



IPv6 Training

KENIC-AFRINIC IPv6 Workshop

17th – 20th June 2008

César Olvera (cesar.olvera@consulintel.es)

Jordi Palet (jordi.palet@consulintel.es)

Alvaro Vives (alvaro.vives@consulintel.es)



Agenda

1. Basic Introduction to IPv6
2. Header Formats & Packet Size Issues
3. Addressing
4. ICMPv6, Neighbor Discovery & DHCPv6
5. Applications
6. IPv6 DNS
7. Security/firewalling
8. Transition and Coexistence
9. Mobility
10. **Routing**



IPv6 Tutorial

10. Routing



Agenda

10.1. Routing concepts

10.2. OSPF

10.3. BGP

10.4. Static Routing



10.1. Routing concepts



Routing General View

- The routers should know where to forward the packets that they receive
- The static routes are not the best solution for both medium and large networks
 - Neither the small networks if there are continuous changes in their topologies
- The routing protocols gives an automatic method to generate the needed routing tables
 - The protocols have in account the changes in the network topologies



Types of Routing Protocols

- Regarding the Scope:
 - IGP Interior Gateway Protocol
 - EGP Exterior Gateway Protocol
- Types of IGPs
 - Regarding propagation
 - Distance-vector routing (examples: RIP, IGRP)
 - Link-state routing (examples: OSPF, IS-IS)
 - Regarding routes
 - Classful
 - Classless



IGP Selection

- Depends on several factors:
 - Network topology
 - Types of routes
 - Convergence
 - Calculation of the route metric
 - Scalability
 - Security
- Further information:
 - <http://www.cisco.com/univercd/cc/td/doc/cisintwk/idg4/nd2003.htm>



IGP Protocols

	VD	LS	Classful	Classless	Security
RIPv1	X		X		
RIPv2	X			X	
IGRP	X		X		
EIGRP	X			X	X
OSPF		X		X	X
IS-IS		X		X	



EGP Protocols

- There are not much alternatives
- BGP
 - Is the “de facto” standard



Routing for IPv6

- IPv6 uses same “longest-prefix match” routing as IPv4 CIDR
- Straightforward changes to existing IPv4 routing protocols to handle bigger addresses
 - unicast: RIPv2, EIGRP, IS-IS, OSPFv3, BGP4+, ...
 - multicast: MOSPF, PIM, ...
- Can use Routing header with anycast addresses to route packets through particular regions
 - e.g., for provider selection, policy, performance, etc.





10.4. OSPF



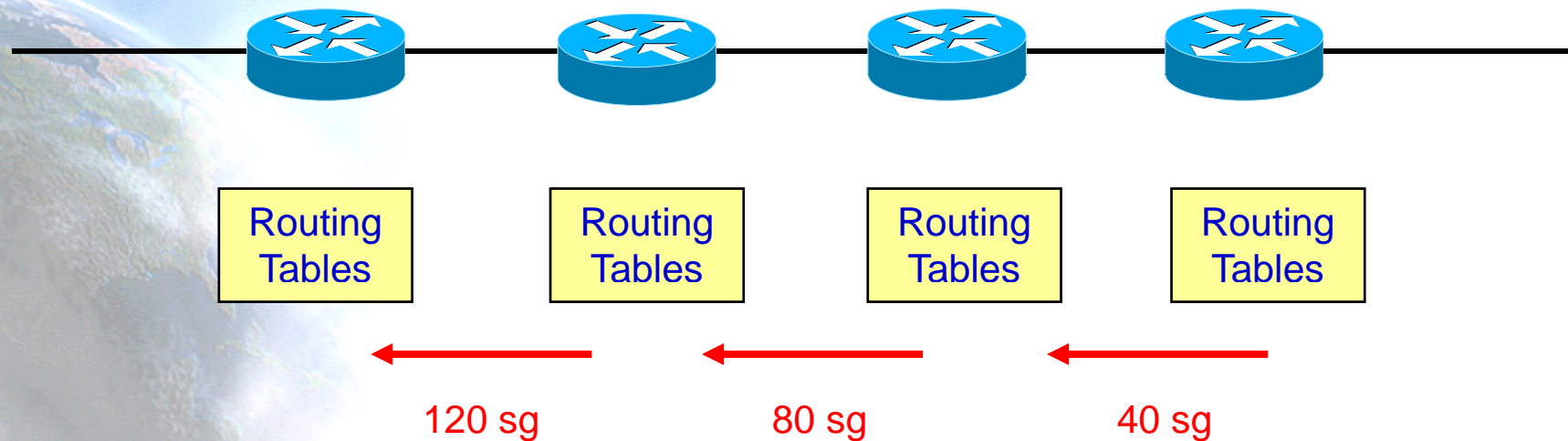
OSPF General View (1)

- OSPF is an IGP routing protocol of Link-state type
- It was developed to fulfill the most advance needed within an Autonomous system. It supports
 - VLSM (Variable Length Subnet Masking)
 - Authentication
 - Quick convergence after changes in the network topology
 - Link-state updates by means of multicast
 - Considers the bandwidth for best route selection
 - etc.
- The routers know the network topology by means of the SPF (Shortest Path First) algorithm



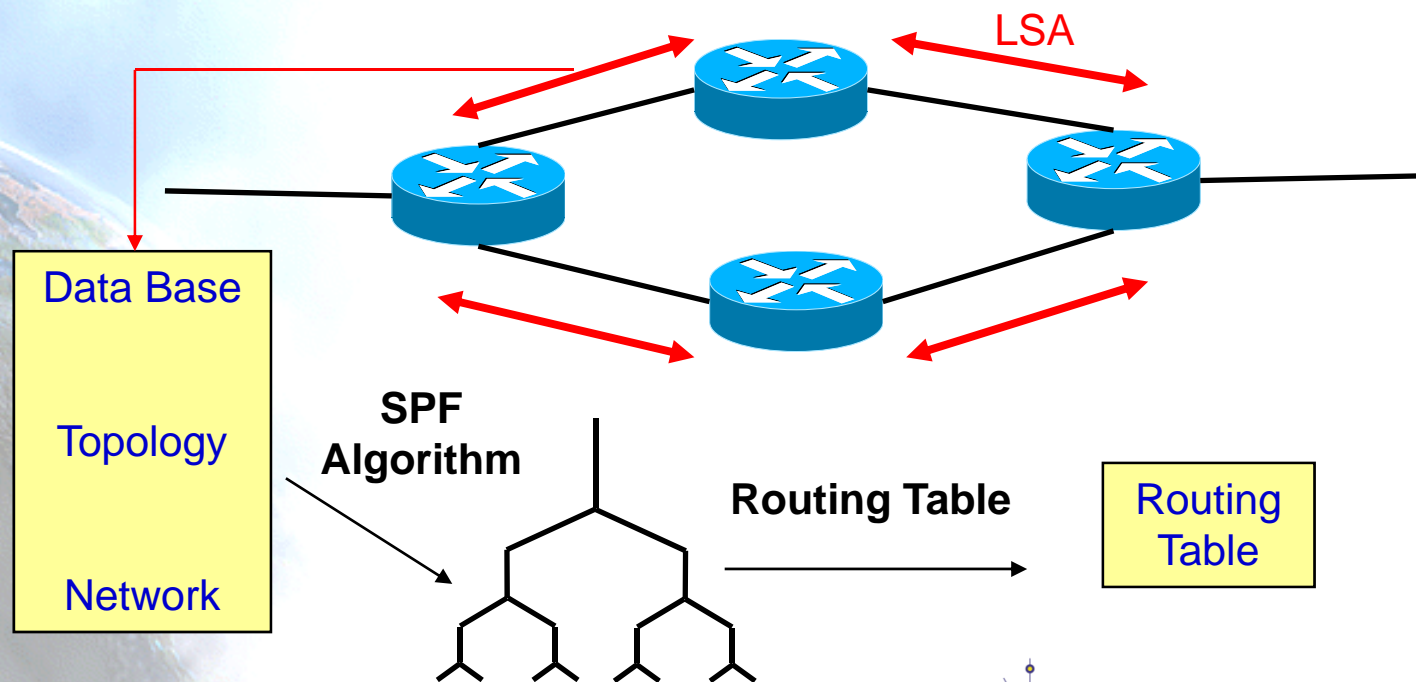
OSPF General View (2)

- The protocols based on Distance vector periodically send the routing table to their neighbors
 - Problems with the bandwidth
 - Problems with the convergence time



OSPF General View (3)

- The protocols based on Link-state only send the changes on the LSA (Link-State Advertisements)
 - Less Bandwidth
 - Quicker convergence
 - Support larger networks



OSPF General View (4)

- OSPF uses the Hello protocol to determine
 - The interfaces that will receive the LSAs
 - The existence of neighbor routers
 - If the neighbor routers are still alive (keepalive)
- The routers send LSAs (Link-State Advertisements) to all routers in the same hierarchy unit by means of a multicast address, including among other information:
 - Network prefix
 - Networks mask
 - Network type
 - Connected routers
 - etc.
- All the routers builds the same topologic database from the received LSAs
 - There is obtained a new routing table from the new topology

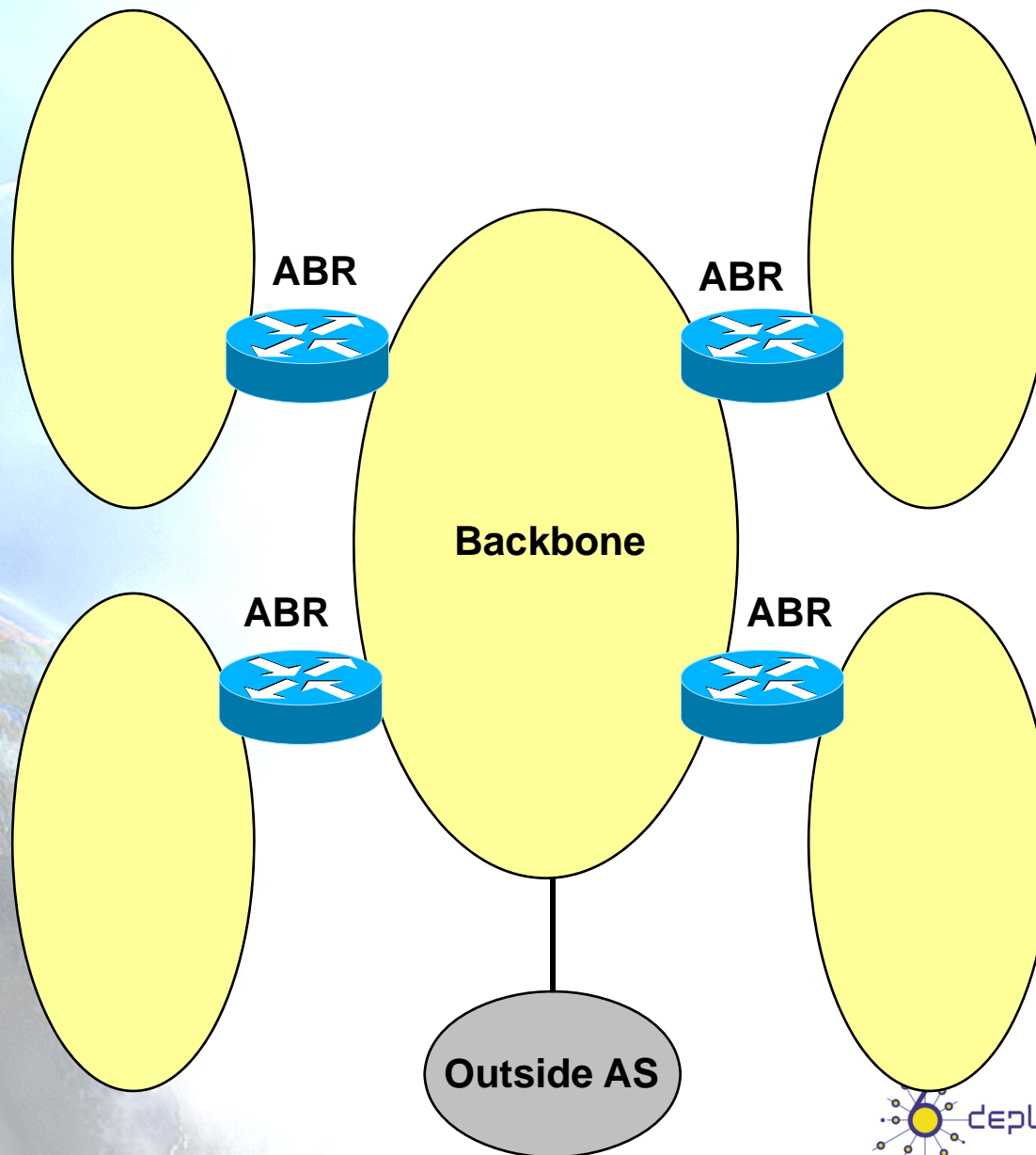


OSPF General View (5)

- The OSPF metric is calculated as the sum of the cost to go through all the nodes up to the final destination
 - Is based on the bandwidth of the interfaces and is user configurable
- OSPF divides the AS in small hierarchy units or areas
 - Each of them are connected to the “backbone area” by means of an ABR (Area Border Router)
 - This reduce the processing load as well the needed memory



OSPF General View (6)



OSPF General View (7)

- The LSAs describe the topology of the hierarchy units or areas
- There exist several types

Code	LSA	Link-State ID
1	Router LSA	Originating router ID of the router
2	Network LSA	Interface IP address of the DR
3	Network summary LSA	Destination network number
4	ASBR summary LSA	Router ID of AS boundary router
5	AS external LSA	External network number
7	NSSA external LSA	External network number



IPv6 OSPF (1)

- The OSPF version 3, for IPv6 (RFC2740), extends the OSPF version 2 (RFC2328) to support the routing of IPv6 prefixes as well the 128 bits addresses
 - Much of the characteristics of IPv6 OSPF are the same than OSPFv2
 - In IPv6 OSPF is not necessary to explicitly create an routing process. When OSPF for IPv6 is enabled in an interface, there will be created the routing process as well its associate configuration
 - In IPv6 OSPF each interface must be enabled using commands in interface configuration mode. This is different from OSPFv2, in which interfaces are indirectly enabled using the router configuration mode
 - In IPv6, there can be configured many address prefixes on an interface. In IPv6 OSPF, all address prefixes on an interface are included by default. Then the users cannot select some address prefixes to be imported into IPv6 OSPF; either all address prefixes on an interface are imported, or no address prefixes on an interface are imported
 - Unlike OSPFv2, multiple instances of IPv6 OSPF can be run on a link
 - When using a nonbroadcast multiaccess (NBMA) interface in IPv6 OSPF, users must manually configure the router with the list of neighbors. Neighboring routers are identified by their router ID



OSPF IPv6 (2)

- Since in IPv6 an network interface could have more than one address , the LSAs are different in OSPv3 than in OSPFv2

Code	LSA	Link-State ID
1	Router LSA	Originating router ID of the router. In IPv6 it does not have information of the network address. It is independent of the network protocol
2	Network LSA	Interface IP address of the DR. In IPv6 it does not have information of the network address. It is independent of the network protocol
3	Inter-Area-prefix LSAs for ABRs	Destination network number. In IPv6 it is given as prefix, prefix length
4	Inter-Area-router LSAs for ASBRs	Router ID of AS boundary router
5	Autonomous System External LSAs	Redistributing routes from another AS. In IPv6 it is given as prefix, prefix length and a default route with zero leght
8	Link LSA	Local-link flooding scope. It informs the link-local addresses of all router within the network segment
9	Intra-Area-Prefix LSA	Describes association to the router LSA.



Differences between OSPFv3 - OSPFv2 (1)

- Protocol processing per-link, not per-subnet
 - OSPFv3 runs per-link instead of the IPv4 behavior of per-IP-subnet. The terms "network" and "subnet" used in the IPv4 OSPF should generally be replaced by link.
 - Likewise, an OSPF interface now connects to a link instead of an IP subnet, etc
 - This change affects the receiving of OSPF protocol packets, and the contents of Hello Packets and Network-LSAs
- Removal of addressing semantics
 - In OSPv3, addressing semantics have been removed from the OSPF protocol packets and the main LSA types, leaving a network-protocol-independent core
- Addition of Flooding scope
 - Flooding scope for LSAs has been generalized and is now explicitly coded in the LSA's LS type field. There are now three separate flooding scopes for LSAs: Link-local, Area and AS scope



Differences between OSPFv3 - OSPFv2 (2)

- Explicit support for multiple instances per link
 - OSPF now supports the ability to run multiple OSPF protocol instances on a single link
 - This may be required on a NAP segment shared between several providers -- providers may be running separate OSPF routing domains that want to remain separate even though they have one or more physical network segments (i.e., links) in common
 - In OSPFv2 this was supported in a haphazard fashion using the authentication fields in the OSPF for IPv4 header
 - Support for multiple instances on a link is accomplished via an "Instance ID" contained in the OSPF packet header and OSPF interface structures
- Use of link-local addresses
 - OSPFv3 assumes that each router has been assigned link-local unicast addresses on each of the router's attached physical segments
 - On all OSPF interfaces except virtual links, OSPF packets are sent using the interface's associated link-local unicast address as source. A router learns the link-local addresses of all other routers attached to its links, and uses these addresses as next hop information during packet forwarding



Differences between OSPFv3 - OSPFv2 (3)

- Authentication

- In OSPFv3, authentication has been removed from OSPF itself

- The "AuType" and "Authentication" fields have been removed from the OSPF packet header, and all authentication related fields have been removed from the OSPF area and interface structures

- OSPF for IPv6 relies on the IP Authentication Header and the IP Encapsulating Security Payload to ensure integrity and authentication/confidentiality of routing exchanges

- Protection of OSPF packet exchanges against accidental data corruption is provided by the standard IPv6 16-bit one's complement checksum, covering the entire OSPF packet and prepended IPv6 pseudo- header



Differences between OSPFv3 - OSPFv2 (4)

- Packet format changes

- OSPFv3 runs directly over IPv6. Aside from this, all addressing semantics have been removed from the OSPF packet headers, making it essentially "network-protocol-independent". All addressing information is now contained in the various LSA types only

- In detail, changes in OSPF packet format consist of the following:

- The OSPF version number has been increased from 2 to 3
 - The Options field in Hello Packets and Database description Packet has been expanded to 24-bits
 - The Authentication and AuType fields have been removed from the OSPF packet header
 - 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, should the router become Designated-Router on the link
 - Two option bits, the "R-bit" and the "V6-bit", have been added to the Options field for processing Router-LSAs during the SPF calculation. If the "R-bit" is clear an OSPF speaker can participate in OSPF topology distribution without being used to forward transit traffic; this can be used in multi-homed hosts that want to participate in the routing protocol. The V6-bit specializes the R-bit; if the V6-bit is clear an OSPF speaker can participate in OSPF topology distribution without being used to forward IPv6 datagrams. If the R-bit is set and the V6-bit is clear, IPv6 datagrams are not forwarded but datagrams belonging to another protocol family may be forwarded
 - The OSPF packet header now includes an "Instance ID" which allows multiple OSPF protocol instances to be run on a single link



Differences between OSPFv3 - OSPFv2 (5)

- LSA format changes
 - All addressing semantics have been removed from the LSA header, and from Router-LSAs and Network-LSAs
 - These two LSAs now describe the routing domain's topology in a network-protocol-independent manner. New LSAs have been added to distribute IPv6 address information, and data required for next hop resolution. The names of some of IPv4's LSAs have been changed to be more consistent with each other.
- Handling unknown LSA types
 - Handling of unknown LSA types has been made more flexible so that, based on LS type, unknown LSA types are either treated as having link-local flooding scope, or are stored and flooded as if they were understood (desirable for things like the proposed External- Attributes-LSA. This behavior is explicitly coded in the LSA Handling bit of the link state header's LS type field
 - The IPv4 OSPF behavior of simply discarding unknown types is unsupported due to the desire to mix router capabilities on a single link. Discarding unknown types causes problems when the Designated Router supports fewer options than the other routers on the link.



Differences between OSPFv3 - OSPFv2 (6)

- Stub area support

- In OSPFv2, stub areas were designed to minimize link-state database and routing table sizes for the areas' internal routers. This allows routers with minimal resources to participate in even very large OSPF routing domains

- In OSPFv3, the concept of stub areas is retained. In IPv6, of the mandatory LSA types, stub areas carry only router-LSAs, network-LSAs, Inter-Area-Prefix-LSAs, Link-LSAs, and Intra-Area-Prefix-LSAs. This is the IPv6 equivalent of the LSA types carried in IPv4 stub areas: router-LSAs, network-LSAs and type 3 summary-LSAs

- However, unlike in IPv4, IPv6 allows LSAs with unrecognized LS types to be labeled "Store and flood the LSA, as if type understood" (U-bit). Uncontrolled introduction of such LSAs could cause a stub area's link-state database to grow larger than its component routers' capacities

- To guard against this, the following rule regarding stub areas has been established: an LSA whose LS type is unrecognized may only be flooded into/throughout a stub area if both

- a) the LSA has area or link-local flooding scope
 - b) and the LSA has U-bit set to 0

- Identifying neighbors by Router ID

- In OSPFv3 the neighboring routers on a given link are always identified by their OSPF Router ID

- This contrasts with the IPv4 behavior where neighbors on point-to-point networks and virtual links are identified by their Router IDs, and neighbors on broadcast, NBMA and Point-to-MultiPoint links are identified by their IPv4 interface addresses

- This change affects the reception of OSPF packets, the lookup of neighbors and the reception of Hello Packets

- The Router ID of 0.0.0.0 is reserved, and should not be used



General Steps to Configure IPv6 OSPF

- Strategy or plan of the IPv6 OSPF network
- Enable IPv6 unicast-routing
- Enable IPv6 OSPF in the interface
- Adequate the configuration to the particular network
- Globally enable CEF for IPv4 (CEFv4) in the router using the command “ip cef global configuration”
- Globally enable CEF for IPv6 (CEFv6) in the router using the command “ipv6 cef global configuration”



Activate IPv6 OSPF

By default the IPv6 OSPF routing is disabled, and is not configured in the interfaces

- Router> enable
- Router# configure terminal
- Router(config)# interface ethernet 0/0
- Router(config-if)# ipv6 ospf process-id area area-id [instance instance-id]
 - Example: Router(config-if)# ipv6 ospf 1 area 0



Check routes in an IPv6 OSPF area

- Router> enable
- Router# configure terminal
- Router(config)# ipv6 router ospf process-id
 - Example: Router(config)# ipv6 router ospf 1
- Router(config-rtr)# area area-id range { ipv6-prefix/prefix-length} [advertise | not-advertise] [cost cost]
 - Example: Router(config-rtr)# area 1 range 2001::/48

See the following routes

2001:DB8:0:7::/64 [110/20] via FE80::A8BB:CCFF:FE00:6F00, Ethernet0/0

2001:DB8:0:8::/64 [110/100] via FE80::A8BB:CCFF:FE00:6F00, Ethernet0/0

2001:DB8:0:9::/64 [110/20] via FE80::A8BB:CCFF:FE00:6F00, Ethernet0/0

Will be summarized as

2001:DB8::/48 [110/100] via FE80::A8BB:CCFF:FE00:6F00, Ethernet0/0

See that the cost of the final route is equal to the cost of the higher one of the summarized routes



Check configuration and operation of IPv6 OSPF

- Router> enable
- Interface Information
 - Router# show ipv6 ospf [process-id] [area-id] interface [interface-type interface-number]
 - Example: Router# show ipv6 ospf interface
- General Information
 - Router# show ipv6 ospf [process-id] [area-id]
 - Example: Router# show ipv6 ospf





10.6. BGP

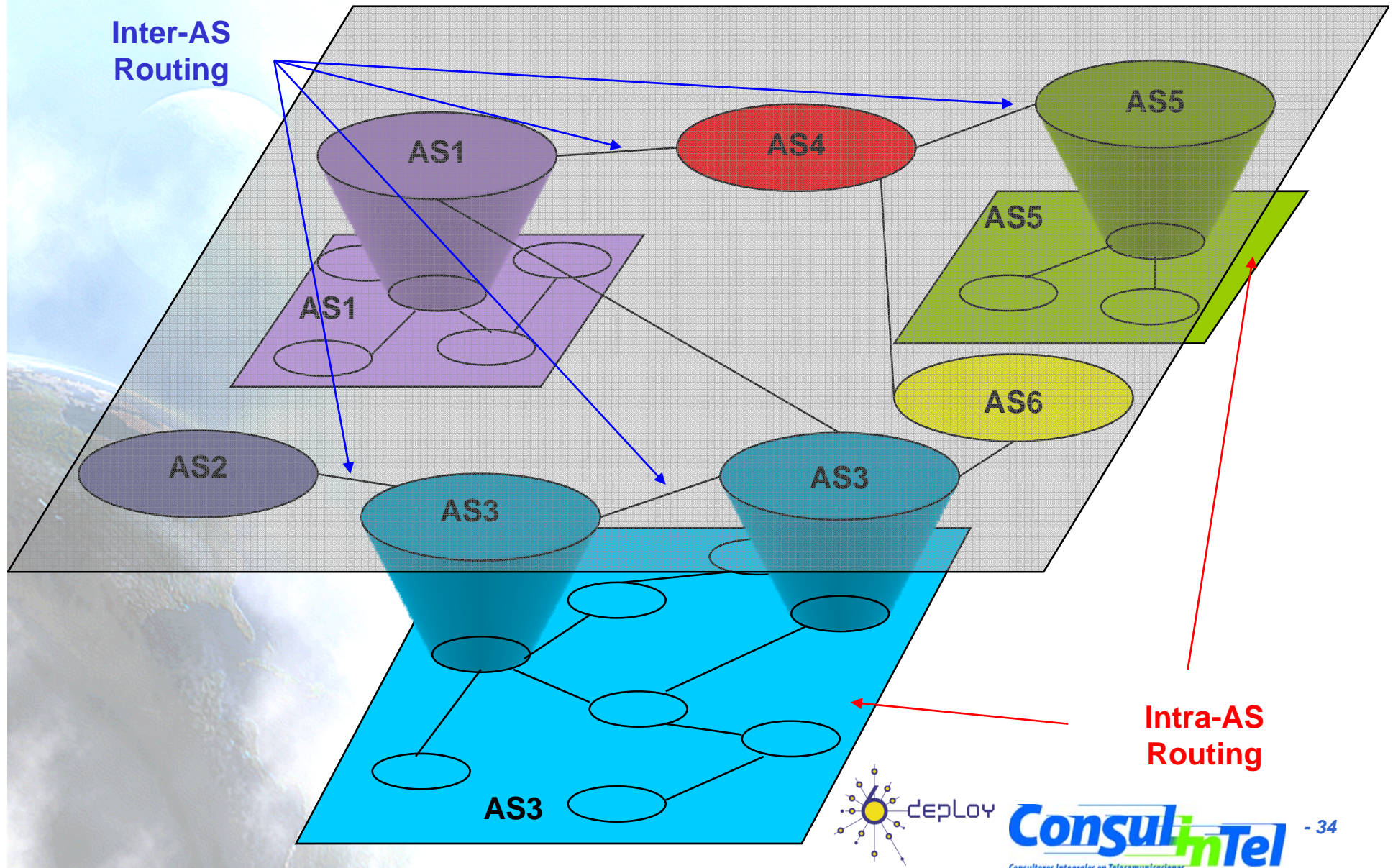


BGP General View (1)

- The routing in Internet is done at two levels
 - Intra-AS => IGP
 - The managements of each Autonomous System is local, which include the type of routing protocol used
 - Inter-AS => EGP
 - Requires an common protocol to assure that all the Autonomous System are reachable
 - BGP is “de facto” standard



BGP General View (2)



BGP General View (3)

- BGP “Border Gateway Protocol”
 - “de facto” standard
- It is based on PVP (Path Vector Protocol)
 - Similar to Distance Vector
 - Each border router sends to its neighbors (“peerings”) the full route to one destination, not just the distance
 - The path is a sequence of ASs up to destination
 - Example: Path(X,Z)=X, Y1, Y2, Y3, Y5, Z



BGP General View (4)

- Path Vector Protocol
 - Router X send its route to Z to its neighbor W
 - If W accepts the route, then $P(W,Z)=X$, $\text{Path}(X,Z)$
 - W could not accept that route
 - To avoid loops
 - Because traffic cost
 - Because violation of internal AS policies
 - etc.
- Is possible to control the traffic that enters into the network announcing or not the AS to the neighbors
- If X does not want to route the traffic to Z, is enough with does not announce the path to Z



BGP General View (5)

- TCP is used to transport BGP messages
 - OPEN – to open a TCP connection
 - UPDATE – announce or confirm a new path
 - KEEPALIVE – is there is not UPDATES, it is used to keep open the TCP connection; also is used as ACK for an OPEN message
 - NOTIFICATION – to inform of errors in messages; also is used to close connections



BGP General View (6)

- Path attributes
 - Well know
 - They are known by all the routers and passed to BGP neighbors
 - Mandatory and are included in the UPDATE messages
 - Optional
 - May not be supported by all BGP implementations
 - The transitive bit determines if an optional attribute is passed to BGP neighbors



BGP for IPv6 (BGP4+) (1)

- The current version of BGP is version 4, i.e. BGP4
 - BGP4 (BGP for IPv4) is described in RFC4271
- The Multiprotocol BGP extensions, i.e. BGP4+, allow BGP4 to be used for different address families, such as IPv6 and Multicast
 - Multiprotocol Extensions for BGP4 (BGP for IPv6) are described in RFC4760
 - This document defines the extensions to BGP4 to enable it to carry routing information for multiple Network Layer protocols (e.g., IPv6, IPX, L3VPN, etc.)



BGP for IPv6 (BGP4+) (2)

- Multiprotocol BGP extensions for IPv6 supports the same features and functionality as IPv4 BGP
 - The IPv6 enhancements to multiprotocol BGP include support for
 - The IPv6 address family and network layer reachability information (NLRI)
 - The next hop (the next router in the path to the destination) attributes that use IPv6 addresses
- Multiprotocol BGP for the IPv6 Multicast Address family feature provides multicast BGP extensions for IPv6 and supports the same features and functionality as IPv4 BGP
 - The IPv6 enhancements to multicast BGP include support for
 - The IPv6 multicast address family and network layer reachability information (NLRI)
 - The next hop (the next router in the path to the destination) attributes that use IPv6 addresses



BGP4+ Characteristics (1)

- The only 3 pieces of information carried by BGP4 that are IPv4 specific are
 1. NEXT_HOP attribute (expressed as an IPv4 address)
 2. AGGREGATOR (contains an IPv4 address)
 3. NLRI (expressed as IPv4 address prefixes)
- RFC4760 assumes that any BGP speaker (including the one that supports multiprotocol capabilities defined in RFC4760) has to have an IPv4 address (which will be used, among other things, in the AGGREGATOR attribute)
- Therefore, to enable BGP4 to support routing for multiple Network Layer protocols, the only two things that have to be added to BGP4 are the
 1. ability to associate a particular Network Layer protocol with the next hop information
 2. ability to associate a particular Network Layer protocol with NLRI
- To identify individual Network Layer protocols associated with the next hop information and semantics of NLRI, RFC4760 uses a combination of
 - Address Family, as defined in “IANA's Address Family Numbers registry“ (<http://www.iana.org/numbers.html>)
 - Subsequent Address Family (as described in RFC4760 itself)



BGP4+ Characteristics (2)

- To provide backward compatibility, as well as to simplify introduction of the multiprotocol capabilities into BGP-4, RFC4760 defines two new attributes
 - Multiprotocol Reachable NLRI (MP_REACH_NLRI), used to carry the set of reachable destinations together with the next hop information to be used for forwarding to these destinations
 - Multiprotocol Unreachable NLRI (MP_UNREACH_NLRI), used to carry the set of unreachable destinations
- Both of these attributes are optional and non-transitive
 - This way, a BGP speaker that doesn't support the multiprotocol capabilities will just ignore the information carried in these attributes and will not pass it to other BGP speakers



General Steps to Configure IPv6 BGP

- Strategy or plan of the IPv6 BGP network
- Enable IPv6 unicast-routing
- Configure an IPv6 BGP Routing process
- Configure BGP Router ID
- Configure IPv6 BGP Peering(s)
- Adequate the configuration to the particular network



Configure a BGP IPv6 Routing process and the BGP Router ID

- Router> enable
- Router# configure terminal
- Router(config)# router bgp autonomous-system-number (Configure the BGP process)
 - Example: Router(config)# router bgp 65000
- Router(config-router)# no bgp default ipv4-unicast (Disable the IPv4 unicast address family for the above BGP process)
 - Example : Router(config-router)# no bgp default ipv4-unicast
- Router(config-router)# bgp router-id ip-address (Optional: Configure u router ID of 32-bits)
 - Example: Router(config-router)# bgp router-id 192.168.99.70



Configure a IPv6 BGP Peering

- Router> enable
- Router# configure terminal
- Router(config)# router bgp autonomous-system-number
 - Example: Router(config)# router bgp 65000
- Router(config-router)# neighbor ipv6-address remote-as autonomous-system-number (By default only IPv4 prefixes are announce)
 - Example: Router(config-router)# neighbor 2001:0DB8:0:CC00::1 remote-as 64600
- Router(config-router)# address-family ipv6 [unicast | multicast]
 - Example: Router(config-router)# address-family ipv6
- Router(config-router-af)# neighbor ipv6-address activate (To perform the announce of IPv6 prefixes on the address-family ipv6)
 - Example: Router(config-router-af)# neighbor 2001:0DB8:0:CC00::1 activate



Configure a IPv6 BGP Peering using Link-local addresses (1)

- Router> enable
- Router# configure terminal
- Router(config)# router bgp autonomous-system-number
 - Example: Router(config)# router bgp 65000
- Router(config-router)# neighbor ipv6-address remote-as autonomous-system-number
 - Example : Router(config-router)# neighbor FE80::BFF:FE0E:A471 remote-as 64600
- Router(config-router)# neighbor ipv6-address update-source interface-type interface-number
 - Example : Router(config-router)# neighbor FE80::BFF:FE0E:A471 update-source fastethernet0



Configure a IPv6 BGP Peering using Link-local addresses (2)

- Router(config-router)# address-family ipv6 [unicast | multicast]
 - Example: Router(config-router)# address-family ipv6
- Router(config-router-af)# neighbor ipv6-address activate ((To perform the announce of IPv6 prefixes on the address-family ipv6)
 - Example: Router(config-router-af)# neighbor FE80::BFF:FE0E:A471 activate
- Router(config-router-af)# neighbor ipv6-address route-map map-name {in | out}
 - Example: Router(config-router-af)# neighbor FE80::BFF:FE0E:A471 route-map nh6 out
- Router(config-router-af)# exit
- Router(config-router)# exit



Configure a IPv6 BGP Peering using Link-local addresses (3)

- Router(config)# route-map map-name [permit | deny] [sequence-number] (Define a route map)
 - Example: Router(config)# route-map nh6 permit 10
- Router(config-route-map)# match ipv6 address prefix-list-name
 - Example: Router(config-route-map)# match ipv6 address prefix-list cisco
- Router(config-route-map)# set ipv6 next-hop ipv6-address [link-local-address] (Overrides the next hop advertised to the peer for IPv6 packets that pass a match clause of a route map for policy routing. The ipv6-address argument specifies the IPv6 global address of the next hop. It need not be an adjacent router. The link-local-address argument specifies the IPv6 link-local address of the next hop. It must be an adjacent router)
 - Example: Router(config-route-map)# set ipv6 next-hop 2001:DB8::1



Configure a Route Map for IPv6 BGP

- Router> enable
- Router# configure terminal
- Router(config)# router bgp 65000
- Router(config-router)# neighbor 2001:DB8:0:cc00::1 remote-as 64600
- Router(config-router)# address-family ipv6
- Router(config-router-af)# neighbor 2001:DB8:0:cc00::1 activate
- Router(config-router-af)# neighbor ipv6-address route-map map-name {in | out} (Applies the map to incoming or out coming routes)
 - Example: Router(config-router-af)# neighbor 2001:0DB8:0:cc00::1 route-map rtp in
- Router(config-router-af)# exit
- Router(config-router)# exit
- Router(config)# route-map map-name [permit | deny] [sequence-number]
 - Example: Router(config)# route-map rtp permit 10
- Router(config-route-map)# match ipv6 address prefix-list prefix-list-name
 - Example: Router(config-route-map)# match ipv6 address prefix-list cisco



Configure an IPv6 BGP Peer Group

- Router> enable
- Router# configure terminal
- Router(config)# router bgp 65000
- Router(config-router)# neighbor peer-group-name peer-group
 - Example: Router(config-router)# neighbor group1 peer-group
- Router(config-router)# neighbor ipv6-address remote-as autonomous-system-number
 - Example: Router(config-router)# neighbor 2001:DB8:0:CC00::1 remote-as 64600
- Router(config-router)# address-family ipv6
- Router(config-router-af)# neighbor {ip-address | peer-group-name | ipv6-address} activate
 - Example1: Router(config-router-af)# neighbor 2001:0DB8:0:CC00::1 activate
 - Example2: Router(config-router-af)# neighbor group1 activate
- Router(config-router-af)# neighbor ipv6-address peer-group peer-group-name
 - Example: Router(config-router-af)# neighbor 2001:0DB8:0:CC00::1 peer-group group1



Announce Routes to IPv6 BGP

- Router> enable
- Router# configure terminal
- Router(config)# router bgp 65000
- Router(config-router)# address-family ipv6 unicast
- Router(config-router-af)# network ipv6-address/
prefix-length (Announce or injects the specified IPv6
prefix into IPv6 BGP. The route must be already in
the routing table of unicast IPv6)
 - Example: Router(config-router-af)# network
2001:DB8::/32



Redistribute Routes to IPv6 BGP

- Router> enable
- Router# configure terminal
- Router(config)# router bgp 65000
- Router(config-router)# address-family ipv6 unicast
- Router(config-router-af)# redistribute protocol [process-id] [level-1 | level-1-2 | level-2] [metric metric-value] [metric-type {internal | external}] [route-map map-name] (Redistribute routes from bgp, connected, isis, rip, o static into IPv6 BGP)
 - Example: Router(config-router-af)# redistribute rip



Check configuration and operation of IPv6 BGP

- Router> show bgp ipv6 {unicast | multicast} [ipv6-prefix/ prefix-length] [longer-prefixes] [labels] (Displays entries in the IPv6 BGP routing table)
 - Example: Router> show bgp ipv6 unicast
- Router> show bgp ipv6 {unicast | multicast} summary (Displays the status of all IPv6 BGP connections)
 - Example: Router> show bgp ipv6 unicast summary
- Router> show bgp ipv6 {unicast | multicast} dampening dampened-paths (Displays IPv6 BGP dampened routes)
 - Example: Router> show bgp ipv6 unicast dampening dampened-paths
- Router> enable
- Router# debug bgp ipv6 {unicast | multicast} dampening [access-list-name] [prefix-list prefix-list-name] (Displays debugging messages for IPv6 BGP dampening packets)
 - Example: Router# debug bgp ipv6 unicast dampening
- Router# debug bgp ipv6 {unicast | multicast} updates [ipv6-address] [prefix-list prefix-list-name] [in | out] (Displays debugging messages for IPv6 BGP update packets)
 - Example: Router# debug bgp ipv6 unicast updates



Clear IPv6 BGP sessions

- Router> enable
- Router# clear bgp ipv6 {unicast | multicast} {* | autonomous-system-number | ip-address | ipv6-address | peer-group-name} [soft] [in | out]
 - Example: Router# clear bgp ipv6 unicast peer-group Intranet soft out



Clear IPv6 BGP information

- Router> enable
- Router# clear bgp ipv6 {unicast | multicast} external [soft] [in | out] (Clears external IPv6 BGP peers)
 - Example: Router# clear bgp ipv6 unicast external soft in
- Router# clear bgp ipv6 {unicast | multicast} peer-group [name] (Clears all members of an IPv6 BGP peer group)
 - Example: Router# clear bgp ipv6 unicast peer-group
- Router# clear bgp ipv6 {unicast | multicast} dampening [ipv6-prefix/ prefix-length] (Clears IPv6 BGP route dampening information and unsuppress the suppressed routes)
 - Example: Router# clear bgp ipv6 unicast dampening 2001:DB8::1/64





10.7. Static Routing



Static Routes

- The minimum IOS version supporting IPv6 Static Routes is the same as those having IPv6 support
 - 12.0(21)ST, 12.0(22)S, 12.2(2)T, 12.2(14)S, 12.3(2)T, 12.3(1)M
- Firstly is needed to enable the IPv6 packet forwarding
 - Router(config)# ipv6 unicast-routing
- Later is needed to IPv6-enable at least one interface
 - As example, configuring an IPv6 address
 - Router(config-if)# ipv6 address



Configuration of Static Routes

- The way to configure static routes is similar to IPv4 but with a minimum change in the syntax
 - Router(config)# ipv6 route ipv6-prefix/prefix-length {ipv6-address | interface-type interface-number [ipv6-address]} [administrative-distance]
- Examples
 - Router(config) # ipv6 route ::/0 serial 2/0
 - Router(config) # ipv6 route 2001:0db8::/32 serial 0
 - Router(config) # ipv6 route 2001:0db8::/32 ethernet 0 fe80::1
 - Router(config) # ipv6 route 2001:0db8::/32 2002:806b:f0fe::1



Check Static Routes

- Command similar to IPv4
 - Router(config)# show ipv6 route [ipv6-address | ipv6-prefix/prefix-length | protocol]
- Example
 - Router# show ipv6 route static

IPv6 Routing Table - 330 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
U - Per-user Static route
I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea

```
S 2001:DB8::/32 [1/0]
    via ::, Tunnel0
S 2001:DB8:C00:8011::/48 [1/0]
    via ::, Null0
S ::/0 [254/0]
    via 2002:806B:F0FE::1, Null
```



Thanks !

Contact:

- Jordi Palet Martínez (Consulintel): jordi.palet@consulintel.es
- Alvaro Vives Martínez (Consulintel): alvaro.vives@consulintel.es

6DEPLOY Project

<http://www.6deploy.org>

The IPv6 Portal:

<http://www.ipv6tf.org>

