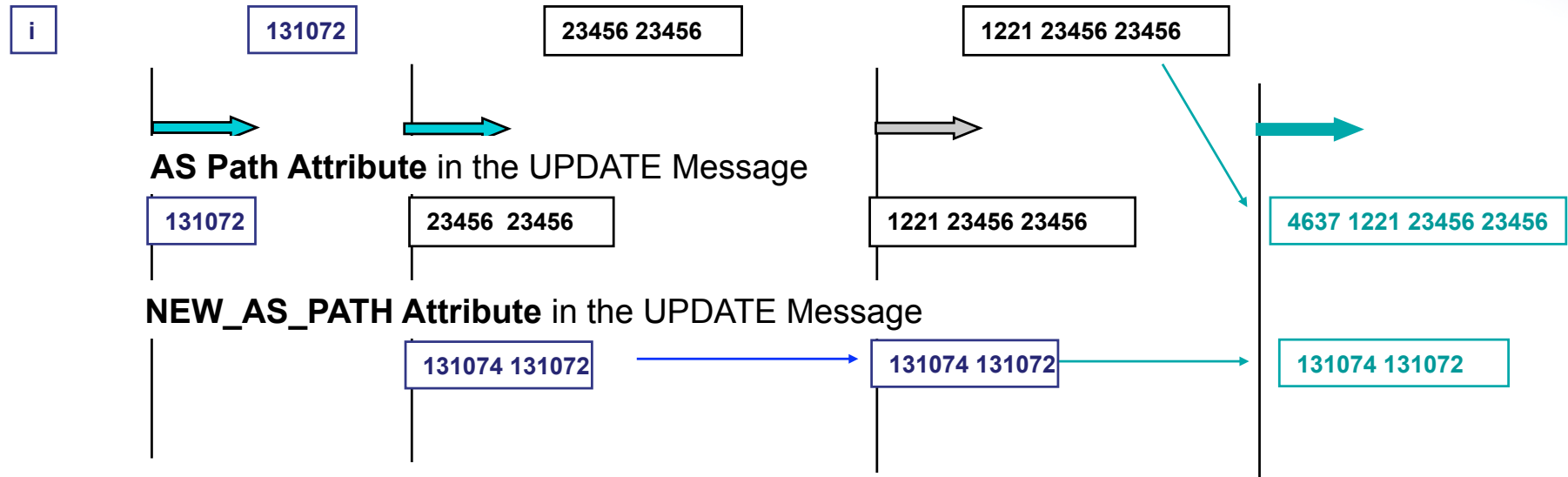
# 32-bit / 16-bit BGP Example



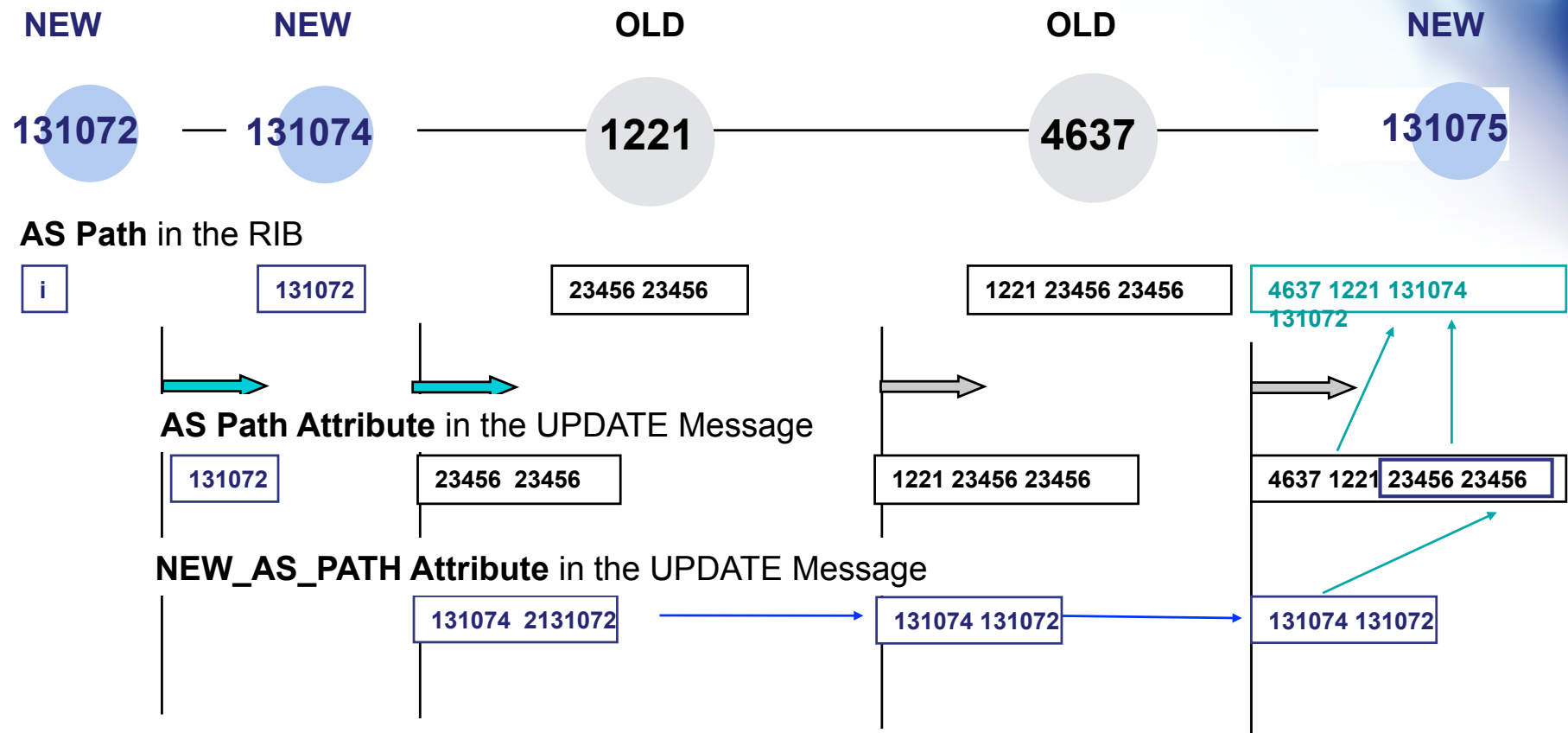
# 32-bit / 16-bit BGP Example



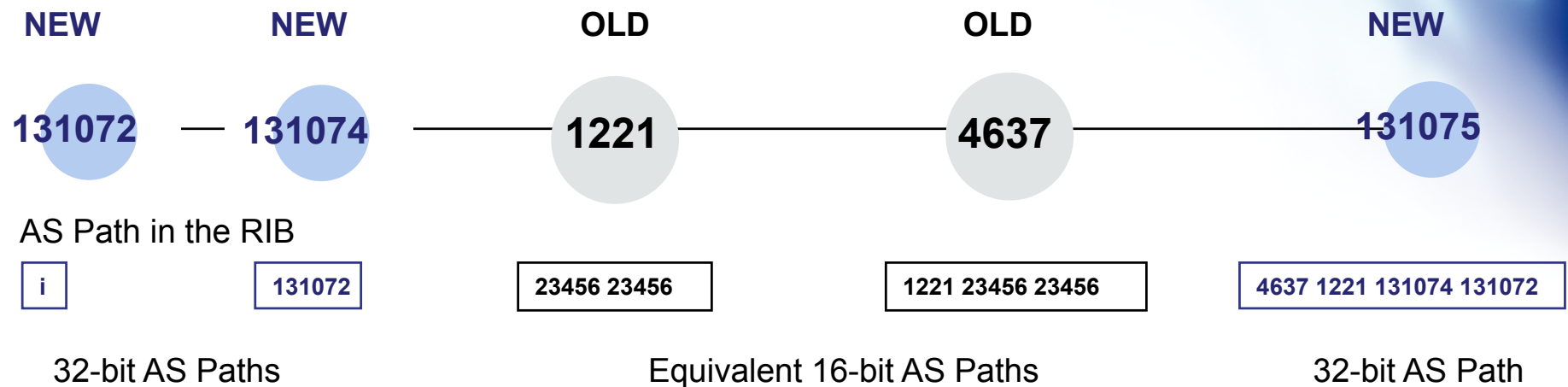
**AS Path in the RIB**



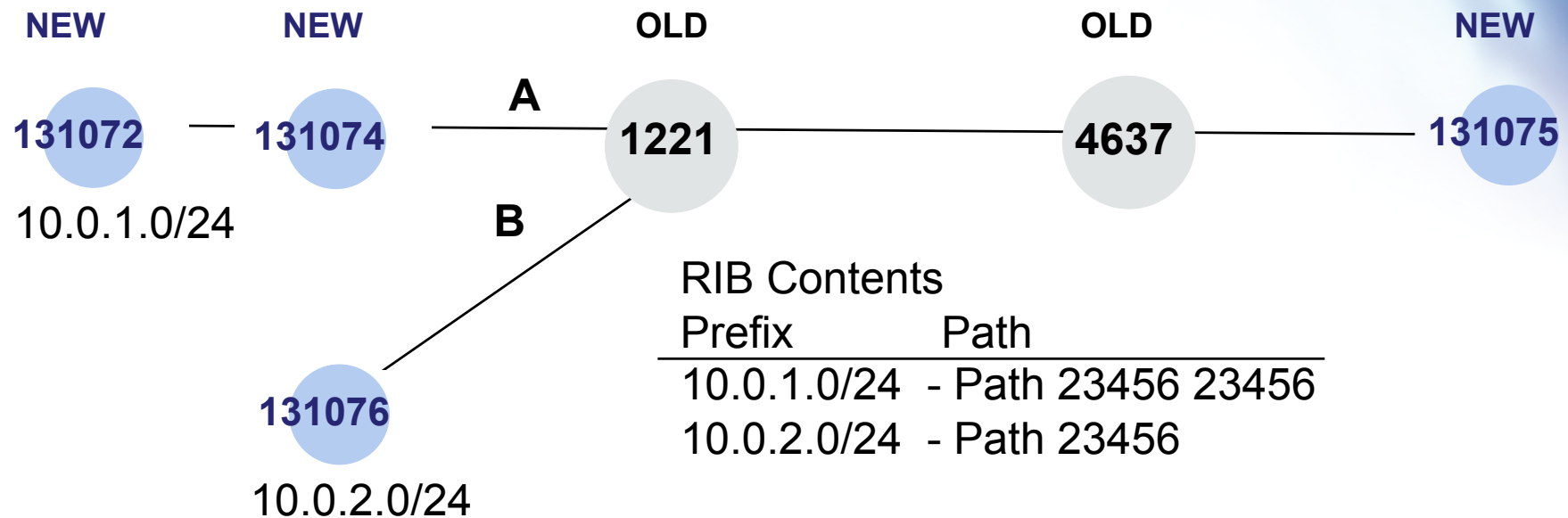
# 32-bit / 16-bit BGP Example



# 32-bit / 16-bit BGP Example



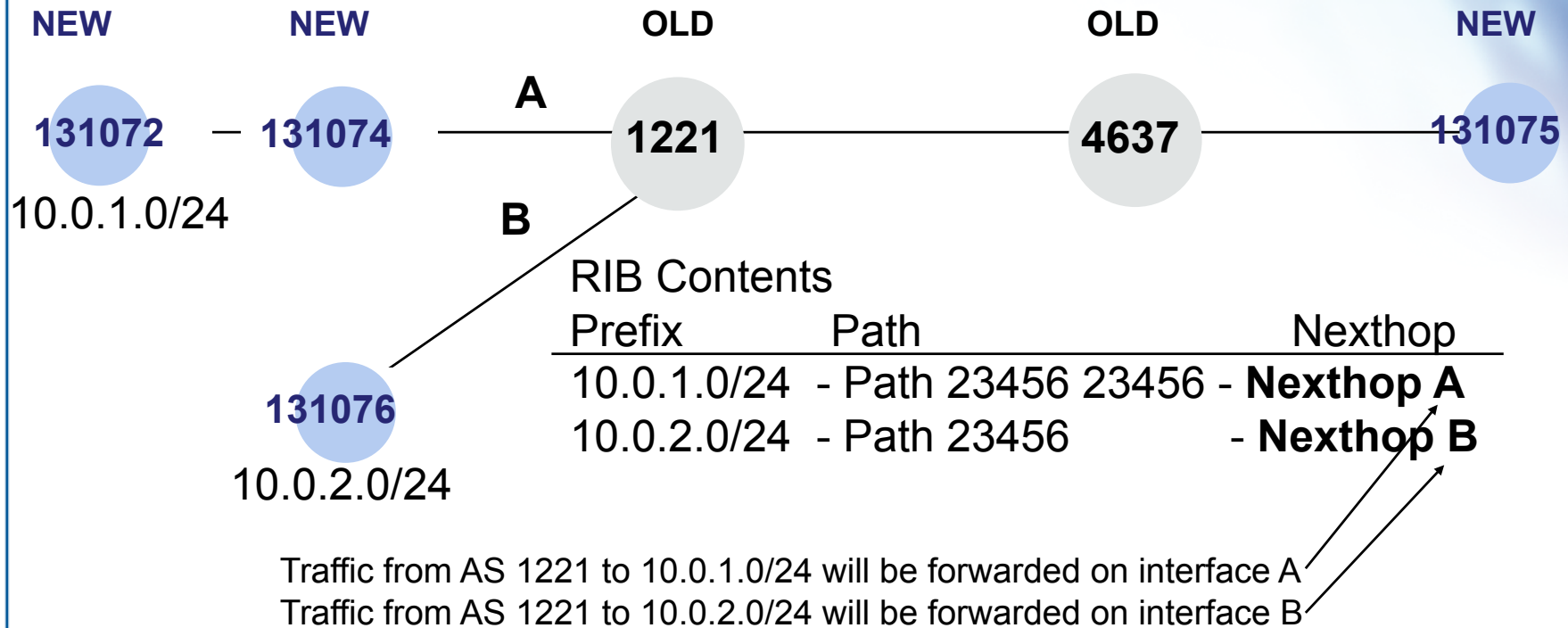
# Can old-BGP get Confused?







# NO! BGP Nexthop is the key!



**This is standard BGP behaviour – nothing changes here for BGP as it is used today**

# NEW\_AS\_PATH Attribute

- BGP speakers in 16-bit AS domains should support **NEW\_AS\_PATH** as a transitive optional attribute in UPDATE messages
  - because that's where the 32-bit path is hiding
  - That's a "SHOULD" not a "MUST",
  - Its better if you do, but nothing fatally breaks if you don't
  - Mixed 2 / 4 Byte loops will get detected in the 16-bit world as a fallback
- Default BGP configurations will do the right thing here

# NEW\_AGGREGATOR Attribute

- BGP speakers in 16-bit AS domains should support **NEW\_AGGREGATOR** as a transitive optional attribute in UPDATE messages
  - because that's where the 32-bit Aggregator AS is hiding
  - That's a "SHOULD" not a "MUST", by the way
  - Its better if you do, but nothing fatally breaks if you don't
- Default BGP configurations should do the right thing here

# AS 23456

- **AS 23456** is going to appear in many 16-bit AS paths – both origin and transit
- This is not an error – it's a 16-bit token holder for a 32-bit AS number

# AS Path and AS4 Path Example

## Router5:

*Network Next Hop Metric LocPrf Weight Path*

*\*> 2001::/32 2406:6400:F:41::1*

*0 23456 38610 6939 i*

*\* i 2406:6400:D::5 0 100 0 45192 4608 4826 6939 i*

*\*> 2001:200::/32 2406:6400:F:41::1*

*0 23456 38610 6939 2500 i*

*\* i 2406:6400:D::5 0 100 0 45192 4608 4826 6939  
2500 i*

# Questions?



# Overview

## IPv6 Deployment Tutorial

- IPv6 Protocol Architecture Overview
- IPV6 Addressing and Sub-netting
- IPv6 Host Configuration
- Training ISP Network Topology Overview
- Deployment of IPV6 in Interior Gateway
- IPv4 to IPv6 Transition technologies
- Planning & Implementation of IPv6 on Exterior Gateway (BGP)
- **Connecting ISP network to an IXP**



# Case study- IXP Configuration

Two type of traffic exchange between ISPs

- Transit
  - Where ISP will pay to send/receive traffic
  - Downstream ISP will pay upstream ISP for transit service
- Peering
  - ISPs will not pay each other to interchange traffic
  - Works well if win win for both
  - Reduce cost on expensive transit link

# IX Peering Model

- **BLPA (Bi-Lateral Peering Agreement)**
  - IX will only provide layer two connection/switch port to ISPs
  - Every ISPs will arrange necessary peering arrangement with others by their mutual business understanding.
- **MLPA (Multi-Lateral Peering Agreement)**
  - IX will provide layer two connection/switch port to ISPs
  - Each ISP will peer with a **route server** on the IX.
  - Route server will collect and distribute directly connected routes to every peers.

# IXP Peering Policy

- BLPA is applicable where different categories of ISPs are connected in an IX
  - Large ISPs can choose to peer with large ISPs (base on their traffic volume)
  - Small ISPs will arrange peering with small ISPs
- Would be preferable for large ISPs
  - They will peer with selected large ISPs (Equal traffic interchange)
  - Will not loose business by peering with small ISP

# IX Peering Policy

- MLPA model works well to widen the IX scope of operation (i.e national IX).
- Easy to manage peering
  - Peer with the **route server** and get all available local routes.
  - Do not need to arrange peering with every ISPs connected to the IX.
- Unequal traffic condition can create not intersected situation to peer with route server

# IX peering Policy

- Both peering model can be available in an IX.
- Member will select peering model i.e either BLPA or MLPA (Route Server Peering)
- IX will provide switch port
- **Mandatory MLPA** model some time not preferred by large ISP (Business Interest)
  - Can create not interested situation to connect to an IX

# IX Operating Cost

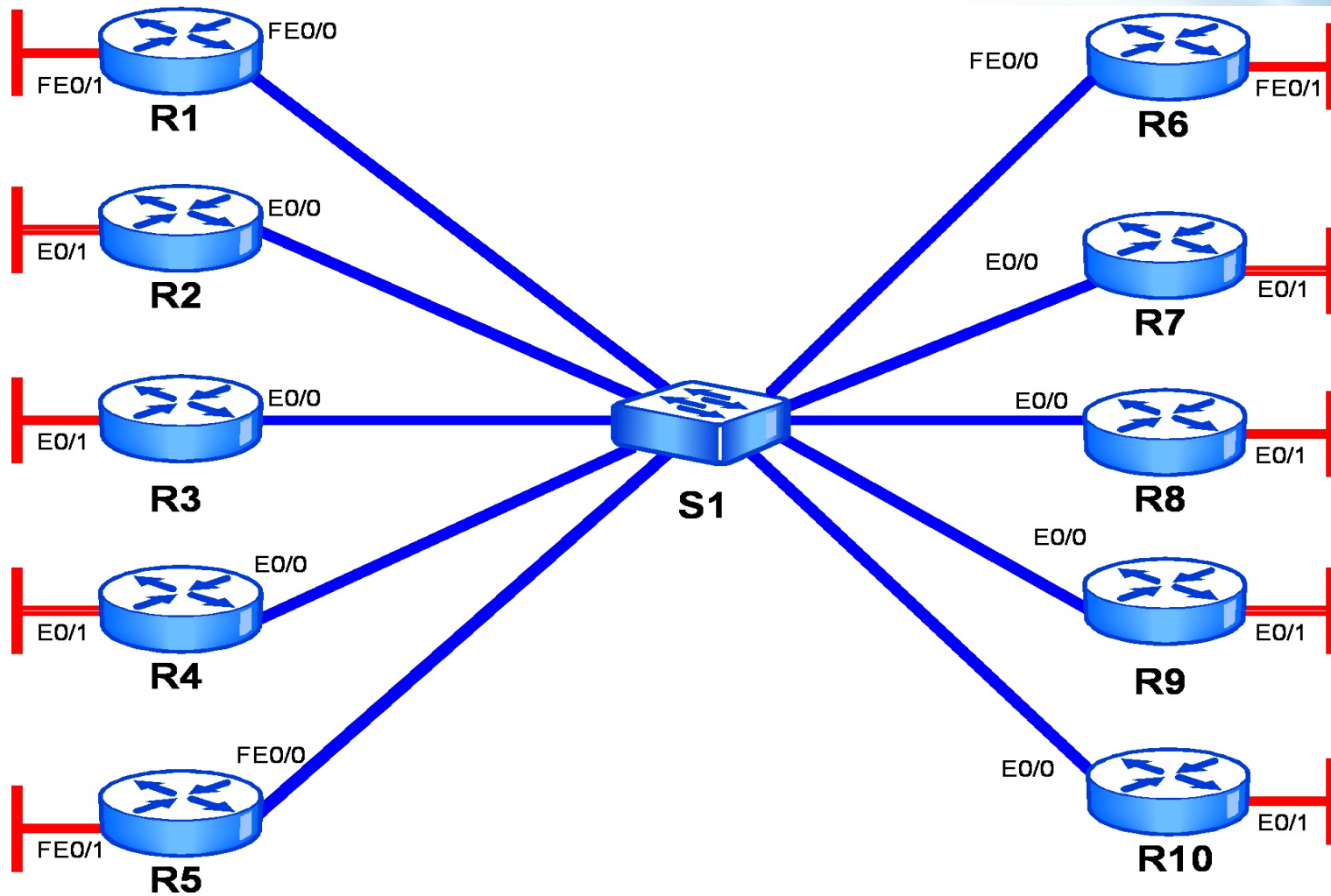
- Access link
- Link maintenance
- Utility
- Administration

# Cost Model

- Not for profit
- Cost sharing
- Membership based
- Commercial IX



# IX Network Diagram





# Case study- IXP Configuration

Required **global & interface** commands to enable IPv6

Router(Config)#ipv6 unicast-routing

Router(Config)#ipv6 cef (optional)

- Configure IPv6 address on interface

Router(Config-if)#ipv6 address *2001:0df0:00aa::1/64*

Router(Config-if)#ipv6 enable

- Verify IPv6 configuration

Router#sh ipv6 interface fa0/0

- Verify connectivity

Router#ping *2001:0df0:00aa::1*

# Case study- IXP Configuration

- Required **BGP** commands to enable IPv6 routing

```
Router(config)# router bgp 1
```

```
Router2(config-router)#bgp router-id 10.0.0.1 (if no 32 bit address on any interface)
```

```
Router(config-router)# address-family ipv6
```

```
Router(config-router-af)# no synchronization
```

```
Router(config-router-af)# neighbor 2001:0df0:00aa::1 remote-as 2 (EBGP)
```

```
Router(config-router-af)#neighbor 2001:0df0:00aa::1 activate
```

```
Router(config-router-af)# network 2001:0df0:00aa::/48
```

- Verify BGP IPv6 configuration

```
Router#sh bgp ipv6 unicast summary (summarized neighbor list)
```

```
Router#sh bgp ipv6 unicast (BGP database)
```

```
Router#sh ipv6 route bgp (BGP routing table)
```

# Case study- IXP Configuration

Required command to add IX prefix filter

- Create prefix filter in global mode

```
Router(config)#ipv6 prefix-list AS1 seq 2 permit 2001:0df0:aa::  
/48
```

- Apply prefix filter in BGP router configuration mode

```
Router(config-router)# address-family ipv6
```

```
Router(config-router-af)#neighbor 2001:0df0:aa::1 prefix-list AS1  
in
```

```
Router(config-router-af)#neighbor 2001:0df0:aa::1 prefix-list AS1  
out
```

# Case study- IXP Configuration

Controlling routing update traffic (Not data traffic)

1. Incoming routing update (Will control outgoing data traffic)
2. Outgoing routing update (Will control incoming data traffic)

# Questions?



# Thank you!