Introduction to IPv6

Administration de Réseaux

Dino Lopez

http://www.i3s.unice.fr/~lopezpac/

- Inspired from
  - The presentation “Introduction to IPv6 and its Security” of Ciscohttp://www.montefiore.ulg.ac.be/~leduc/cours/ISIR/ipv6_sec.pdf
  - The presentation “APNIC Tutorial: IPv6 Essentials” https://training.apnic.net/docs/TIP601.pdf
- Request For Comments (RFCs)

A bit of history

- 1968 – DARPA
  - (Defense Advanced Research Projects Agency) contracts with BBN to create ARPAnet
- 1969 – First four nodes
- 1970 - Five nodes:
  - UCLA – Stanford - UC Santa Barbara - U of Utah – BBN
- 1971 – 15 nodes, 23 hosts connected
- 1974 – TCP specification by Vint Cerf & Bob Kahn
- 1983 – TCP/IP
  - On January 1, the Internet counts 1000

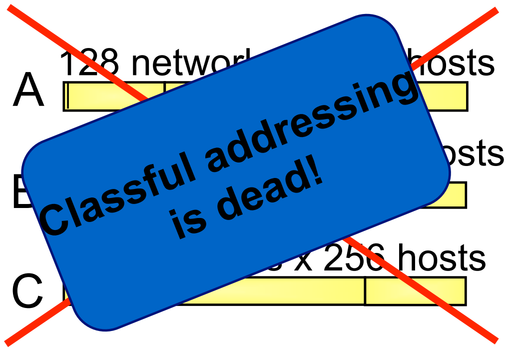
Address architecture

- Initially, only 256 networks in the Internet
- Network classes : A, B, C
- 1992 – Internet challenges
  - Address space depletion
  - Wasteful allocations
  - Routing chaos, since no aggregation

Classful vs Classless

Best Current Practice

Classful



Obsolete
- *inefficient*
- *depletion of B space*
- *too many routes from C space*

Classless

Addresses	Prefix	Classful	Net Mask
...	...	...	...
8	/29		255.255.255.248
16	/28		255.255.255.240
32	/27		255.255.255.224
64	/26		255.255.255.192
128	/25		255.255.255.128
256	/24	1 C	255.255.255.0
...	...	...	...
4096	/20	16 C's	255.255.240.0
8192	/19	32 C's	255.255.224
16384	/18	64 C's	255.255.192
32768	/17	128 C's	255.255.128
65536	/16	1 B	255.255.0.0
...	...	...	... *

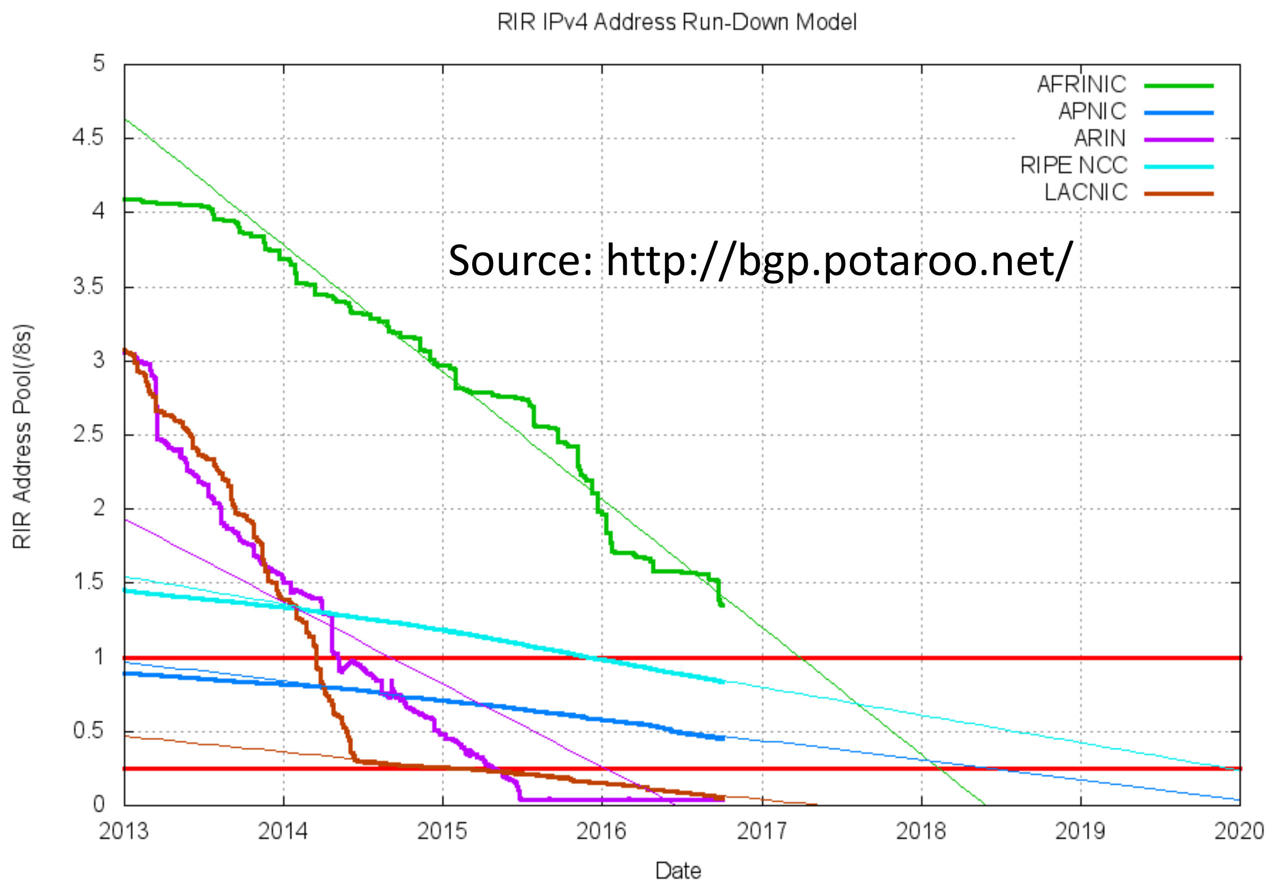
Management by RIRs

Source: https://www.apnic.net/

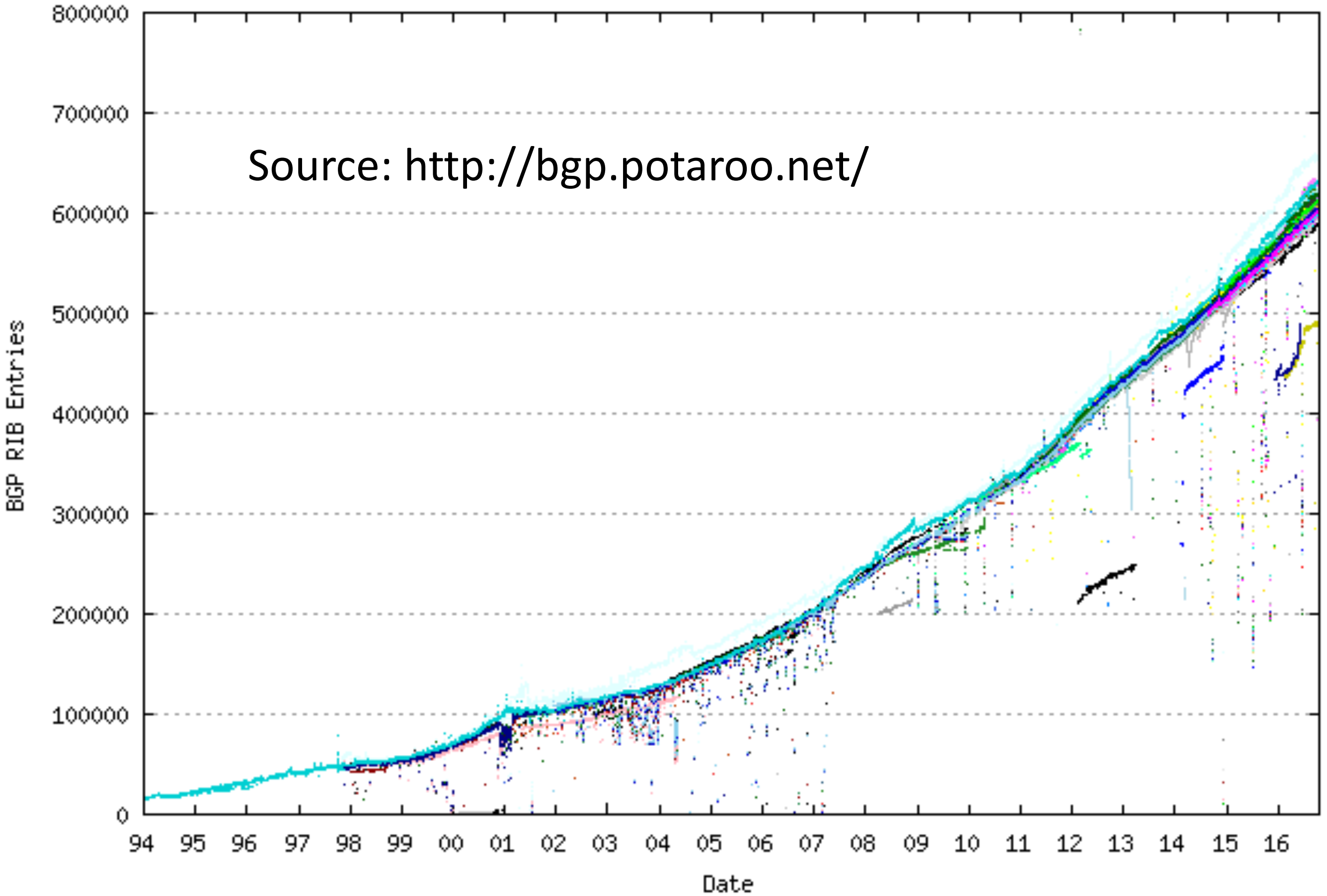
IPv6 history

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

IPv4 exhaustion



IPv4 BGP Table

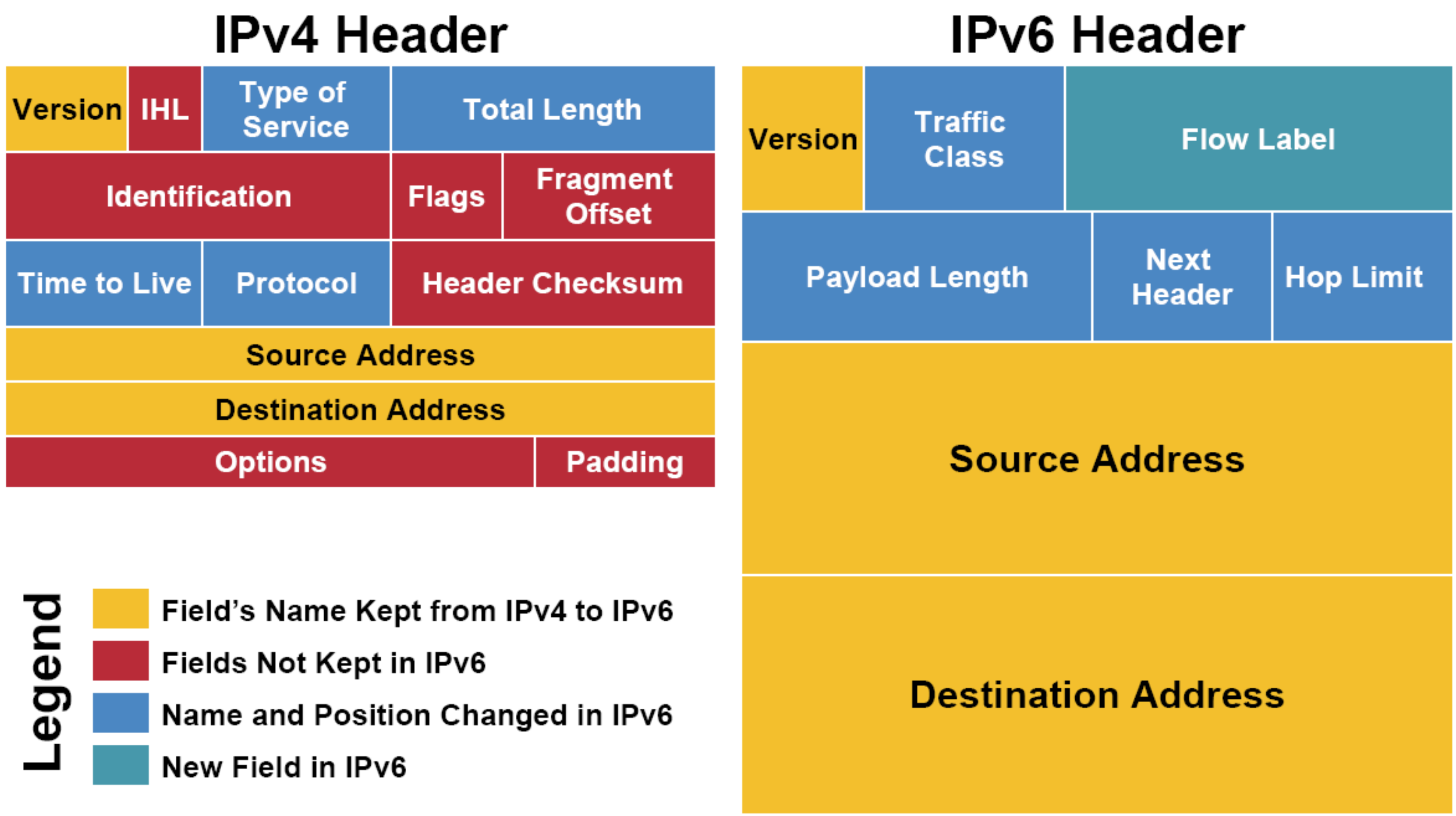


Source: http://bgp.potaroo.net/

IPv6 improvements

- Large address space → 128-bit addresses
- Management
  - Stateless autoconfiguration
- Multicast
  - Builting features for multicast groups and new “anycast” group
- Mobile IP
  - Eliminate triangular routing through Route Optimization
- Virtual Private Networks
  - Built-in support for ESP/AH encrypted/authenticated VPN protocols
- Built-in support for QoS tagging
- No more broadcast
- No hop-by-hop segmentation
  - Path MTU discovery

IPv4 vs IPv6 header



Source: https://343networks.files.wordpress.com/

IPv6 extension header

- The use of an extension header makes it easy to add new features in IPv6
- The number of extension headers is variable
- The extension header is placed between the main IPv6 header and the payload
- Some values for the Next Header field
  - 0 Hop-by-hop option
  - 2 ICMP
  - 6 TCP
  - 17 UDP
  - 43 Source routing
  - 44 Fragmentation
  - 50 Encrypted security payload – 51 Authentication
  - 59 Null (No next header)
  - 60 Destination option

Order of Extension Header

- Source node place the EH in the following order
  - 1. Hop-by-hop
  - 2. Routing
  - 3. Fragment
  - 4. Authentication
  - 5. Encapsulating security payload
  - 6. Destination option
  - 7. Upper-layer
- Order is important
  - Only hop-by-hop must be processed at every intermediate node
  - Routing headers must be processed by intermediate routers
  - At the destination, fragmentation must be handled before others

The MTU in IPv6

- Path MTU discovery
  - The sender sends first a packet as big as required by the application layer
  - If the device receives an ICMP packet with a too big message signaling, the sender must apply the provided MTU size
  - Each source needs to track the MTU size for each session
- Minimum MTU is 1280
- Most efficient MTU is 1500
- Maximum datagram size is 64k

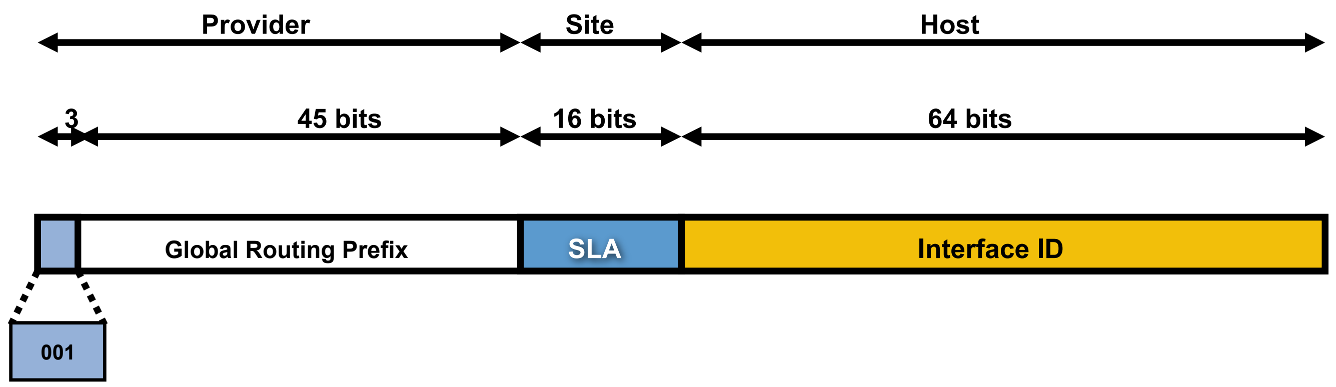
IPv6 addresses representation

- Format
  - x:x:x:x:x:x:x:x where x is 16 bits hexadecimal field
  - 2001:0000:130F:0000:0000:09C0:876A:130B
  - Case insensitive
- Leading zeros in a field are optional:
  - 2001:0:130F:0:0:9C0:876A:130B
- Successive fields of 0 are represented as ::, but only once in an address:
  - 2001:0:130F::9C0:876A:130B
  - FF01:0:0:0:0:0:0:1 => FF01::1
  - 0:0:0:0:0:0:0:1 => ::1
  - 0:0:0:0:0:0:0:0 => ::
- RFC5952 recommends to use double colons for the rightmost set of :0:
- In an URL
  - http://[2001:db8:4f3a::206:ae14]:8080/index.html
- Prefix representation
  - Just like in IPv4 CIDR

IPv6 addressing model

- Unicast
  - An identifier for a single interface
- Multicast
  - Identifier for a group of nodes
- Anycast
  - Identifier for a set of interfaces
  - Anycast routing → find the closest peer

Global Unicast addresses

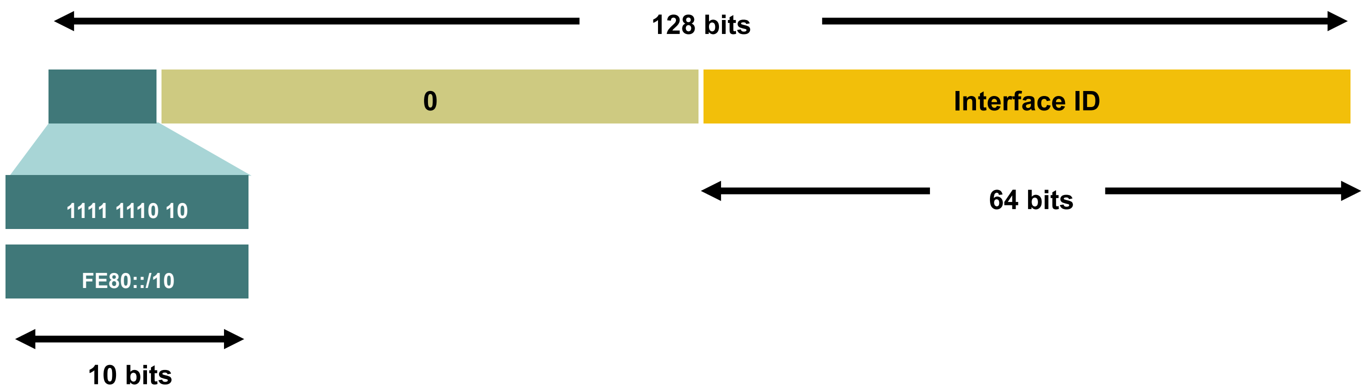


- Global Unicast Range
  - Currently: 2000::/3

Global Unicast Addresses

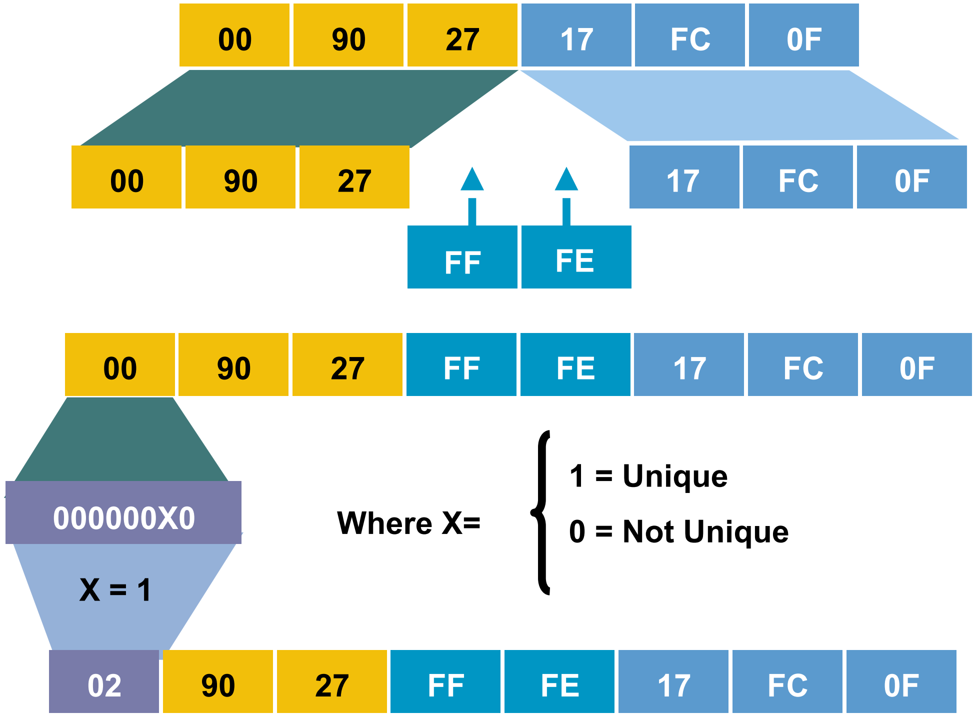
- All five RIRs are given a /12 from the /3 to further distribute within the RIR region
  - APNIC - 2400:0000::/12
  - ARIN - 2600:0000::/12
  - AfriNIC - 2C00:0000::/12
  - LACNIC - 2800:0000::/12
  - Ripe NCC - 2A00:0000::/12
- 6to4 Addresses
  - **2002::/16**
  - Designed for a special tunneling mechanism [RFC 3056] to connect IPv6 Domains via IPv4 Clouds
  - Automatic tunnel transition Mechanisms for IPv6 Hosts and Routers
  - Need 6to4 relay routers in ISP network

IPv6 Local Addresses



- Link-local addresses
  - Have a limited scope of the link
  - Allow to communicate with anyone on the link (e.g. host or router)
  - Mandatory and automatically configured
  - fe80::/10

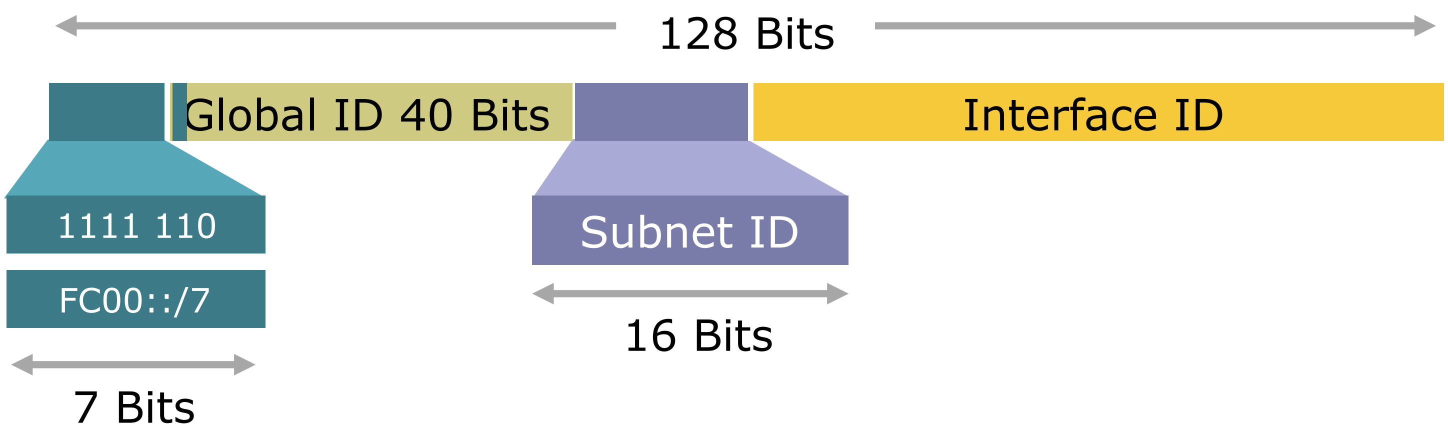
Extended Unique Identifier (EUI-64) format



Local addresses

- In 1995, RFC 1884 reserved the block fec0::/10 for site-local addresses
  - RFC 3879 (September 2004) deprecated this address range.
- In October 2005, RFC 4193 was published, reserving the address block fc00::/7 for use in private IPv6 networks.
- fc00::/7 is divided into 2 groups
  - The block fc00::/8 has not been defined yet. Proposed to be managed by an allocation authority.
  - The block fd00::/8 is defined for /48 prefixes, formed by setting the 40 least-significant bits of the prefix to a randomly generated bit string. This results in the format fdxx:xxxx:xxxx::
  - Some webtools to generate ULA prefix
    - http://www.sixxs.net/tools/grh/ula/
    - http://www.goebel-consult.de/ipv6/createLULA

Local addresses



- Unique-Local Addresses used For:
  - Local communications & inter-site VPNs
  - Local devices such as printers, telephones, etc
  - Site Network Management systems connectivity
- Not routable on the Internet

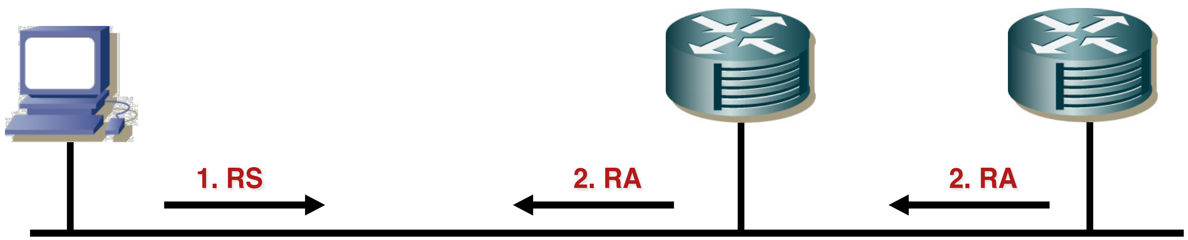
Examples and Documentation Prefix

- For exemple 3fff:ffff::/32
- For documentation 2001:0DB8::/32

Special addresses

- ::/128
  - Unspecified address
  - Equivalent to 0.0.0.0 in IPv4
- ::/0
  - default route
- ::1/128
  - Loopback address
  - Equivalent to 127.0.0.0/8 in IPv4

Stateless autoconfiguration

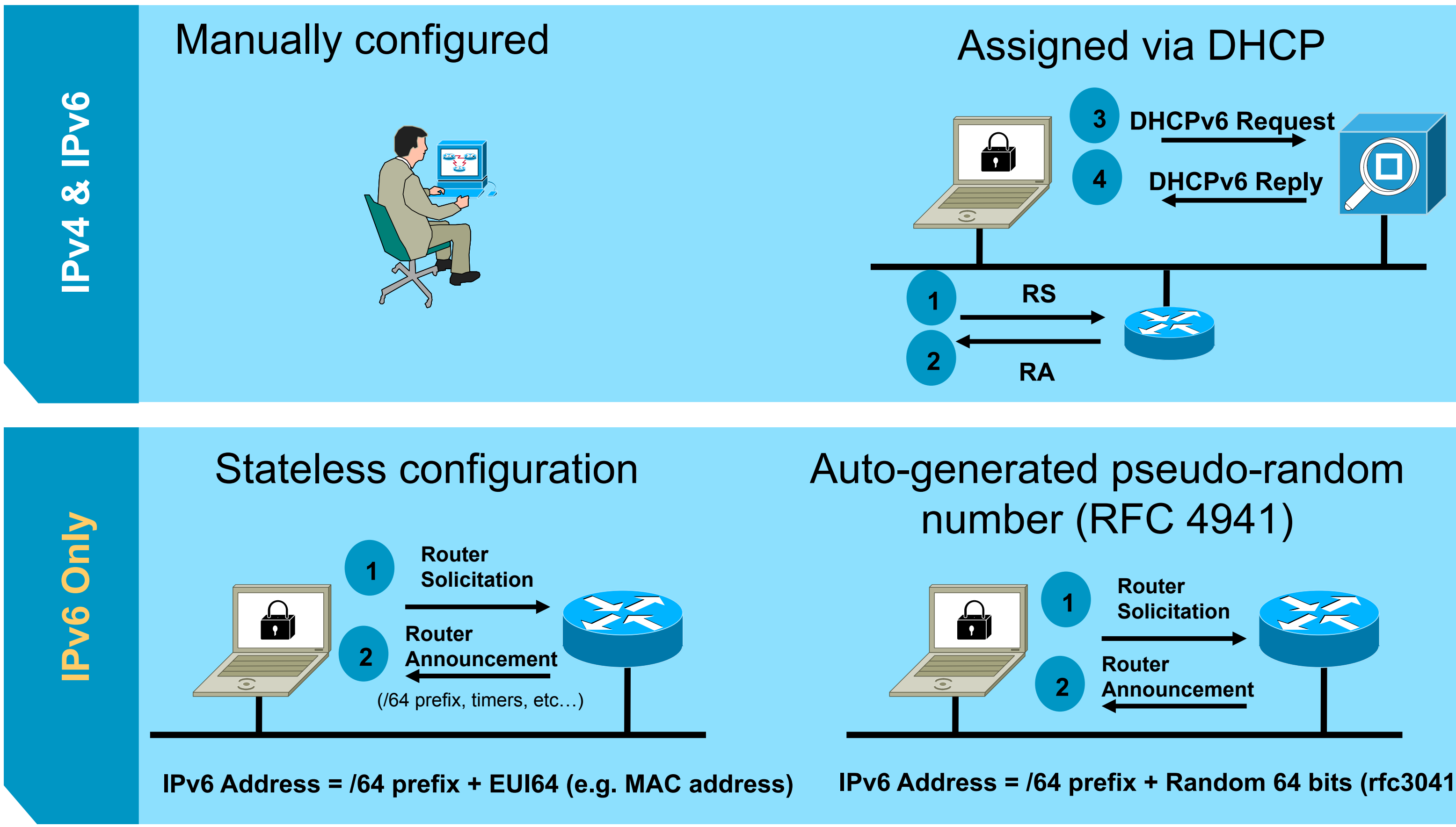



```
1. RS:
ICMP Type = 133

Src = ::
Dst = All-Routers multicast Address
query= please send RA
```

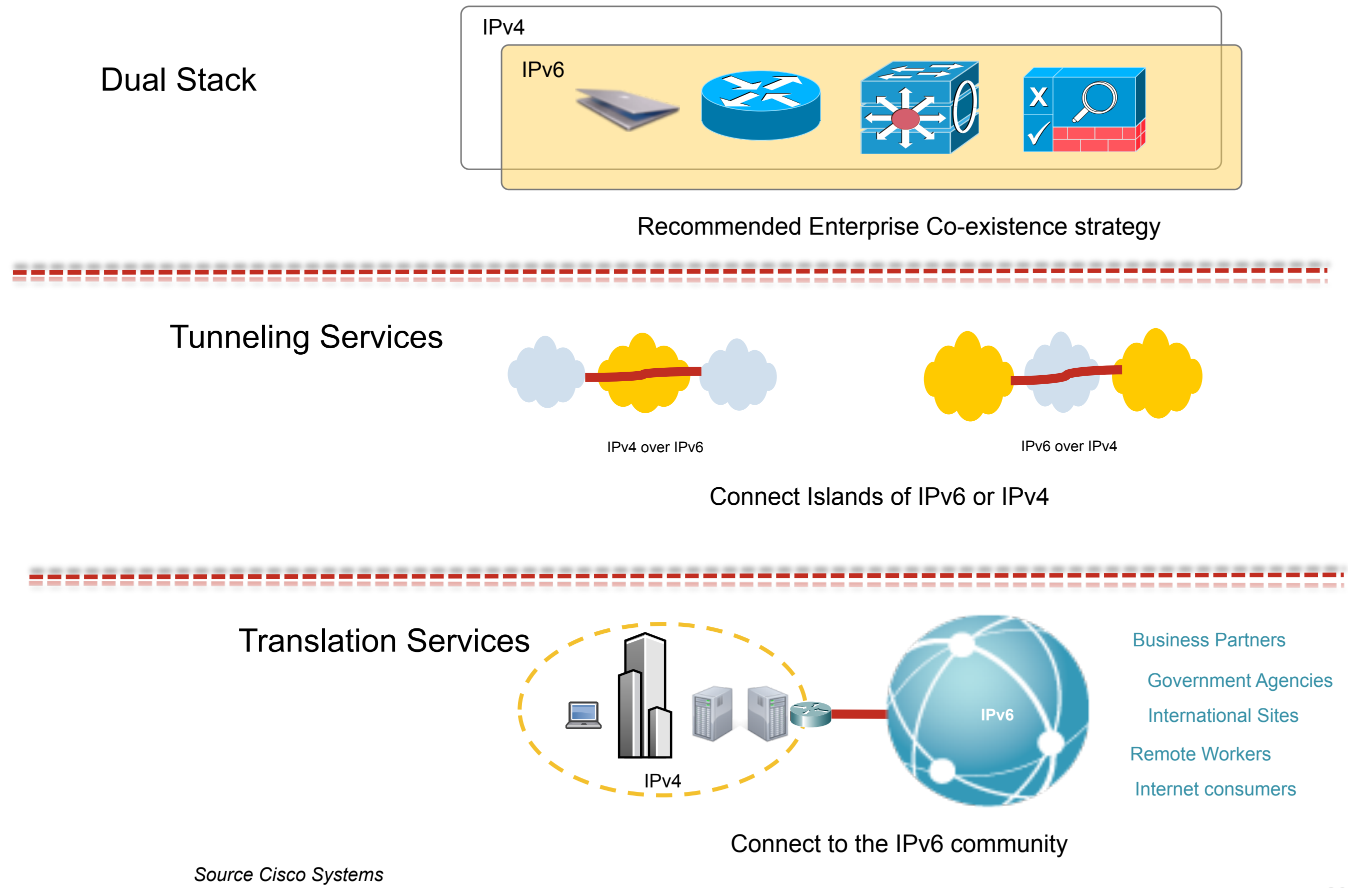
```
2. RA:
ICMP Type = 134

Src = Router Link-local Address
Dst = All-nodes multicast address
Data= options, prefix(es), lifetime,  autoconfig
flag (no managed flag)
```

- RS = Router Solicitation, RA = Router Advertisement

IPv6 Address assignment

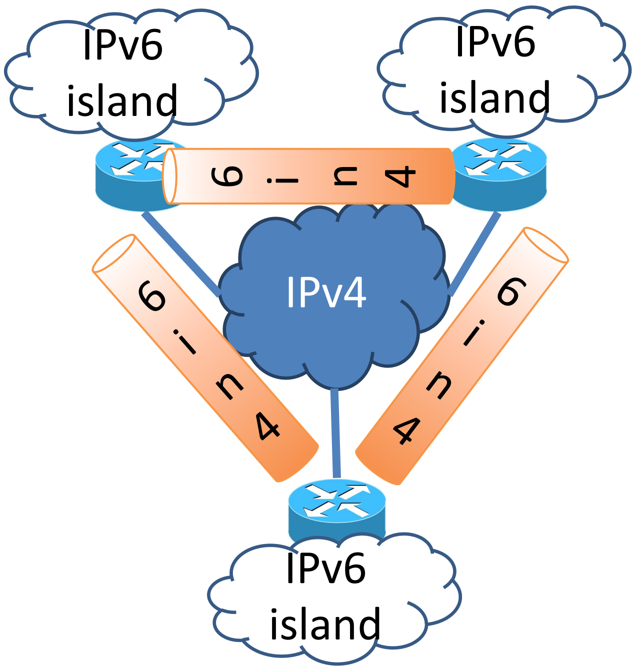


IPv4 – IPv6 coexistence



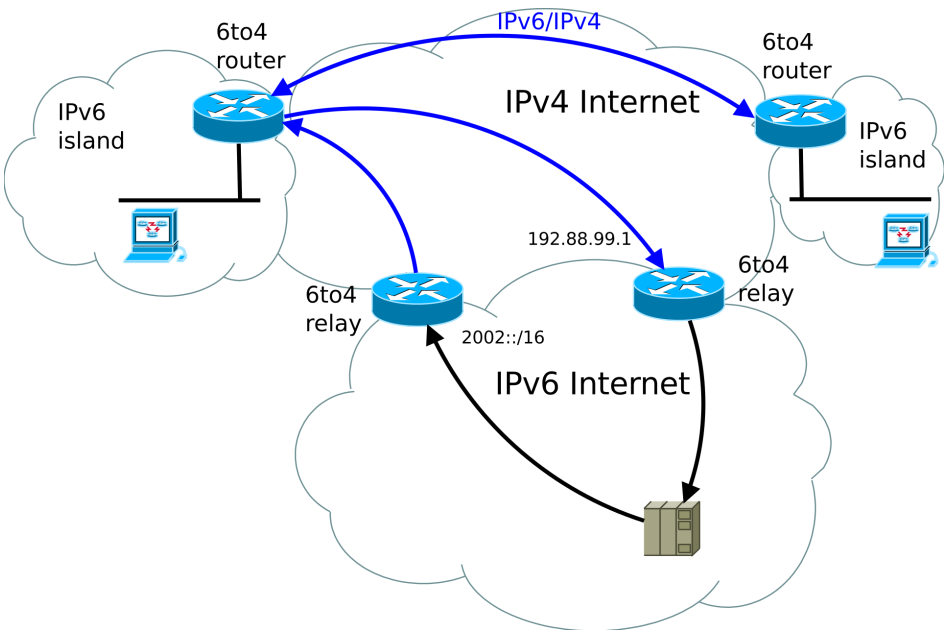
6in4

- Encapsulate IPv6 packets into IPv4 packets
  - Interconnect IPv6 islands
- Needs as many tunnels as IPv6 islands we need to reach
  - And as many routes as needed using the tunnel devices as the output interface
  - Difficult to set up



6to4

- Transfer IPv6 packets over IPv4 networks
  - Connect IPv6 islands
  - Migration from IPv4 to IPv6
- Does not need one explicit tunnel per IPv6 island
- 6to4 transition solutions need to employ 6to4 IPv6 addresses
  - 6to4 network -> 2002::/16
- One IPv6 network connects to IPv4 networks with 6to4 routers
  - 6to4 relays to connect to native IPv6 networks



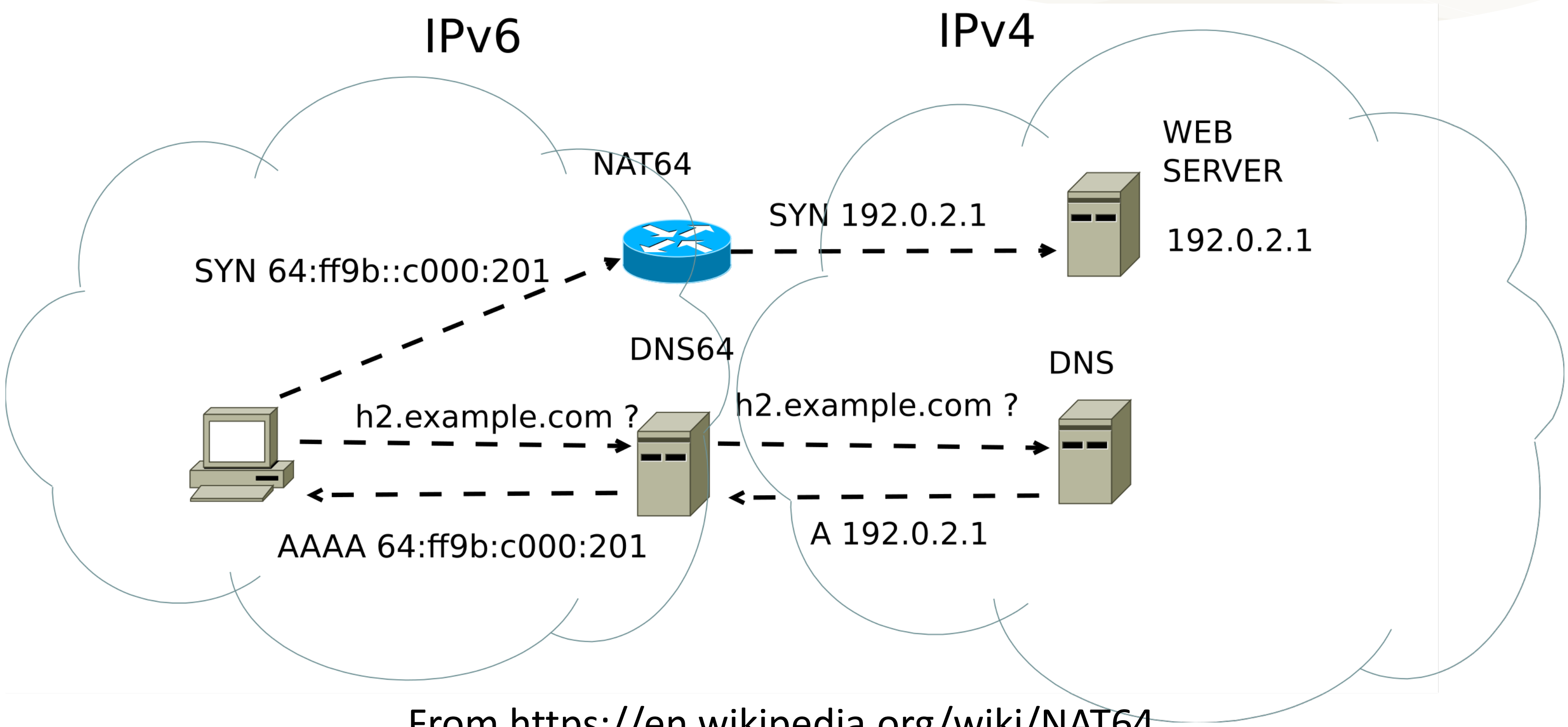
6to4 addressing

- 6to4 networks need to employ 6to4 addresses
  - The 6to4 tunnel can be configured with an IPv6 address prefixed with 2002:xxxx:xxxx::/16, where the “x” characters represent the IPv4 address of the outgoing device
  - The 6to4 network can be build on a 2002:xxxx:xxxx::/48 prefix
  - ipv4="1.2.3.4"; printf "2002:%02x%02x:%02x%02x::1\n" `echo $ipv4 | tr "." " "`
- The route entry to reach a remote IPv6 island must indicate as a gateway, the IPv6 address of the remote 6to4 tunnel device
  - ip -6 route add default via ::192.88.99.1 dev tun6to4 metric 1

IPv6 - Some useful Linux commands

- Add an IPv6 address
  - ip -6 addr add 2001:db8::1/64 dev eth0
- Add a route
  - ip -6 route add 2001:db8:2::/64 via 2001:db8:1::1 dev eth0
- Enable IPv6 forwarding
  - sysctl net.ipv6.conf.all.forwarding=1
- Create a 6in4 or 6to4 tunnel
  - Create a sit mode tunnel
  - For a 6in4 tunnel, specify the remote IPv4 address
  - For a 6to4, you do not specify the remote IPv4 address. Use “any” instead
  - ip tunnel add tun6in4 mode sit remote 192.168.0.2 local 192.168.0.1 ttl 64
- Show the IPv6 routing table
  - Ip -6 r s
- Show the neighbors table
  - ip -6 neigh show

NAT64 – DNS64



From https://en.wikipedia.org/wiki/NAT64