Showed & Shared
IPv6 Transition Mechanism CLI

Andy Chien (錢小山)

Customer Solutions Architect

Cisco Systems

Dec-7-2012

Agenda

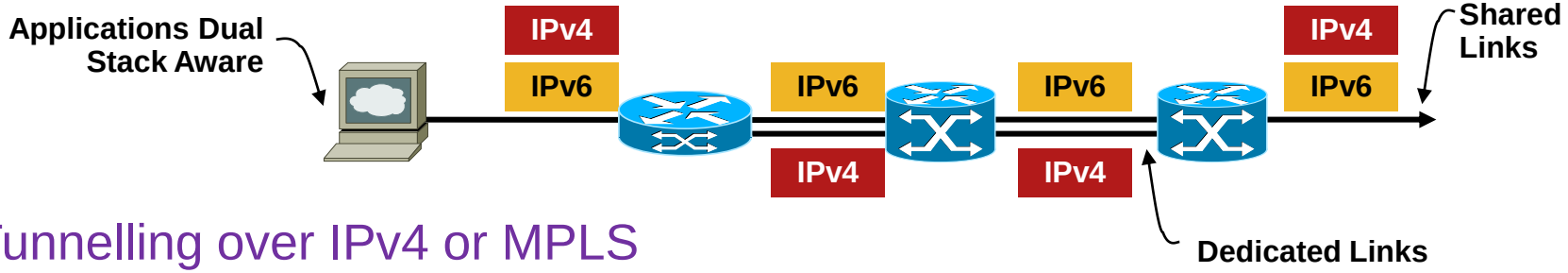
- IPv6 Deployment
  - Dual Stack
  - Tunnelling Techniques
  - MPLS Solutions
  - NAT Protocol Translation

IPv6 Deployment

IPv6 Deployment Options

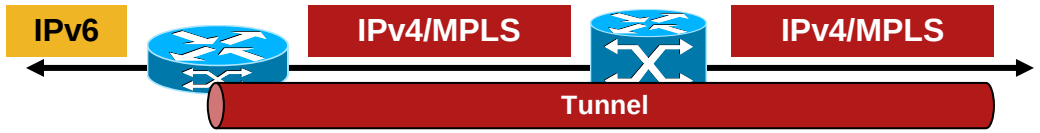
- Dual Stack (in devices/hosts and networks)

  IPv4 and IPv6 operate in tandem over shared or dedicated links



- Tunnelling over IPv4 or MPLS

  IPv6 confined to the edge of the IPv4 / MPLS core



- IPv6 Only

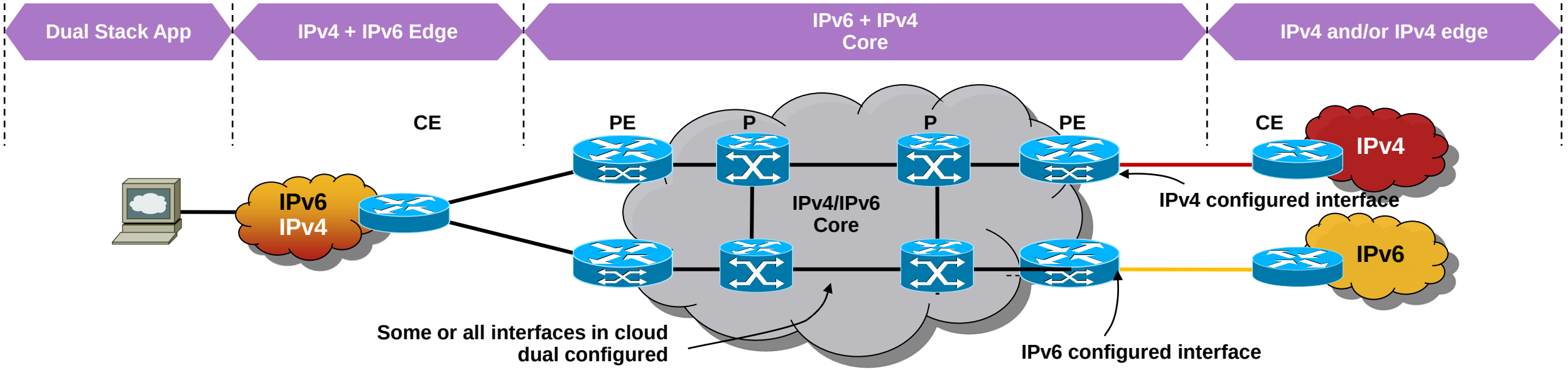
  IPv6 is the only protocol operating in the network

- 6to4 Protocol Translation (BEHAVE IETF Working Group)

  Allow IPv6-only devices to communicate with IPv4-only devices

Dual Stack Technique

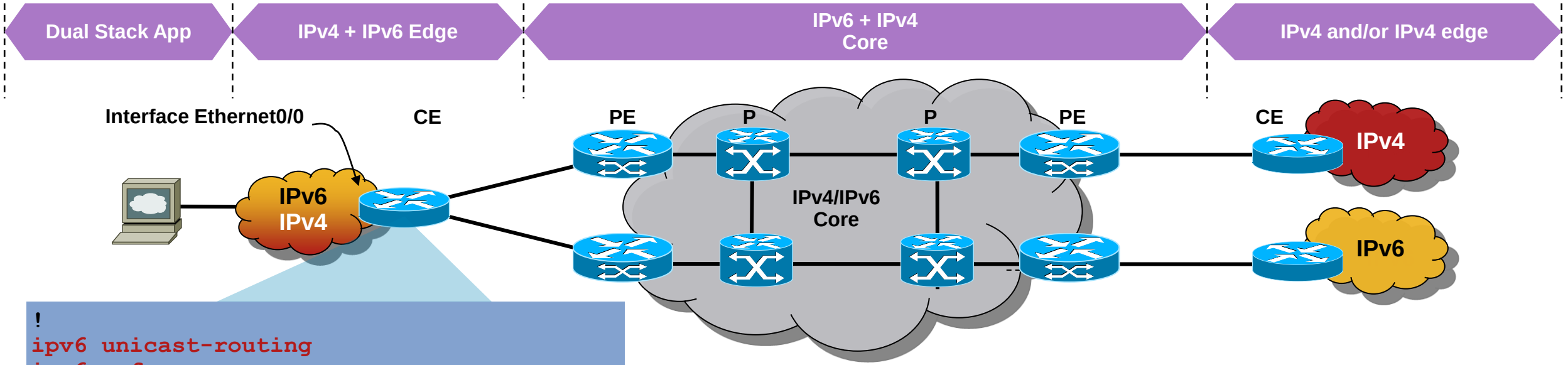
IPv6 using Dual Stack Backbone



- All P + PE routers are capable of IPv4+IPv6 support
- Two IGPs supporting IPv4 and IPv6
- Memory considerations for larger routing tables
- Native IPv6 multicast support
- All IPv6 traffic routed in global space
- Good for content distribution and global services (Internet)

Dual Stack Configuration

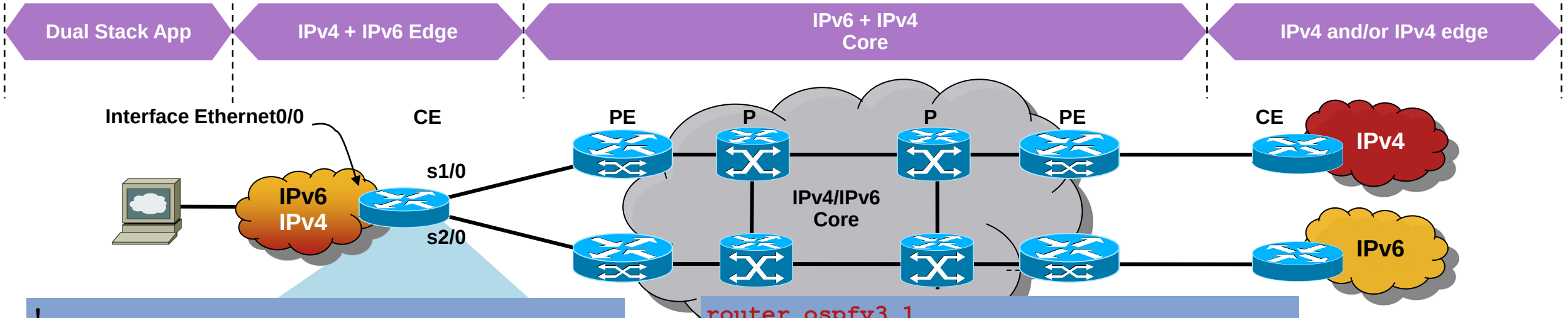
The Basics




```
!
ipv6 unicast-routing
ipv6 cef
!
interface Ethernet0/0
 ip address 192.168.99.1 255.255.255.0
 ipv6 address 2001:db8:213:1::1/64
!
```

Dual Stack Configuration

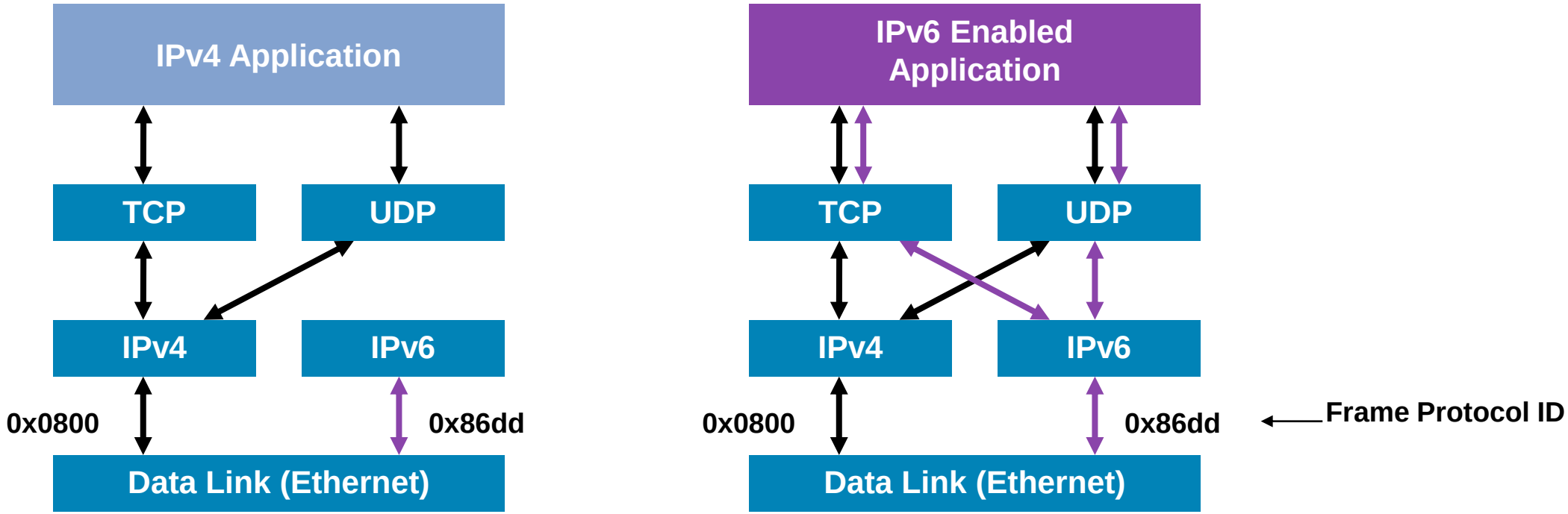
More Realistic




```
!
interface Ethernet0/0
 ip address 192.168.99.1 255.255.255.0
 ipv6 address 2001:db8:213:1::1/64
 ospfv3 1 area 0 ipv6
!
interface Serial 1/0
 ip address 192.0.2.1 255.255.255.252
 ipv6 address 2001:db8:ffff:1::1/64
!
interface Serial 2/0
 ip address 192.0.2.5 255.255.255.252
 ipv6 address 2001:db8:ffff:2::1/64
!
```

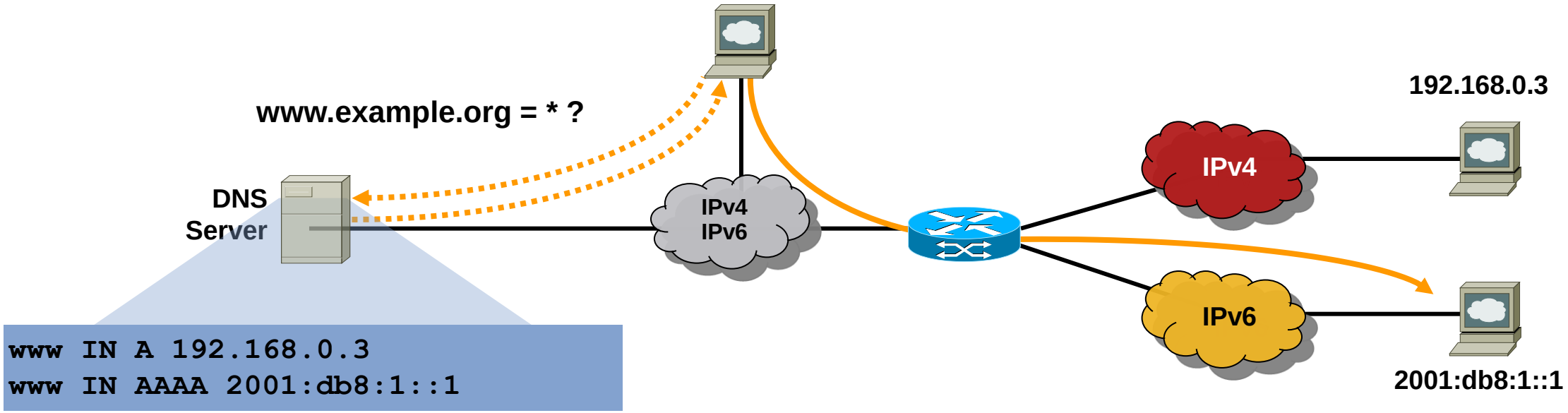
```
router ospfv3 1
 address-family ipv6 unicast
 exit-address-family
!
router bgp 65000
...
address-family ipv4
neighbor 192.0.2.2 activate
neighbor 192.0.2.6 activate
!
address-family ipv6
neighbor 2001:db8:ffff:1::2 activate
neighbor 2001:db8:ffff:2::2 activate
...
```

Application Dual Stack Approach



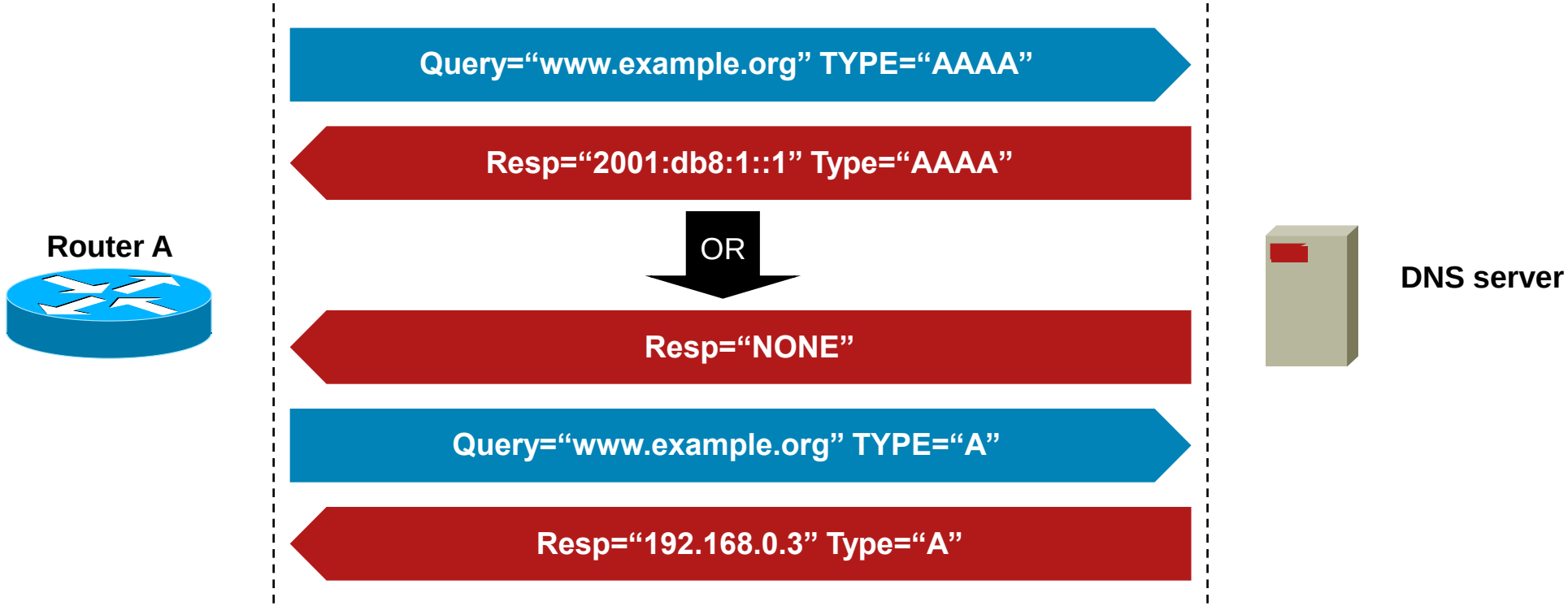
- Dual stack in a device means
  - Both IPv4 and IPv6 stacks enabled
  - Applications can talk to both
  - Choice of the IP version is based on DNS and application preference
- Dual stack at edge does not necessarily mean dual stack backbone

Dual Stack Approach & DNS



- In a dual stack network an application that is IPv4 and IPv6-enabled:
  - Can query the DNS for IPv4 records (A) and/or IPv6 (AAAA) records
  - The transport used for the lookup is not related to the resource record required. e.g. Use IPv4 transport to ask for AAAA records
  - Chooses one address and, for example, connects to the IPv6 address

DNS query in IOS



- DNS resolver picks IPv6 AAAA record first
- IPv6 stacks on Windows XP, W7, Linux, FreeBSD, MacOS etc also pick IPv6 address before IPv4 address if both exist

DNS Query on Windows 7 (Dual Stack)

mSecs between last packet sent

Domain name with IPv6 address only

mSecs	Source	Destination	Prot	Info
0.000	64.104.197.141	64.104.200.248	DNS	Standard query A ipv6.google.com
0.158	64.104.200.248	64.104.197.141	DNS	Standard query response CNAME ipv6.l.google.com
0.000	64.104.197.141	64.104.200.248	DNS	Standard query AAAA ipv6.google.com
0.135	64.104.200.248	64.104.197.141	DNS	Standard query response CNAME ipv6.l.google.com AAAA 2404:6800:8004::68

Initial Query over IPv4 for IPv4 A record

DNS response refers to an alias/canonical address

Host immediately sends a request for AAAA record (original FQDN)

IPv6 address of canonical name returned

Domain name with both addresses

mSecs	Source	Destination	Prot	Info
0.000	64.104.197.141	64.104.200.248	DNS	Standard query A www.apnic.net
0.017	64.104.200.248	64.104.197.141	DNS	Standard query response A 202.12.29.211
0.000	64.104.197.141	64.104.200.248	DNS	Standard query AAAA www.apnic.net
0.017	64.104.200.248	64.104.197.141	DNS	Standard query response AAAA 2001:dc0:2001:11::211
0.001	2001:420:1:fff:2	2001:dc0:2001:11::211	ICMPv6	Echo request (Unknown (0x00))
0.023	2001:dc0:2001:11::211	2001:420:1:fff::2	ICMPv6	Echo reply (Unknown (0x00))

Initial Query over IPv4 for IPv4 A record

IPv4 address returned

Host immediately sends a request for AAAA record

IPv6 address of FQDN returned

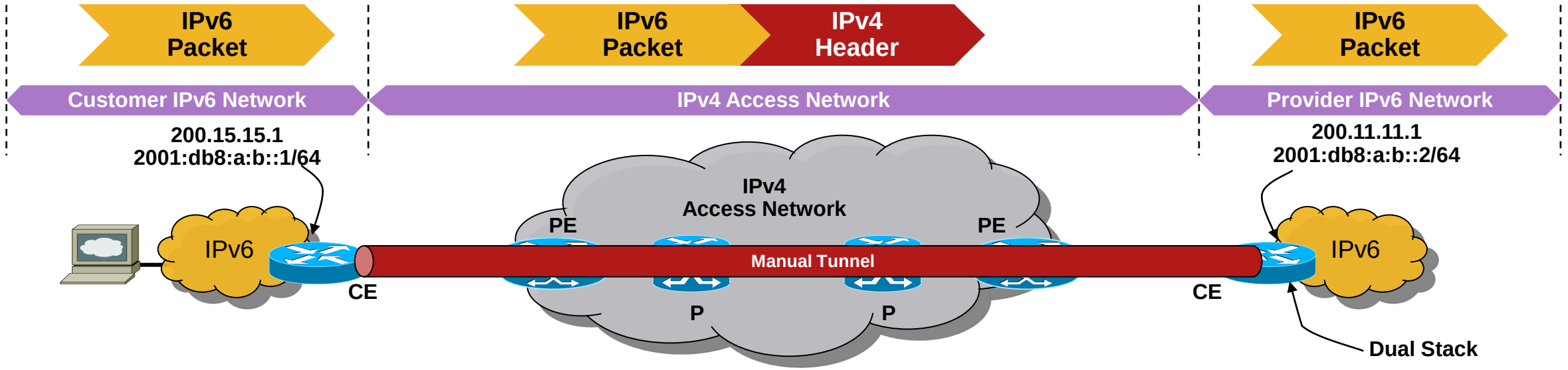
Hosts prefers IPv6 address (configurable)

Tunneling Techniques

Using Tunnels for IPv6 Deployment

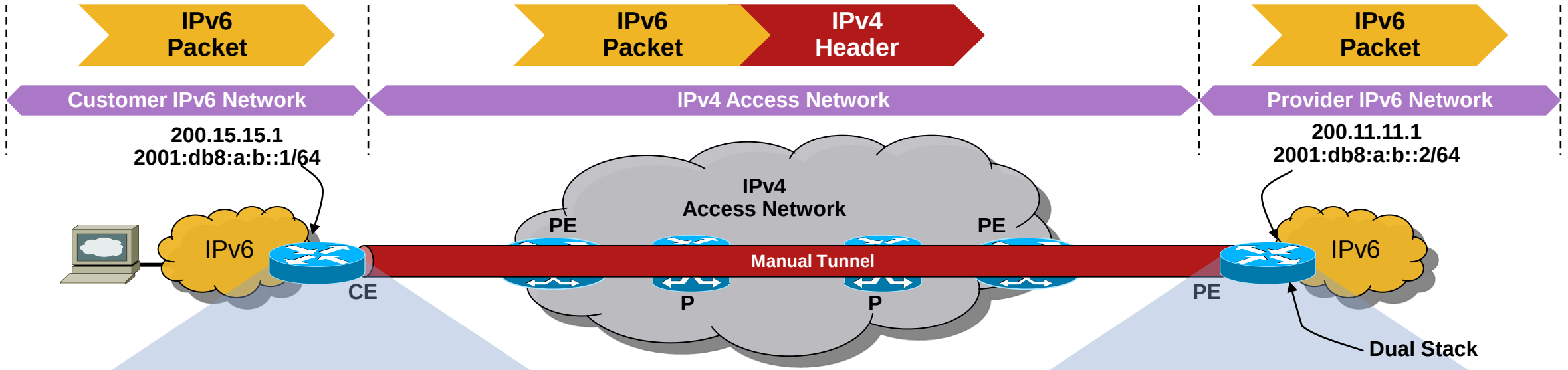
- Tunnelling encapsulates an IPv6 packet into an IPv4 packet
  - Host to Router, Router to Router
    Router to Host, Host to Host
- Manually configured tunnels
  - Manual Tunnel (RFC 2893)
  - IPv6 over GRE (RFC 2473)
- Semi-automated tunnels
  - Tunnel broker (RFC 3053)
- Automatic tunnels
  - 6to4 (RFC 3056)
  - ISATAP (RFC 5214)
  - Dynamic Multipoint VPN
  - 6rd (RFC5969)
  - LISP (IETE Working Group & Internet Draft)

Manual Tunnel (RFC 2893)



- One of the first transition mechanisms developed for IPv6
  - Static P2P tunnel, IP protocol type = 41, no additional header, NAT breaks
- Terminates on dual stack end points
  - IPv4 end point address must be routable
  - IPv6 prefix configured on tunnel interface
- Difficult to scale and manage
  - For link few sites in fixed long term topology
  - Use across IPv4 access network to reach IPv6 Provider

Manual Tunnel Configuration

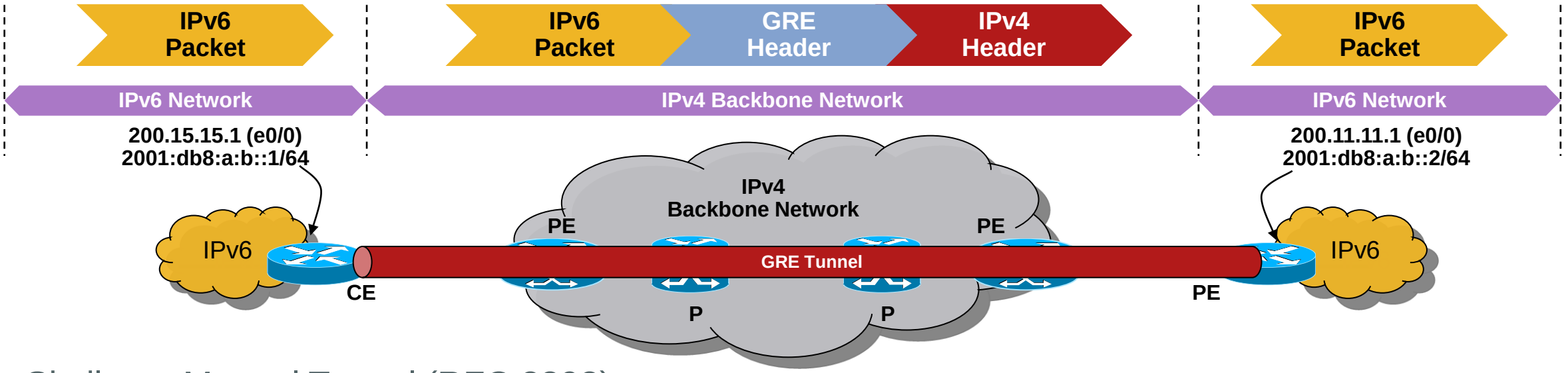



```
interface tunnel 100
 ipv6 address 2001:db8:a:b::1/64
 no ipv6 nd ra suppress
 tunnel source 200.15.15.1
 tunnel destination 200.11.11.1
 tunnel mode ipv6ip
```

```
interface tunnel 100
 ipv6 address 2001:db8:a:b::2/64
 no ipv6 nd ra suppress
 tunnel source 200.11.11.1
 tunnel destination 200.15.15.1
 tunnel mode ipv6ip
```

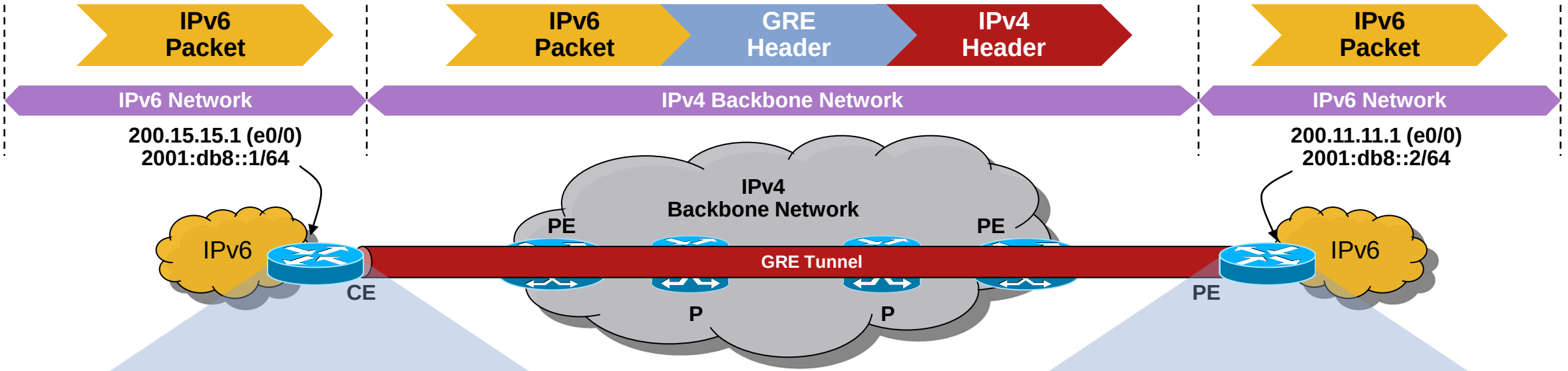
- Only supports routing protocols that use IP encapsulation
  - ISIS is itself a network layer protocol (not dependant upon IP)
  - Therefore will not work over IP Protocol-Type=41

IPv6 over GRE Tunnel



- Similar to Manual Tunnel (RFC 2893)
  - But can transport non IP packets
  - Hence can be used to support ISIS across the tunnel
- GRE header uses 0x86DD to identify IPv6 payload
- Similar scale and management issues
- L2TPv3 is another tunnelling option

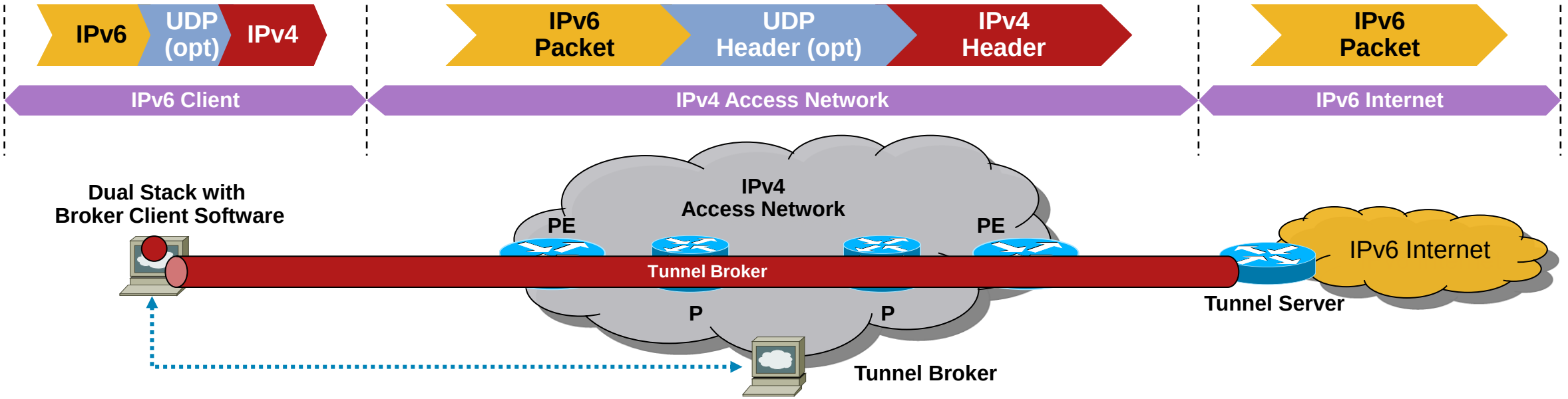
IPv6 over GRE Tunnel Configuration




```
interface tunnel 100
 ipv6 address 2001:db8:a:b::1/64
 tunnel source e0/0
 tunnel destination 200.11.11.1
 tunnel mode gre ip
 ipv6 router isis
```

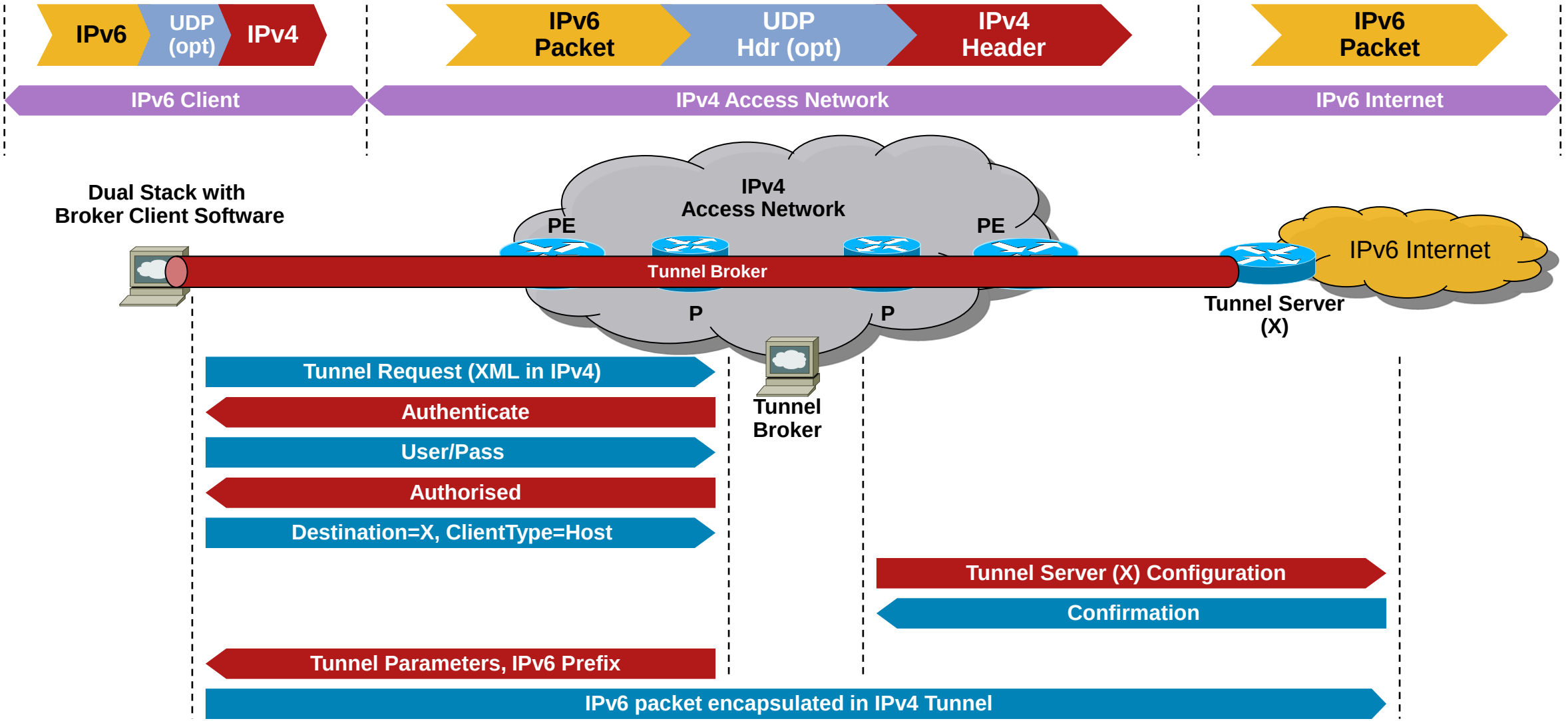
```
interface tunnel 100
 ipv6 address 2001:db8:a:b::2/64
 tunnel source e0/0
 tunnel destination 200.15.15.1
 tunnel mode gre ip
 ipv6 router isis
```

Tunnel Broker (RFC 3053)



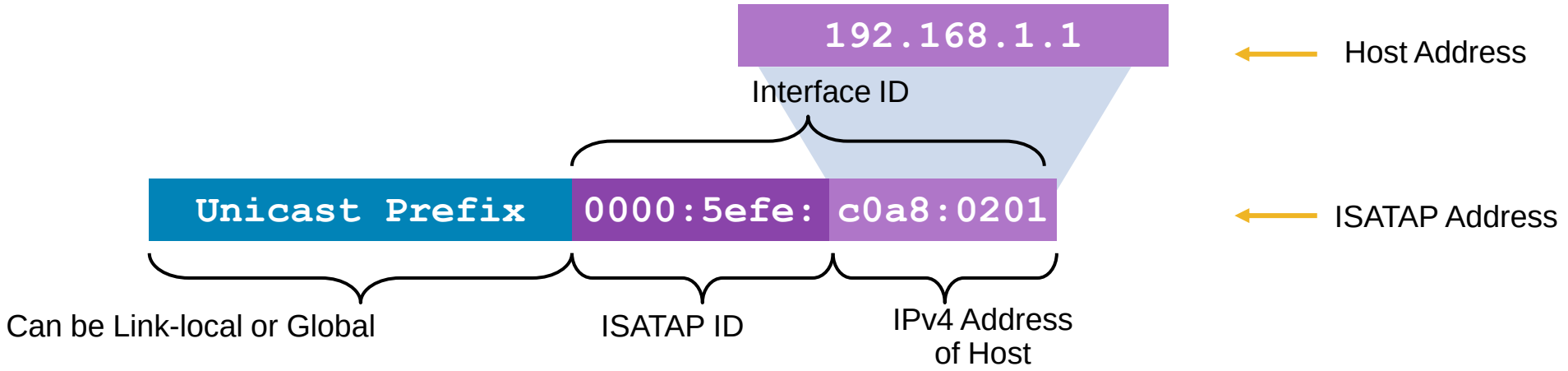
- Tunnel Broker servers manage tunnel requests from users
  - Binds tunnel between host and IPv6 server or router
- Particularly suited to isolated IPv6 hosts connected via an IPv4 Internet
- Client software loaded on to host
  - Linux/MAC/Windows
- Cisco routers do not support tunnel broker function
  - Tunnel brokers can remotely configure Cisco routers

Tunnel Broker Operation

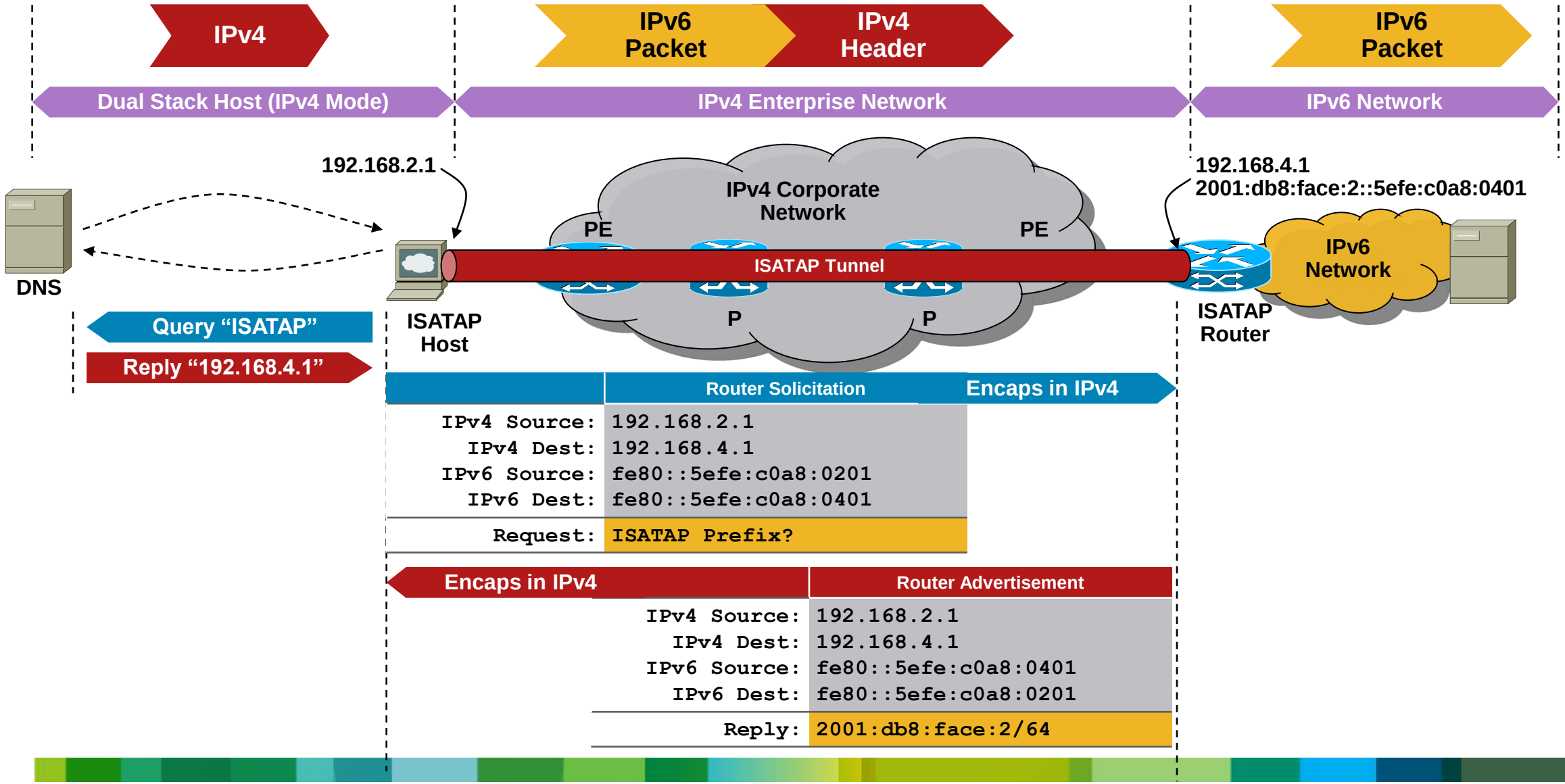


ISATAP Address Format

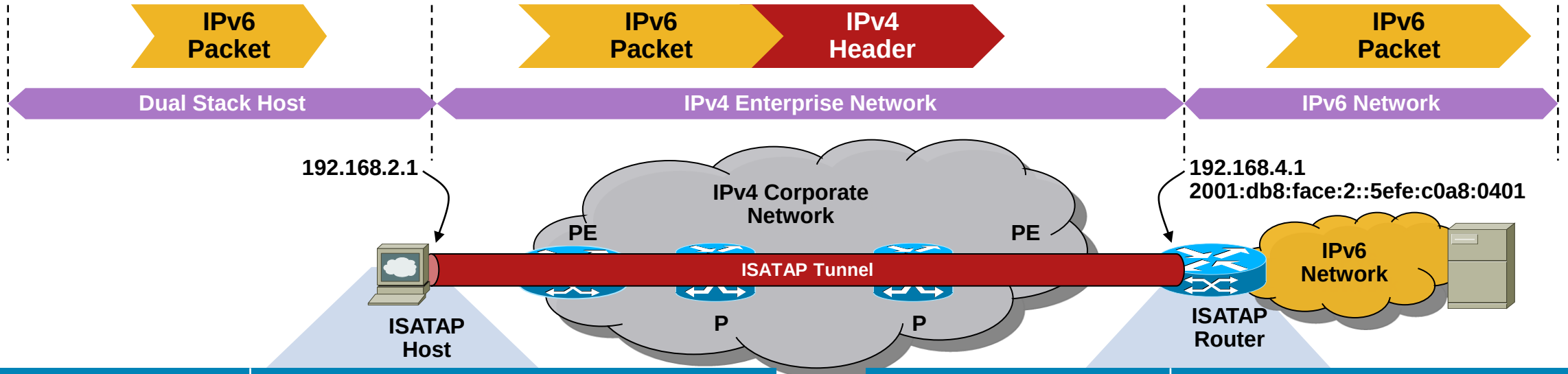
- ISATAP hosts use a special IPV6 address format
- Interface ID carries information
  - Rightmost 32 bits contains the host IPv4 address
  - Leftmost 32 bits contains “0000:5efe”
- Global prefix provided by ISATAP router
  - Interface ID portion remain static for all packets
  - Link-Local addresses used for solicitation of global address



ISATAP prefix advertisement



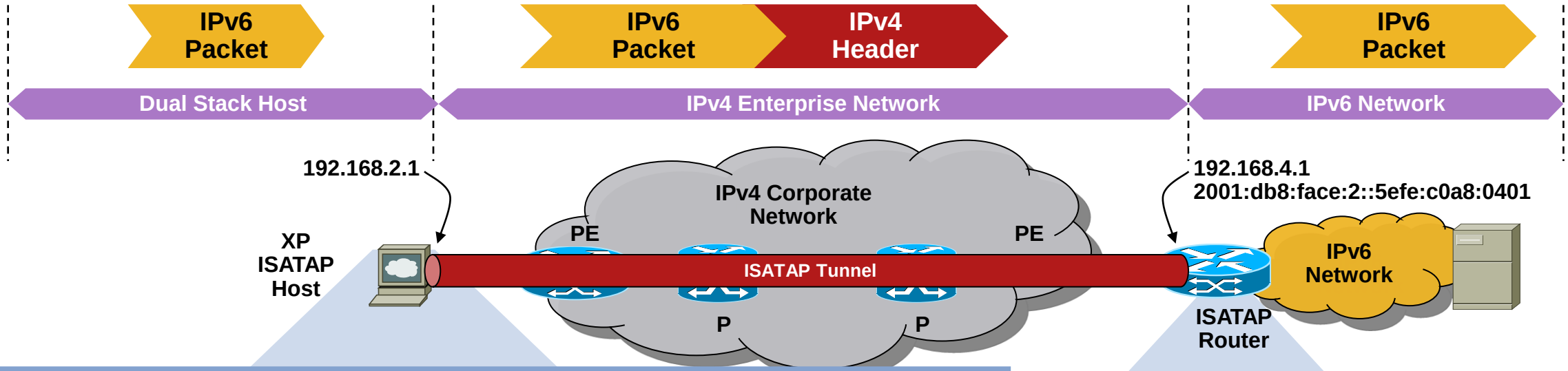
ISATAP Nodes Use 3 Addresses



Address	Value
IPv4 Global:	192.168.2.1
IPv6 Link-Local:	fe80::5efe:c0a8:0201
IPv6 Global:	2001:db8:face::5efe:c0a8:0201

Address	Value
IPv4 Global:	192.168.4.1
IPv6 Link-Local:	fe80::5efe:c0a8:0401
IPv6 Global:	2001:db8:face::5efe:c0a8:0401

ISATAP Configuration (XP)

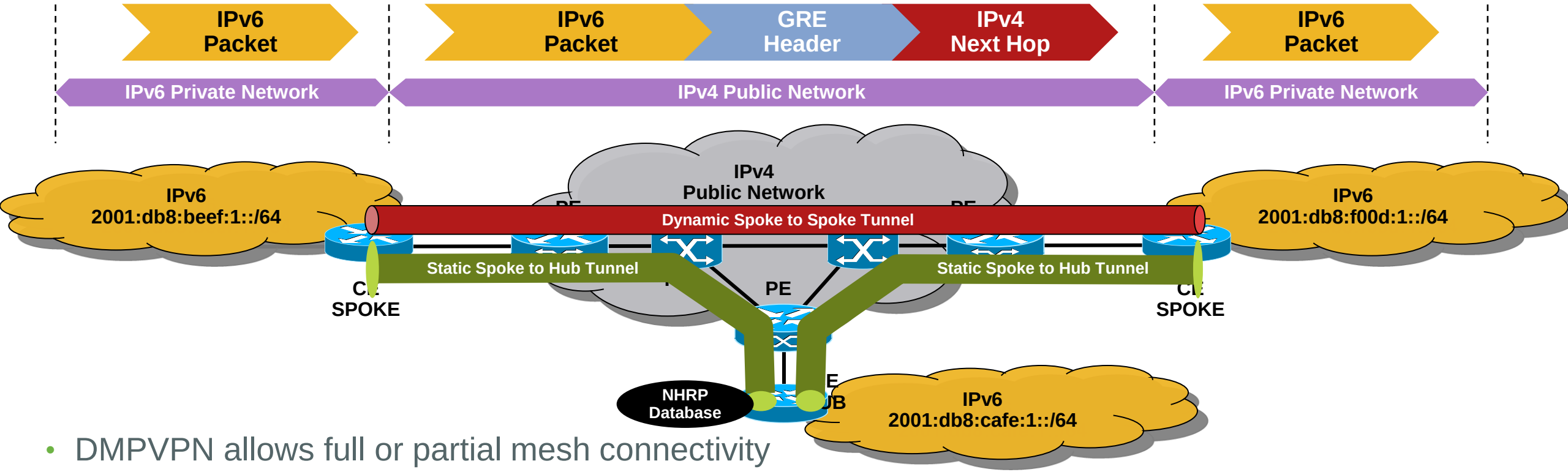



```
ipv6 install
netsh interface ipv6 isatap set router 192.168.4.1
```

- PC config does not use DNS
- EUI-64 allows router to generate Link ID portion
- Turn off ND message suppression

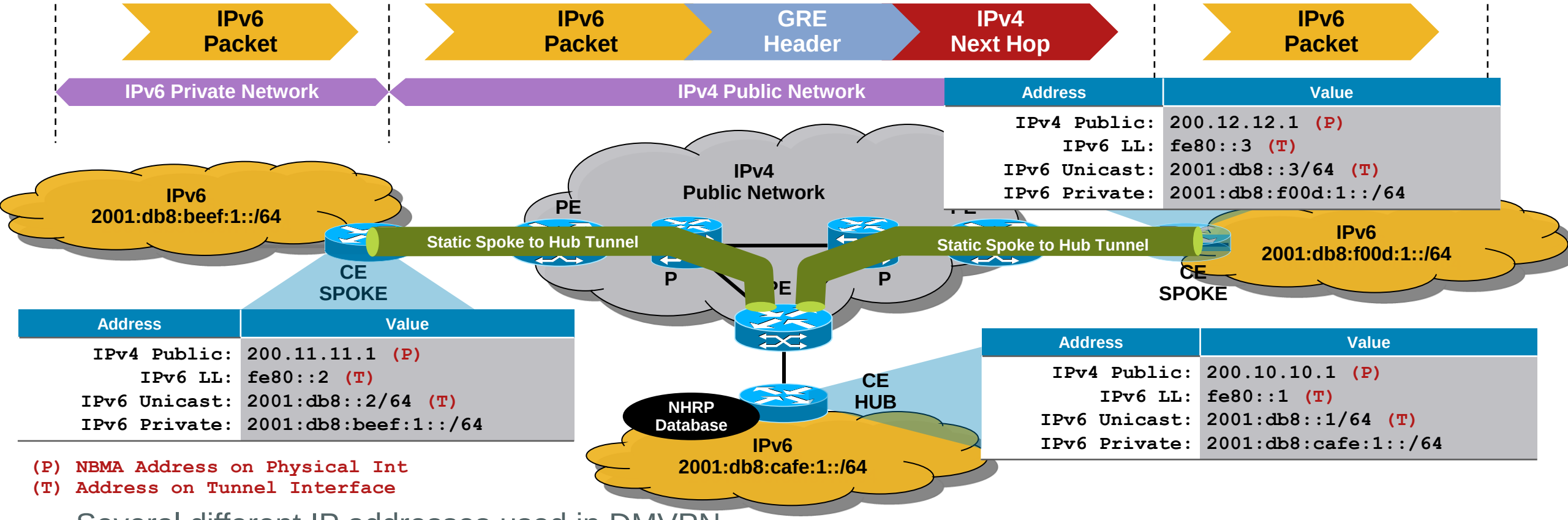
```
interface Ethernet0
 ip address 192.168.4.1 255.255.255.0
!
interface Tunnel0
 ipv6 address 2001:db8:face:2::/64 eui-64
 no ipv6 nd suppress-ra
 tunnel source Ethernet0
 tunnel mode ipv6ip isatap
```

DMVPN for IPv6



- DMPVPN allows full or partial mesh connectivity
  - Static mGRE tunnel spokes to hub
  - Dynamic mGRE tunnels spoke to spoke
- mGRE Tunnels can be encrypted
- mGRE relies on NHRP for destination address (NBMA)

DMVPN Addressing



Address	Value
IPv4 Public:	200.12.12.1 (P)
IPv6 LL:	fe80::3 (T)
IPv6 Unicast:	2001:db8::3/64 (T)
IPv6 Private:	2001:db8:f00d:1::/64

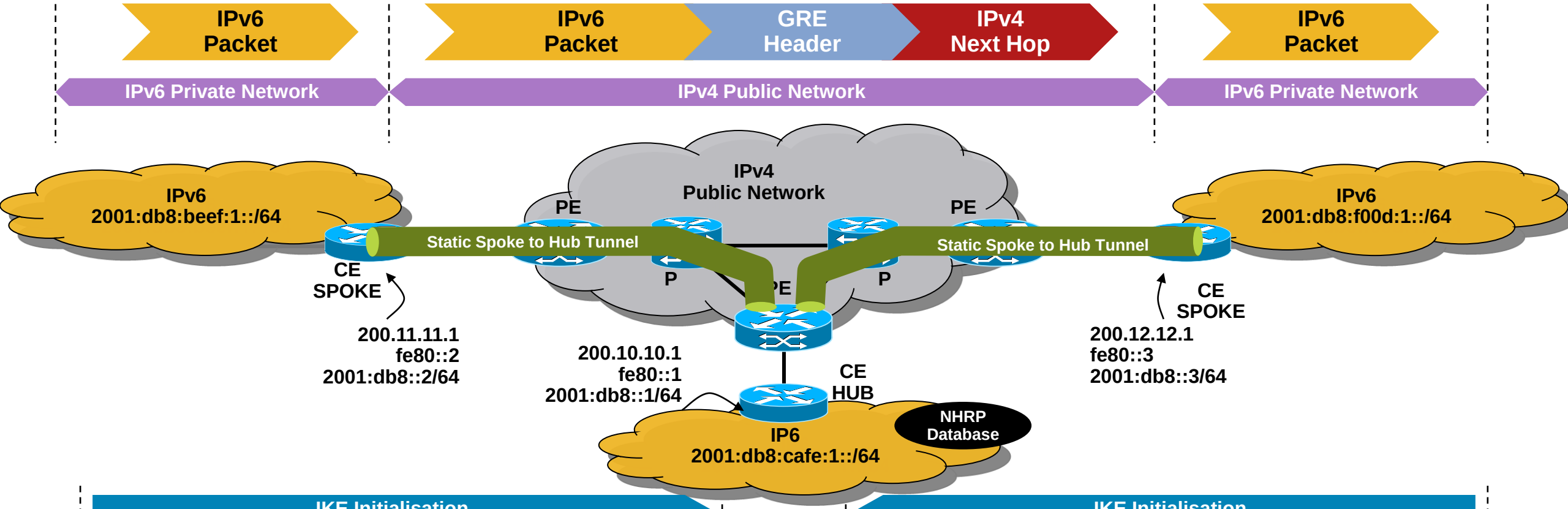
Address	Value
IPv4 Public:	200.11.11.1 (P)
IPv6 LL:	fe80::2 (T)
IPv6 Unicast:	2001:db8::2/64 (T)
IPv6 Private:	2001:db8:beef:1::/64

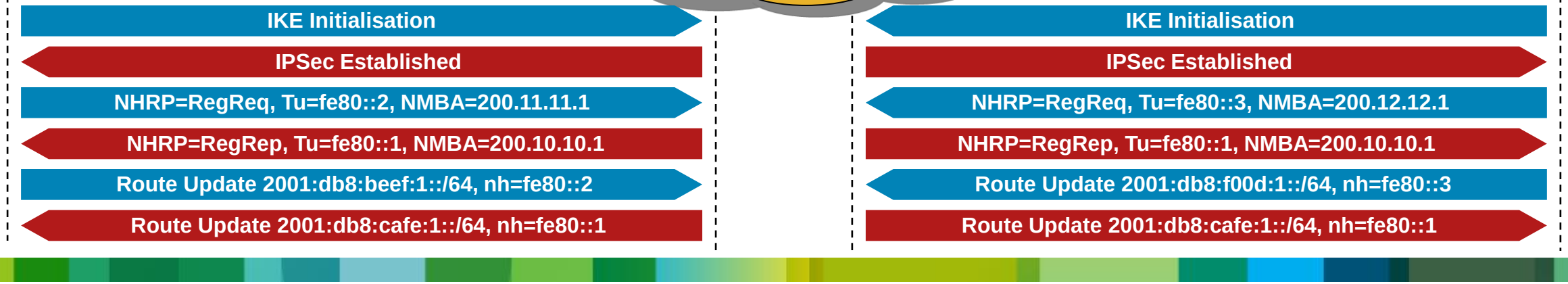
Address	Value
IPv4 Public:	200.10.10.1 (P)
IPv6 LL:	fe80::1 (T)
IPv6 Unicast:	2001:db8::1/64 (T)
IPv6 Private:	2001:db8:cafe:1::/64

(P) NBMA Address on Physical Int
(T) Address on Tunnel Interface

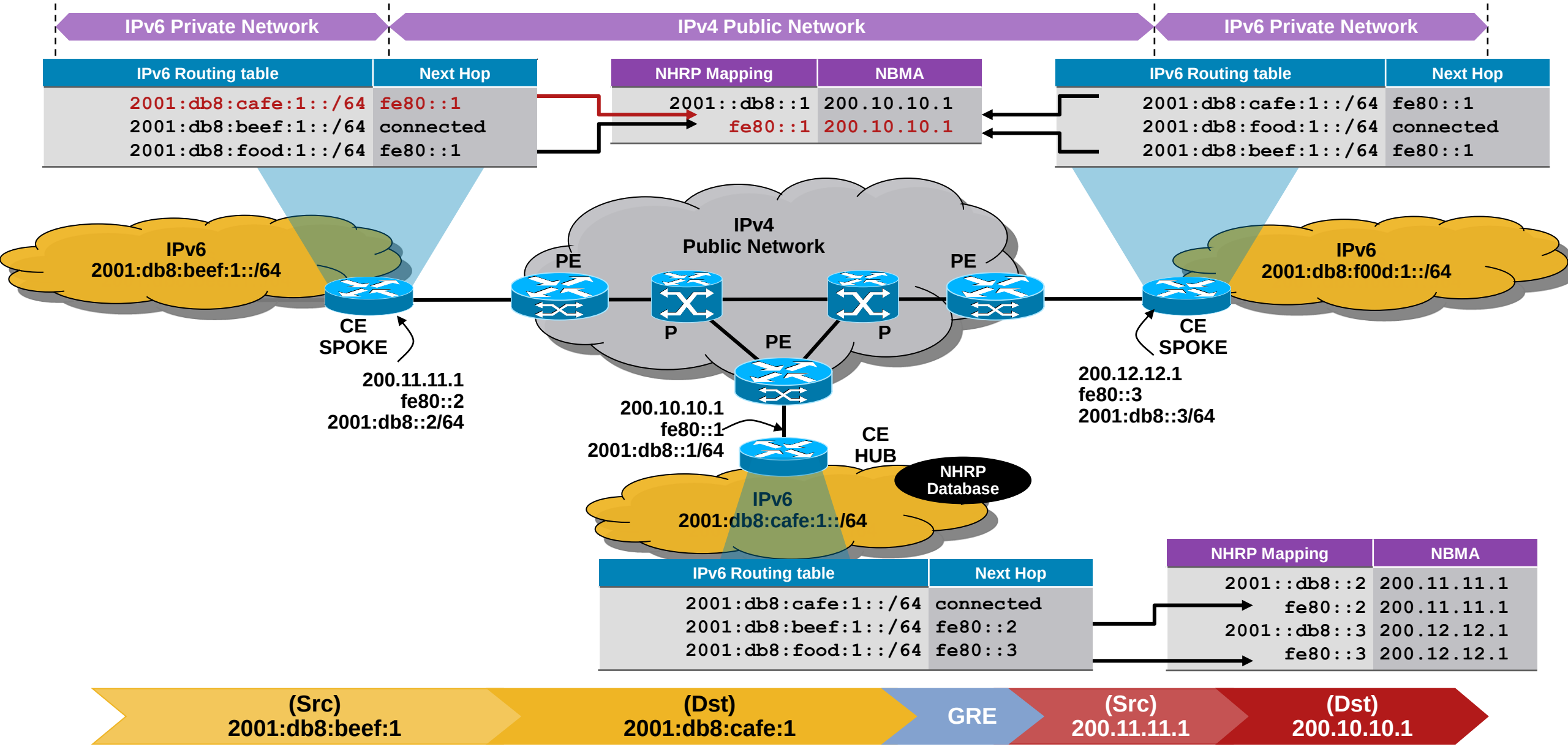
- Several different IP addresses used in DMVPN
  - Public IPv4 address – Resolves IPv6 to NBMA Next Hop address
  - Link-Local IPv6 address – Next hop in IPv6 routing table tunnel-end point
  - Unicast IPv6 address - Mandatory on tunnel to communicate with hub
  - Private IPv6 Network – The IPv6 subnets that are being connected

DMVPN Addressing

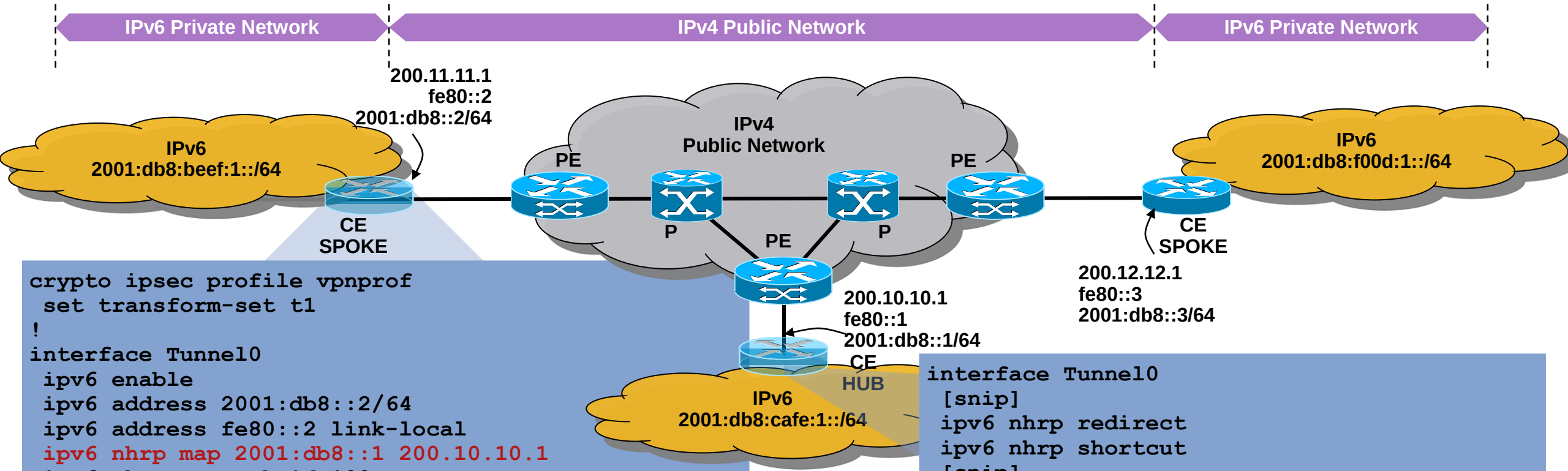




DMVPN Routing and NHRP Tables



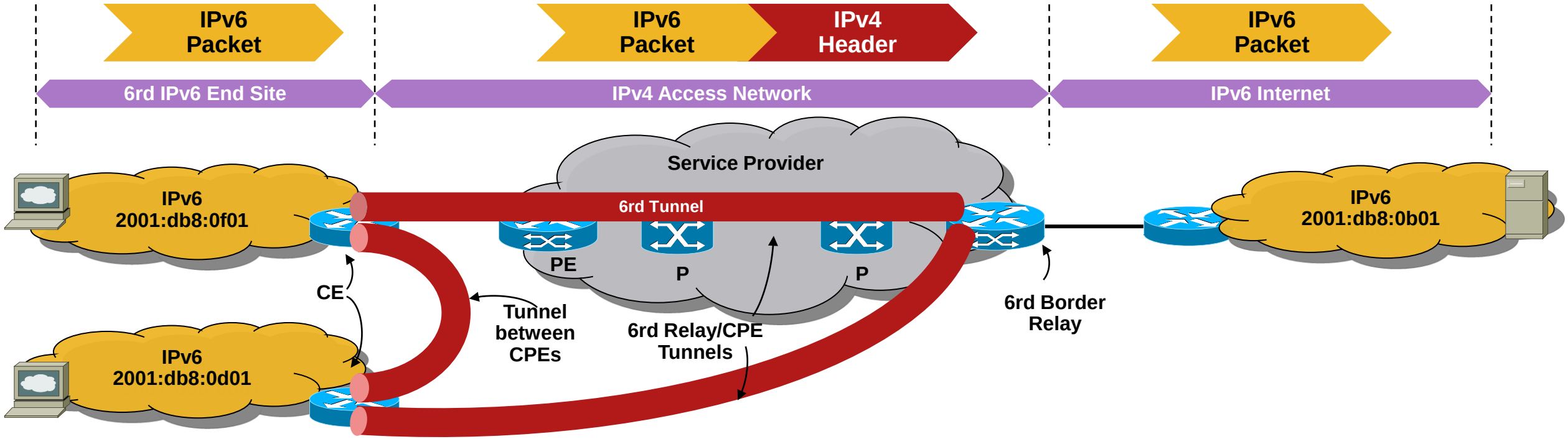
DMVPN IPv6 Configuration




```
crypto ipsec profile vpnprof
 set transform-set t1
!
interface Tunnel0
 ipv6 enable
 ipv6 address 2001:db8::2/64
 ipv6 address fe80::2 link-local
 ipv6 nhrp map 2001:db8::1 200.10.10.1
 ipv6 nhrp network-id 100
 ipv6 nhrp nhs 2001:db8::1
 ipv6 nhrp map multicast 200.10.10.1
 tunnel source Ethernet 0/0
 tunnel mode gre multipoint
 tunnel protection ipsec profile vpnprof
!
interface Ethernet 0/0
 ip address 200.11.11.1 255.255.0.0
```

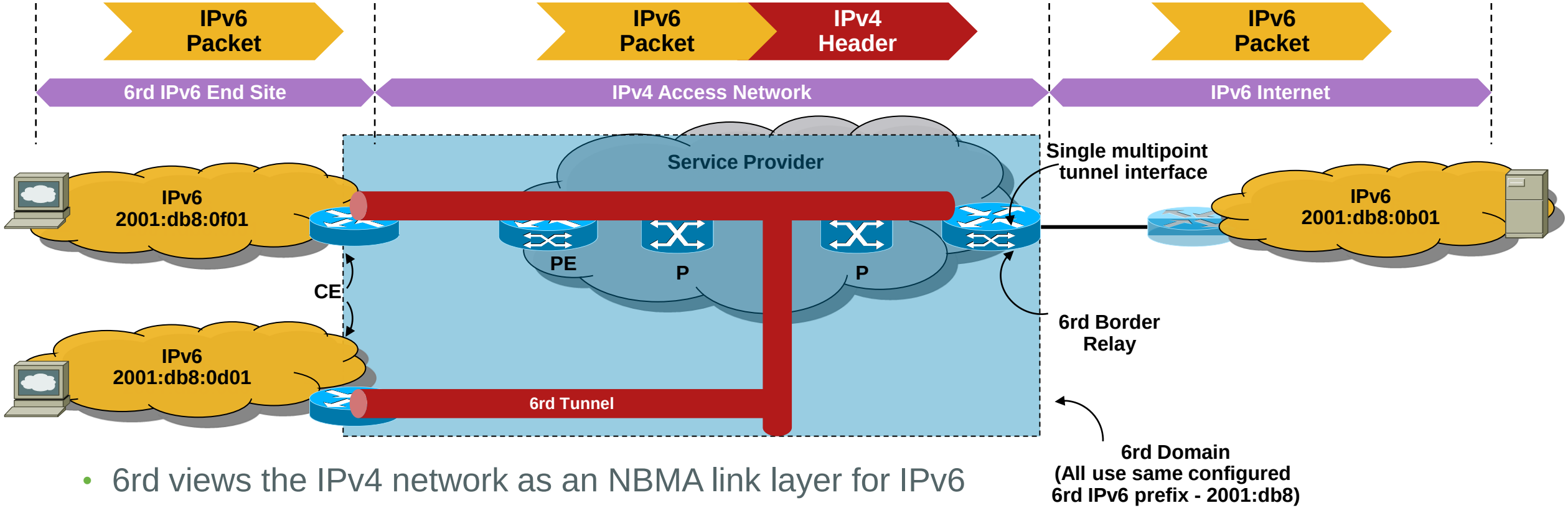
```
interface Tunnel0
 [snip]
 ipv6 nhrp redirect
 ipv6 nhrp shortcut
 [snip]
!
interface Ethernet 0/0
 ip address 200.10.10.1 255.255.0.0
```

6rd Tunnels (RFC 5969)



- Native dual-stack IP service to the end site
- Simple, stateless, automatic IPv6-in-IPv4 encap and decap functions
- Embedded IPv4 address needs to match IPv4 address in Tunnel header for security
- IPv6 traffic automatically follows IPv4 Routing (IPv4 address used as tunnel endpoint)
- BRs placed at IPv6 edge, addressed via anycast for load-balancing and resiliency

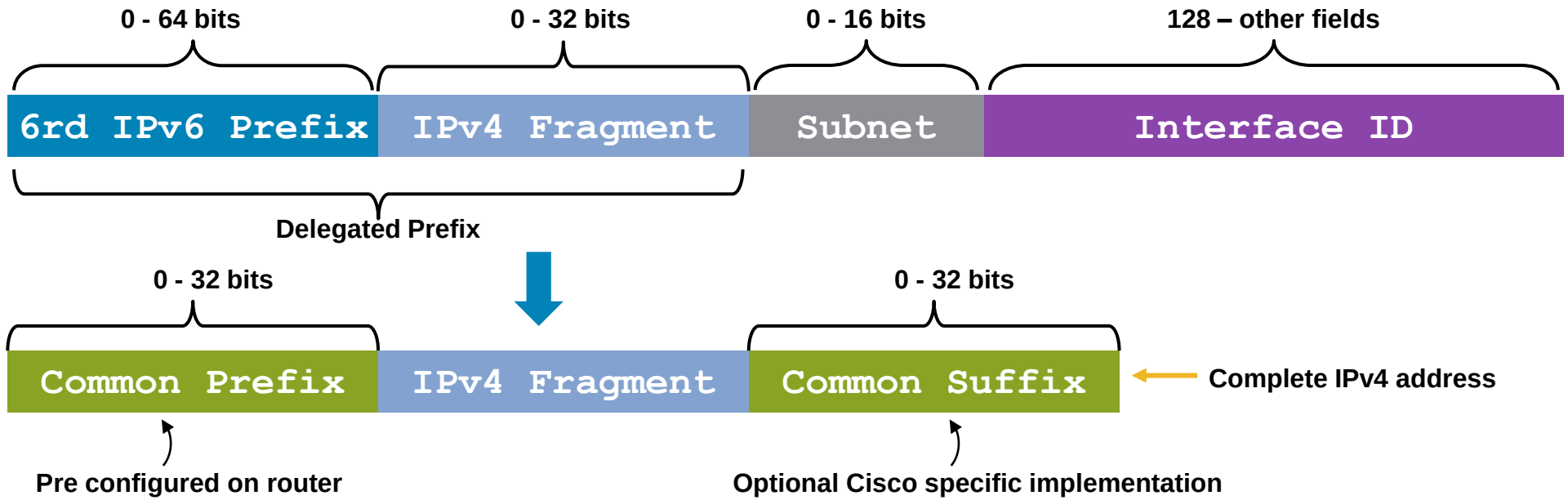
6rd Logical NBMA Behaviour



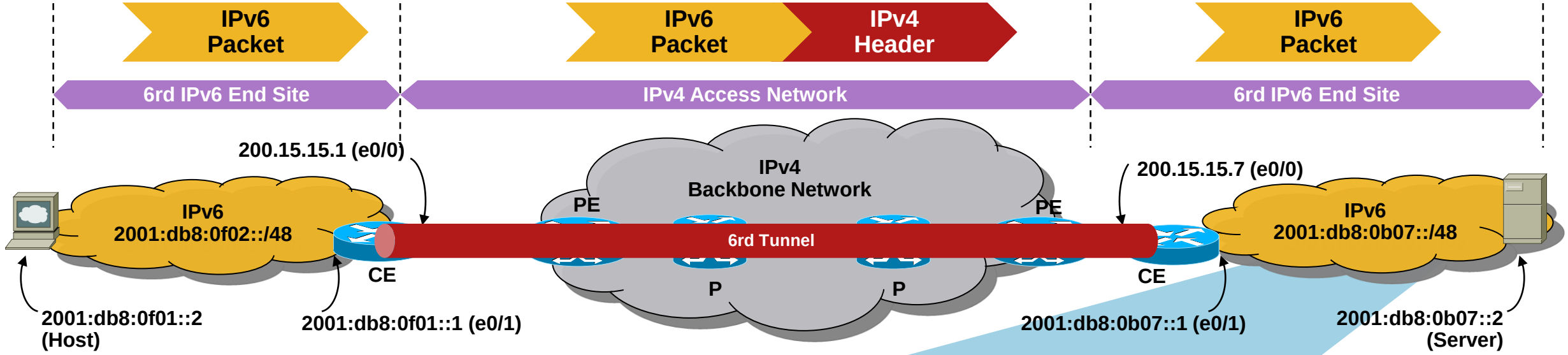
- 6rd views the IPv4 network as an NBMA link layer for IPv6
- Border Relay serves has a single multipoint interface
  - No per user state, serves all users in 6rd domain

6rd Delegated Prefix

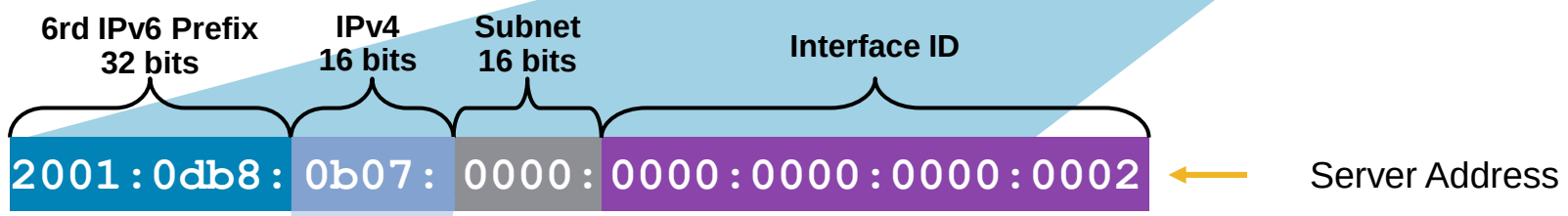
- Every customer site is assigned a 6rd delegated prefix
- Delegated prefix is created by
  - Combining the SP's 6rd prefix and all or part of the CE IPv4 address
- Not all 32 bits of IPv4 address need be carried
  - Common IPv4 prefix and suffix can be pre-configured

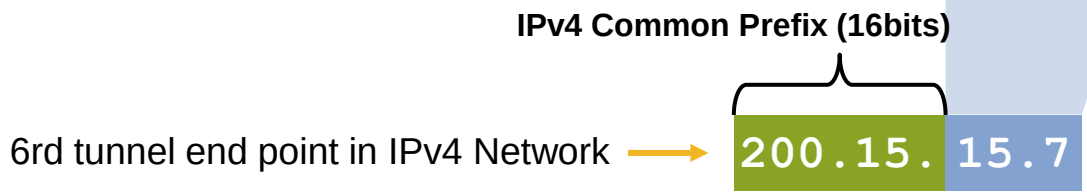


Destination Dynamically Computed Example

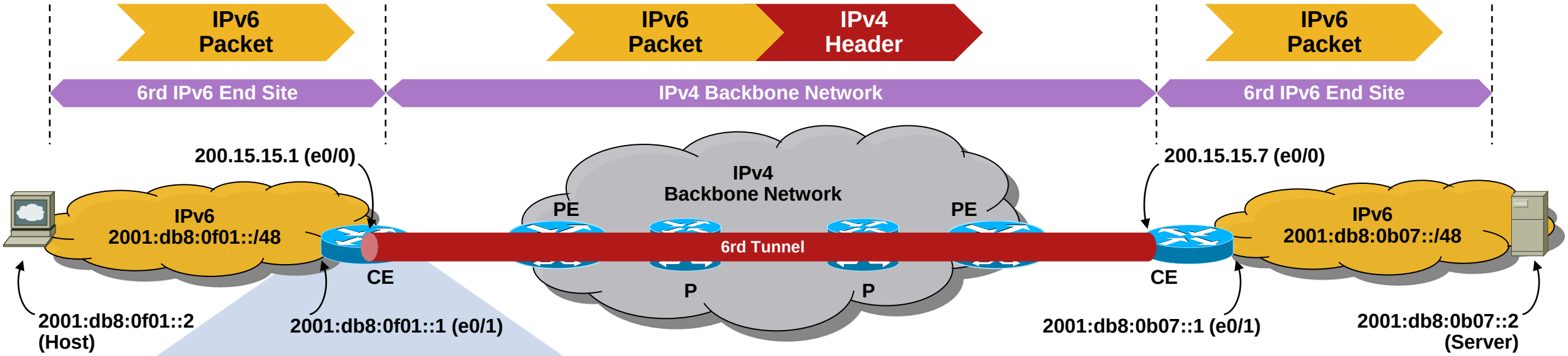


6rd Parameter	Value
6rd Prefix	2001:db8::/32
IPv4 Common Prefix	200.15/16
IPv4 Common Suffix	0/0 (Cisco specific)





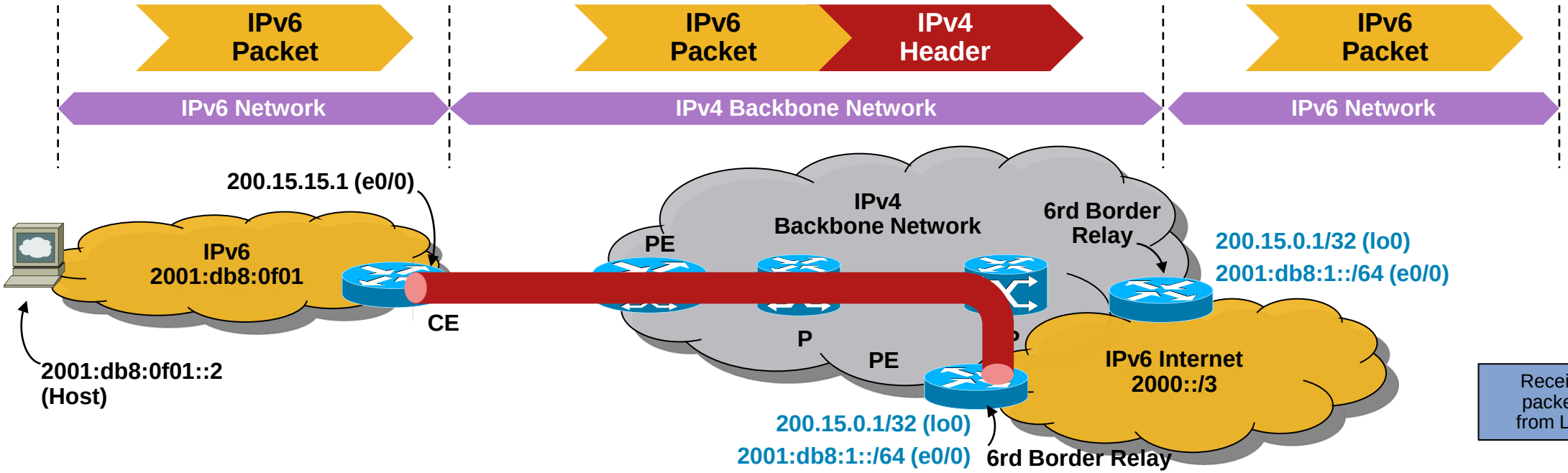
6rd CE Configuration (IOS)



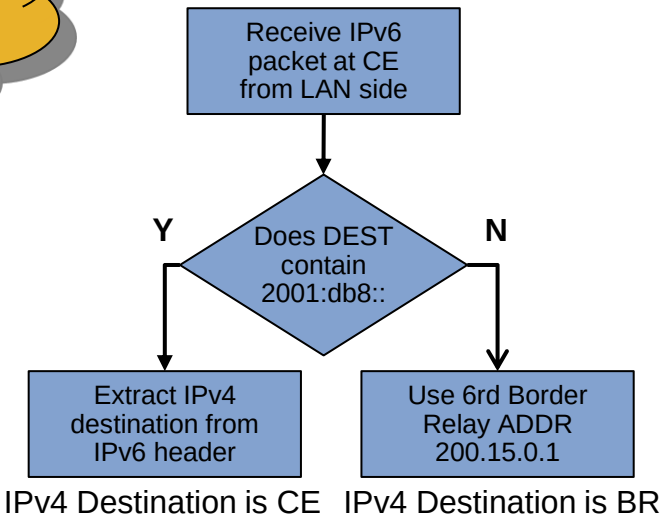

```
!
ipv6 general-prefix 6rd-prefix 6rd Tunnel1
ipv6 unicast-routing
ipv6 cef
!
interface Tunnel1
 ipv6 enable
 tunnel source Ethernet0/0
 tunnel mode ipv6ip 6rd
 tunnel 6rd prefix 2001:db8::/32
 tunnel 6rd ipv4 prefix-len16
 tunnel 6rd br 200.15.0.1 ← Config to Border Relay
```

```
!
interface Ethernet0/0
 description Shared IPv4 infrastructure
 ip address 200.15.15.1 255.255.255.0
!
interface Ethernet1/0
 description End Site LAN
 ipv6 address 6rd-prefix ::1/64
!
ipv6 route 2001:db8::/32 tunnel1
ipv6 route ::/0 Tunnel1 2001:db8:1:: ←Default to BR
```

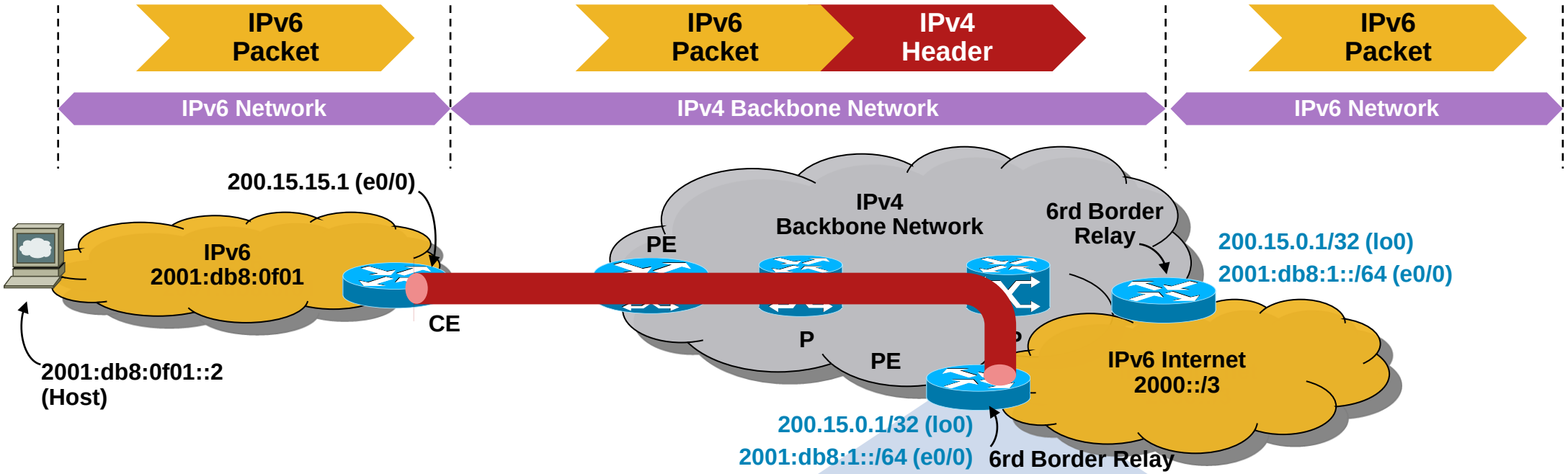
Internet Access through 6rd Border Relay



- 6rd Border Relay allows access to IPv6 global Internet
  - If IPv6 destination outside of 6rd prefix then tunnel packet to border relay
- Can use tunnel pre-provisioned IPv4 Anycast address
  - 6rd CE router finds closest 6rd BR router based on IGP
- Default route to IPv6 Internet, usually pre-provisioned by the ISP



6rd Border Relay Configuration (IOS)




```
!
ipv6 general-prefix 6rd-prefix 6rd Tunnel1
ipv6 unicast-routing
ipv6 cef
!
interface Tunnel1
 ipv6 enable
 tunnel source Loopback0
 tunnel mode ipv6ip 6rd
 tunnel 6rd prefix 2001:db8::/32
 tunnel 6rd ipv4 prefix-len16
```

```
interface Ethernet0/0
 description IPv6 Internet
 ipv6 address 2001:db8:1::/64
!
interface Loopback0
 description Shared IPv4 infrastructure
 ip address 200.15.0.1 255.255.255.0
!
ipv6 route 2001:db8::/32 tunnel1
ipv6 route ::/0 2001:db8:2:: ! Or use routing protocol
```

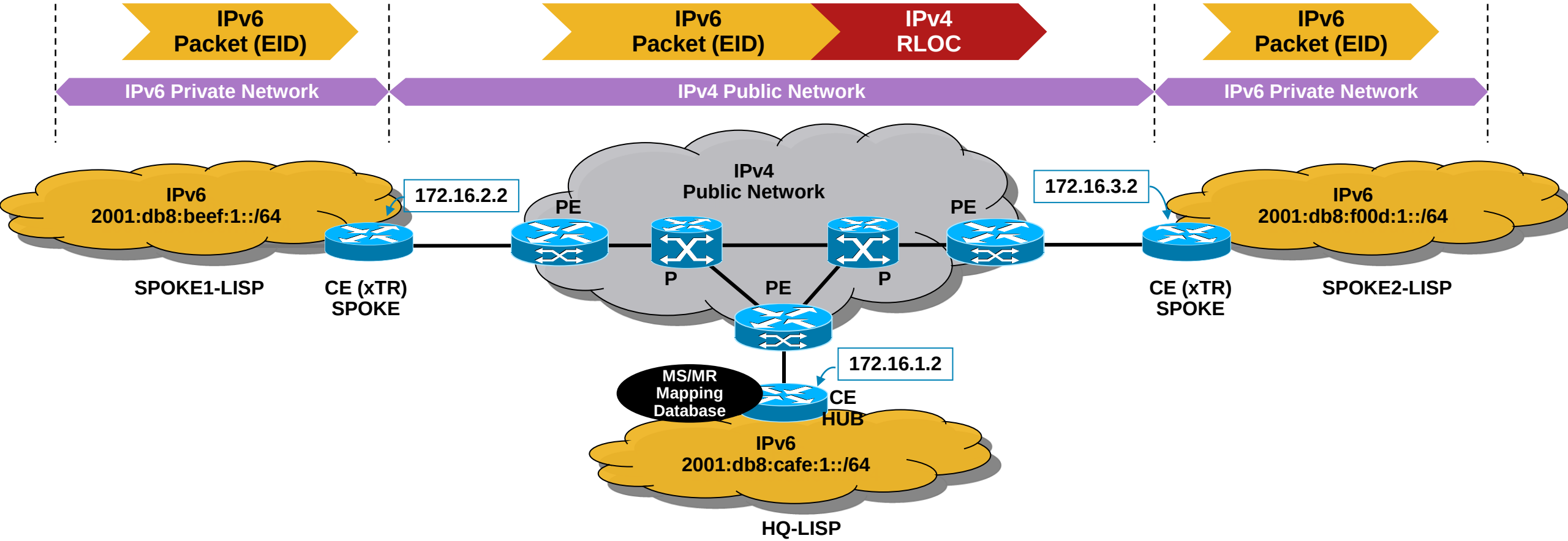
LISP Overview

- Locator/Identity Split creates a “Level of indirection” by using two namespaces – EID and RLOC
  - EID = Endpoint Identifier
  - RLOC = Routing Locator
- LISP creates two Name Spaces:
- EID (Endpoint Identifier) is the host IP address
  - Same as today – it’s what is used in DNS
  - In LISP, the EID can move independently of the RLOC.
- RLOC (Routing Locator) is the infrastructure IP address of the LISP router
  - Routed in the Internet just like today!
  - Globally routed and aggregated along Internet connectivity topology
- EID packets are encapsulated inside RLOC packets and forwarded over the RLOC infrastructure (Internet / WAN cloud / Enterprise)

LISP Overview

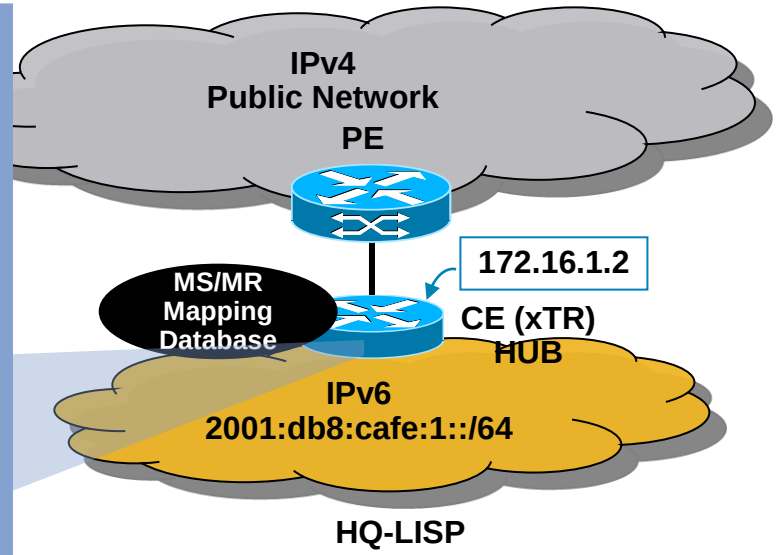
- Only affects ‘edge’ devices
  - LISP ITR = LISP Ingress Tunnel Router
  - LISP ETR = LISP Egress Tunnel Router
  - Devices that undertake both roles are termed LISP xTR
- Map Server (MS)
  - ETRs register EID prefixes with the MS (just like authoritative DNS)
- Map Resolver (MR)
  - ITRs send LISP Map Requests to the MR to be resolved (just like a DNS resolver)

LISP Network Topology

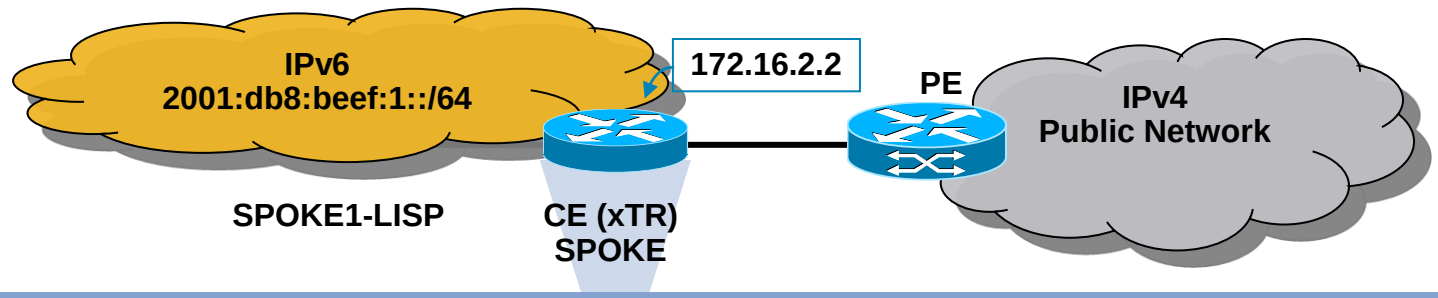


LISP Mapping Database

```
vrf definition lisp
 rd 1:1
!
interface LISP0
!
lisp site HQ-LISP
 description LISP HQ Site
 authentication-key s3cr3t-hq
 eid-prefix 2001:db8:cafe:1::/48
lisp site Spoke1-LISP
 description LISP Spoke Site 1
 authentication-key s3cr3t-1
 eid-prefix 2001:db8:beef:1::/48
lisp site Spoke2-LISP
 description LISP Spoke Site 2
 authentication-key s3cr3t-2
 eid-prefix 2001:db8:f00d:1::/48
!
ipv6 lisp map-server
ipv6 lisp map-resolver
ipv6 lisp alt-vrf lisp
!
ipv6 lisp database-mapping 2001:db8:cafe:1::/48 172.16.1.2 priority 1 weight 100
ipv6 lisp itr map-resolver 172.16.1.2
ipv6 lisp itr
ipv6 lisp etr map-server 172.16.1.2 key s3cr3t-hq
ipv6 lisp etr
!
```



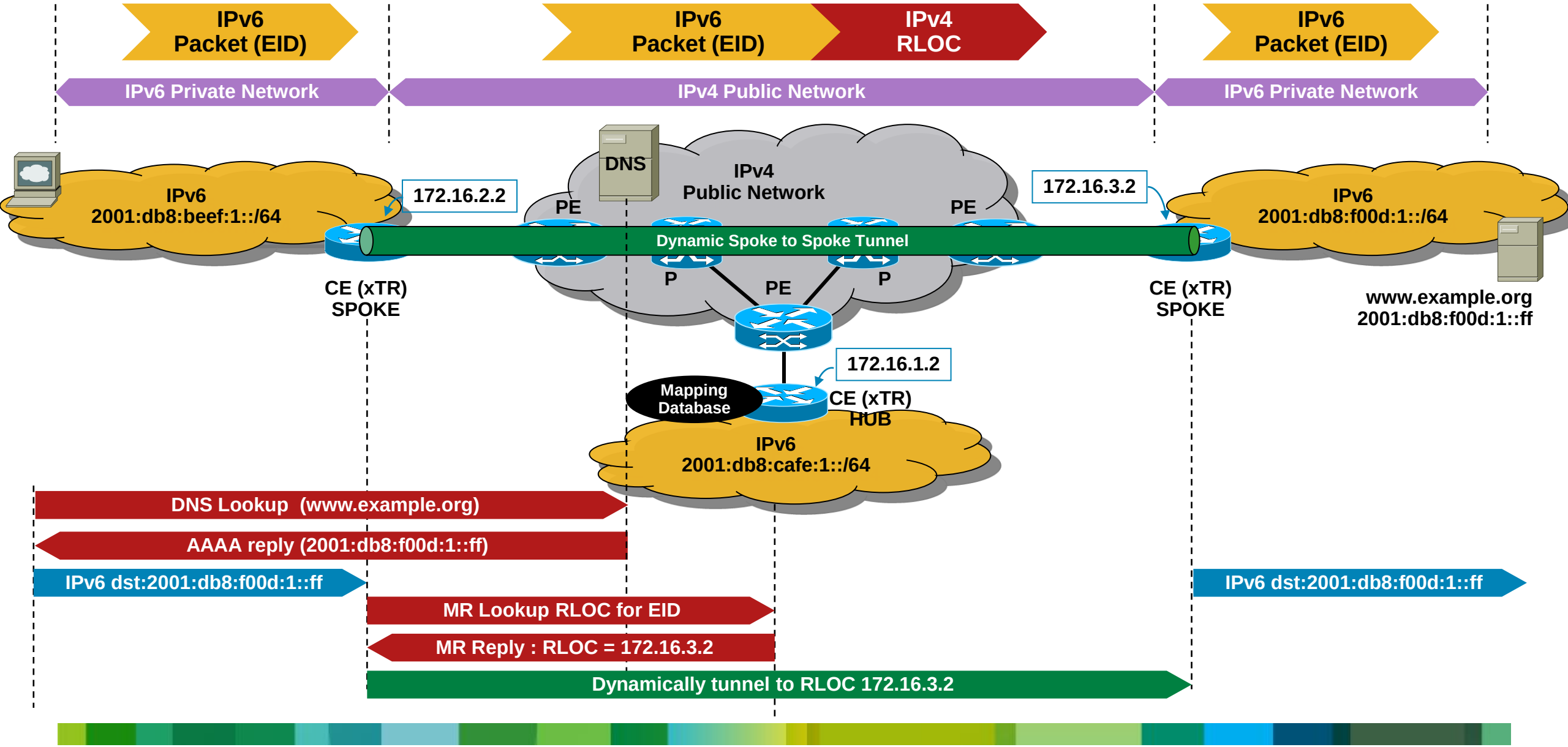
LISP Spoke Setup




```
!
ipv6 unicast routing
ipv6 cef
!
interface LISP0
!
ipv6 lisp database-mapping 2001:db8:beef:1::/48 172.16.1.2 priority 1 weight 100
!
ipv6 lisp itr map-resolver 172.16.1.2
ipv6 itr
ipv6 etr map-server 172.16.1.2 key s3cr3t-1
ipv6 lisp etr
!
ipv6 route ::/0 Null0
```

- Same config would be applied to SPOKE2-LISP
  - Only change of IPv6 prefix & map-server key

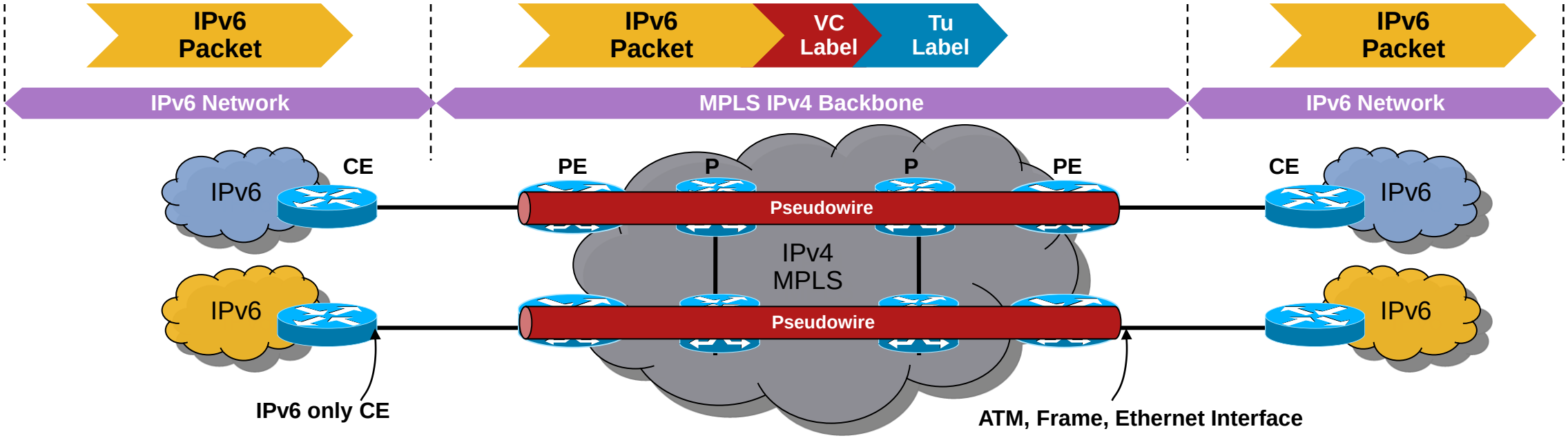
LISP Traffic Flow



IPv6 Solutions using MPLS

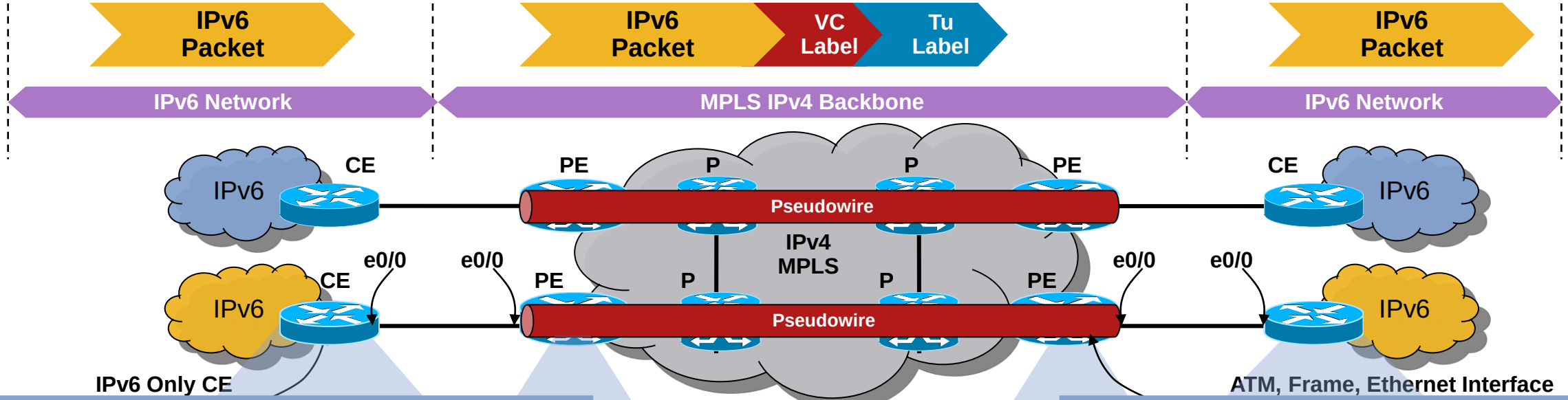
43

IPv6 over MPLS Pseudowires



- No impact on existing MPLS Core (IPv6 unaware)
- CE routers need only be single stack IPv6 aware
- P2P PWs or P2MP PWs (VPLS) supported at PE
- Good for interconnecting discrete customer islands
- Sub-Optimal Multicast support, Full Mesh required
- Many CE interface typed supported – Ethernet, ATM, Frame Relay

MPLS Pseudowire Configuration



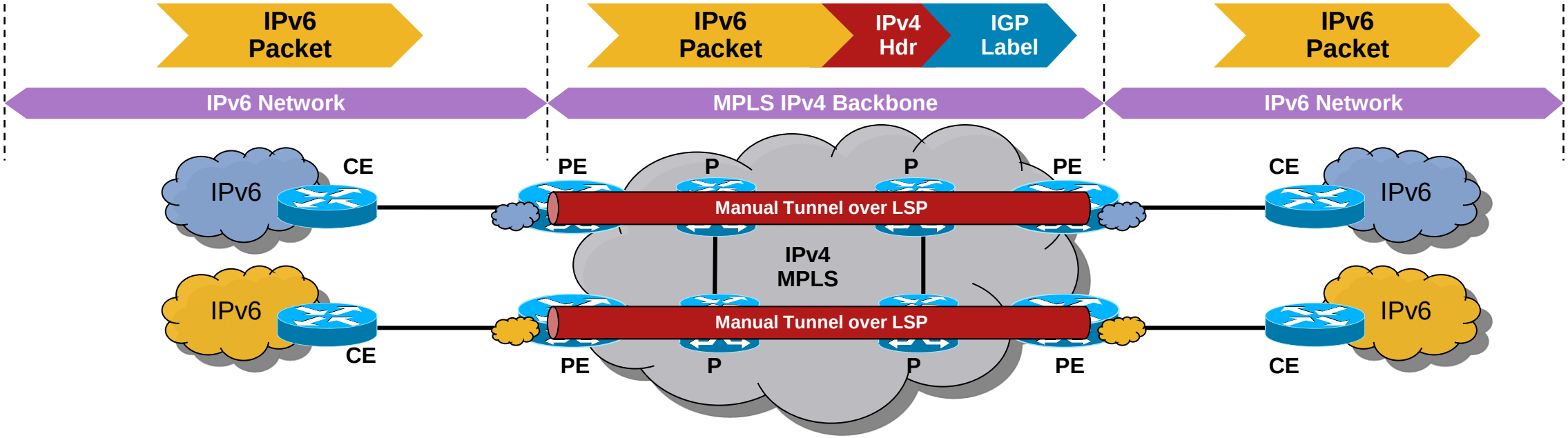

```
interface ethernet0/0.10
 encapsulation dot1q 10
 ipv6 address 2001:100::72A/64
```

```
interface loopback0
 ip address 200.10.10.1 255.255.255.0

Interface ethernet0/0.10
 encapsulation dot1q 10
 xconnect 200.11.11.1 100 encaps mpls
```

```
interface ethernet0/0.10
 encapsulation dot1q 10
 ipv6 address 2001:100::72E/64
```

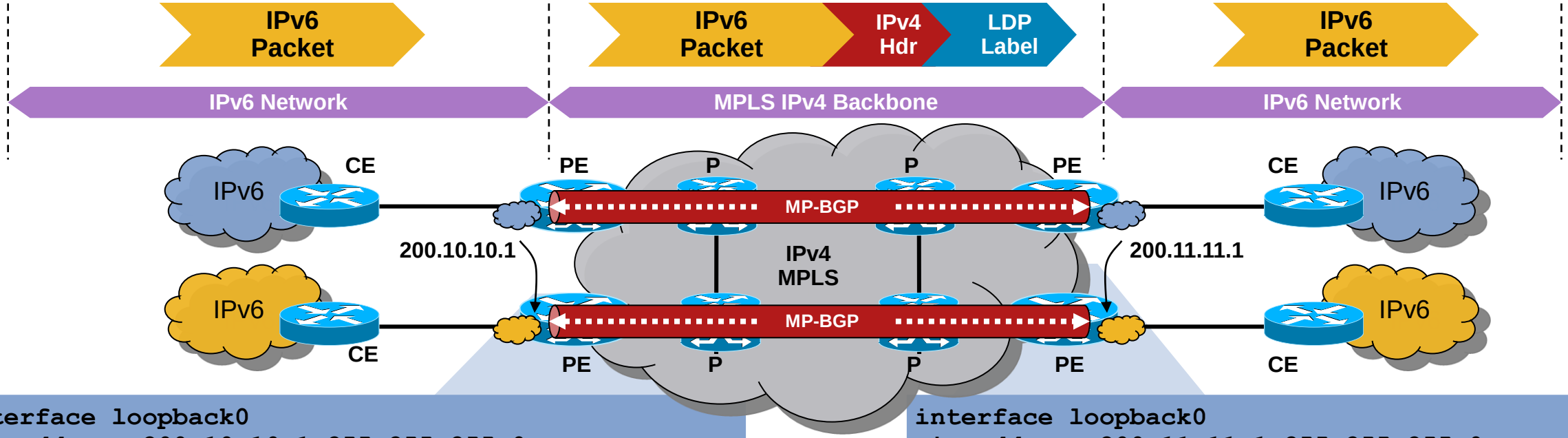
```
interface loopback0
 ip address 200.11.11.1 255.255.255.0

Interface ethernet0/0.10
 encapsulation dot1q 10
 xconnect 200.10.10.1 100 encaps mpls
```

Manual Tunnels Over MPLS



- Uses manual tunnel configuration as previously discussed
- Tunnel in Tunnel Method (P2P tunnels)
  - Manual Tunnel running over Label Switch Path (protocol=41)
  - Double encapsulation, first IPv4 then into MPLS frame (tunnel overhead)
  - Tunnels terminate on PE
- CE routers single stack, PE must be dual stack
- MP-BGP can be run over tunnel to distribute IPv6 routes between PE routers
- Good for interconnecting discrete customer islands

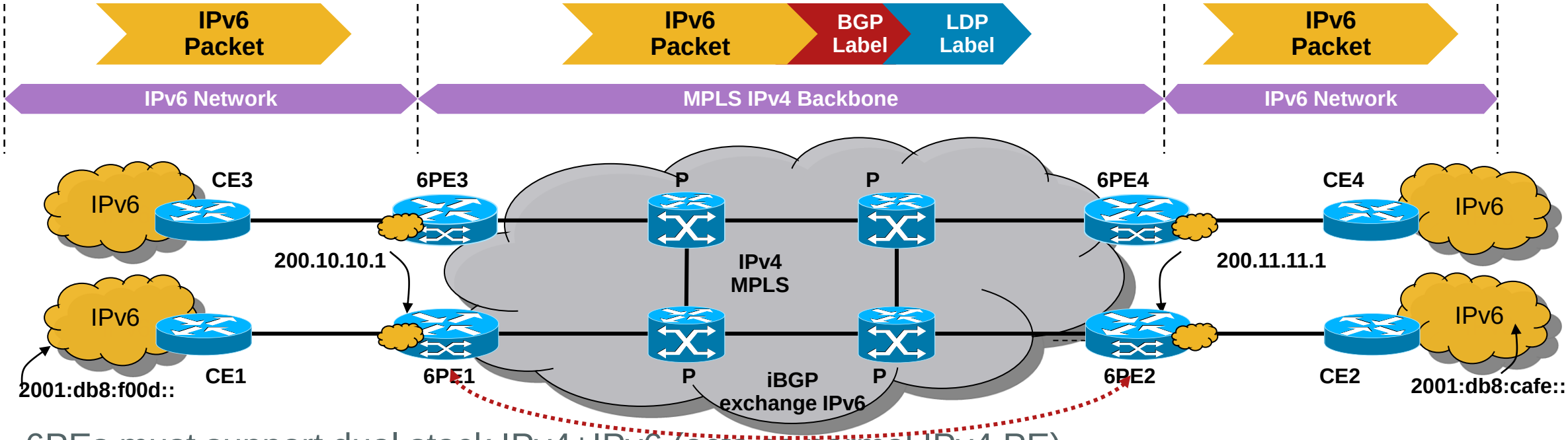
Manual Tunnels Over MPLS Configuration




```
interface loopback0
 ip address 200.10.10.1 255.255.255.0

interface tunnel100
 ipv6 address 2001:db8:a:b::1/64
 tunnel source loopback0
 tunnel destination 200.11.11.1
 tunnel mode ipv6ip

router bgp 100
 neighbor 2001:db8:a:b::2 remote-as 100

 address-family ipv6
 neighbor 2001:db8:a:b:2 activate
```

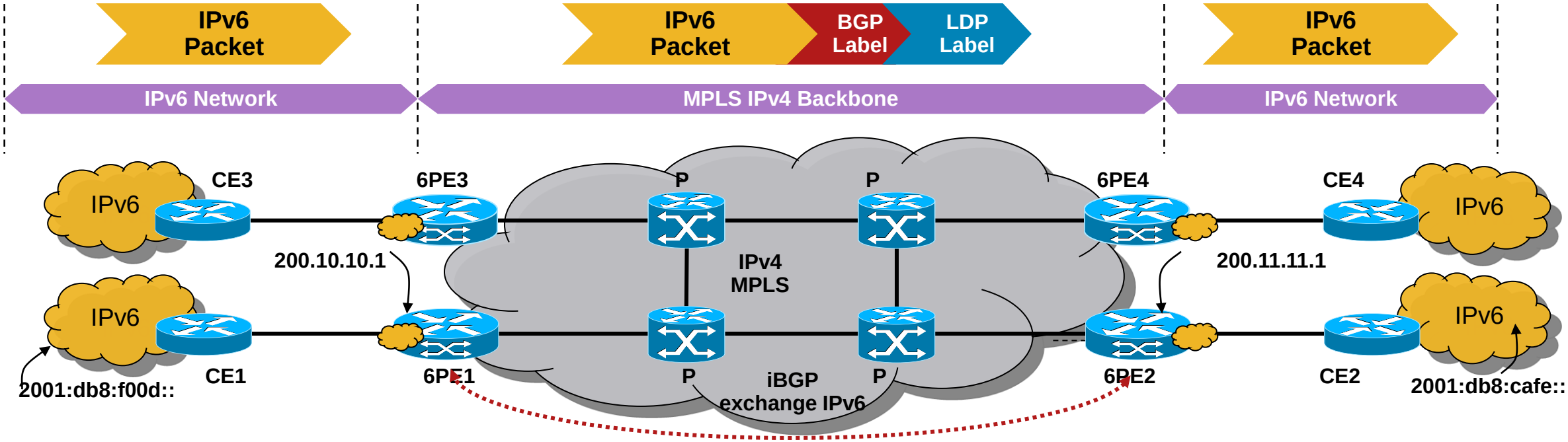
```
interface loopback0
 ip address 200.11.11.1 255.255.255.0

interface tunnel100
 ipv6 address 2001:db8:a:b::2/64
 tunnel source loopback0
 tunnel destination 200.10.10.1
 tunnel mode ipv6ip

router bgp 100
 neighbor 2001:db8:a:b::1 remote-as 100

 address-family ipv6
 neighbor 2001:db8:a:b::1 activate
```

IPv6 Transit using MPLS 6PE (RFC 4798)



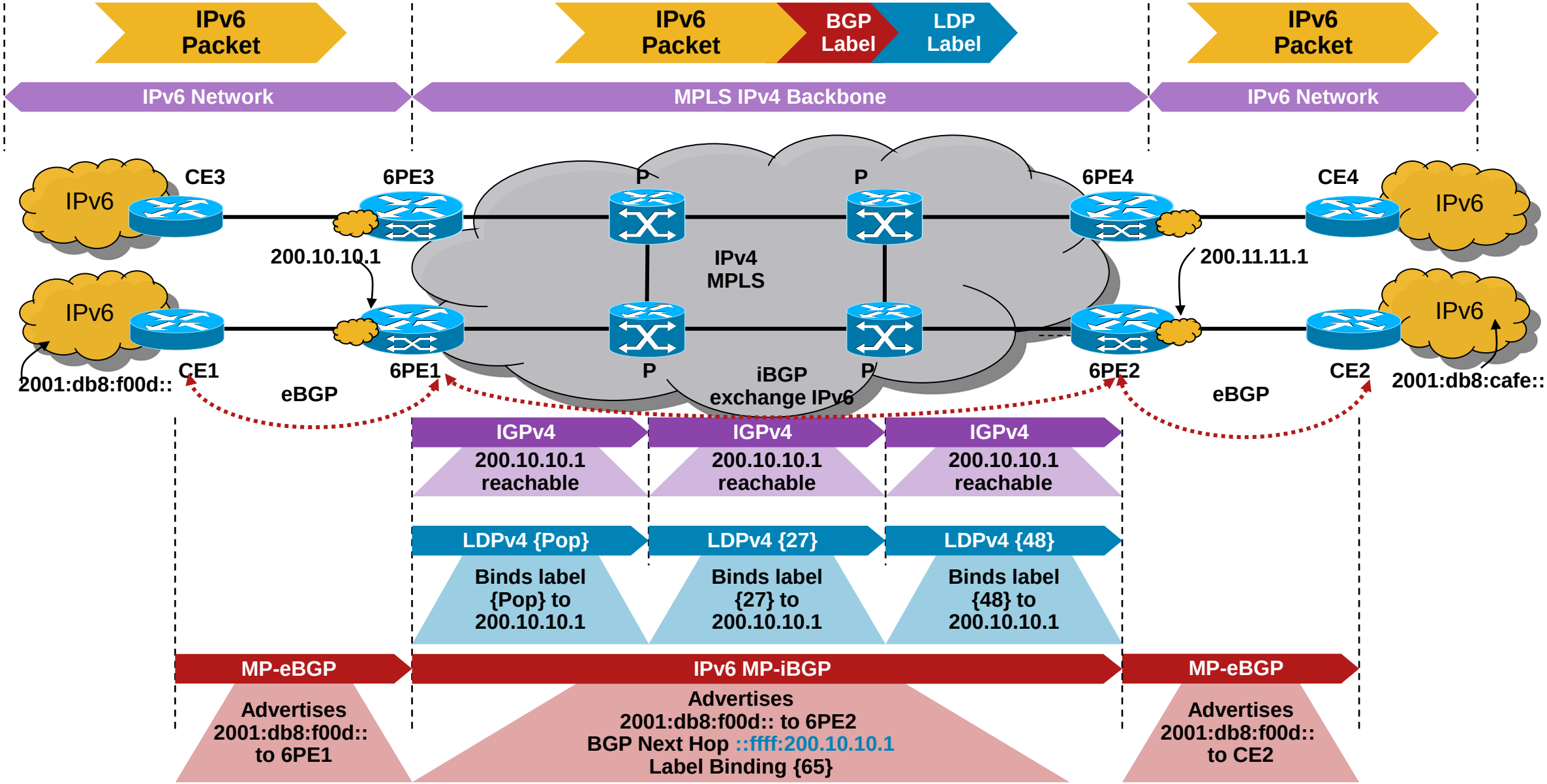
- 6PEs must support dual stack IPv4+IPv6 (acts as normal IPv4 PE)
- IPv6 packets transported from 6PE to 6PE over Label Switch Path
- IPv6 addresses exist in global table of PE routers only
  - IPv6 addresses exchanged between 6PE using MP-BGP session
- Core uses IPv4 control plane (LDPv4, TEv4, IGPv4, MP-BGP)
- Benefits from MPLS features such as FRR, TE

Services using MPLS 6PE

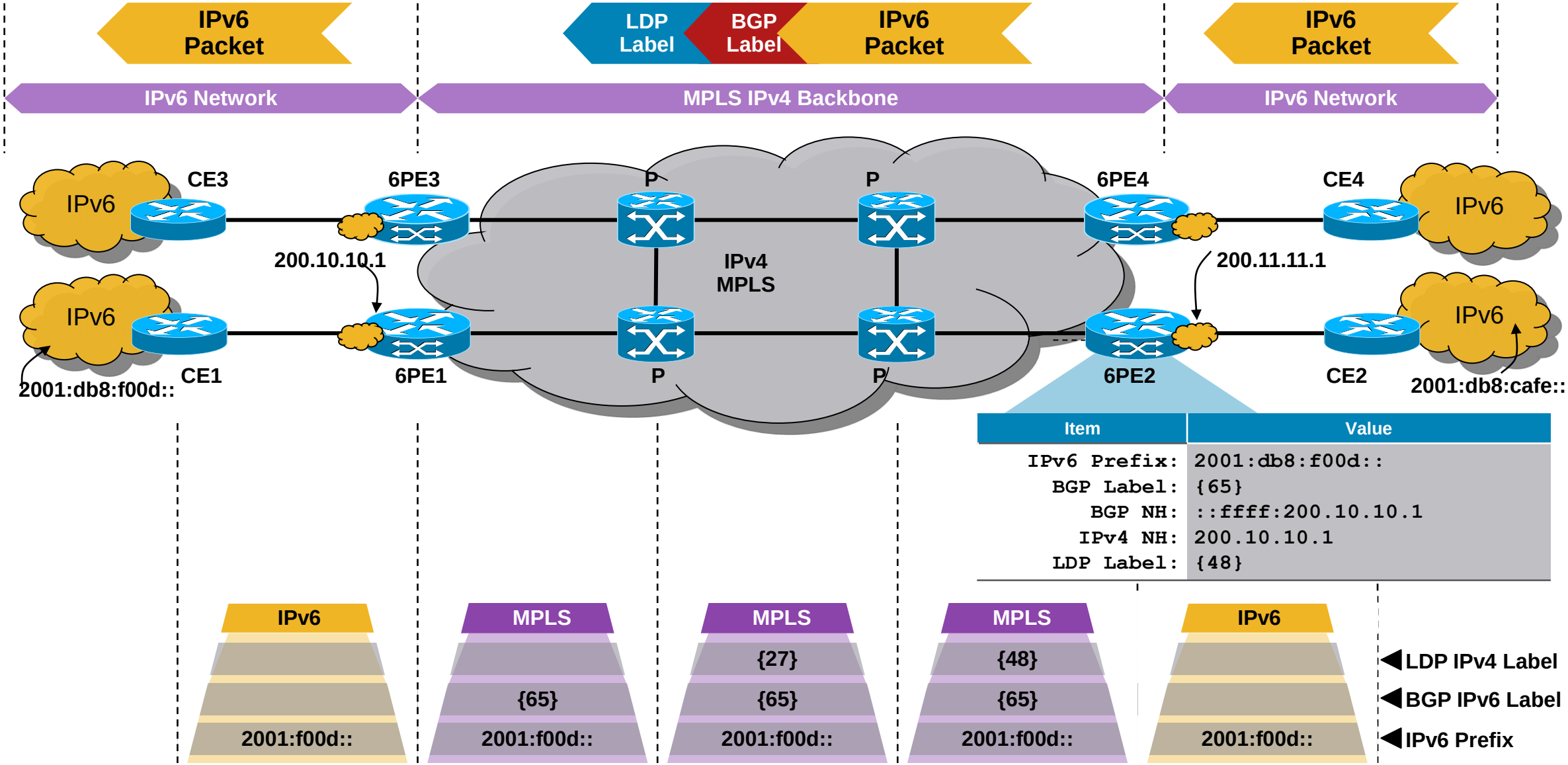


- Connects IPv6 islands over MPLS core (Transits edge to edge)
- Transition mechanism for providing unicast IPv6 access
- Coexistence mechanism for combining IPv4 and IPv6 services
- As other IPv6 “tunnel” technologies, enables services such as
  - IPv6 Internet Access
  - Peer-to-peer connectivity
  - Access to IPv6 services supplied by the SP itself

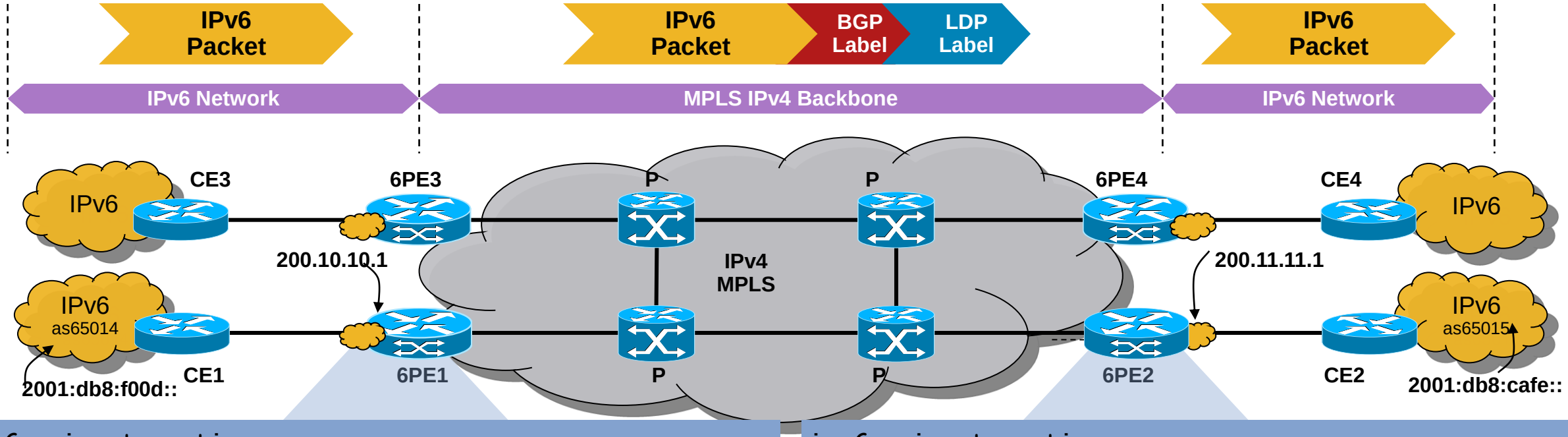
6PE Routing And Label Distribution Example



6PE Label Forwarding



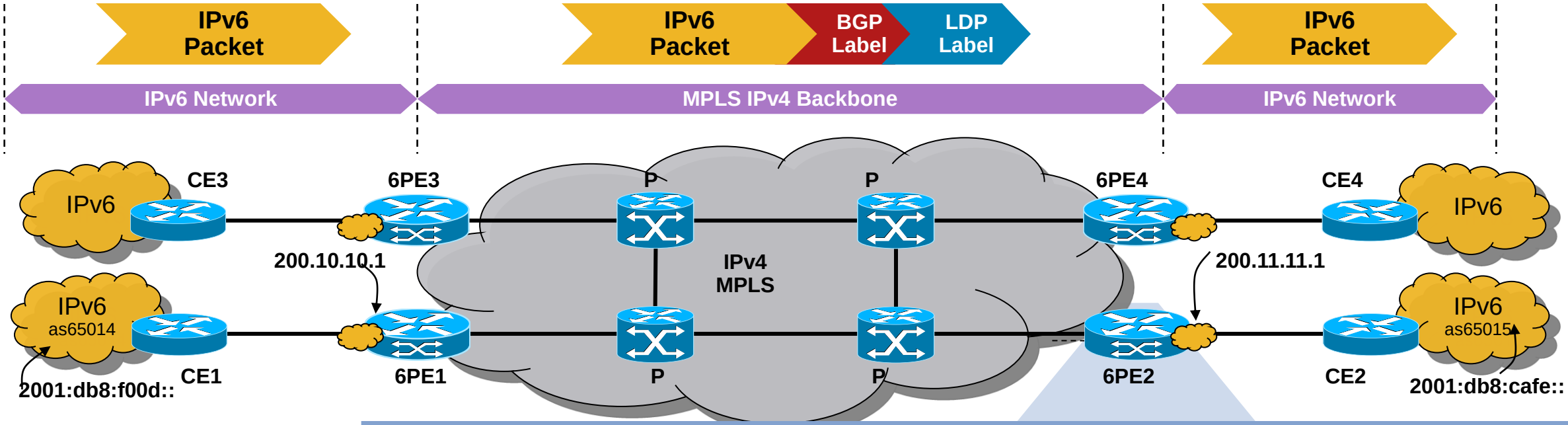
6PE Configuration




```
ipv6 unicast-routing
!
interface loopback0
 ip address 200.10.10.1 255.255.255.0
!
router bgp 65000
 neighbor 2001:db8:f00d:1::1 remote-as 65014
 neighbor 200.11.11.1 remote-as 65000
 neighbor 200.11.11.1 update-source lo0
!
address-family ipv6
 neighbor 200.11.11.1 activate ← 6PE2
 neighbor 200.11.11.1 send-label
 neighbor 2001:db8:f00d:1::1 activate ← CE1
```

```
ipv6 unicast-routing
!
interface loopback0
 ip address 200.11.11.1 255.255.255.0
!
router bgp 65000
 neighbor 2001:db8:cafe:1::1 remote-as 65015
 neighbor 200.10.10.1 remote-as 65000
 neighbor 200.10.10.1 update-source lo0
!
address-family ipv6
 neighbor 200.10.10.1 activate ← 6PE1
 neighbor 200.10.10.1 send-label
 neighbor 2001:db8:cafe:1::1 activate ← CE2
```

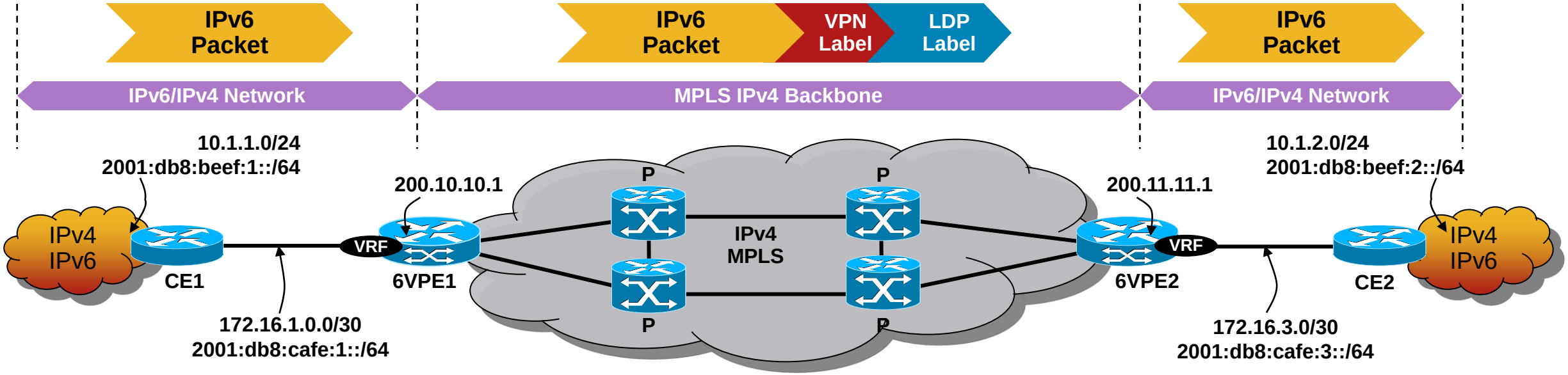
6PE IPv6 Route




```
6PE-2#show ipv6 route
B  2001:db8:f00d::/48 [200/0]
   via ::ffff:200.10.10.1, IPv6-mpls

6PE-1#show ipv6 cef internal
[snip]
2001:F00D::/64,
        nexthop ::ffff:200.10.10.1
 fast tag rewrite with F0/1, 10.12.0.1, tags imposed {48 65}
```

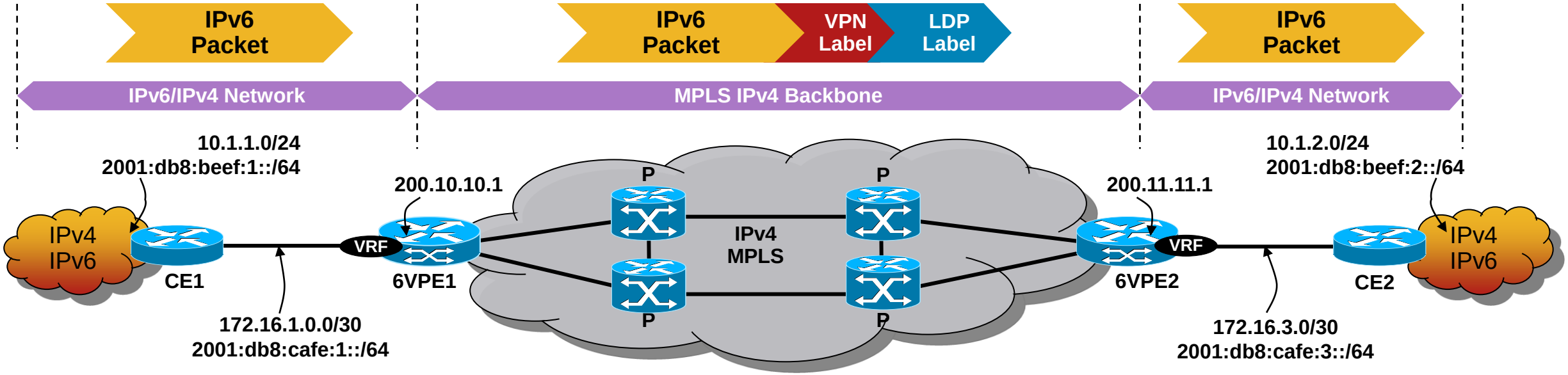
LDP BGP

IPv6 VPN 6VPE (RFC 4659)



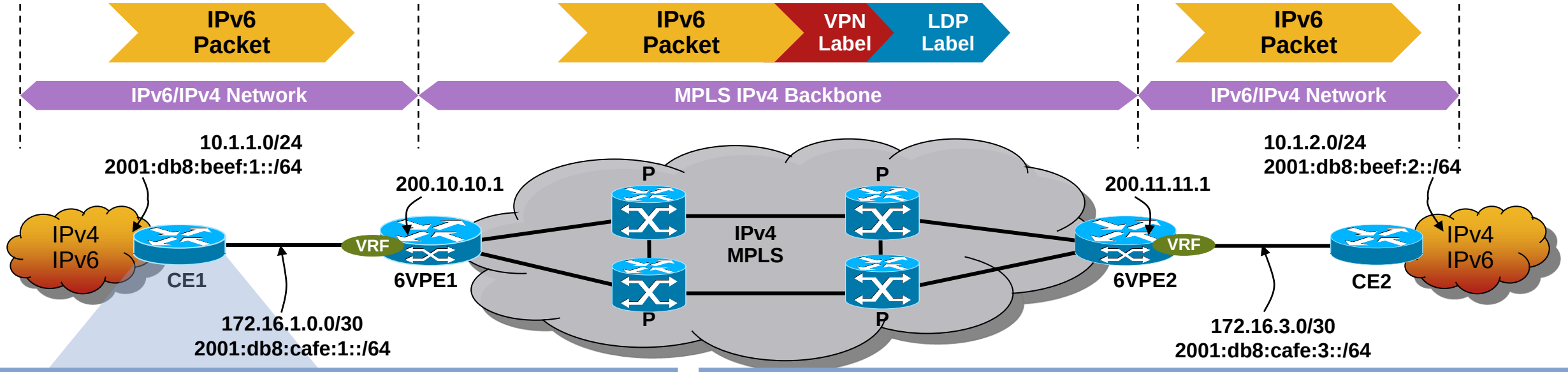
- 6VPE uses existing IPv4 MPLS infrastructure to provide IPv6 VPN
- Core uses IPv4 control plane (LDPv4, TEv4, IGPv4)
- PEs must support dual stack IPv4+IPv6
- Offers same architectural features as MPLS-VPN for IPv4
  - RTs, VRFs, RDs are appended to IPv6 to form VPNv6 address
  - MP-BGP distributed both VPN address families
  - BGP NH uses IPv4 to IPv6 mapped address format ::ffff:A.B.C.D
- VRF can contain both VPNv4 and VPNv6 routes

Services Using 6VPE



- For VPN customers, IPv6 VPN service is exactly as IPv4 VPN service
- 6PE is “like VPN” but prefixes are in global table, 6VPE is true VPN
- 6VPE enables services such as
  - IPv6 VPN Access
  - Carriers Supporting Carrier
  - Access to IPv6 services supplied by the SP itself

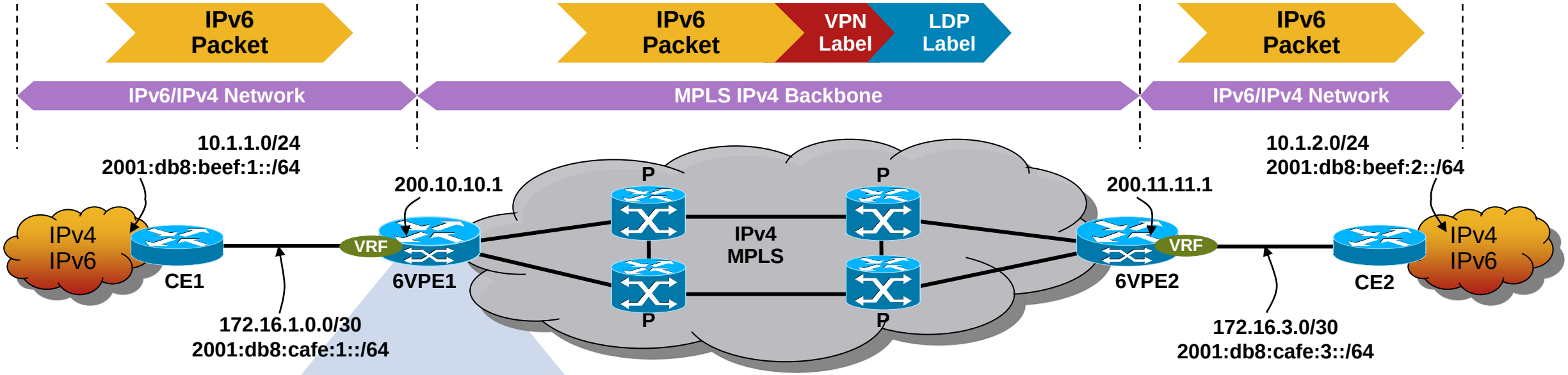
CE1 Configuration




```
ipv6 unicast-routing
ipv6 cef
!
interface Ethernet0/0
 description Link to PE1
 ip address 172.16.1.1 255.255.255.0
 ipv6 address 2001:db8:cafe:1::1/64
!
interface Ethernet1/0
 description to GREEN LAN
 ip address 10.1.1.1 255.255.255.0
 ipv6 address 2001:db8:beef:1::1/64
 ipv6 rip GREEN enable
```

```
router bgp 500
 neighbor 2001:db8:cafe:1::2 remote-as 100
 neighbor 172.16.1.2 remote-as 100
 !
 address-family ipv4
 redistribute eigrp 100
 neighbor 172.16.1.2 activate ← 6VPE1
 exit-address-family
 !
 address-family ipv6
 neighbor 2001:db8:cafe:1::2 activate ← 6VPE1
 redistribute rip GREEN
 exit-address-family
```

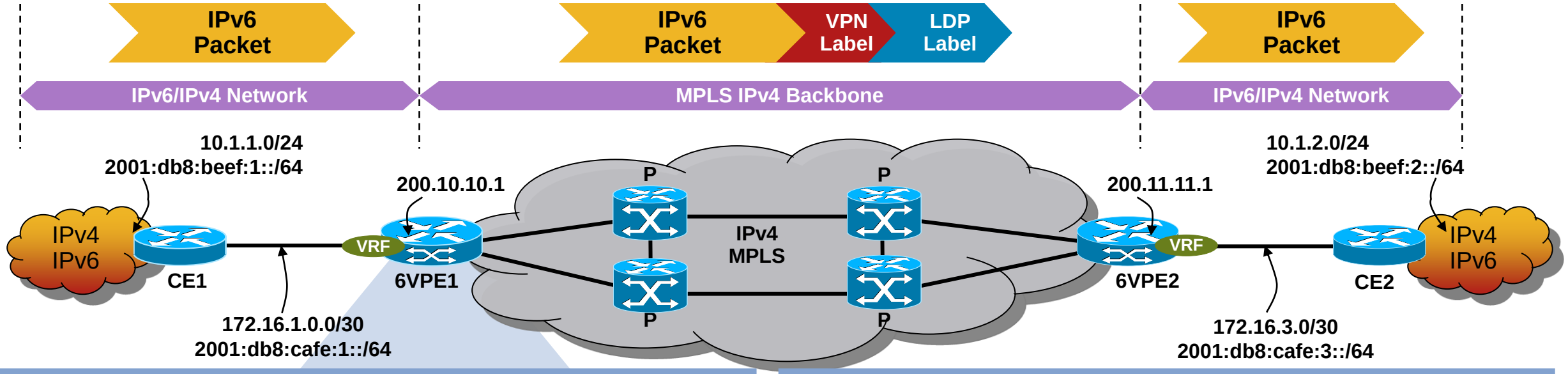
New Multi-AF VRF Configuration




```
vrf definition GREEN
 rd 200:1
  ! Common RT policies go here
 address-family ipv4
  route-target export 200:1
  route-target import 200:1
 exit-address-family
 !
 address-family ipv6
  route-target export 200:1
  route-target import 200:1
 exit-address-family
```

- New VRF AF definition
  - Allows address-families
  - Each with unique or common policies
- `vrf upgrade-cli multi-af-mode {common-policies | non-common-policies} [vrf <name>]`
  - This command can update existing VRF definitions

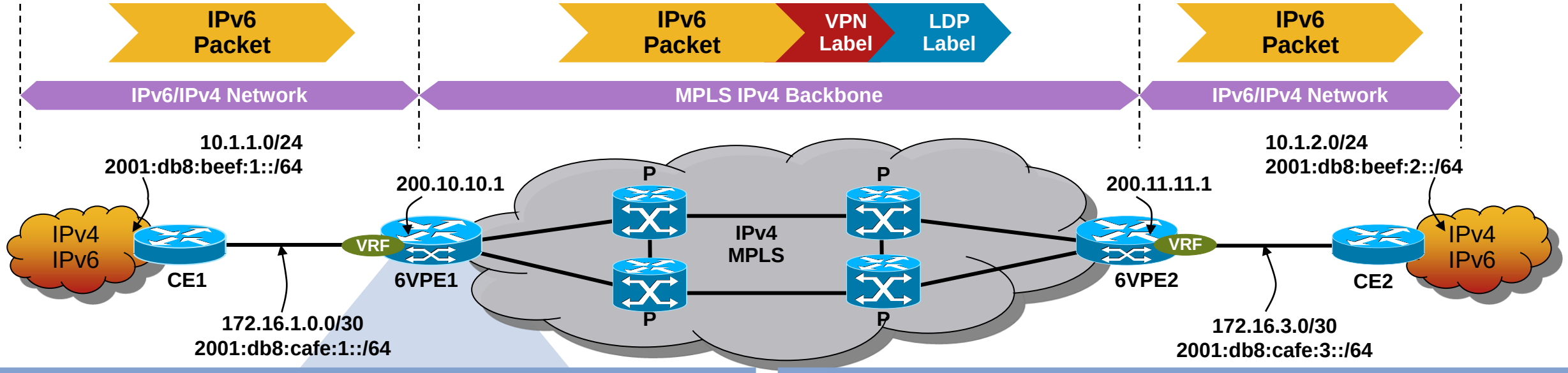
6VPE1 General Configuration




```
ipv6 unicast-routing
ipv6 cef
!
interface Loopback0
 ip address 200.10.10.1  255.255.255.255
!
interface Ethernet0/0
 Description Link to CE1
 vrf forwarding GREEN
 ip address 172.16.1.2 255.255.255.0
 ipv6 address 2001:db8:cafe:1::2/64
```

```
!
interface Ethernet2/0
 description Link to Core Network
 ip address 192.168.1.1 255.255.255.252
 mpls ip
!
router ospf 1
 log-adjacency-changes
 redistribute connected subnets
 passive-interface Loopback0
 network 192.168.1.0 0.0.0.255 area 0
```

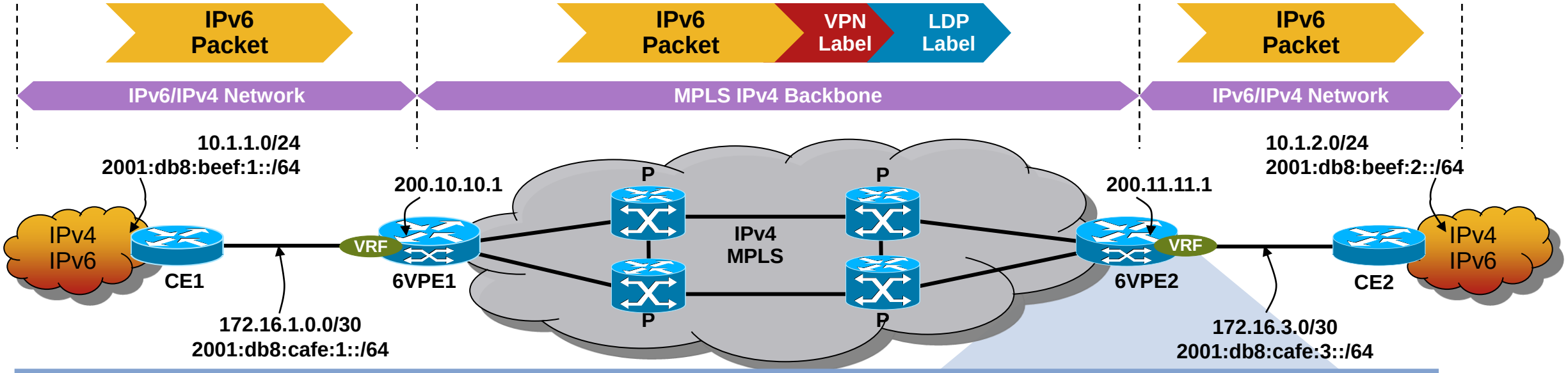
6VPE1 BGP Configuration




```
router bgp 100
 neighbor 200.11.11.1 remote-as 100
 neighbor 200.11.11.1 update-source lo0
 !
 address-family ipv4 ← Internet Routes
 neighbor 200.11.11.1 activate
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family vpnv4 ← To 6VPE2
 neighbor 200.11.11.1 activate
 neighbor 200.11.11.1 send-community ext
 exit-address-family
```

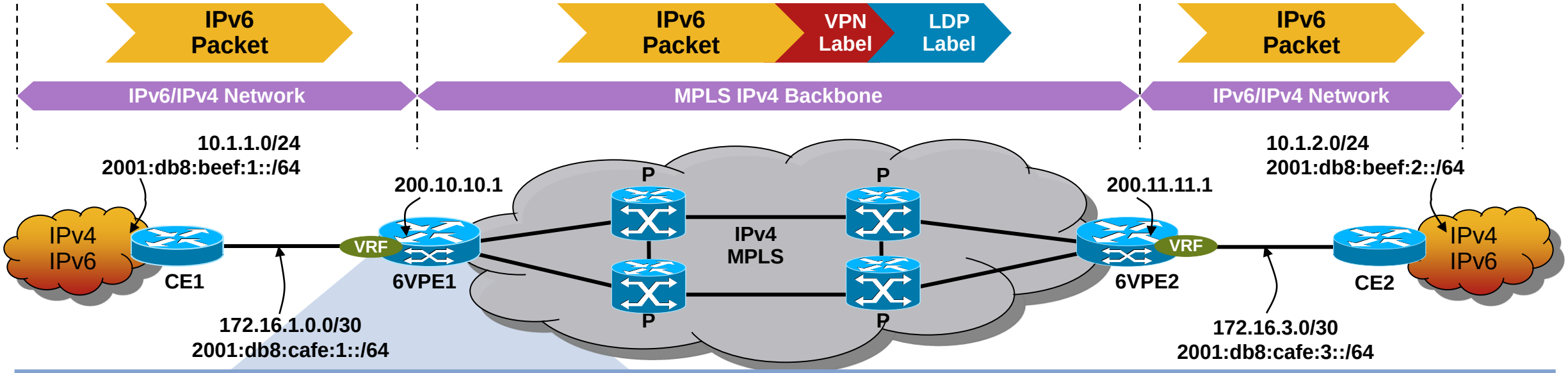
```
 address-family vpnv6 ← To 6VPE2
 neighbor 200.11.11.1 activate
 neighbor 200.11.11.1 send-community ext
 exit-address-family
!
 address-family ipv4 vrf GREEN ← To CE1
 redistribute connected
 neighbor 172.16.1.1 remote-as 500
 neighbor 172.16.1.1 activate
 exit-address-family
 !
 address-family ipv6 vrf GREEN ← To CE1
 neighbor 2001:db8:cafe:1::1 remote-as 500
 neighbor 2001:db8:cafe:1::1 activate
 exit-address-family
```

6VPE2 IPv6 VRF Routes



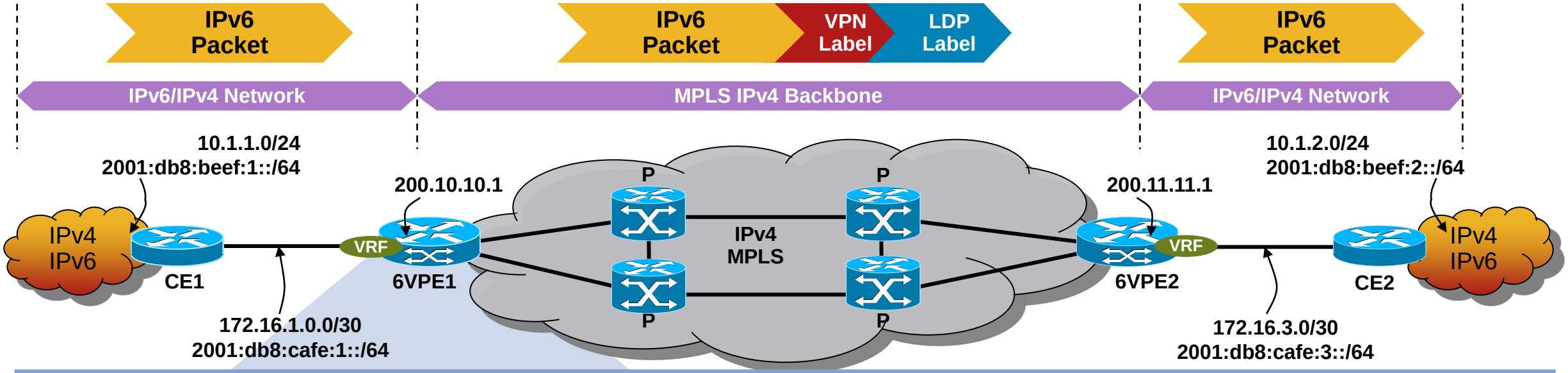

```
6VPE2#show ipv6 route vrf GREEN
B   2001:db8:beef:1::/64 [200/0]
    via 200.10.10.1
B   2001:db8:beef:2::/64 [20/0]
    via FE80::A8BB:CCFF:FE01:FA00, Ethernet1/0
B   2001:db8:cafe:1::/64 [200/0]
    via 200.10.10.1
C   2001:db8:cafe:3::/64 [0/0]
    via Ethernet1/0, directly connected
L   2001:db8:cafe:3::2/128 [0/0]
    via Ethernet1/0, receive
L   FF00::/8 [0/0]
    via Null0, receive
```

6VPE1 BGP VPNv6 Table




```
6VPE1#show bgp vpnv6 unicast all
Network              Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 200:1 (default for vrf GREEN)
*> 2001:db8:beef:1::/64
                     2001:db8:cafe:1::1  ← Route from CE1
                                          0             0 500 ?
*>i2001:db8:beef:2::/64
                     ::FFFF:200.11.11.1  ← Route from CE2 via 6VPE2
                                          0    100      0 506 ?
*>i2001:db8:cafe:3::/64
                     ::FFFF:200.11.11.1  ← PE/CE Connected route from 6VPE2
                                          0    100      0 ?
```

6VPE1 LFIB




```
6VPE1#show mpls forwarding
Local   Outgoing      Prefix            Bytes Label   Outgoing    Next Hop
Label   Label or VC   or Tunnel Id      Switched      interface
16      Pop Label     192.168.1.4/30    0             Et2/0       192.168.1.2
17      16            192.168.1.8/30    0             Et2/0       192.168.1.2
18      Pop Label     200.12.12.1/32    0             Et2/0       192.168.1.2
19      18            200.13.13.1/32    0             Et2/0       192.168.1.2
20      19            200.11.11.1/32    0             Et2/0       192.168.1.2
21      No Label      10.1.1.0/24[V]    0             Et0/0       172.16.1.1
22      Aggregate     172.16.1.0/24[V]  570           GREEN
25      No Label      2001:db8:beef:1::/64[V]    \
                                        570           Et0/0       FE80::A8BB:CCFF:FE01:F400
26      Aggregate     2001:db8:cafe:1::/64[V]    \
                                        35456         GREEN
```

Cisco Carrier-Grade IPv6

63

Cisco Carrier-Grade IPv6 (CGv6)

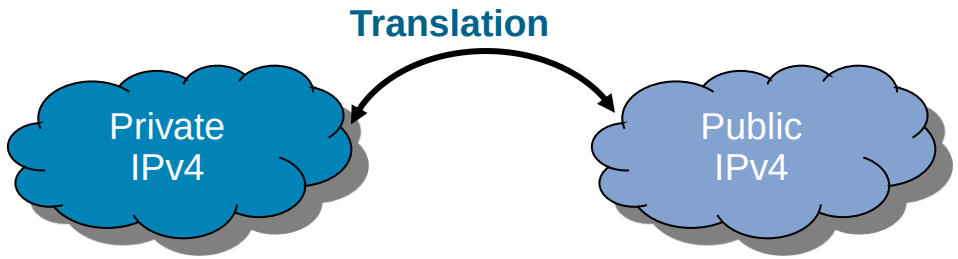
- CGV6 is a Cisco framework for moving to IPv6
  - Consisting of three pronged approach Preserve, Prepare, Prosper
- Preserve IPv4 investments and assets
  - Objective: Support continued use of IPv4 after address exhaustion
  - Solutions: Carrier Grade NAT (Millions NAT44, NAT444)
  - This is a short to medium term solution
- Prepare to deliver interoperable IPv6 services
  - Objective: Upgrade to IPv6 whilst co-existing with IPv4
  - Solutions: Dual-Stack, 6PE/6VPE, Softwire Mesh, Translation (AFT)
- Prosper from accelerated growth and innovation
  - Objective: IPv6 end-to-end for most devices/things
  - Solution: Mostly dual-stack and standalone IPv6 clouds

Cisco Carrier-Grade IPv6 (CGv6)

- Built on carrier-scale address translation and protocol tunneling capability

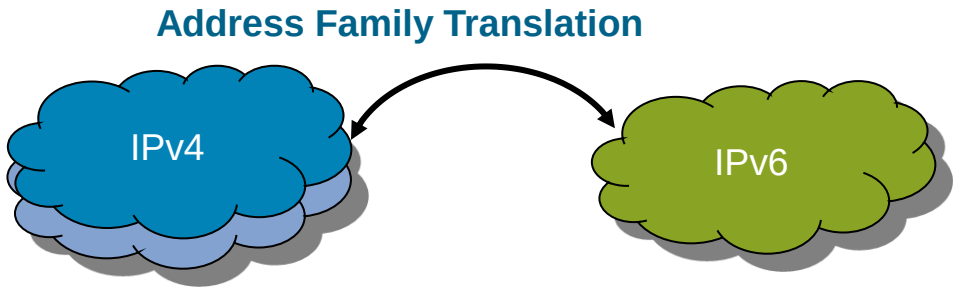
- Preserve

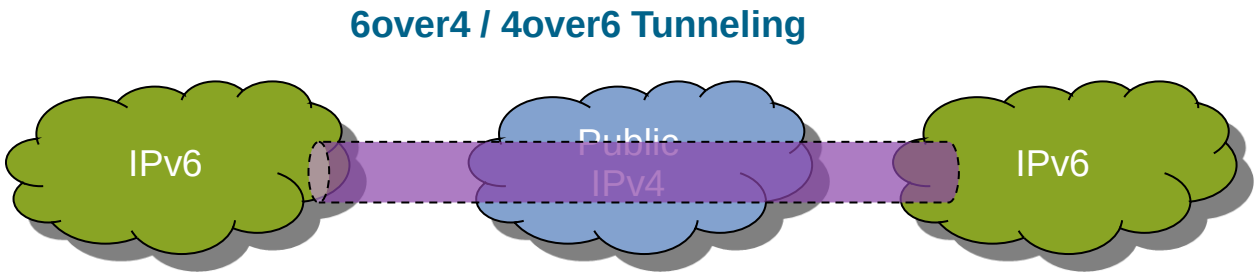
  By extending Private IP into the IP-NGN for continued growth



- Prepare

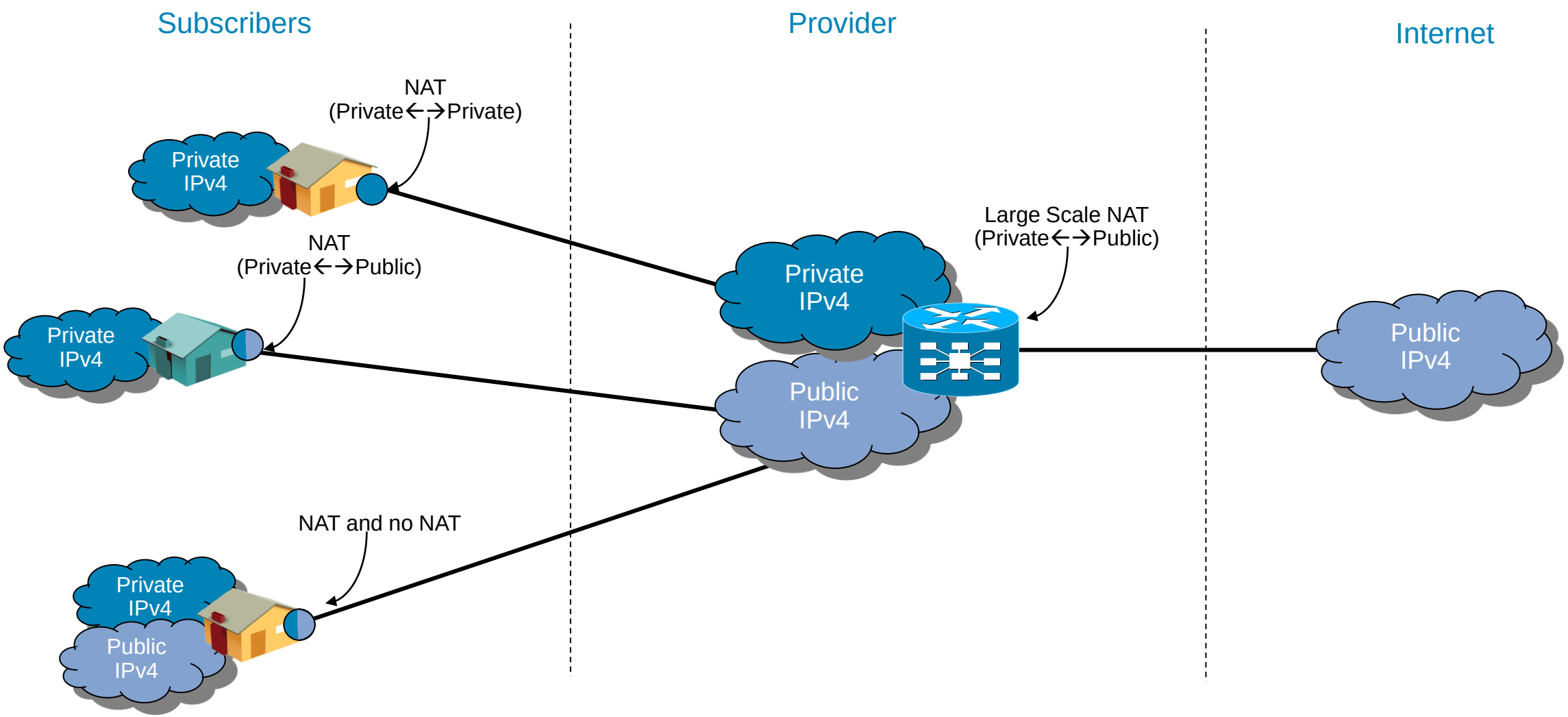
  By enabling interoperable IPv4/IPv6 services over existing IP-NGN infrastructure





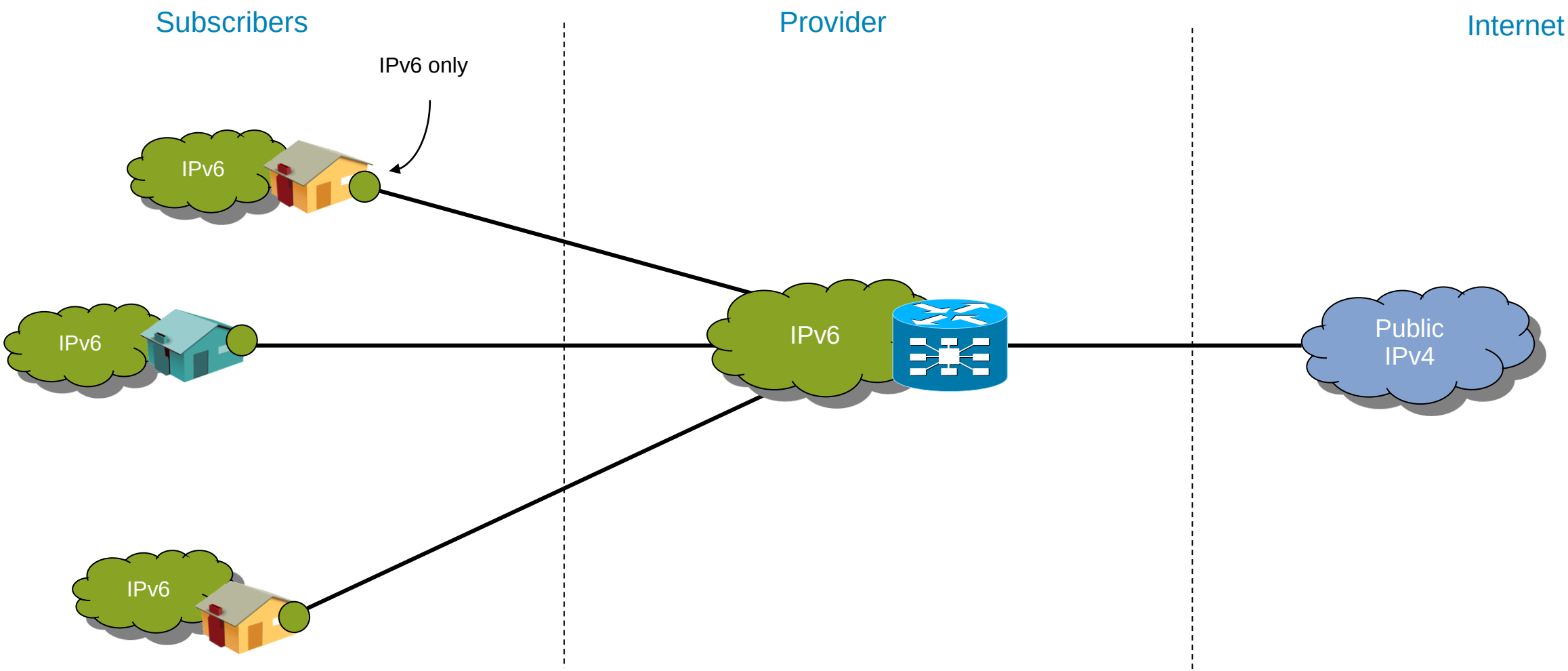
Preserve - Double NAT

- Provides additional IPv4 space for the short term



Prepare - Address Family Translation (AFT)

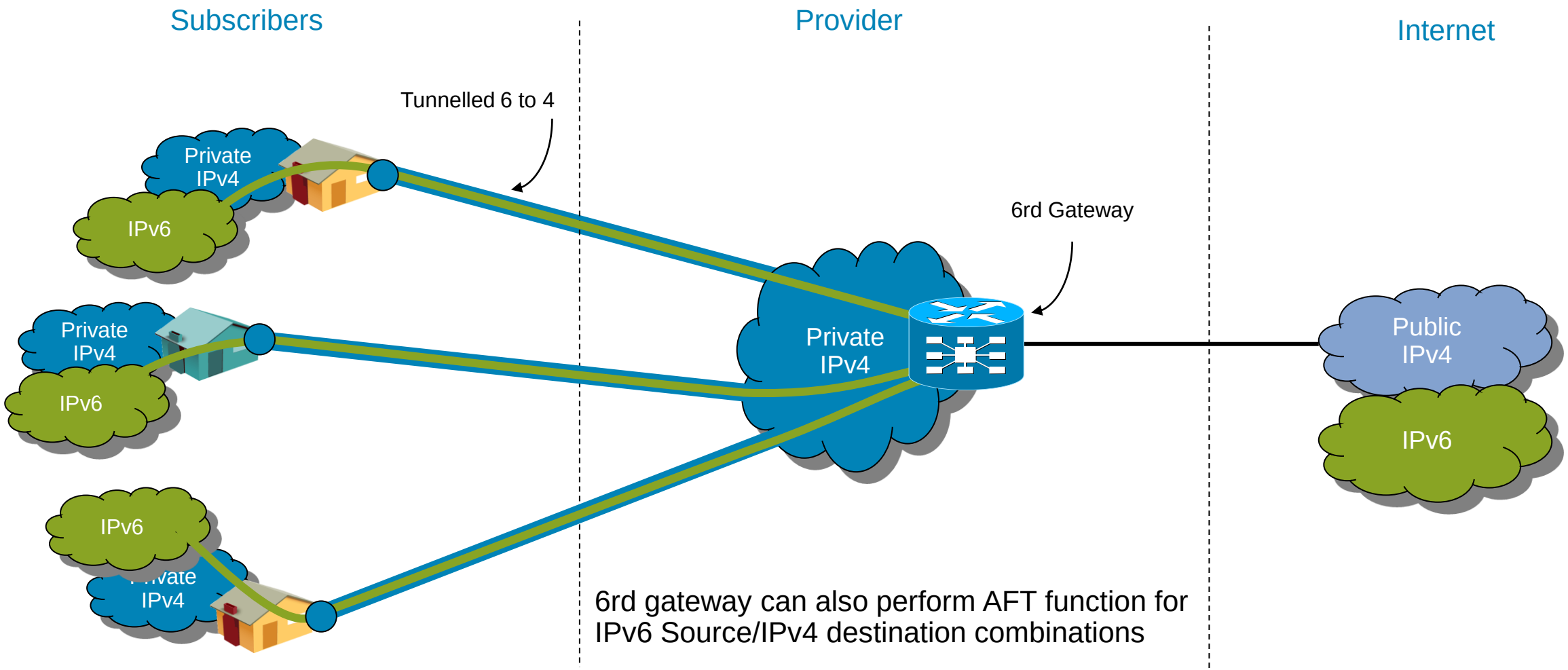
- Allows access between IPv6 and IPv4 networks (IETF BEHAVE)



IETF BEHAVE working group on AFT for NAT64 and DN64

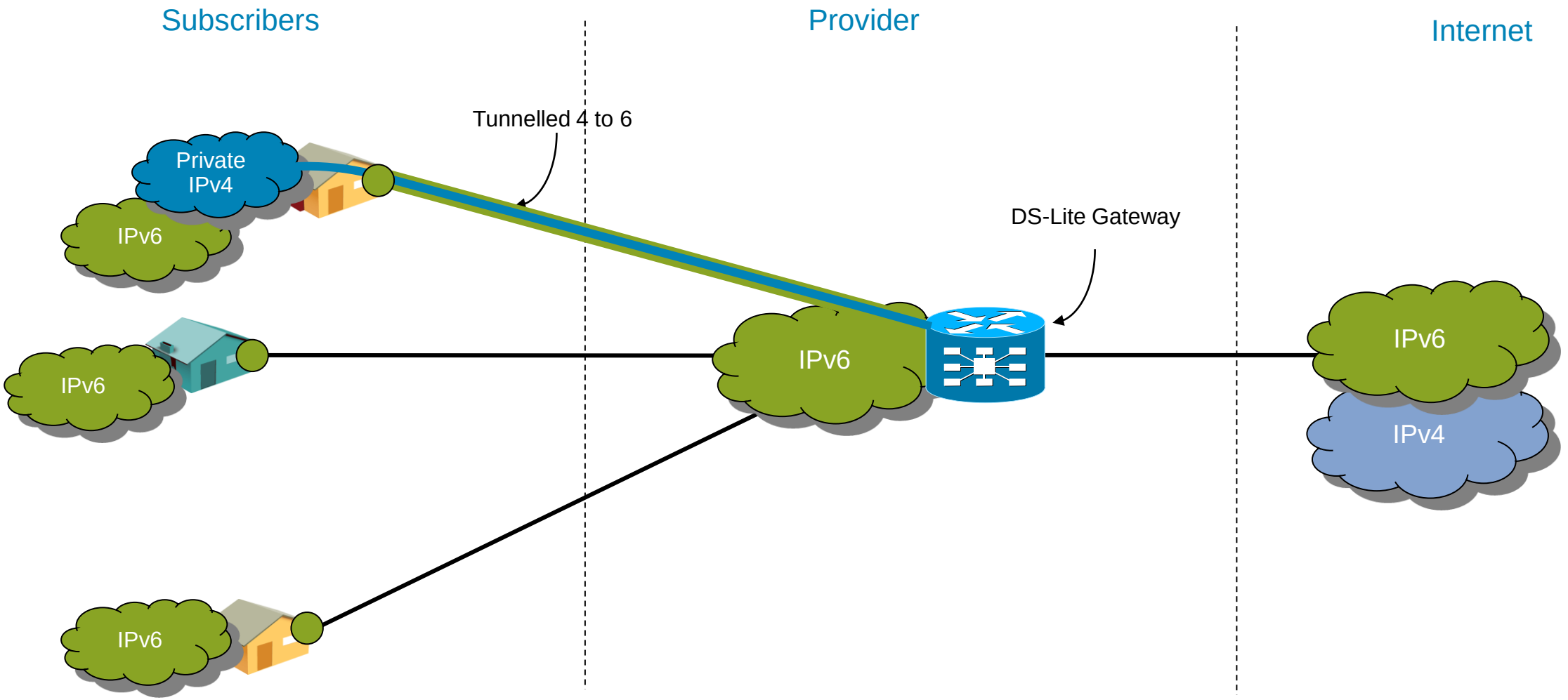
Prepare - IPv6 Rapid Deployment (6rd)

- Tunnel IPv6 over IPv4 access network to 6rd gateway



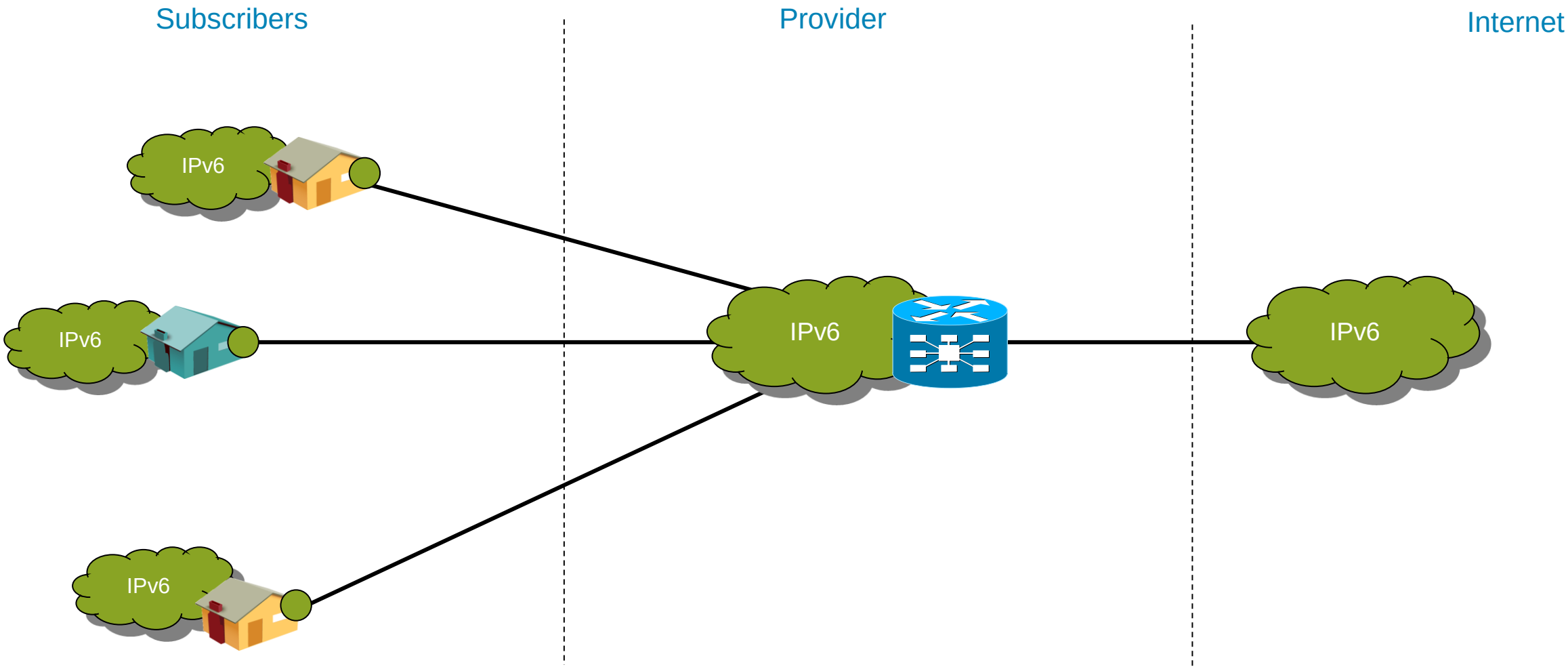
Prepare - Dual-Stack Lite (4Over6, 4rd)

- Tunnel IPv4 over IPv6 access network to DS gateway (more v6 than v4)



Prosper - All IPv6

- All IPv6 is most likely decades away



The Carrier-Grade Services Engine

- CGSE - engine for massive Cisco CGv6 deployments
  - 20+ million active translations
  - 100s of thousands of subscribers
  - 1+ million connections per second
  - 20Gb/s of throughput per CGSE

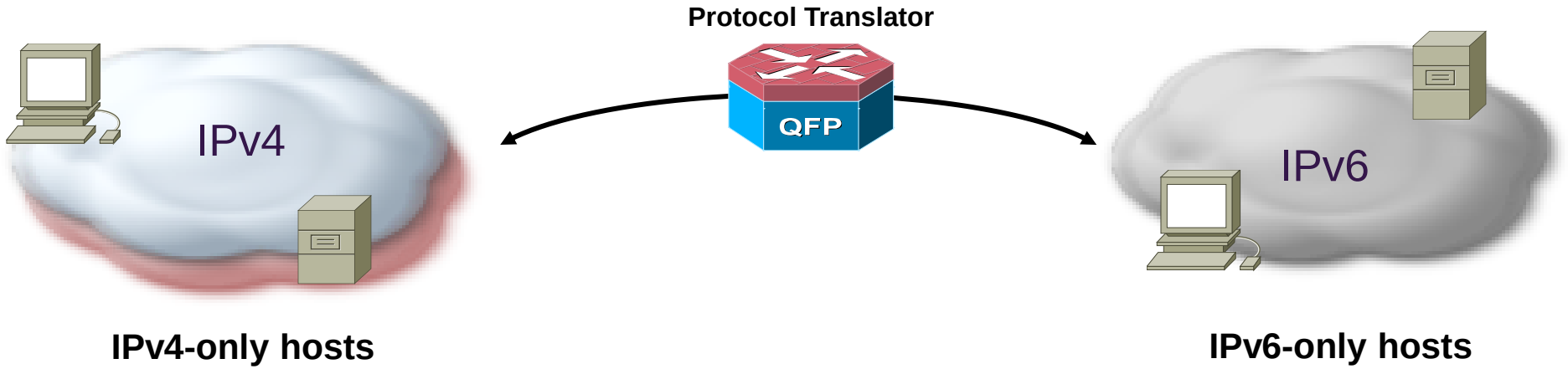
Cisco CGSE

Cisco CRS

CGN (NAT44/NAT64/6rd) also supported on Cisco ASR1k

NAT64 Summary

- Translate between IPv6 and IPv4



- Choice of Translation or Tunneling?
  - Always choose tunnel
  - Translation (NAT) has worse side effects

http://datatracker.ietf.org/wg/behave/

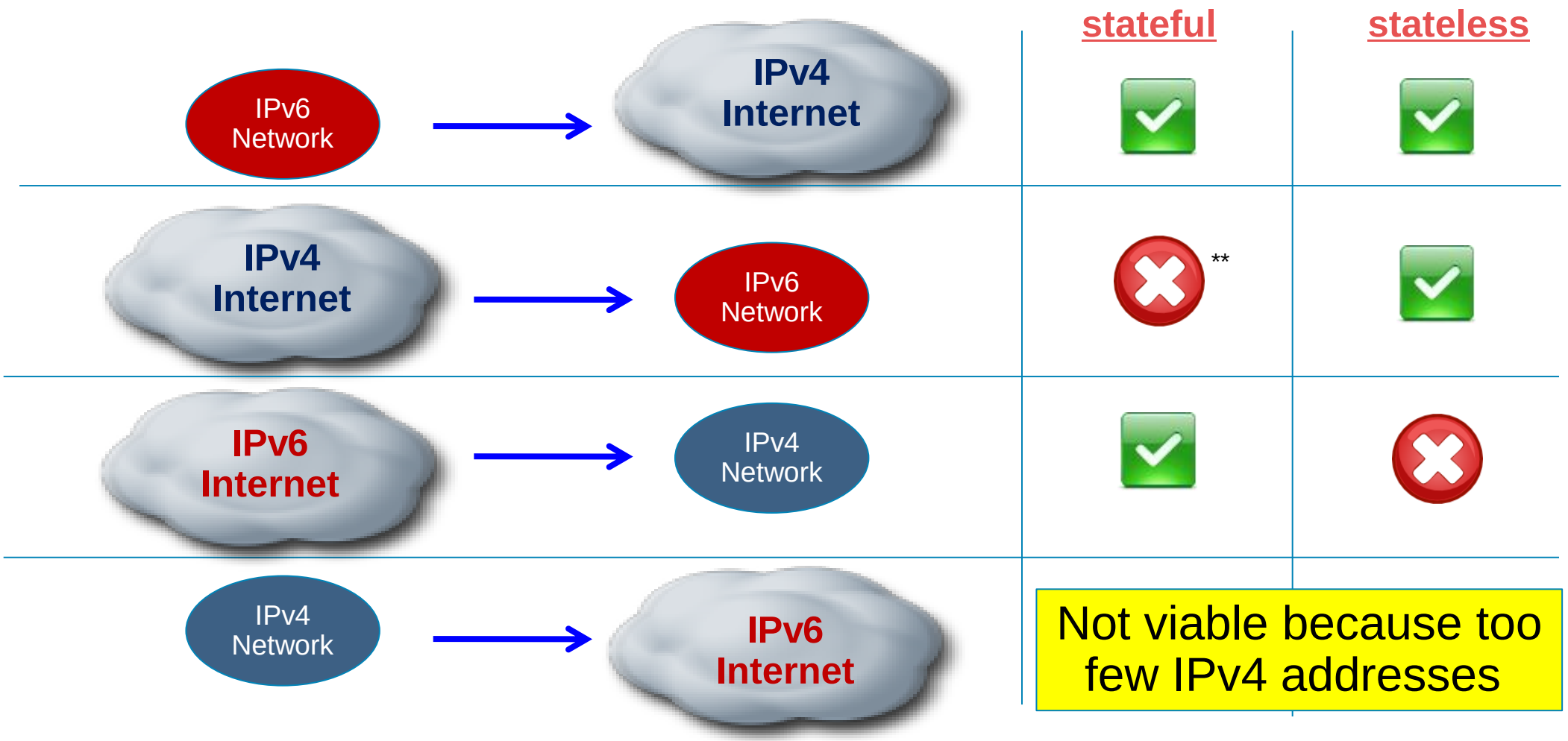
NAT-PT: The Historic Way to Translate 6/4

- NAT-PT combined all scenarios
  - IPv4 to IPv6 is problematic; IPv6 space is bigger
  - Broke DNSSEC
- RFC4966 said IPv6/IPv4 translation causes other side effects
  - And some are not solvable

But:

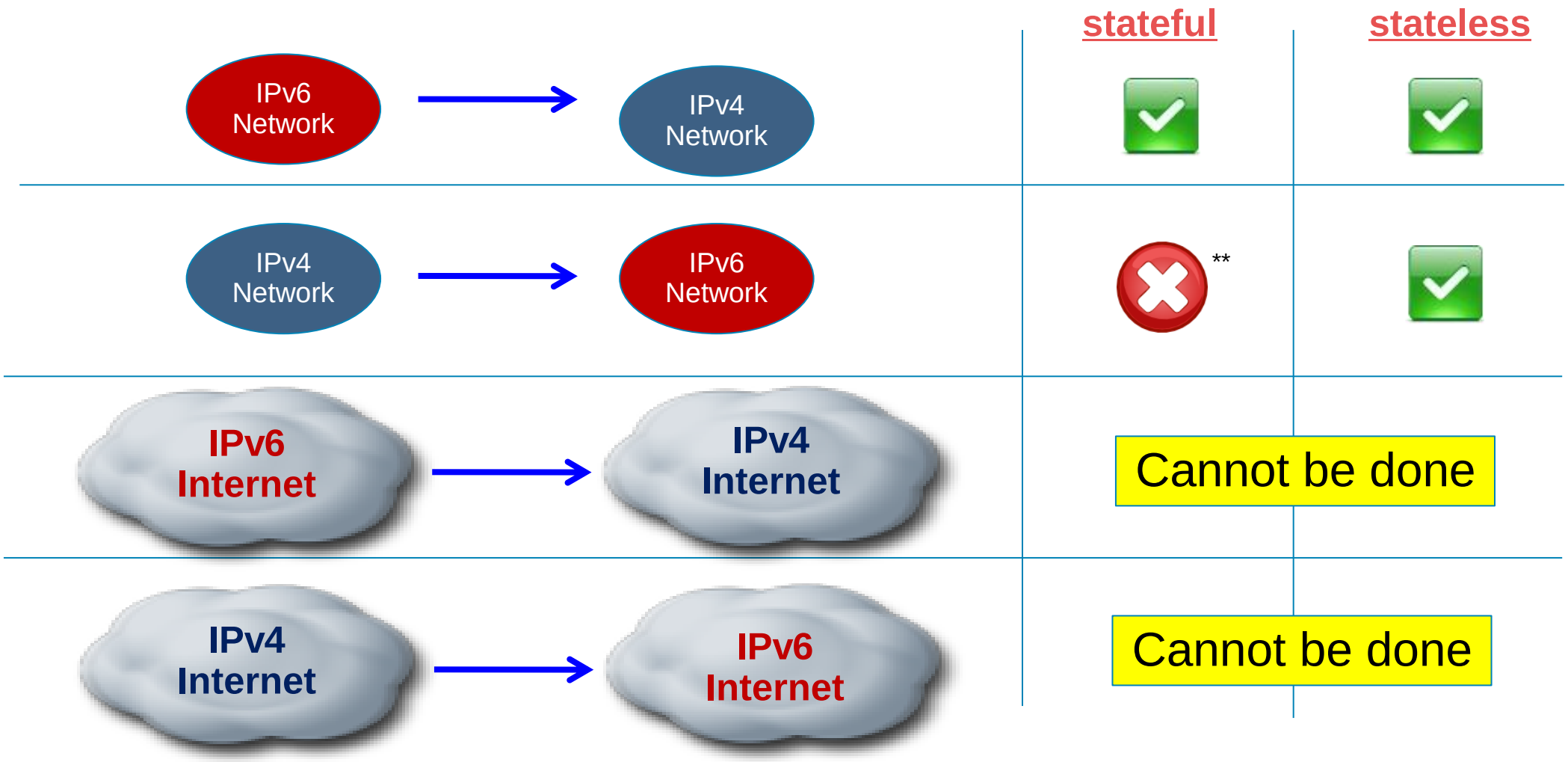
- IPv4 addresses near exhaustion
- Effectively no IPv6 Internet access and no IPv6 content anywhere in the world
- We can’t tunnel everywhere

IPv4/IPv6 Translation Framework Scenarios



**Possible with nat64 v4v6 static mappings

IPv4/IPv6 Translation Framework Scenarios



**Possible with nat64 v4v6 static mappings

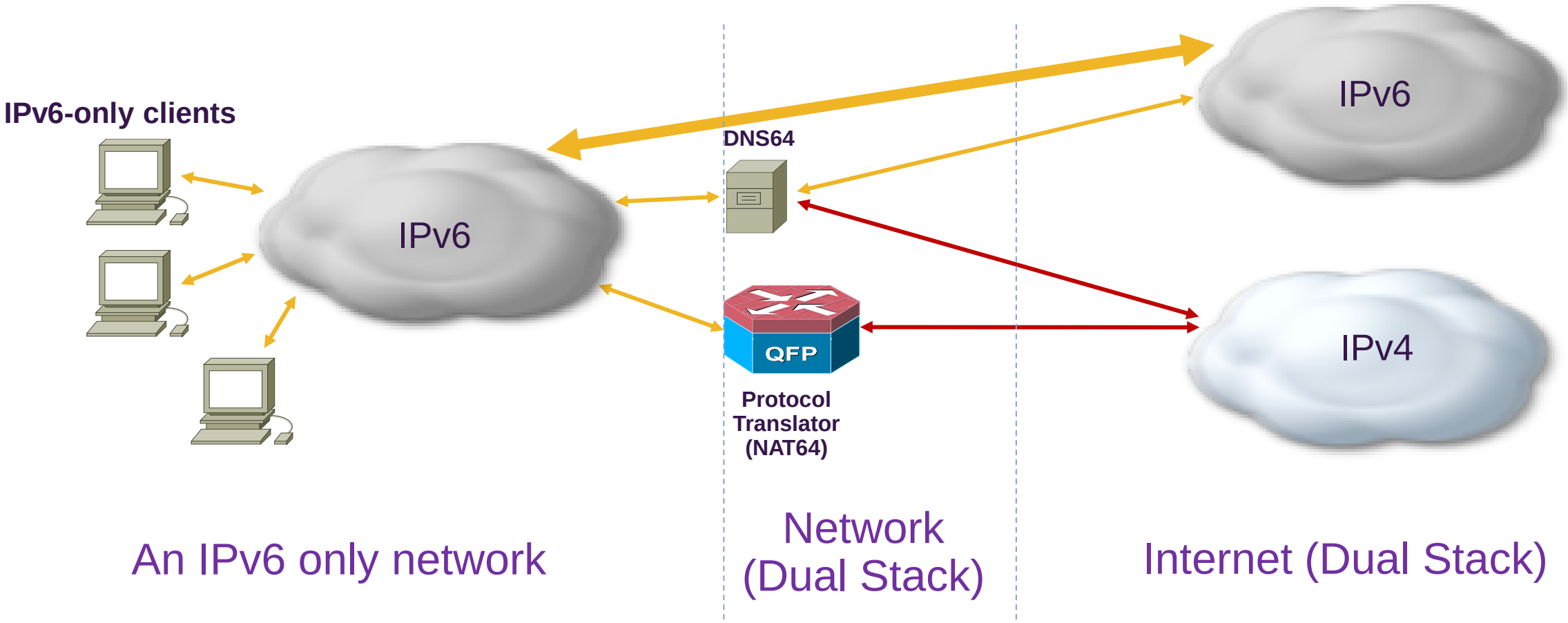
IPv6/IPv4 Translation: Two Scenarios

- Connecting an IPv6 network to the IPv4 Internet
  - You built an IPv6-only network, and want to access servers on the IPv4 Internet
  - Example: IPv6-only LTE handsets
- Connecting the IPv6 Internet to an IPv4 network
  - You have IPv4 servers, and want them available to the IPv6 Internet
  - Example: IPv4-only datacenter (HTTP servers)

IPv6/IPv4 Translation

Stateless	Stateful
1:1 translation	N:1 translation
NAT	NAPT
Any Protocol	TCP, UDP, ICMP
Helps ensure end-to-end address transparency and scalability	Uses address overloading; hence lacks end-to-end address transparency
No state or bindings created on the translation	State or bindings created on every unique translation
Session can be initiated from either side	Session must be initiated from IPv6 side
Requires IPv4-translatable IPv6 address assignment (mandatory requirement)	No requirement for the characteristics of IPv6 address assignment
Requires either manual or Domain Host Configuration Protocol Version 6 (DHCPv6)-based address assignment for IPv6 hosts	Capability to choose any mode of IPv6 address assignment: manual, DHCPv6, or stateless address auto-configuration (SLAAC)
No IPv4 address savings (Just like NAT)	Saves IPv4 addresses

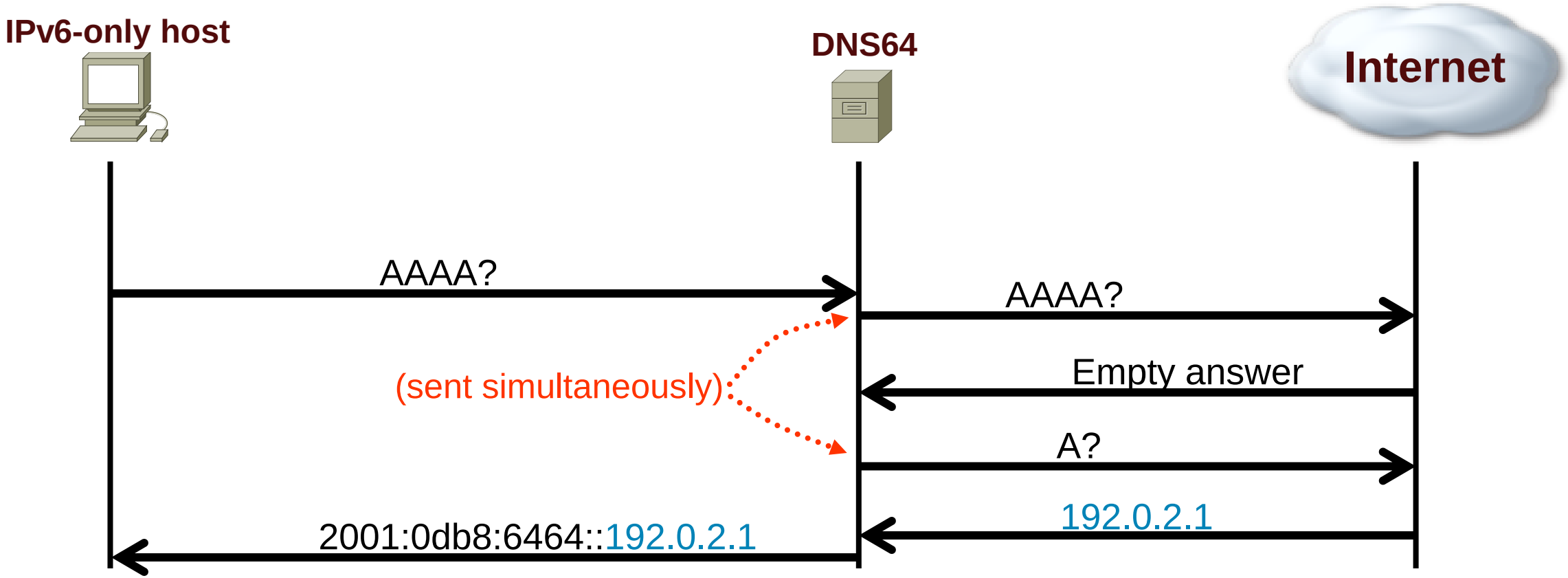
Connecting an IPv6 Network to the IPv4 Internet



http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6553/white_paper_c11-676278.html

DNS64

- Synthesizes AAAA records when AAAA are not present in the DNS
  - With IPv6 prefix of NAT64 translator



DNS64

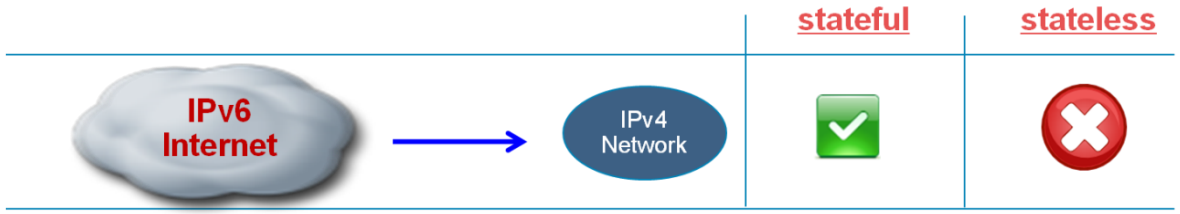
- Works for applications that do DNS queries
  - http://www.example.com
  - Well over 80% of applications.
- Breaks for applications that don't do DNS queries
  - http://1.2.3.4
  - SIP, RTSP, H.323, etc. – IP address literals
- Solutions:
  - Application-level proxy for IP address literals (HTTP proxy)
  - IPv6 application learns NAT64's prefix

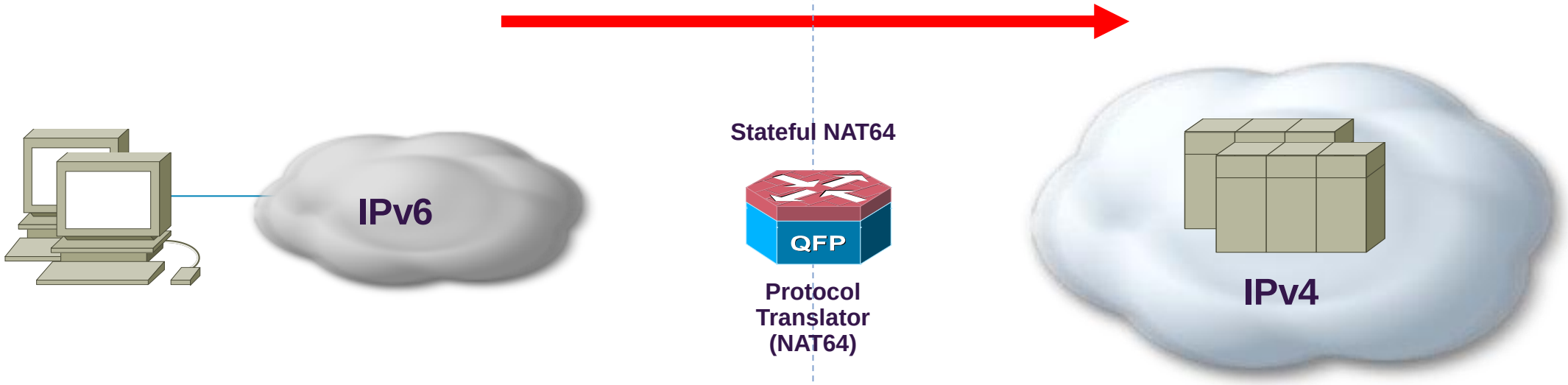
IPv6/IPv4 Translation Issues

- IPv4 address literals
  - http://1.2.3.4, SIP, RTSP, etc.
- Application Layer Gateway, or application proxy
  - FTP (EPSV, PASV)
  - RTSP in mobile environments (3G)
  - Others applications?

draft-ietf-behave-ftp64

IPv6 Into IPv4-Only Datacenter



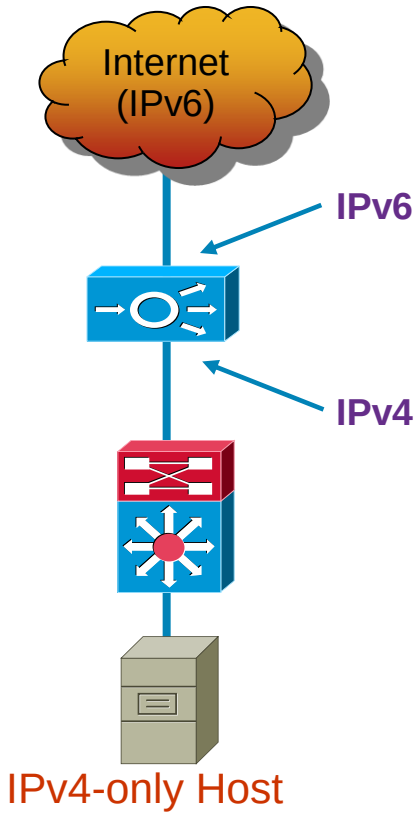


IPv6 clients

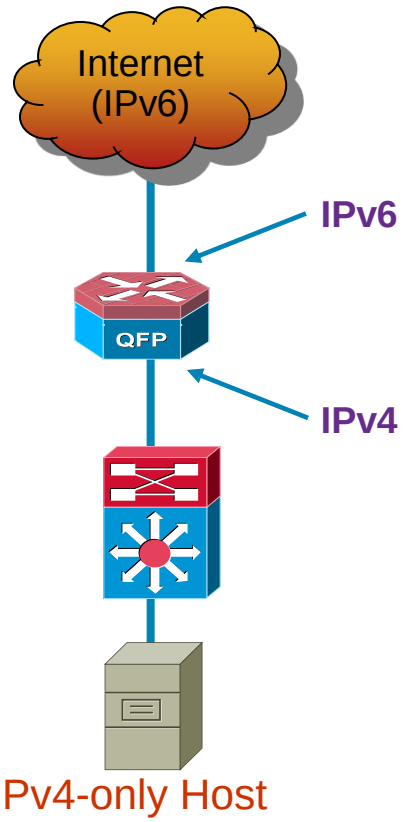
IPv4 Datacenter

What if I can’t Dual Stack my edge ?

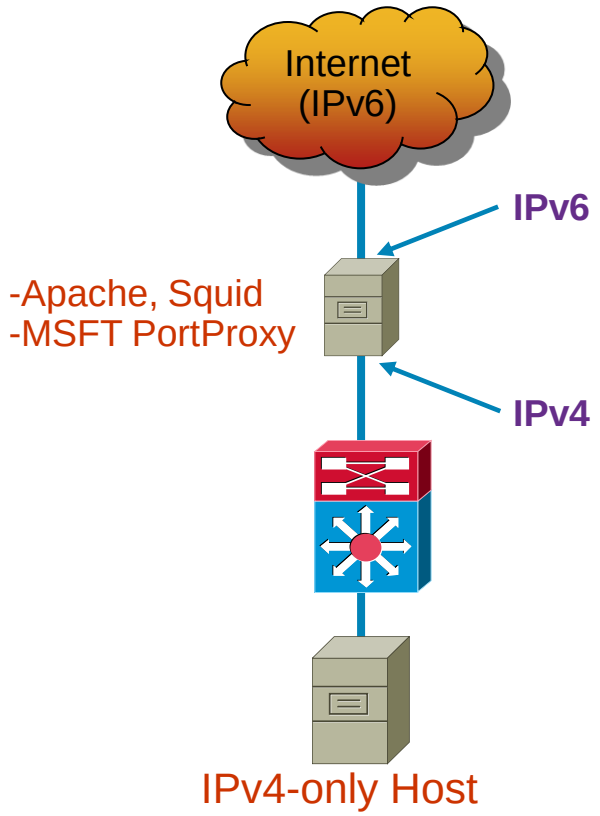
Server Load Balancer



Stateful NAT64

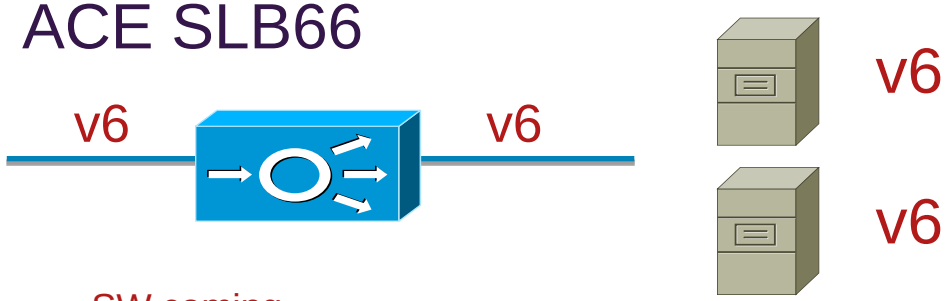


Proxy

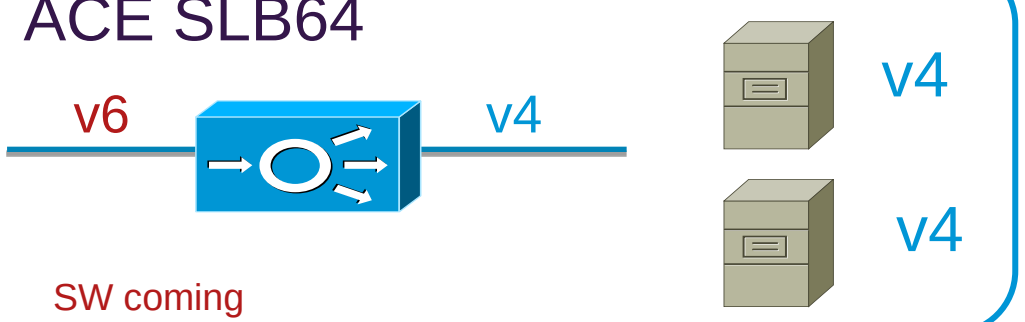


ACE + IPv6 / ASR + NAT64

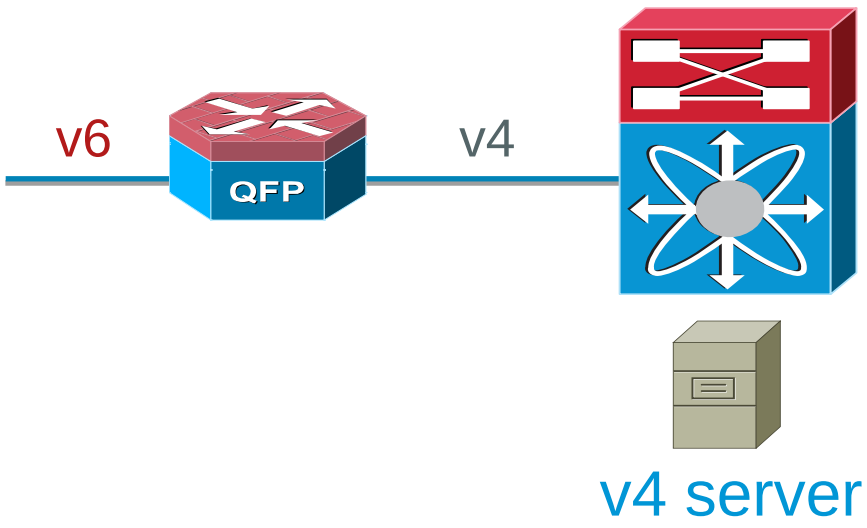
ACE SLB66



ACE SLB64



Stateful NAT64

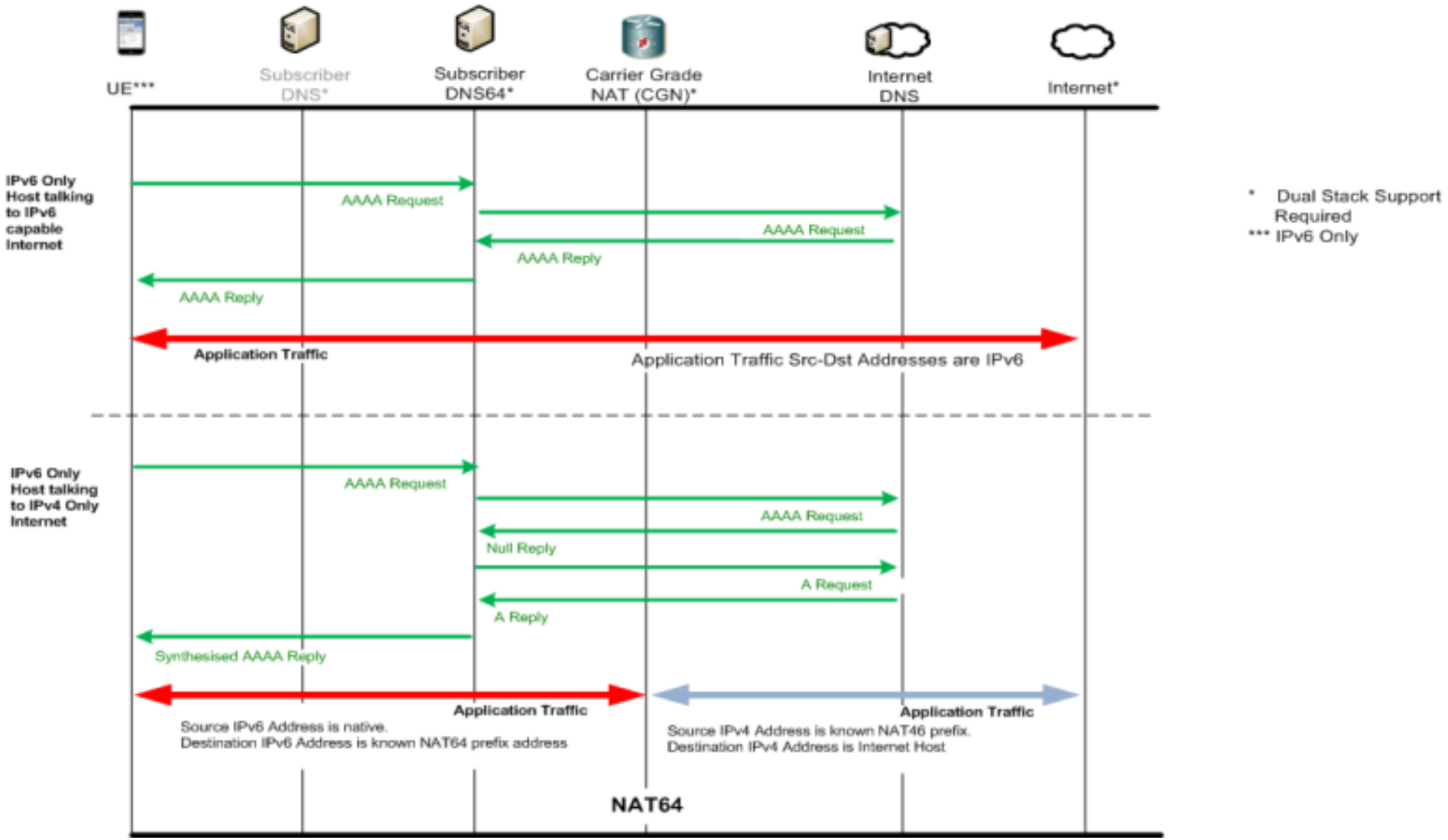


- ACE SLB64 offers HTTP header rewrite for ‘X-Forwarded-For’
  - Server can log IPv6 address of source host
- Router based NAT64 has no HTTP header rewrite support

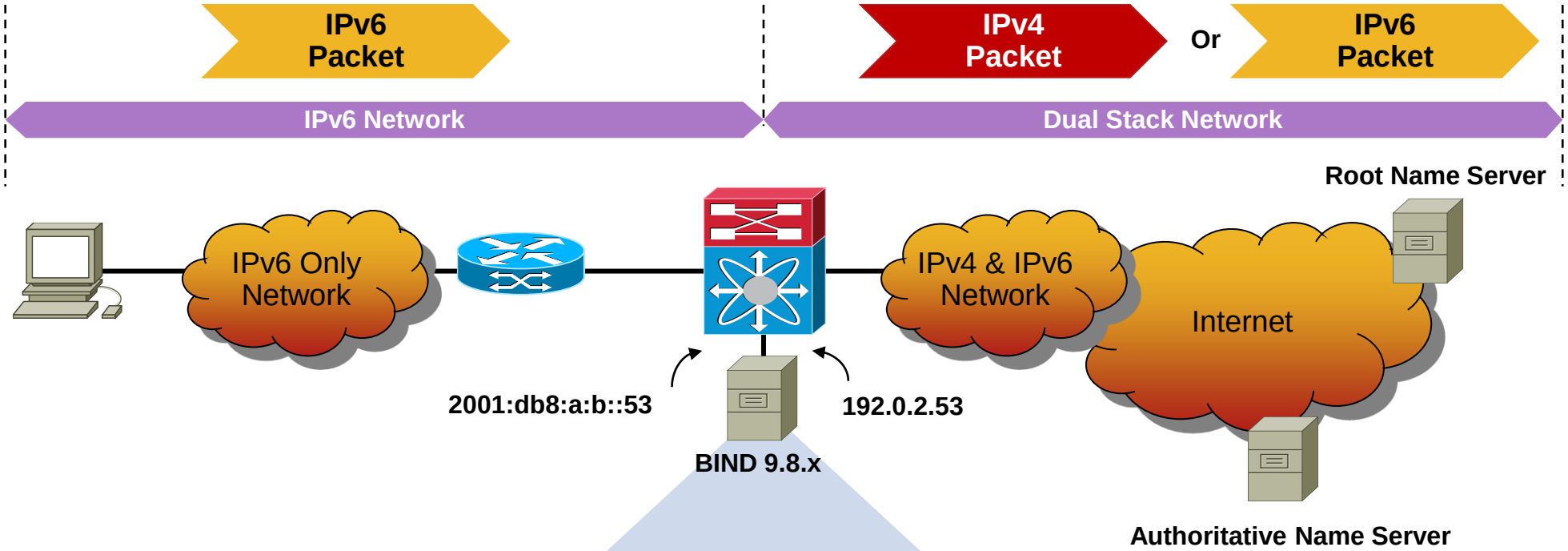
Issues With NAT64 for Data Center

- Makes IPv4-only servers accessible on the IPv6 Internet
- Requires stateful translation
  - Because IPv6 Internet is bigger than IPv4
  - (can’t represent every address in IPv4)
- All connections come from translator’s IPv4 address
  - Problem for abuse logging
  - Lack of X-Forwarded-For: header
- Maybe application proxy is superior?
  - e.g., lighthttpd
  - But has poor TLS interaction

DNS64 flows



DNS64 example Topology




```
options {
    dns64 2001:db8:6464::/96 {
        clients { any; };
        mapped { any; };
        exclude { 64:FF9B::/96; ::ffff:0000:0000/96; };
        suffix ::;
    };
};
```

The NAT64 ‘Well Known Prefix’

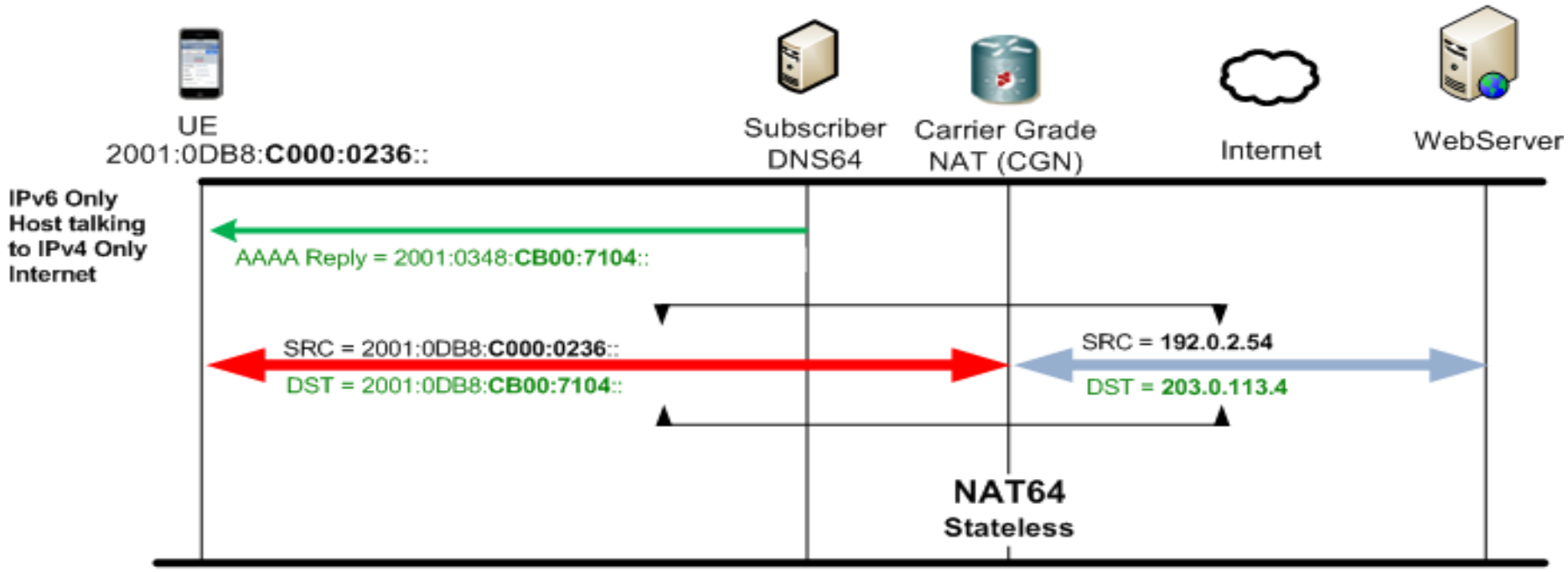
The address of clients allowed to use DNS64

The list of IPv4 addresses which can be mapped into AAAA records

The list of addresses that should not be mapped

The filler after the synthesis point. Only applies where WKP is < 96 bits

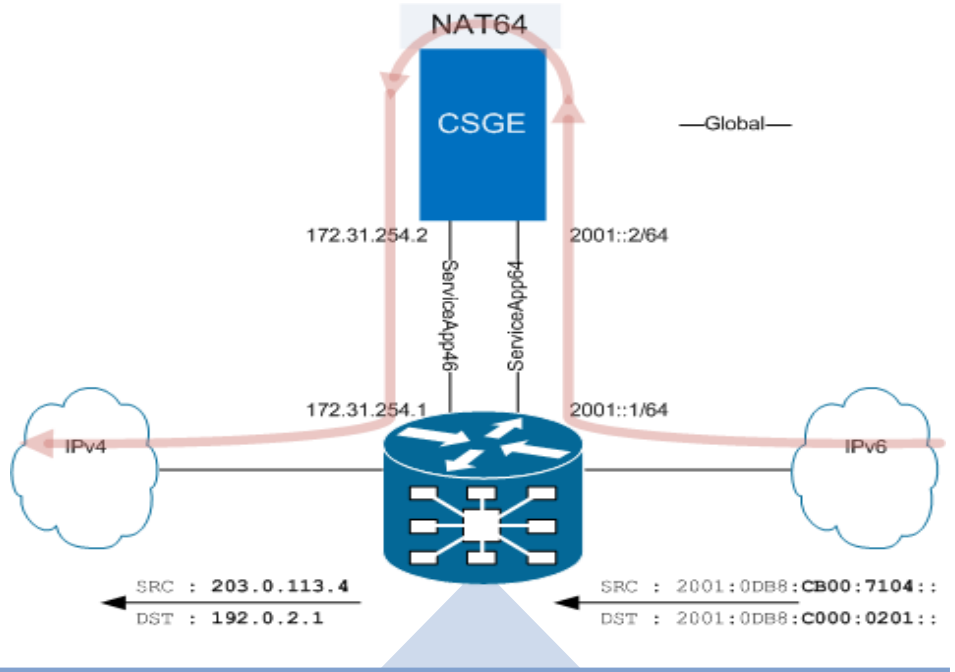
NAT64 Stateless



Key:
192.0.2.54 = C000:0236
203.0.113.4 = CB00:7104

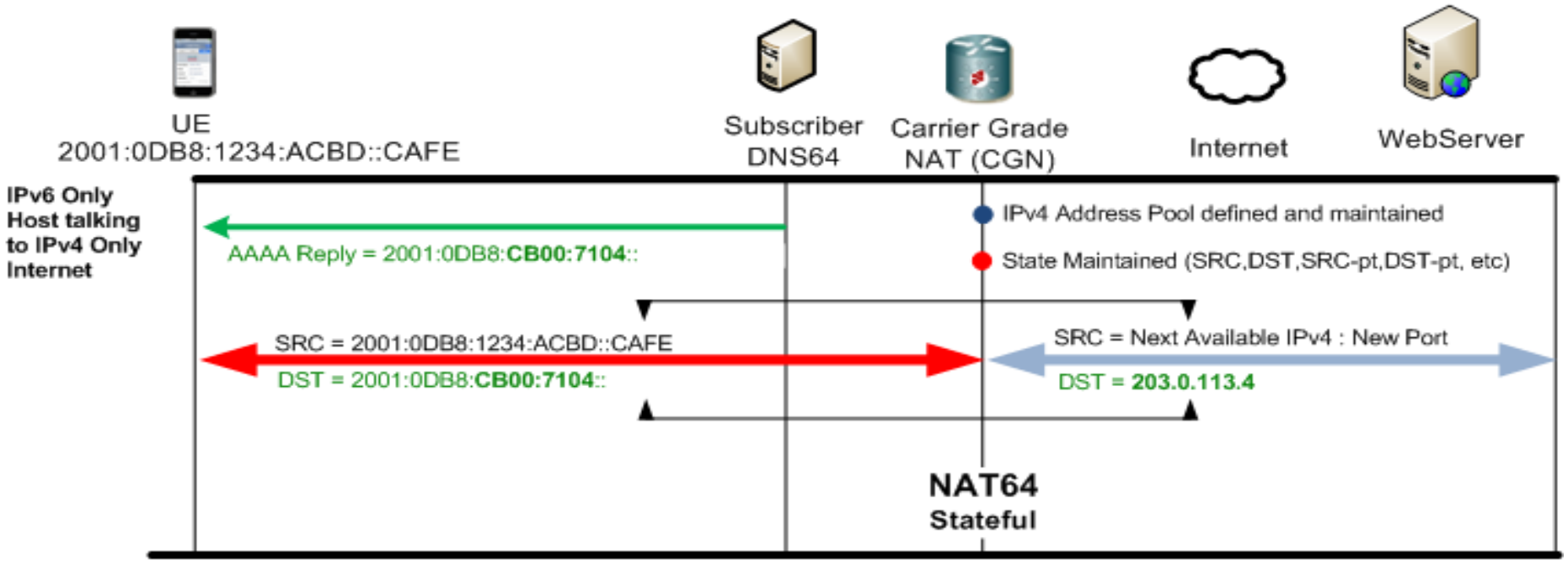
Stateless NAT64 example Topology
Using CGSE

```
!
service cgn CGN
 service-location preferred-active 0/3/CPU0
!
service-type nat64 stateless xlat1
  ipv6-prefix 2001:db8:6464::/96
  address-family ipv4
   interface ServiceApp46
  !
  address-family ipv6
   interface ServiceApp64
!
interface ServiceApp46
 description the IPv4 side NAT64 interface
 ipv4 address 172.31.254.1 255.255.255.252
 service cgn CGN service-type nat64 stateless
!
interface ServiceApp64
 description the IPv6 side NAT64 interface
 ipv6 address 2001:db8:a:b::1/64
 service cgn CGN service-type nat64 stateless
!
```



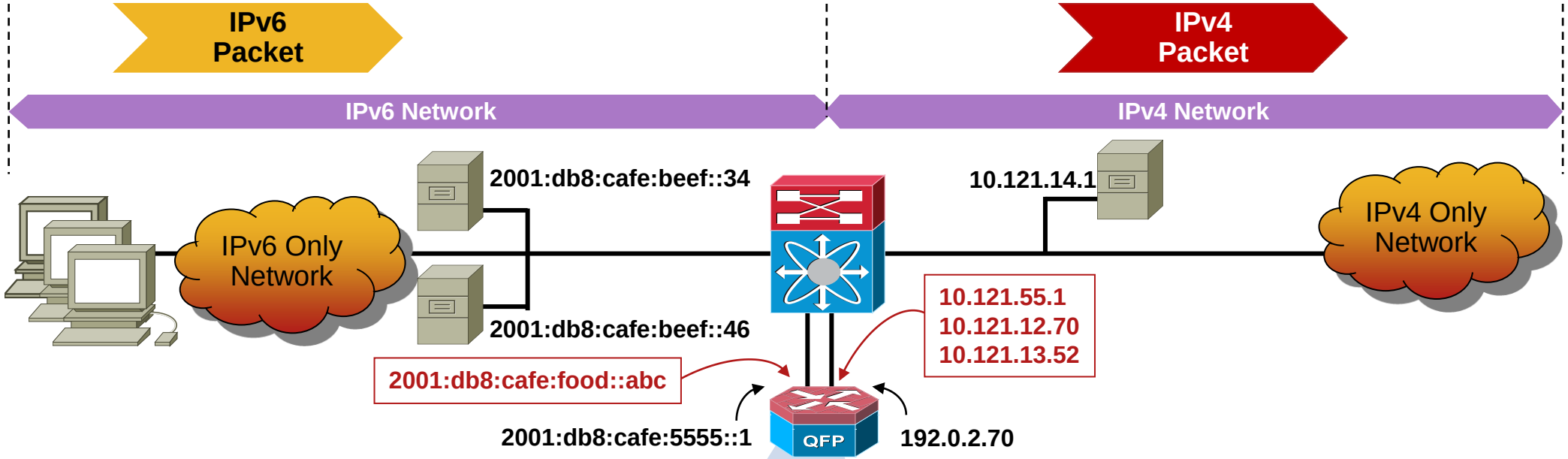

```
router static
 address-family ipv4 unicast
  192.0.2.0/24 ServiceApp46
 !
 address-family ipv6 unicast
  2001:db8:6464::/96 ServiceApp64 2001:db8:a:b::2
  2a01:db8:6464::/64 <LAN interface>
```

NAT64 Stateful



Key:
203.0.113.4 = CB00:7104

Stateful NAT64 example Topology




```
interface GigabitEthernet0/0/0
 description to 6k-dmz-1 Outside
 no ip address
 ipv6 address 2001:db8:cafe:1::1/64
 nat64 enable
!
interface GigabitEthernet0/0/1
 description to 6k-dmz-1 Inside
 ip address 192.0.2.70 255.255.255.0
 nat64 enable
```

```
ipv6 access-list EDGE_ACL
 permit ipv6 any any
!
nat64 prefix stateful 2001:db8:cafe:beef::/96
nat64 v4 pool EDGE 10.121.55.1 10.121.55.1
nat64 v6v4 list EDGE_ACL pool EDGE overload
!
nat64 v4v6 static 10.121.12.70 2001:db8:cafe:beef::46
nat64 v4v6 static 10.121.13.52 2001:db8:cafe:beef::34
!
nat64 v6v4 static 2001:db8:cafe:food::abc 10.121.14.1
```

http://tools.cisco.com/squish/A77F8

References

- CCO IPv6

  http://www.cisco.com/ipv6

- The ABC of IPv6

  http://www.cisco.com/en/US/products/sw/iosswrel/products_abc_ios_overview.html

- IPv6 e-Learning [requires CCO username/password]

  http://www.cisco.com/warp/customer/732/Tech/ipv6/elearning/

- IPv6 Access Services

  http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6553/whitepaper_C11-472610.html

- ICMPv6 Packet Types and Codes TechNote

  http://www.cisco.com/warp/customer/105/icmpv6codes.html

- Enterprise and SP Deployment Scenarios:

  ISP IPv6 Deployment Scenarios in Broadband Access Networks (RFC 4779)

  Scenarios and Analysis for Introducing IPv6 into ISP Networks (RFC 4029)

  IPv6 Enterprise Network Scenarios (RFC 4057)

  Procedures for Renumbering an IPv6 Network without a Flag Day (RFC 4192)

Recommended Reading

- These books are excellent reference material