Transition from IPv4 to IPv6

Mario Baldi – Fulvio Risso

Politecnico di Torino
Nota di Copyright

This set of transparencies, hereinafter referred to as slides, is protected by copyright laws and provisions of International Treaties. The title and copyright regarding the slides (including, but not limited to, each and every image, photography, animation, video, audio, music and text) are property of the authors specified on page 1.

The slides may be reproduced and used freely by research institutes, schools and Universities for non-profit, institutional purposes. In such cases, no authorization is requested.

Any total or partial use or reproduction (including, but not limited to, reproduction on magnetic media, computer networks, and printed reproduction) is forbidden, unless explicitly authorized by the authors by means of written license.

Information included in these slides is deemed as accurate at the date of publication. Such information is supplied for merely educational purposes and may not be used in designing systems, products, networks, etc. In any case, these slides are subject to changes without any previous notice. The authors do not assume any responsibility for the contents of these slides (including, but not limited to, accuracy, completeness, enforceability, updated-ness of information hereinafter provided).

In any case, accordance with information hereinafter included must not be declared.

In any case, this copyright notice must never be removed and must be reported even in partial uses.
Introduction
Assumption (and problem to solve)

IPv4 and IPv6 will coexist (at least for a while)
Dual Stack Approach

- Both IPv4 and IPv6 capabilities
  - In all hosts and routers supporting IPv6
  - IPv4 support can be (gradually) removed (and included in new hosts) once all hosts have IPv6
- Hosts communicate natively with both
- Complete duplication of all protocol stack components
  - Routing protocols
  - Routing table
  - Access lists

![Diagram showing IPv4 and IPv6 layers with Data-link and Physical layers]
Dual Stack Limitations

- It does not reduce the need for IPv4 addresses
  - Each host still needs an IPv4 address to use IPv4
- Applications have the responsibility whether to use IPv4 or IPv6
Dual Layer Approach

- An alternative
- Applications are not responsible for choosing the protocol to use
- Less modifications in the applications

Less common than dual stack
Dual stack vs. Dual Layer

IPv4-mapped address (e.g. ::FFFF:a.b.c.d)

Applicazioni

TCP/UDP
IPv4
IPv6
Data-link
Physical

IPv4 address
IPv6 address

TCP/UDP
IPv4
IPv6
Data-link
Physical

IPv4 address
IPv6 address

IPv4-IPv6 program

IPv4-IPv6 program

Copyright: see page 2
Not all hosts will be dual stack

- IPv6 hosts shall communicate with IPv6 hosts through an IPv4 network
  - Same for IPv4 hosts through an IPv6 network
- IPv6 hosts shall communicate with IPv4 hosts
  - Translation mechanisms must be used
  - Not targeting IPv4 hosts contacting IPv6 ones
    - Difficult to map the large IPv6 address space on smaller IPv4 address space
First stage: isolated IPv6 networks

IPv6 in IPv4 Tunnel
Traversing a IPv4-only Network

- Tunnelling
  - Encapsulation of IP (v6) packets into IP (v4) packets
  - Emulates a “direct” link among IPv6 devices
Tunnelling

- End points: hosts and routers

- Protocols
  - GRE (Generic Routing Encapsulation)
  - IPv6 in IPv4
    - Protocol type = 41

- Set up: manual and automatic
  - IPv4-compatible addresses, 6over4 (RFC 2529), 6to4, Tunnel Broker (RFC 3053), ISATAP, Teredo
Host-centered Solutions

*Dual stack hosts* exchange IPv6 packets through an IPv4 network.
IPv4-compatible Addresses

- Sometimes improperly called “automatic tunneling”
  - Same name is used for other solutions
- IPv4-compatible addresses (::/96) are used for IPv6 communication
IPv4-compatible Addresses

- Application sends IPv6 packets to IPv6 addresses
  - E.g., ::2.2.2.2
- Static route forwards packets to ::/96 through pseudo-interface
  - Automatic Tunneling Pseudo-Interface
- Pseudo-interface encapsulates IPv6 packets in IPv4 packets and sends them out
End-to-end Tunnel

Encapsulating IPv4 packets are sent to IPv4 address corresponding to IPv6 destination
IPv4-compatible Address with Dual Stack Router

- Pseudo-interface in sending host is configured to terminate tunnels on router
- Encapsulating IPv4 packets are sent to IPv4 address of dual stack router
6over4

- IPv4 network emulates a virtual LAN
  - Broadcast multiple access data link
  - IP Multicasting used for the purpose
- Neighbor and router discovery enabled
- IPv4 address is used for automatic IPv6 Interface ID generation of link local address
- Not very used because IPv4 multicast support is not widespread
6over4 Neighbor Discovery

- IPv6 multicast addresses are mapped on IPv4 multicast addresses
  - 239.192.x.y
  - x.y are the last 2 bytes of the IPv6 address

- An example
  - IPv4 address: 1.1.1.1
  - IPv6 Link local address: fe80::101:101
  - Solicited node multicast address: ff02::1:ff00:1:101
  - 6over4 multicast address: 239.192.1.1
ISATAP: Intra-site Automatic Tunnel Addressing Protocol

- IPv4 network as Non-Broadcast Multiple Access (NBMA) data link
  - No IP multicast support needed
- Interface ID derived from IPv4 address
  - Prefixed by 0000:5efe
  - E.g., fe80::5efe:0101:0101 for 1.1.1.1
(Lack of) Neighbor Discovery

- Not needed for data-link address discovery as IPv4 address is embedded in IPv6 address
  - Last 4 bytes
- PRL (Potential Router List) must be provided
  - Router discovery not possible
- By configuration
- Automatically acquired from DNS
  - Hostname not mandated
  - E.g., isatap.polito.it
Automatic Configuration

- IPv4 address, DNS address and domain name obtained through DHCPv4
- Generation IPv6 link-local address
  - Interface ID from IPv4 address
- DNS query to obtain PRL
  - If not provided by DHCPv4 (proprietary)
- Periodic Router Discovery to each router
  - On-link prefixes for autoconfiguration
Network-centered Solutions

Dual stack and native IPv6 hosts exchange IPv6 packets through an IPv4 network
6to4

6to4 router or relay

- Relay address embedded in IPv6 prefix

```
3  | 13  | 32  | 16  | 64
---|-----|-----|-----|-----
001 | TLA | V4ADDR | SLA ID | Interface ID
0x002 |     |        |        |      
```

2002::/16

2002:x.y:z.w/48
Basic 6to4 Scenario

Not meant for IPv4 host to IPv6 host communication

IPv4 Internet

IPv6 in IPv4 tunnel

Address of 6to4 router

192.1.2.3

9.254.2.252

2002:c001:0203::/48

2002:09fe:02fc::/48
Mixed 6to4 Scenario

6to4 Relay must be default gateway of 6to4 hosts

Address of 6to4 Relay (predefined *anycast* address)
Teredo

- Architecture similar to 6to4
- IPv6 packets are encapsulated in IP/UDP to be compatible with NAT
  - NAT cannot use ports with IPv6 in IPv4
Tunnel Broker

- Communication with a *tunnel broker server*
  - Identifies *tunnel server* and mediates tunnel setup
- IPv6 in IPv4 (a.k.a. proto-41) tunnels
- Tunnel Setup Protocol (TSP) or Tunnel Information Control (TIC) protocol used to setup tunnels
Tunnel Broker Architecture

IPv6 island

IPv6 island

Global IPv6 Internet

Tunnel Broker Server

IPv4 Internet

Tunnel Server

Copyright: see page 2
Scalable, Carrier-grade Solutions

Native IPv6 (IPv4) hosts exchange IPv6 (IPv4) packets through an IPv4 (IPv6) network.
Goals

- Single host support not so relevant
  - Beyond experimentation by individuals
- We need to migrate networks
- Still need to support IPv4 host IPv6 host communication

Several Options

- DS-Lite
- A+P (DS-Lite evolution)
- NAT64
- 6PE (MPLS-based)
NAT is widely used today

IPv4 RFC 1918 (private addressing)

Carrier Grade NAT
Large Scale NAT

CGN/LSN NAT 44

RFC 1918

IPv4 Provider Private

IPv4 Internet

Copyright: see page 2
How do we like NAT?

- Problematic with inbound sessions
  - E.g., servers
  - NAT + STUN/TURN may be ok for peer-to-peer sessions
- Bottleneck and single point of failure

Nevertheless

- Several (independent) cascaded instances of NAT are now very common
  - Starting from virtual machines
- Difficult to do without due to scarce addresses
Same Architecture with IPv6

- **DS-Lite**
  - RFC 1918
  - IPv6
  - IPv4 Internet
  - CPE
  - AFTR

- **A+P**
  - RFC 1918
  - IPv6
  - IPv4 Internet
  - CPE
  - AFTR

- **NAT64**
  - IPv6
  - IPv4 Internet
  - CPE
  - NAT64

CPE: Customer Premises Equipment
AFTR: Address Family Transition Router

Note where the NAT function is located!
AFTR: Address Family Transition Router

- Allows IPv4 host to communicate with IPv4 hosts over an IPv6 carrier infrastructure
  - Residential host and current providers
- Features an IPv6 tunnel concentrator and possibly a large-scale NAT
- In use in DS-Lite and A+P
DS-Lite (Dual-Stack Lite)

- Dual-stack at the edge
- IPv6-only Service Provider backbone

- IPv4 traffic is decapsulated, IPv4 addresses are translated into public addresses, and forwarded to the IPv4 Internet
- IPv4 traffic is tunneled to the CGN
- IPv6 traffic goes natively to the IPv6 Internet
- DS-Lite CPE tunnels packets toward the AFTR
- IPv6 traffic is tunefully to the IPv6 Internet
- IPv6-only NSP
- IPv6 Internet
- IPv4 Internet
- IPv4 NAT44
- IPv6-only Service Provider backbone
- IPv4 + IPv6
- IPv6 Internet
- DS-Lite CPE
- NSP-assign private IPv4 space (if plain NAT44 in AFTR)
Properties

- Reduces requirement for IPv4 addresses compared to dual-stack approach
  - Dual-stack requires public IPv4 address per host
- Extended NAT enables customer assigned (i.e., overlapping) addressing
  - IPv6 address of CPE in NAT table
Limitation

- NAT is not under control of customer
  - Same problem as with CGN

- Problematic with servers
  - Static mapping and port forwarding cannot be configured
A+P (Address plus Port)

- NAT is under control of customer
- Ranges of TCP/UDP ports are assigned to each customer
- Only ports used by NAT on outside

The NAT translates private IPv4 addresses into a public IPv4 address; ports are translated to a set of pre-assigned ports; packets are tunneled toward the AFTR.
Features

- No problem with overlapping private address spaces at customers’
- Ports can be assigned automatically to CPE using the Port Control Protocol (PCP)
  - CPE can negotiate more ports any time
- AFTR is just a IPv4-in-IPv6 (proto-41) tunnel terminator
  - NAT44 is no longer needed in the AFTR
NAT64 + DNS64: Deployment Scenarios

IPv6

IPv4 Internet

NAT64

DNS64

IPv6

IPv4 + IPv6

IPv4 Internet

NAT64

DNS64
NAT64 + DNS64: Principles

- An IPv6 prefix is dedicated to mapped IPv4 addresses
  - Either well-known or network-specific
- DNS64 maps A records into AAAA using NAT64 prefix, then serves A and AAAA records to the client
- NAT64 router advertises NAT64 prefix into IPv6 network to attract traffic toward IPv4 hosts
DNS64: Name Resolution

IPv6

IPv4 + IPv6

IPv4

www.example.com

DNS (example.com)

NAT64

AAAA for www.example.com?

Name error

www.example.com is 64:FF9B::20.2.2.2

Well-known or deployment specific prefix (configured on DNS64 and possibly NAT64)

www.example.com is 20.2.2.2

A for www.example.com?

AAAA for www.example.com?
NAT64: Packet Forwarding

- **NAT64 (outbound)**
  - Translates IPv6 address and packet into IPv4
  - Picks a free IPv4 address/port from its pool
  - Builds NAT session entry

- **IPv6**
  - 2^{32} IPv6 addresses

- **IPv4 + IPv6**
  - Public IPv4 address pool (30.3.3.3)

- **IPv4**
  - www.example.com

- **DNS (example.com)**
  - DNS64

- TCP SYN for 64:FF9B::20.2.2.2
- TCP ACK from 64:FF9B::20.2.2.2
- TCP SYN for 20.2.2.2
- TCP ACK for 30.3.3.3

- Copyright: see page 2
A Specific Use Case: Individual IPv4 Server Reachability

No need for DNS64: a regular DNS is enough

Network service provider advertises address: 2002:FF9B::/96

IPv6
IPv4 + IPv6
IPv4

DNS (example.com)

www.example.com       A       20.2.2.2
                        AAAA   2002:FF9B::20.2.2.2
NAT64 Deeper Dive

- NAT64 prefix
  - Any /32, /40, /48, /56, /64 or /96 prefix
  - Well-Known Prefix (WKP): 64:FF9B::/96
  - /64 recommended for Network Service Providers

- Stateful NAT64
  - Very similar to PAT (stateful NAT44)
  - Individual TCP and UDP sessions + ICMP replies are translated
  - Source IPv6 address + port number used in the lookup

- Stateless NAT64
  - Each IPv6 address is translated into one IPv4 address
    - Either static mapping or IPv6 hosts have NAT64 prefix
  - Only ICMP packets and IP headers are translated
  - Limited use: few IPv6 hosts
    - E.g., IPv6 only servers, which requires static mapping
NAT64 + DNS64 Limitations

- Only when the DNS is involved
  - I.e., hostnames are used
  - E.g., it does not work in case the user directly specifies an IPv4 address
    - E.g., ping 1.2.3.4

- No DNSSEC
  - In DNSSEC authoritative DNS signs record
  - But DNS64 modifies records
MPLS-based Solutions

- Meant for interconnecting IPv6 islands through an IPv4 MPLS backbone
  - Also dually (IPv4 islands IPv6 backbone)
- Most network service providers deploy MPLS anyway

IPv6

Provider (P) Router

MPLS network

Provider Edge (PE) Router

Customer Edge (CE) Router

IPv6

IPv6

IPv6
IPv6 and MPLS

- Data plane is agnostic to IPv4/IPv6
  - Forwarding is based on labels
- Control Plane is not
  - Destinations are identified with an IP address
Native IPv6 over MPLS

Full Control Plane upgrade

IPv6 support in

- Routing
- Label Distribution Protocol
- Management

Possibly dual (IPv4 and IPv6) support
IPv6 over Circuit Transport

- MPLS as layer 2 connectivity
  - LSP -> L2 tunnel
- PEs (i.e., label edge routers) are IPv6 aware
  - Static routes to IPv6 destinations
  - Routing with remote PEs
- No changes needed to P routers
- Scalability problems
  - L2 tunnels and routes configured manually
  - Possibly mesh topologies of L2 tunnels
6PE

- Only PE routers need to be changed
- Same mechanisms as MPLS-based VPN
  - VPN services can be offered on the same backbone
- Very scalable
- Minimum configuration required
- As more customers require IPv6 connectivity, provider might consider migrating to native support
Architecture

2001:1::/64

Dual stack IPv4-IPv6 with MP-BGP support

IPv4-only routers

2001:3::/64

CE-1

PE-1

P-1

P-2

PE-2

CE-3

CE-2

CE-4

IPv4-only routers

Dual stack IPv4-IPv6 with MP-BGP support

2001:1::/64

2001:3::/64
Announcing IPv6 Networks

- CE and 6PE connect through **native** IPv6 interfaces
- Routing information is exchanged
  - Any routing protocol (RIP, OSPF, BGP)
  - Static routes

Network 2001:3::/64 is this way

IGP (e.g., RIP)

Copyright: see page 2
Propagating IPv6 Routes

- 6PEs exchange routing information through MP-BGP sessions over IPv4
- 6PE IPv4-mapped address as the BGP Next-Hop

Network 2001:3::/64 is reachable via BGP Next Hop ::FFFF:20.2.2.2 (and is bound to the MPLS label 2001:3::)
Redistributing IPv6 Routes

- PEs propagate advertisements to IPv6 networks through IGP
  - This may not be needed if an IPv6 default route is used in CE-1

```
+----------------+ +----------------+ +----------------+ +----------------+
|     CE-1      |   PE-1     |     P-1      |   PE-2     |
|   CE-2       |   PE-1     |     P-1      |   PE-2     |
|+---------------+-------------+--------------+-------------+
| 2001:1::/64   | Network 2001:3::/64 is this way | 2001:3::/64 |
```
Internal Routing

- 6PE and P routers advertise their IPv4 prefixes with
- Labels are bound to each PE
  - Topology-based label binding

2001:1::/64  
CE-1  
PE-1  
P-1  
P-2  
PE-2  
CE-3

2001:3::/64  
CE-2  
P-2  
P-1  
CE-4  

IGP (e.g., OSPF) propagates reachability of 20.2.2.2 (corresponding network)

LDPv4 binds a label to 20.2.2.2

IPv4 address:
20.2.2.2

Copyright: see page 2
Label Distribution

- For IPv4 destinations among P and 6PE routers
  - LDP or RSVP
- For IPv6 destination directly among 6PEs using MP-iBGPv4
IPv6 Packet Forwarding

- CE routers send IPv6 packets to 6PE routers
  - Static (default) route or dynamic route from IGP

```
2001:1::/64

CE-1

CE-2

PE-1

P-1

P-2

PE-2

2001:3::/64

PE-2

CE-3

CE-4
```
IPv6 Packet Forwarding

- 6PE has label mapping for destination
  - Label as distributed by iBGP (e.g., 2001:3::)
  - Next hop: 6PE identified the BGP Next Hop
    - IPv4-mapped IPv6 address (e.g., ::FFFF:20.2.2.2)

2001:1::/64

Aggregate label (iBGP) (e.g., 2001:3::)

2001:3::/64
IPv6 Packet Forwarding

- Next hop is reachable through IPv4
  - Advertised by IGP
- Label mapping distributed with LDP/RSVP
  - Pushed

```
2001:1::/64

CE-1
IPv6

PE-1
IPv6

P-1
IPv6

I Pv6

P-2
IPv6

PE-2
IPv6

CE-3
IPv6

 Aggregate label (iBGP) (internal)

2001:3::/64

CE-2
IPv6

Penultimate hop popping

Copyright: see page 2
```
A closer look to the labels

- Inner label in principle not required for
  - It keeps the solution the same as for VPN
  - Penultimate hop should be able to forward IPv6 packet
    - Or PHP should not be performed
References

