istic

IPv6 Courses

November 13, 2015

Table of Contents

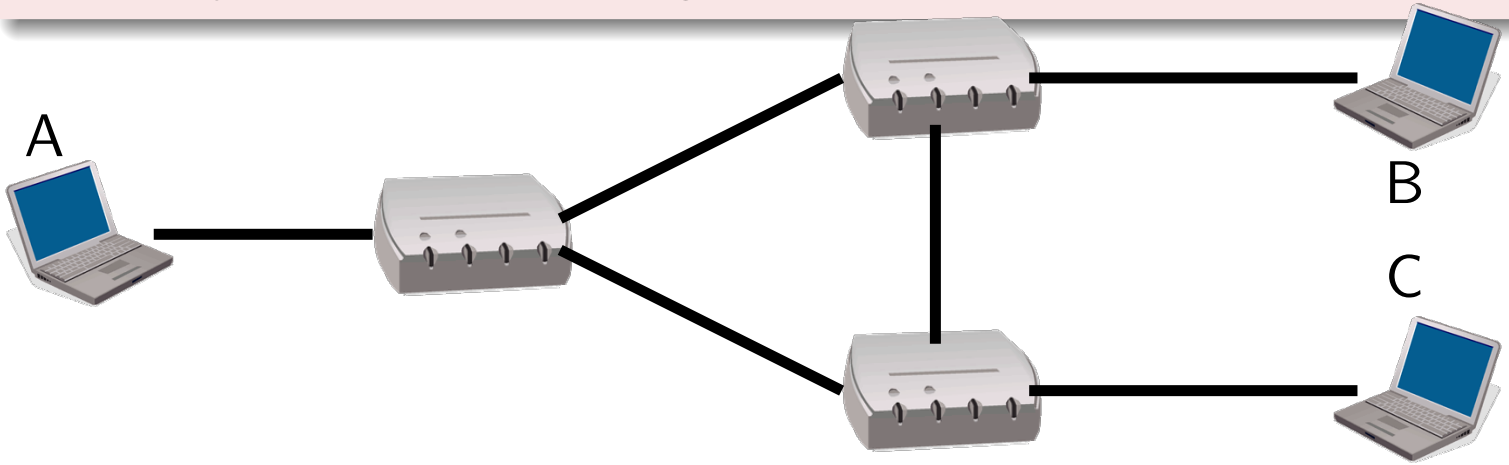
Concepts

Datagram

What Is A Datagram

Definition

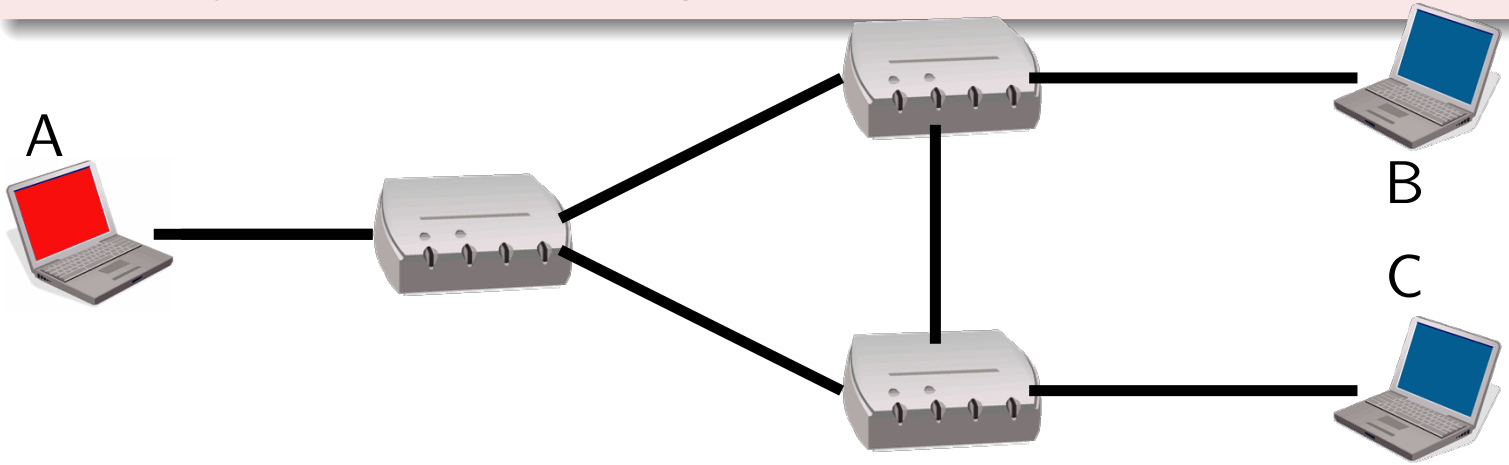
1. Every packet is processed separately
2. No state in the network
3. Destination address MUST be repeated in each packet
4. Every equipment MUST agree on a common header format



What Is A Datagram

Definition

1. Every packet is processed separately
2. No state in the network
3. Destination address MUST be repeated in each packet
4. Every equipment MUST agree on a common header format

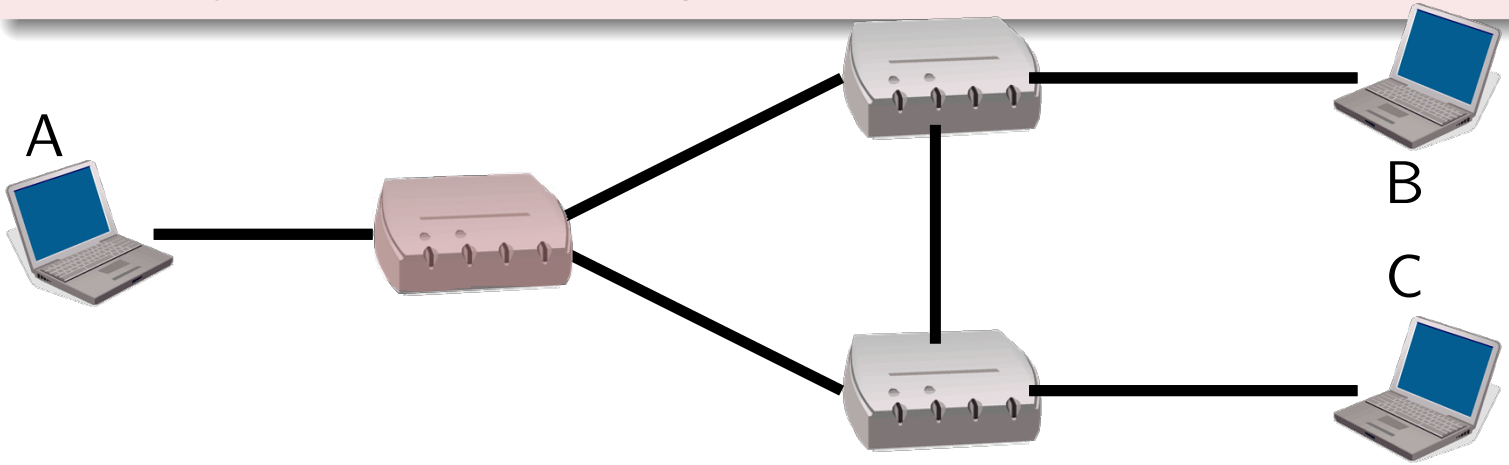


A sends a packet to B

What Is A Datagram

Definition

1. Every packet is processed separately
2. No state in the network
3. Destination address MUST be repeated in each packet
4. Every equipment MUST agree on a common header format

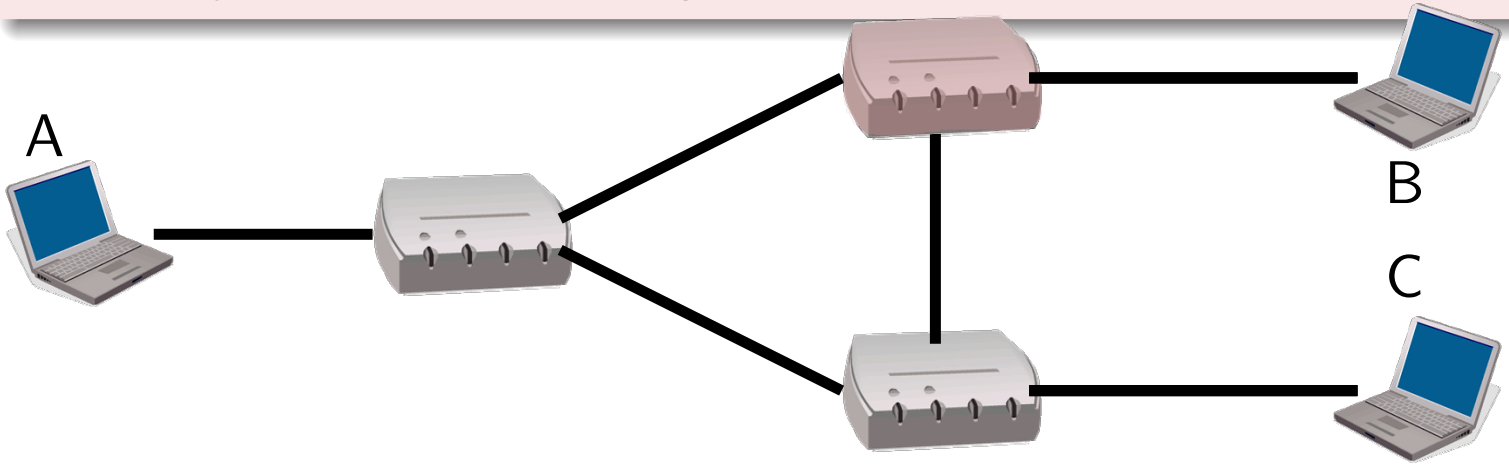


The first router looks at the header to find the exit interface

What Is A Datagram

Definition

1. Every packet is processed separately
2. No state in the network
3. Destination address MUST be repeated in each packet
4. Every equipment MUST agree on a common header format

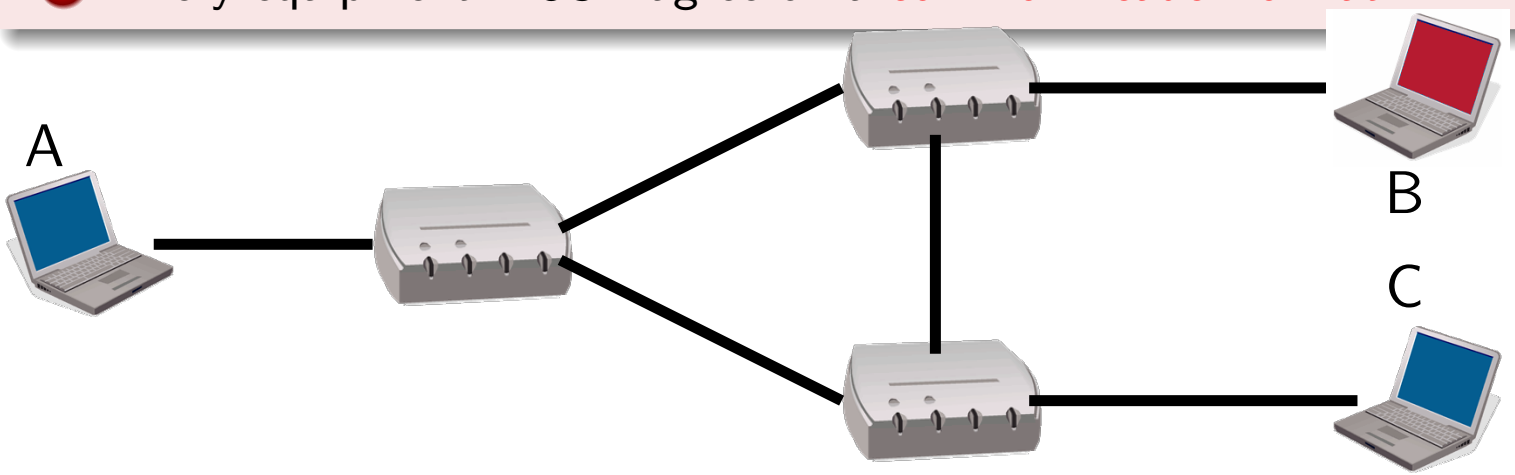


The second router looks at the header to find the exit interface

What Is A Datagram

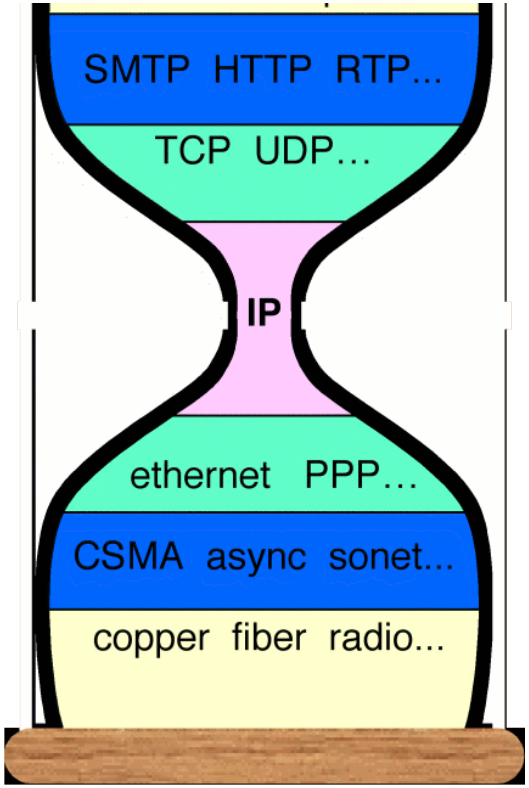
Definition

1. Every packet is processed separately
2. No state in the network
3. Destination address MUST be repeated in each packet
4. Every equipment MUST agree on a common header format



B accepts the packet

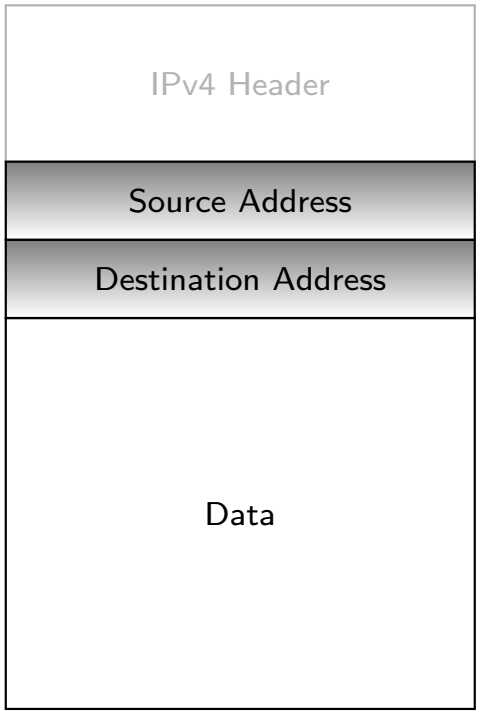
IP Layer



- IP is kept simple
  - Forwards packet towards destination
- IP on everything
  - Adapt IP protocol on every layer 2
- Everything on IP
  - Write applications to use IP layer (through L4: TCP, UDP)
- IP must facilitate network interconnection
  - Avoid ambiguities on addresses

http://www.ietf.org/proceedings/01aug/slides/plenary-1/index.html Steve deering, Watching the Waist of the Protocol Hourglass, IETF 51, London

Destination Address Processing



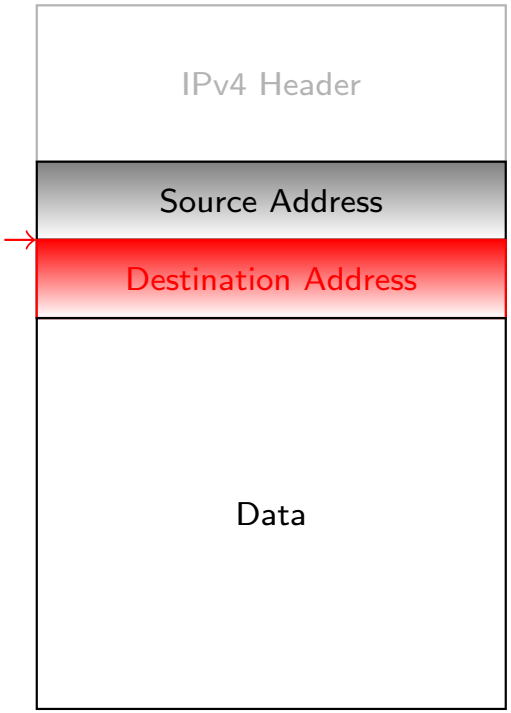
The destination address must be easily accessible:

- Fixed location
- Fixed size
- Aligment in memory

RFC 791 (Sept 1981)

Addresses are fixed length of four octets (32 bits)

Destination Address Processing



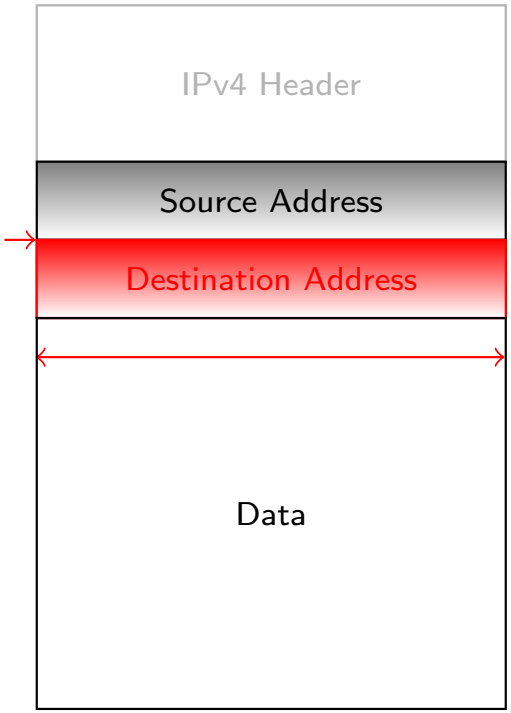
The destination address must be easily accessible:

- Fixed location
- Fixed size
- Aligment in memory

RFC 791 (Sept 1981)

Addresses are fixed length of four octets (32 bits)

Destination Address Processing



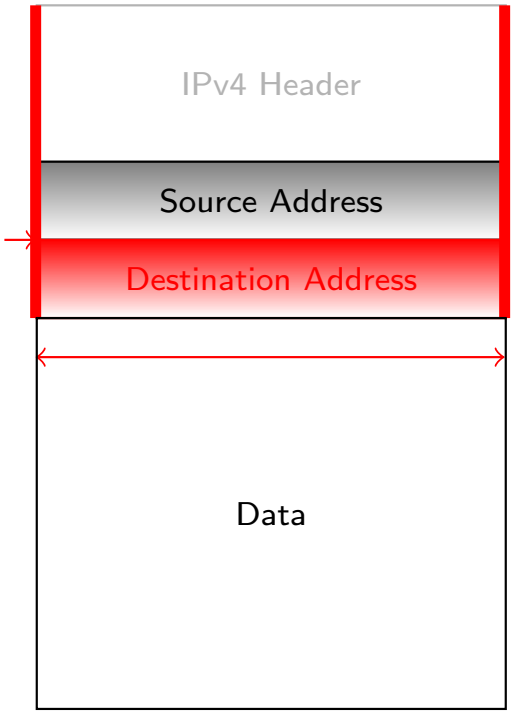
The destination address must be easily accessible:

- Fixed location
- Fixed size
- Aligment in memory

RFC 791 (Sept 1981)

Addresses are fixed length of four octets (32 bits)

Destination Address Processing



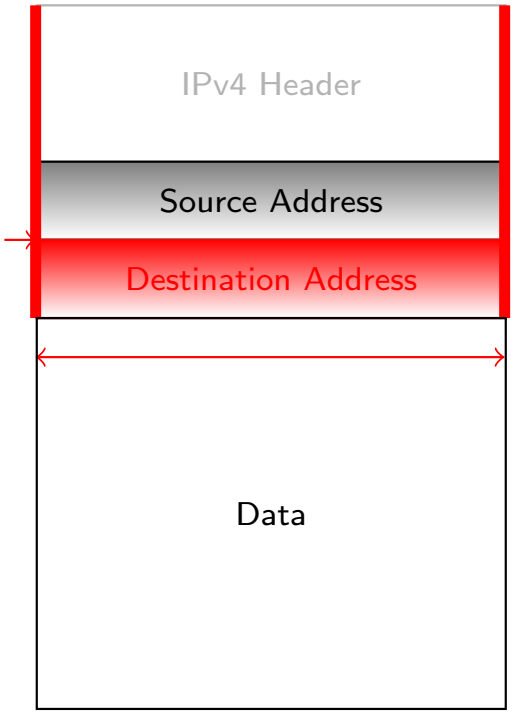
The destination address must be easily accessible:

- Fixed location
- Fixed size
- Aligment in memory

RFC 791 (Sept 1981)

Addresses are fixed length of four octets (32 bits)

Destination Address Processing



The destination address must be easily accessible:

- Fixed location
- Fixed size
- Aligment in memory

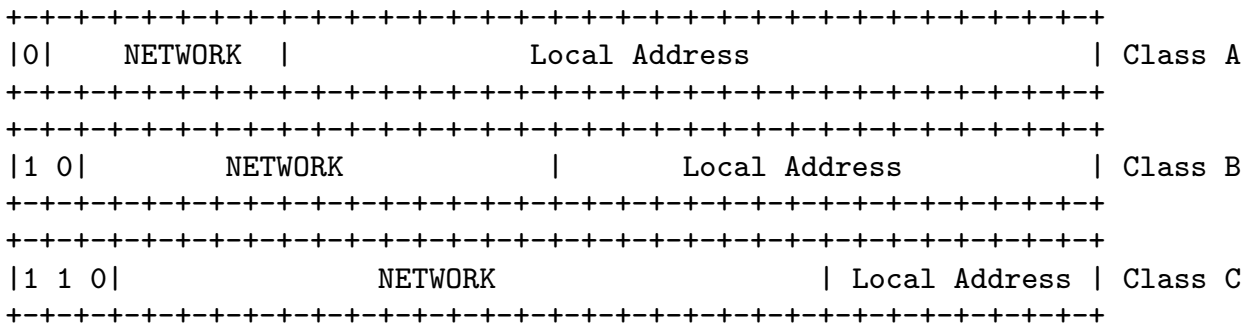
RFC 791 (Sept 1981)

```
Addresses are fixed length of four octets
(32 bits)
```

Concepts

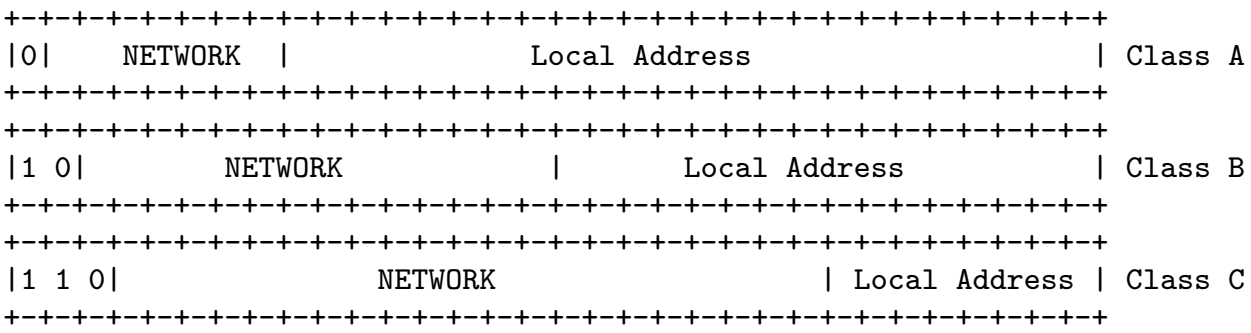
Addresses

IPv4 address allocation (originally)



- The address is split into two parts:
  - Network part
  - Host part
- Initially the boundary was given by a prefix
  - 3 boundaries called classes
  - 1 class (D) for mutlicast added later
  - 1 class (E) reserved (never used)
- An authority used to give unique prefix to sites
- This plan was developed to guarantee address uniqueness

IPv4 address allocation (originally)

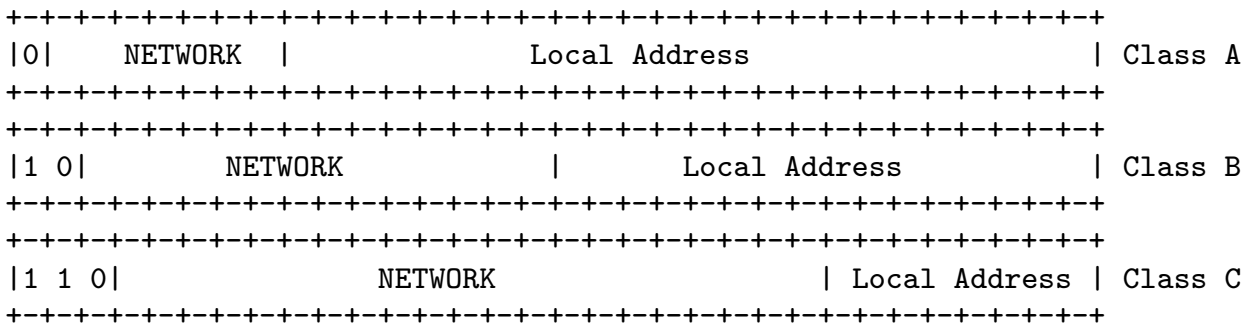


- The address is split into two parts:
  - Network part
  - Host part
- Initially the boundary was given by a prefix
  - 3 boundaries called classes
  - 1 class (D) for mutlicast added later
  - 1 class (E) reserved (never used)
- An authority used to give unique prefix to sites
- This plan was developed to guarantee address uniqueness

Facts on Addresses

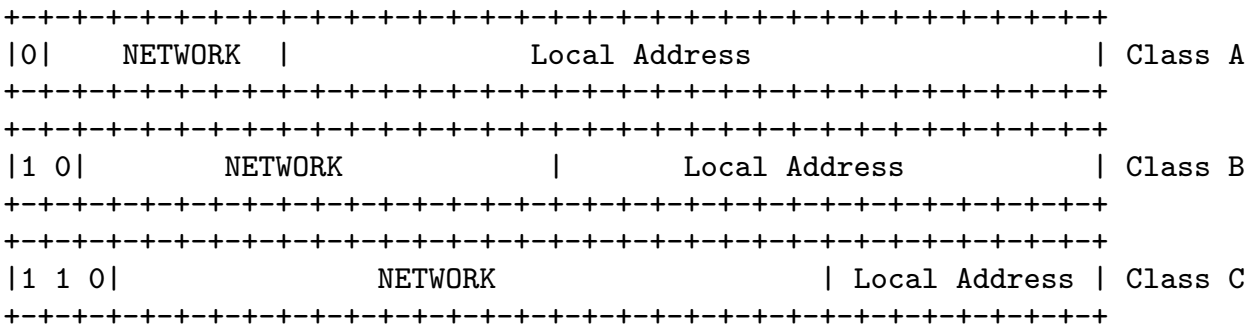
Historical view

IPv4 address allocation (originally)



- The address is split into two parts:
  - Network part
  - Host part
- Initially the boundary was given by a prefix
  - 3 boundaries called classes
  - 1 class (D) for mutlicast added later
  - 1 class (E) reserved (never used)
- An authority used to give unique prefix to sites
- This plan was developed to guarantee address uniqueness

IPv4 address allocation (originally)



- The address is split into two parts:
  - Network part
  - Host part
- Initially the boundary was given by a prefix
  - 3 boundaries called classes
  - 1 class (D) for mutlicast added later
  - 1 class (E) reserved (never used)
- An authority used to give unique prefix to sites
- This plan was developed to guarantee address uniqueness

Historical facts

- 1983 : Research network for about 100 computers
- 1992 : Commercial activity
  - Exponential growth
- 1993 : Exhaustion of the class B address space
  - Allocation in the class C space
  - Require more information in routers memory
- Forecast of network collapse for 1998!
  - 1999 : Bob Metcalfe ate his Infoworld 1995 paper where he made this prediction

Facts on Addresses

Emergency Measures

Emergency Measures: Better Addresses Management

RFC 1517 - RFC 1520 (Sept 1993)

- Ask the internet community to give back allocated prefixes (RFC 1917)
- Re-use class C address space
- CIDR (Classless Internet Domain Routing)
  - network address = prefix/prefix length
  - less address waste
  - recommend aggregation (reduce routing table length)
- Introduce private prefixes (RFC 1918)

Facts on Addresses

NAT

Emergency Measures: Private Addresses (RFC 1918 BCP)

- Allow private addressing plans
- Addresses are used internally
- Similar to security architecture with firewalls
- Use of proxies or NAT to go outside
  - RFC 1631, RFC 2663 and RFC 2993
- NAPT is the most commonly used of NAT variations

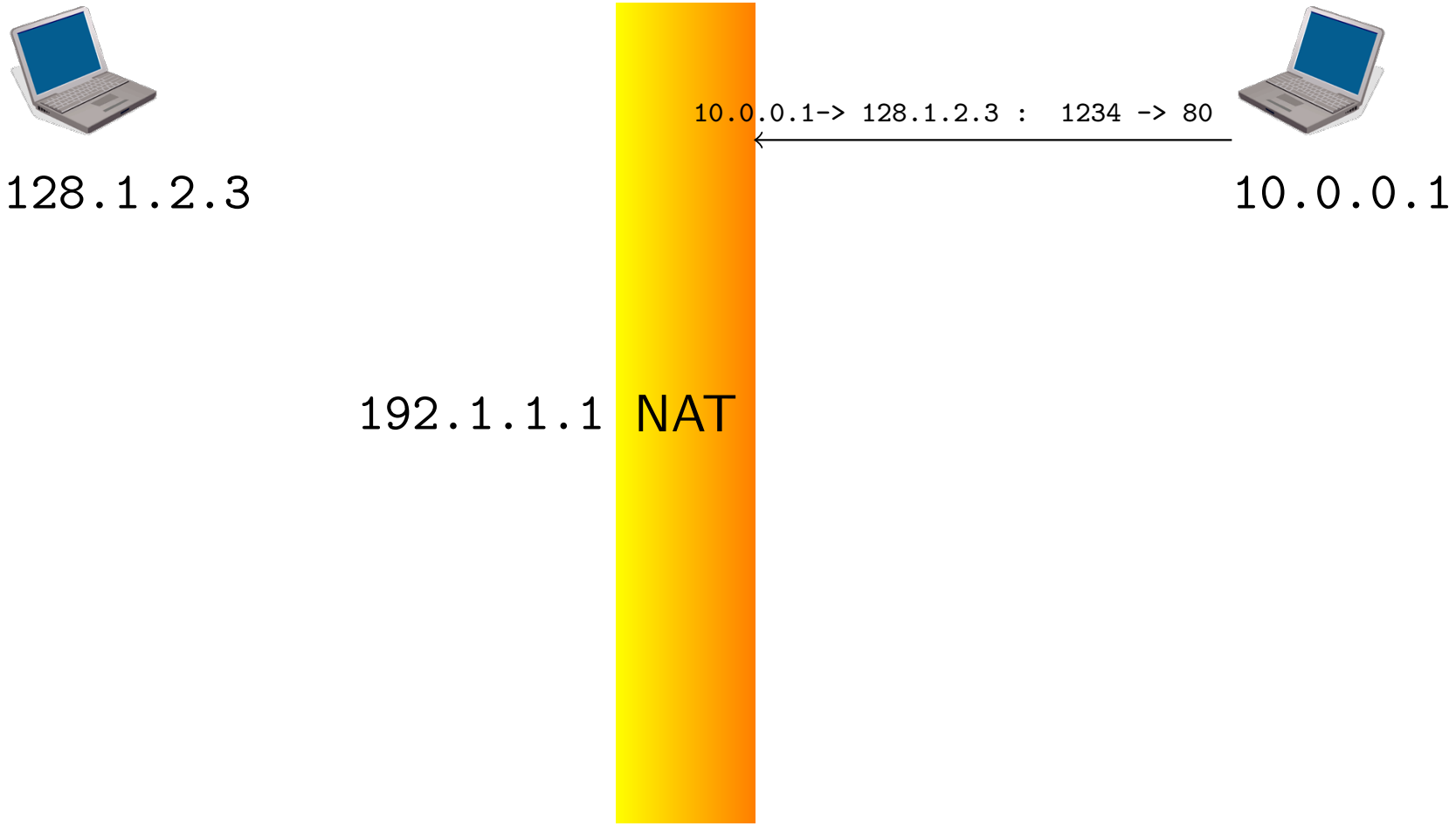
How NAT with Port Translation Works

128.1.2.3

10.0.0.1

192.1.1.1 NAT

How NAT with Port Translation Works



How NAT with Port Translation Works

128.1.2.3

10.0.0.1-> 128.1.2.3 : 1234 -> 80

10.0.0.1

192.1.1.1 NAT

7890 : 10.0.0.1 & 1234

How NAT with Port Translation Works

10.0.0.1-> 128.1.2.3 : 1234 -> 80

128.1.2.3

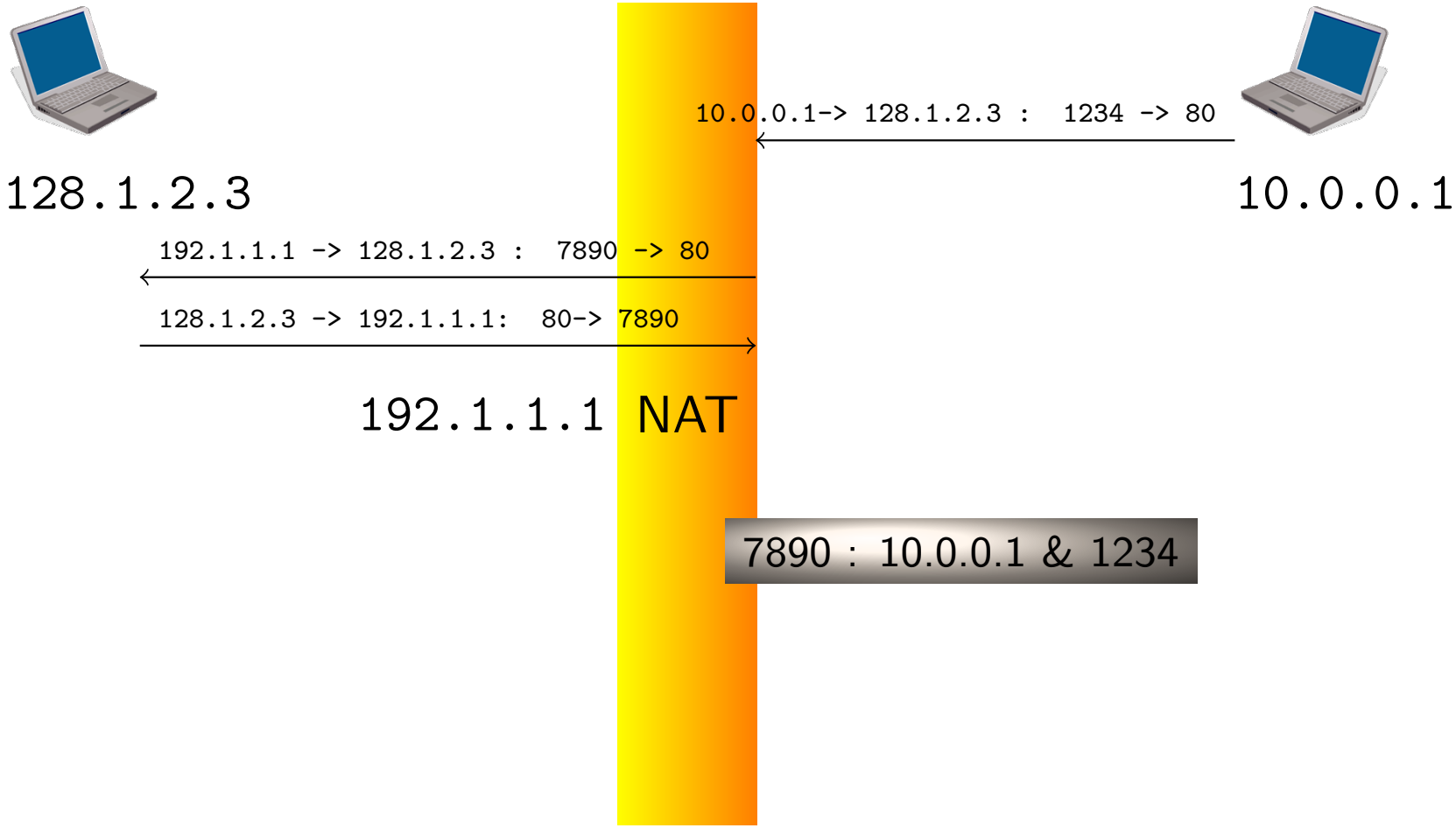
10.0.0.1

192.1.1.1 -> 128.1.2.3 : 7890 -> 80

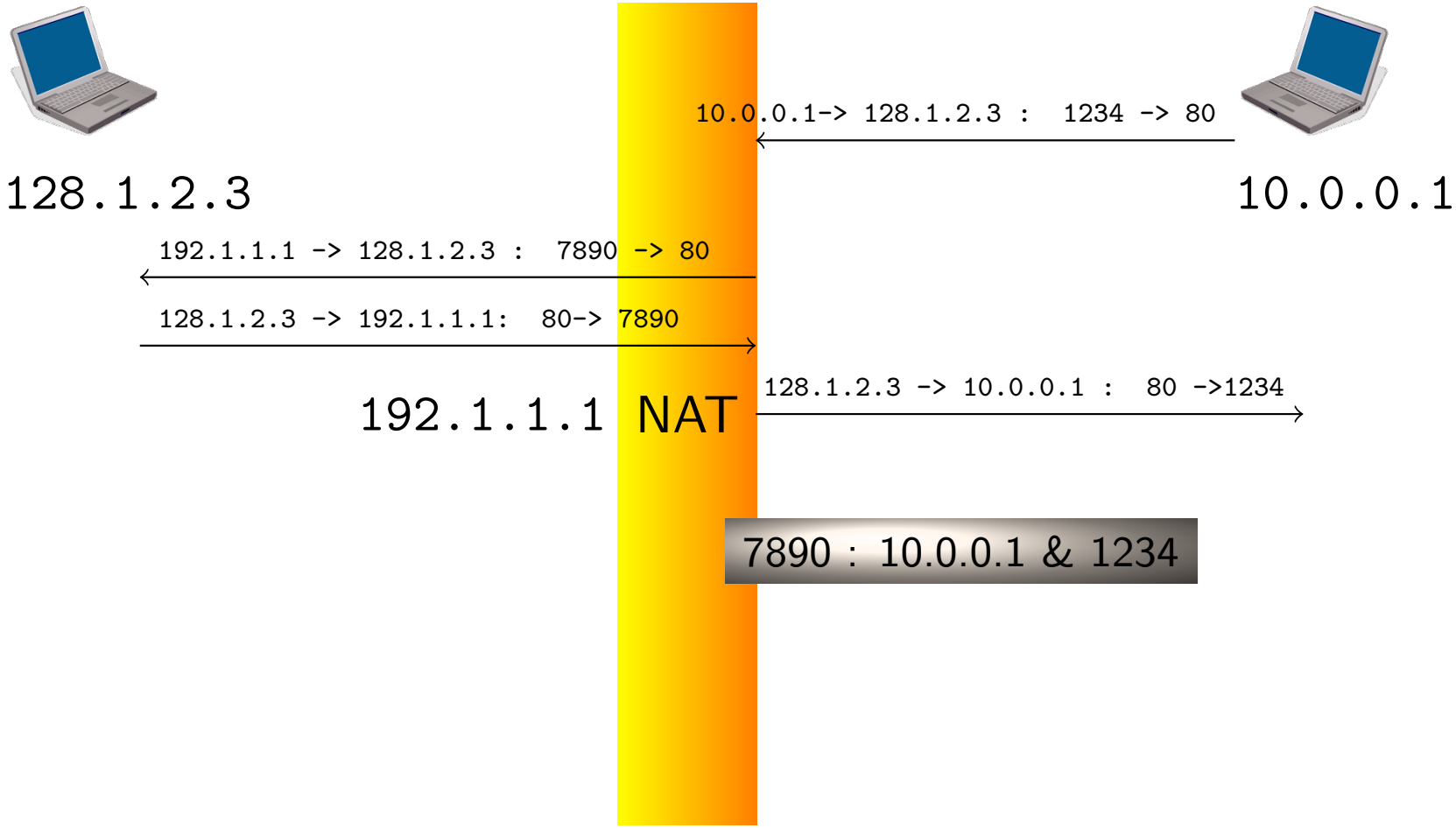
192.1.1.1 NAT

7890 : 10.0.0.1 & 1234

How NAT with Port Translation Works



How NAT with Port Translation Works



NAT Impact

first consequence

The application does not know its public name.

second consequence

It is difficult to contact a NATed equipment from outside

- Security feeling
- Solutions for NAT traversal exist

third consequence

There is no standardized behavior for NAT yet

Facts on Addresses

Prefixes delegation

What Has Changed

Classful Addressing

1. Ensure uniqueness
2. Facilitate administrative allocation
   - One central entity

Class-Less (CIDR)

1. Facilitate administrative allocation (hierarchical)
   - Nowadays 5 regional entities
2. Facilitate host location in the network
3. Allocate the minimum pool of addresses

What Has Changed

Classful Addressing

1. Ensure uniqueness
2. Facilitate administrative allocation
   - One central entity

Class-Less (CIDR)

1. Facilitate administrative allocation (hierarchical)
   - Nowadays 5 regional entities
2. Facilitate host location in the network
3. Allocate the minimum pool of addresses

What Has Changed

Classful Addressing

1. Ensure uniqueness
2. Facilitate administrative allocation
   - One central entity

Class-Less (CIDR)

1. Facilitate administrative allocation (hierarchical)
   - Nowadays 5 regional entities
2. Facilitate host location in the network
3. Allocate the minimum pool of addresses

What Has Changed

Classful Addressing

1. Ensure uniqueness
2. Facilitate administrative allocation
   - One central entity

Class-Less (CIDR)

1. Facilitate administrative allocation (hierarchical)
   - Nowadays 5 regional entities
2. Facilitate host location in the network
3. Allocate the minimum pool of addresses

What Has Changed

Classful Addressing

1. Ensure uniqueness
2. Facilitate administrative allocation
   - One central entity

Class-Less (CIDR)

1. Facilitate administrative allocation (hierarchical)
   - Nowadays 5 regional entities
2. Facilitate host location in the network
3. Allocate the minimum pool of addresses

CIDR Administrative Point of View

- A hierarchy of administrative registries
  - IANA/ICANN at the top
- 5 Regional Internet Registries (RIR)
  - APNIC (Asia Pacific Network Information Centre)
  - ARIN (American Registry for Internet Numbers)
  - LACNIC (Regional Latin-American and Caribbean IP Address Registry)
  - RIPE NCC (Réseaux IP Européens - Network Coordination Center)
    - Europe, Middle east.
  - AfriNIC (Africa)
- Providers get prefixes allocation from RIR

RIR Regions

Prefix

- CIDR can be viewed as an extension of the netmask concept
- It is called classless since IP addresses are no longer interpreted as belonging to a given Class (A, B, C) based on the value of the 1-4 leading bits
- The prefix length must be added to the 32 bit word to indicate what is the network part.
  - Lookup complexity in the FIB (Forwarding Information Base) is increased:
  - Best prefix match rule

Exhaustion of IPv4 Prefix Pool

- IANA Unallocated Address Pool Depleted: February, 1st 2011
  - See: http://www.nro.net/news/ipv4-free-pool-depleted
- RIR Unallocated Address Pool Exhaustion
  - APNIC (Asia) : April 2011
  - RIPE-NCC (Europe) : September 2012
  - LACNIC (South America) : June 2014
  - ARIN (North Amercia) : September 2015 (completely exhausted!)
    - See: http://www.potaroo.net/tools/ipv4/
    - See als: http://www.ipv4depletion.com/

Genesis of a new version of IP

Preliminary works between 1991 and 1994

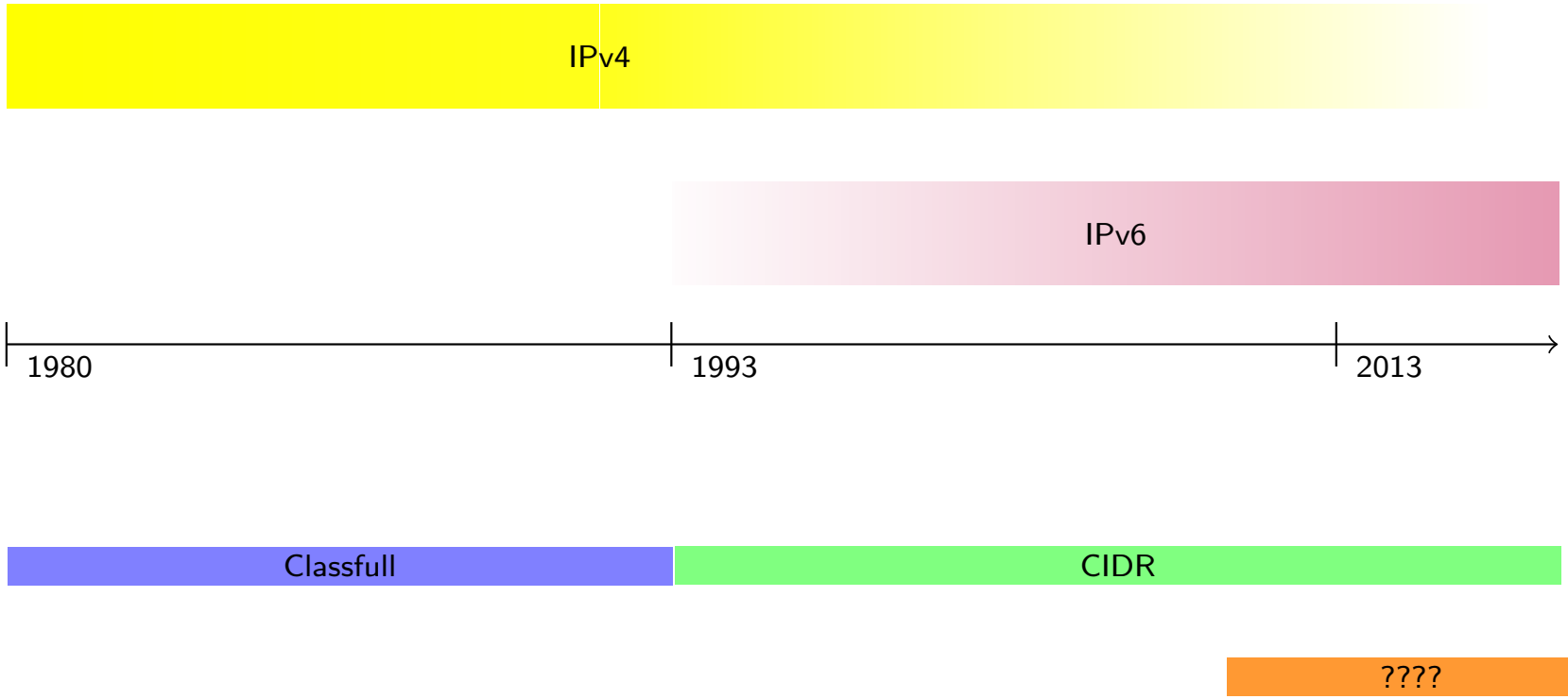
- In 1991 IAB proposed an ISO-like solution (CNLP), refused by IETF
- An IPng area is created, initiated a call for tender
- Between 1992 and 1994, several propositions emerged

During IETF'30 (Toronto, July 1994), the SIPP+ solution is adopted

- Keep the fundamentals of IPv4
- Larger address space (16-byte addresses)
- Simpler header

IPv6 is formalized in RFC 1883 in december 1995 (updated with RFC 2460). First deployments followed (6bone, G6).

Addresses versus Packet Format



IPv6 Benefits

- Larger address space from 2^{32} to 2^{128}
  - Allow different addressing scheme
- Stateless auto-configuration of hosts
  - Layer 3 "Plug & Play" Protocol
- Simple header $\Rightarrow$ Efficient routing
  - No checksum
  - No fragmentation by routers
  - Enhanced extension system

IPv6 Benefits

- Larger address space from 2^{32} to 2^{128}
  - Allow different addressing scheme
- Stateless auto-configuration of hosts
  - Layer 3 "Plug & Play" Protocol
- Simple header $\Rightarrow$ Efficient routing
  - No checksum
  - No fragmentation by routers
  - Enhanced extension system
- end to end, but...
-
-
-

IPv6 Benefits

- Larger address space from 2^{32} to 2^{128}
  - Allow different addressing scheme
- Stateless auto-configuration of hosts
  - Layer 3 "Plug & Play" Protocol
- Simple header $\Rightarrow$ Efficient routing
  - No checksum
  - No fragmentation by routers
  - Enhanced extension system
- end to end, but...
- ~~Quality of service~~
-
-

IPv6 Benefits

- Larger address space from 2^{32} to 2^{128}
  - Allow different addressing scheme
- Stateless auto-configuration of hosts
  - Layer 3 "Plug & Play" Protocol
- Simple header $\Rightarrow$ Efficient routing
  - No checksum
  - No fragmentation by routers
  - Enhanced extension system
- end to end, but...
- ~~Quality of service~~
- ~~Better support of mobility~~
-

IPv6 Benefits

- Larger address space from 2^{32} to 2^{128}
  - Allow different addressing scheme
- Stateless auto-configuration of hosts
  - Layer 3 "Plug & Play" Protocol
- Simple header $\Rightarrow$ Efficient routing
  - No checksum
  - No fragmentation by routers
  - Enhanced extension system
- end to end, but...
- ~~Quality of service~~
- ~~Better support of mobility~~
- ~~IPsec~~

Addresses

Notation

IPv6 addresses


```
F2C:544:9E::2:EF8D:6B7 F692::  A:1455::A:6E0 D:63:D::4:3A:55F B33:C::F2 7:5059:3D:C0::

9D::9BAC:B8CA:893F:80 1E:DE2:4C83::4E:39:F35:C875 2::  A:FDE3:76:B4F:D9D::  D6::

369F:9:F8:DBF::2 DD4:B45:1:C42F:BE6:75::  9D7B:7184:EF::3FB:BF1A:D80 FE9::B:3

EC:DB4:B:F:F11::E9:090 83:B9:08:B5:F:3F:AF:B84 E::35B:8572:7A3:FB2 99:F:9:8B76::BC9

D64:07:F394::BDB:DF40:08EE:A79E AC:23:5D:78::233:84:8 F0D:F::F4EB:0F:5C7

E71:F577:ED:E:9DE8::  B::3 1D3F:A0AA::  70:8EA1::8:D5:81:2:F302 26::8880:7 93::  F::9:0

E:2:0:266B::  763E:C:2E:1EB:F6:F4:14:16 E6:6:F4:B6:A888:979E:D78:09

9:754:5:90:0A78:A1A3:1:7 2:8::  97B:C4::C36 A40:7:5:7E8F:0:32EC:9A:D0 8A52::575

D::4CB4:E:2BF:5485:8CE 07:5::41 6B::A9:C 94FF:7B8::D9:51:26F 2::E:AE:ED:81 8241::  5F97::

AD5B:259C:7DB8:24:58:552A::  94:4:9FD:4:87E5::  5A8:2FF:1::CC EA:8904:7C::

7C::D6B7:A7:B0:8B DC:6C::34:89 6C:1::5 7B3:6780:4:B1::E586 412:2:5E1:6DE5:5E3A:553:3::

7F0::  B39::1:B77:DB 9D3:1F1:4B:3:B4E6:7681:09:D4A8 61:520::E0 1:28E9:0:095:DF:F2::

1B61:4::1DE:50A 34BC:99::E9:9EFB E:EF::  BDC:672A:F4C8:A1::4:7:9CB7 C697:56AD:40:8:0::62
```

Don't Worry

Addresses are not random numbers. . . they are often easy to handle and even to memorize sometimes

Notation

- Base format (a 16-octet Global IPv6 Address):
  - `2001:0db8:beef:0001:0000:0000:cafe:deca`
- Compact Format:

2001:0db8:beef:0001:0000:0000:cafe:deca

1. Remove 0 on the left of each word
2. To avoid ambiguity, substitute ONLY one sequence of zeros by `::`

- an IPv4 address may also appear : `::ffff:192.0.2.1`

Warning:

`2001:db8:3::/40` is in fact `2001:db8:0003::/40` and not `2001:db8:0300::/40`

Notation

- Base format (a 16-octet Global IPv6 Address):
  - `2001:0db8:beef:0001:0000:0000:cafe:deca`
- Compact Format:

2001:0db8:beef:0001:0000:0000:cafe:deca

1. Remove 0 on the left of each word
2. To avoid ambiguity, substitute ONLY one sequence of zeros by `::`
   - an IPv4 address may also appear : `::ffff:192.0.2.1`

Warning:

`2001:db8:3::/40` is in fact `2001:db8:0003::/40` and not `2001:db8:0300::/40`

Notation

- Base format (a 16-octet Global IPv6 Address):
  - `2001:0db8:beef:0001:0000:0000:cafe:deca`
- Compact Format:

2001:db8:beef:1:0:0:cafe:deca

1. Remove 0 on the left of each word
2. To avoid ambiguity, substitute ONLY one sequence of zeros by `::`

- an IPv4 address may also appear : `::ffff:192.0.2.1`

Warning:

`2001:db8:3::/40` is in fact `2001:db8:0003::/40` and not `2001:db8:0300::/40`

Notation

- Base format (a 16-octet Global IPv6 Address):
  - `2001:0db8:beef:0001:0000:0000:cafe:deca`
- Compact Format:

2001:db8:beef:1:0:0:cafe:deca

1. Remove 0 on the left of each word
2. To avoid ambiguity, substitute ONLY one sequence of zeros by `::`

- an IPv4 address may also appear : `::ffff:192.0.2.1`

Warning:

`2001:db8:3::/40` is in fact `2001:db8:0003::/40` and not `2001:db8:0300::/40`

Notation

- Base format (a 16-octet Global IPv6 Address):
  - `2001:0db8:beef:0001:0000:0000:cafe:deca`
- Compact Format:

2001:db8:beef:1::cafe:deca

1. Remove 0 on the left of each word
2. To avoid ambiguity, substitute ONLY one sequence of zeros by `::`

- an IPv4 address may also appear : `::ffff:192.0.2.1`

Warning:

`2001:db8:3::/40` is in fact `2001:db8:0003::/40` and not `2001:db8:0300::/40`

Notation

- Base format (a 16-octet Global IPv6 Address):
  - `2001:0db8:beef:0001:0000:0000:cafe:deca`
- Compact Format:

2001:db8:beef:1::cafe:deca

1. Remove 0 on the left of each word
2. To avoid ambiguity, substitute ONLY one sequence of zeros by `::`

- an IPv4 address may also appear : `::ffff:192.0.2.1`

Warning:

`2001:db8:3::/40` is in fact `2001:db8:0003::/40` and not `2001:db8:0300::/40`

Is it enough for the future ?

- Address length
  - About $3.4x10^{38}$ addresses
  - 60 000 trillion trillion addresses per inhabitant on earth
  - Addresses for every grain of sands in the world
  - IPv4: 6 addresses per US inhabitant, 1 in Europe, 0.01 in China and 0.001 in India
- Justification of a fixed-length address

Warning:

- An address for everything on the network and not an address for everything
- No addresses for the whole life:
  - Depends on your position on the network
  - ISP Renumbering may be possible

Is it enough for the future ?

- Hop Limit:
  - Should not be a problem
  - Count the number of routers used to reach a destination
  - Growth will be in-width more than in-depth
- Payload Length
  - 64 Ko is not a current hard limit
  - Ethernet is limited to 1.5 Ko, evolution can use until 9Ko.
  - Use Jumbogram for specific cases

Addresses

Addressing scheme

Addressing scheme

- RFC 4291 defines current IPv6 addresses
  - loopback (`::1`)
  - `link local (fe80::/10)`
  - `global unicast (2000::/3)`
  - `multicast (ff00::/8)`
- Use CIDR principles:
  - Prefix / prefix length notation
  - `2001:db8:face::/48`
  - `2001:db8:face:bed:cafe:deca:dead:beef/64`
- Interfaces have several IPv6 addresses
  - at least a link-local, and one or more global unicast addresses

Addressing Space Utilization


```
0000::/8 Reserved by IETF [RFC4291]
0100::/8 Reserved by IETF [RFC4291]
0200::/7 Reserved by IETF [RFC4048]
0400::/6 Reserved by IETF [RFC4291]
0800::/5 Reserved by IETF [RFC4291]
1000::/4 Reserved by IETF [RFC4291]
2000::/3 Global Unicast [RFC4291]
4000::/3 Reserved by IETF [RFC4291]
6000::/3 Reserved by IETF [RFC4291]
8000::/3 Reserved by IETF [RFC4291]
a000::/3 Reserved by IETF [RFC4291]
c000::/3 Reserved by IETF [RFC4291]
e000::/4 Reserved by IETF [RFC4291]
f000::/5 Reserved by IETF [RFC4291]
F800::/6 Reserved by IETF [RFC4291]
fc00::/7 Unique Local Unicast [RFC4193]
fe00::/9 Reserved by IETF [RFC4291]
fe80::/10 Link Local Unicast [RFC4291]
fec0::/10 Reserved by IETF [RFC3879]
ff00::/8 Multicast [RFC4291]
```

http://www.iana.org/assignments/ipv6-address-space

Addressing Space Utilization


```
0000::/8 Reserved by IETF [RFC4291]
0100::/8 Reserved by IETF [RFC4291]
0200::/7 Reserved by IETF [RFC4048]
0400::/6 Reserved by IETF [RFC4291]
0800::/5 Reserved by IETF [RFC4291]
1000::/4 Reserved by IETF [RFC4291]
2000::/3 Global Unicast [RFC4291]
4000::/3 Reserved by IETF [RFC4291]
6000::/3 Reserved by IETF [RFC4291]
8000::/3 Reserved by IETF [RFC4291]
a000::/3 Reserved by IETF [RFC4291]
c000::/3 Reserved by IETF [RFC4291]
e000::/4 Reserved by IETF [RFC4291]
f000::/5 Reserved by IETF [RFC4291]
F800::/6 Reserved by IETF [RFC4291]
fc00::/7 Unique Local Unicast [RFC4193]
fe00::/9 Reserved by IETF [RFC4291]
fe80::/10 Link Local Unicast [RFC4291]
fec0::/10 Reserved by IETF [RFC3879]
ff00::/8 Multicast [RFC4291]
```

http://www.iana.org/assignments/ipv6-address-space

Addressing Space Utilization


```
0000::/8 Reserved by IETF [RFC4291]
0100::/8 Reserved by IETF [RFC4291]
0200::/7 Reserved by IETF [RFC4048]
0400::/6 Reserved by IETF [RFC4291]
0800::/5 Reserved by IETF [RFC4291]
1000::/4 Reserved by IETF [RFC4291]
2000::/3 Global Unicast [RFC4291]
4000::/3 Reserved by IETF [RFC4291]
6000::/3 Reserved by IETF [RFC4291]
8000::/3 Reserved by IETF [RFC4291]
a000::/3 Reserved by IETF [RFC4291]
c000::/3 Reserved by IETF [RFC4291]
e000::/4 Reserved by IETF [RFC4291]
f000::/5 Reserved by IETF [RFC4291]
F800::/6 Reserved by IETF [RFC4291]
fc00::/7 Unique Local Unicast [RFC4193]
fe00::/9 Reserved by IETF [RFC4291]
fe80::/10 Link Local Unicast [RFC4291]
fec0::/10 Reserved by IETF [RFC3879]
ff00::/8 Multicast [RFC4291]
```

http://www.iana.org/assignments/ipv6-address-space

Addressing Space Utilization


```
0000::/8 Reserved by IETF [RFC4291]
0100::/8 Reserved by IETF [RFC4291]
0200::/7 Reserved by IETF [RFC4048]
0400::/6 Reserved by IETF [RFC4291]
0800::/5 Reserved by IETF [RFC4291]
1000::/4 Reserved by IETF [RFC4291]
2000::/3 Global Unicast [RFC4291]
4000::/3 Reserved by IETF [RFC4291]
6000::/3 Reserved by IETF [RFC4291]
8000::/3 Reserved by IETF [RFC4291]
a000::/3 Reserved by IETF [RFC4291]
c000::/3 Reserved by IETF [RFC4291]
e000::/4 Reserved by IETF [RFC4291]
f000::/5 Reserved by IETF [RFC4291]
F800::/6 Reserved by IETF [RFC4291]
fc00::/7 Unique Local Unicast [RFC4193]
fe00::/9 Reserved by IETF [RFC4291]
fe80::/10 Link Local Unicast [RFC4291]
fec0::/10 Reserved by IETF [RFC3879]
ff00::/8 Multicast [RFC4291]
```

http://www.iana.org/assignments/ipv6-address-space

Addressing Space Utilization


```
0000::/8 Reserved by IETF [RFC4291]
0100::/8 Reserved by IETF [RFC4291]
0200::/7 Reserved by IETF [RFC4048]
0400::/6 Reserved by IETF [RFC4291]
0800::/5 Reserved by IETF [RFC4291]
1000::/4 Reserved by IETF [RFC4291]
2000::/3 Global Unicast [RFC4291]
4000::/3 Reserved by IETF [RFC4291]
6000::/3 Reserved by IETF [RFC4291]
8000::/3 Reserved by IETF [RFC4291]
a000::/3 Reserved by IETF [RFC4291]
c000::/3 Reserved by IETF [RFC4291]
e000::/4 Reserved by IETF [RFC4291]
f000::/5 Reserved by IETF [RFC4291]
F800::/6 Reserved by IETF [RFC4291]
fc00::/7 Unique Local Unicast [RFC4193]
fe00::/9 Reserved by IETF [RFC4291]
fe80::/10 Link Local Unicast [RFC4291]
fec0::/10 Reserved by IETF [RFC3879]
ff00::/8 Multicast [RFC4291]
```

http://www.iana.org/assignments/ipv6-address-space

Addressing Space Utilization


```
0000::/8 Reserved by IETF [RFC4291]
0100::/8 Reserved by IETF [RFC4291]
0200::/7 Reserved by IETF [RFC4048]
0400::/6 Reserved by IETF [RFC4291]
0800::/5 Reserved by IETF [RFC4291]
1000::/4 Reserved by IETF [RFC4291]
2000::/3 Global Unicast [RFC4291]
4000::/3 Reserved by IETF [RFC4291]
6000::/3 Reserved by IETF [RFC4291]
8000::/3 Reserved by IETF [RFC4291]
a000::/3 Reserved by IETF [RFC4291]
c000::/3 Reserved by IETF [RFC4291]
e000::/4 Reserved by IETF [RFC4291]
f000::/5 Reserved by IETF [RFC4291]
F800::/6 Reserved by IETF [RFC4291]
fc00::/7 Unique Local Unicast [RFC4193]
fe00::/9 Reserved by IETF [RFC4291]
fe80::/10 Link Local Unicast [RFC4291]
fec0::/10 Reserved by IETF [RFC3879]
ff00::/8 Multicast [RFC4291]
```

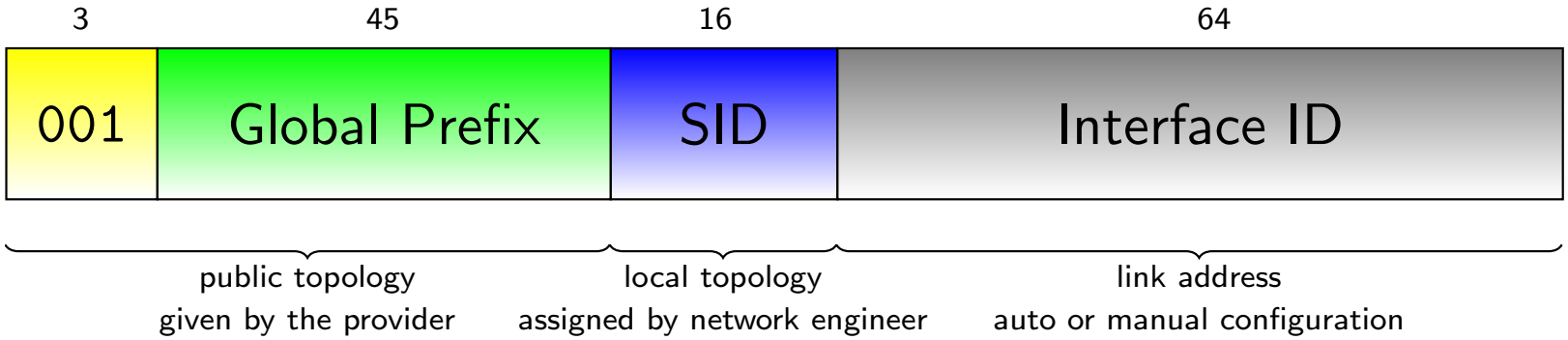
http://www.iana.org/assignments/ipv6-address-space

Addresses

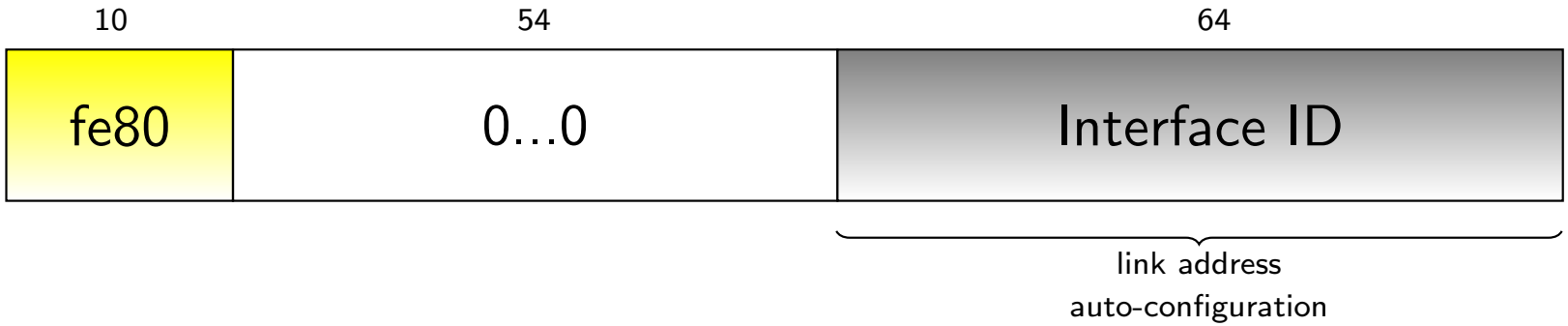
Address Format

Address Format

Global Unicast Address:



Link-Local Address:



Global Unicast Addresses

Used for communication between hosts of the IPv6 Internet (≈ public IPv4 addresses)
Composed by 2 parts

- a 64-bit **Global Prefix**, identifying the network of the host
- a 64-bit **Interface ID**, identifying the host in the network

The **Global Prefix** is defined by network topology.
The **Interface ID** can be selected by the host itself.
Note: **The 64-bit border is hard-coded !**

SID Values

- 16-bit length up to 65 535 subnets
  - Large enough for most companies
  - Too large for home network ?
  - May be a /56 or /60 GP will be allocated depending on the ISP
- There is no strict rules to structure SID:
  - sequencial : 1, 2, ...
  - use VLAN number
  - include usage to allow filtering, for instance, for a University:

Interface Identifier

Interface ID can be selected differently

- Derived from a Layer 2 ID (I.e. MAC address) :
  - for Link Local address
  - for Global Address : plug-and-play hosts
- Assigned manually :
  - to keep same address when Ethernet card or host is changed
  - to remember easily the address
    - 1, 2, 3, ...
    - last digit of the v4 address
    - the IPv4 address (for nostalgic system administrators)
    - ...

Interface Identifier

Interface ID can be selected differently

- Derived from a Layer 2 ID (I.e. MAC address) :
  - for Link Local address
  - for Global Address : plug-and-play hosts
- Assigned manually :
  - to keep same address when Ethernet card or host is changed
  - to remember easily the address
    - 1, 2, 3, ...
    - last digit of the v4 address
    - the IPv4 address (for nostalgic system administrators)
    - ...

Interface Identifier

Interface ID can be selected differently

- Random value :
  - Changed frequently (e.g, every day, per session, at each reboot...) to guarantee anonymity
- Hash of other values (experimental) :
  - To link address to other properties
  - Public key
  - List of assigned prefixes
  - ...

Interface Identifier

Interface ID can be selected differently

- Random value :
  - Changed frequently (e.g, every day, per session, at each reboot...) to guarantee anonymity
- Hash of other values (experimental) :
  - To link address to other properties
  - Public key
  - List of assigned prefixes
  - ...

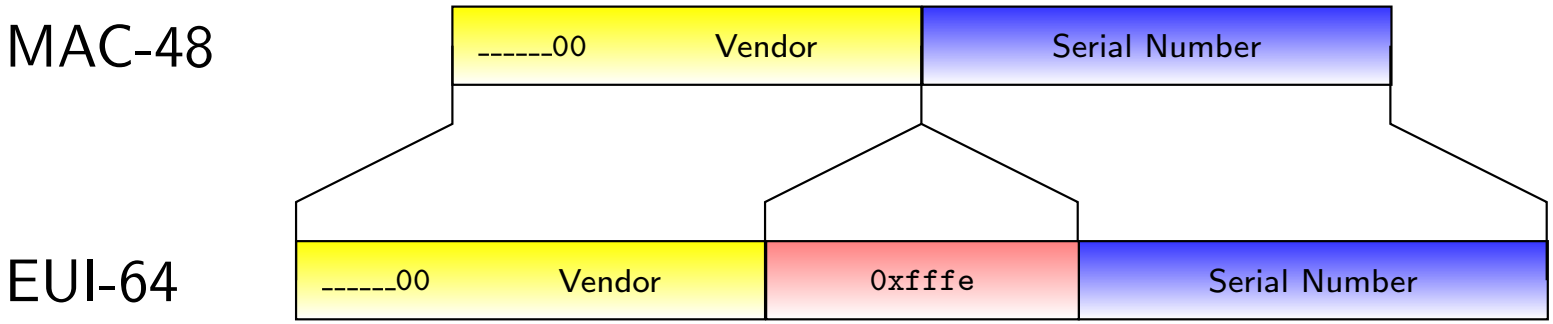
How to Construct an IID from MAC Address

- 64 bits is compatible with EUI-64 (i.e. IEEE 1394 FireWire, ...)
- IEEE propose a way to transform a MAC-48 to an EUI-64
- U/L changed for numbering purpose

- There is no conflicts if IID are manually numbered: 1, 2, 3, ...

How to Construct an IID from MAC Address

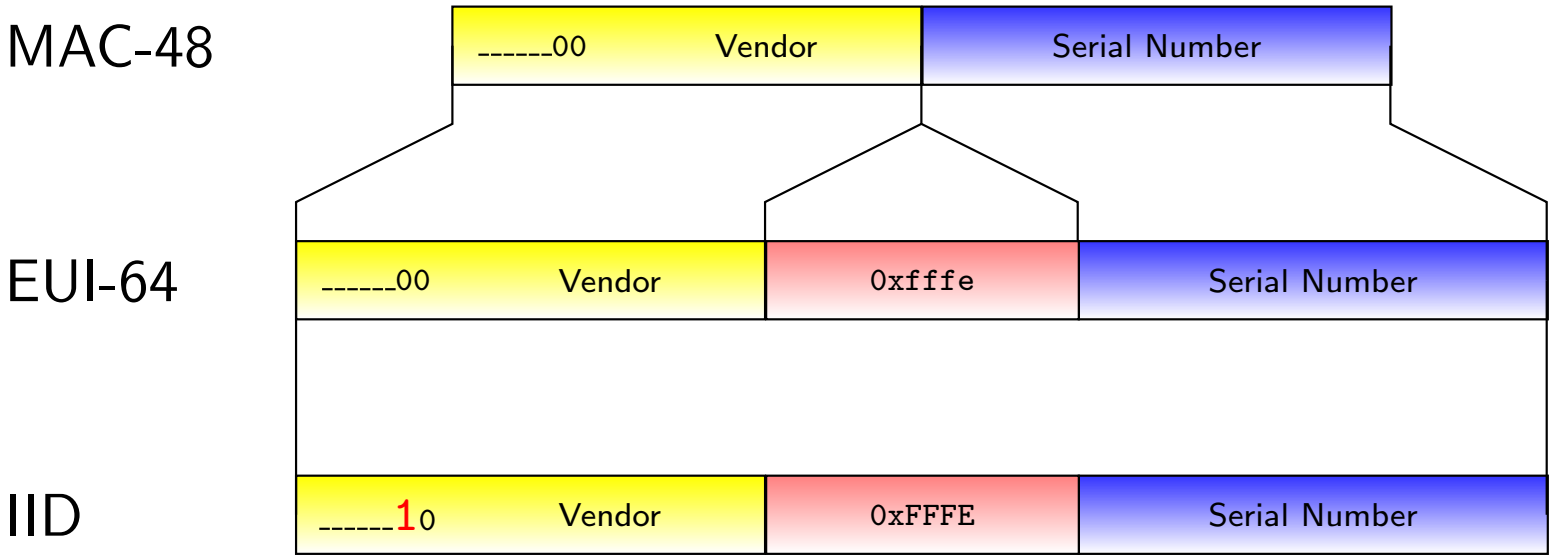
- 64 bits is compatible with EUI-64 (i.e. IEEE 1394 FireWire, ...)
- IEEE propose a way to transform a MAC-48 to an EUI-64
- U/L changed for numbering purpose



- There is no conflicts if IID are manually numbered: 1, 2, 3, ...

How to Construct an IID from MAC Address

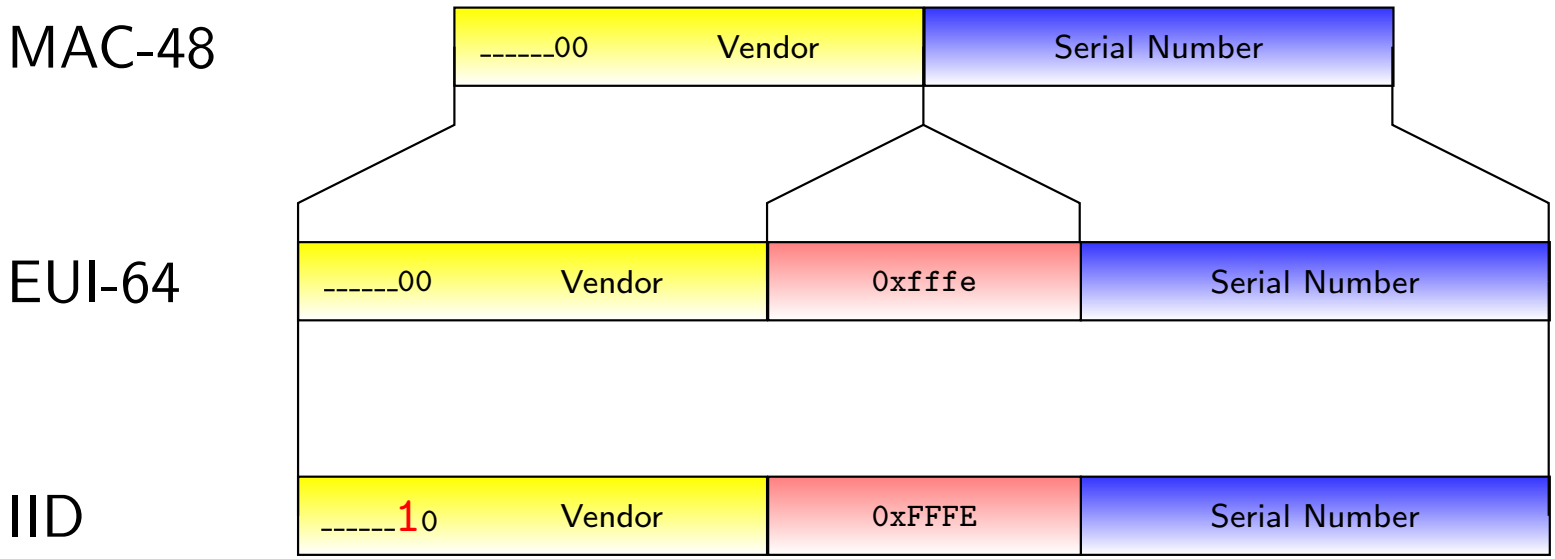
- 64 bits is compatible with EUI-64 (i.e. IEEE 1394 FireWire, ...)
- IEEE propose a way to transform a MAC-48 to an EUI-64
- U/L changed for numbering purpose



- There is no conflicts if IID are manually numbered: 1, 2, 3, ...

How to Construct an IID from MAC Address

- 64 bits is compatible with EUI-64 (i.e. IEEE 1394 FireWire, ...)
- IEEE propose a way to transform a MAC-48 to an EUI-64
- U/L changed for numbering purpose

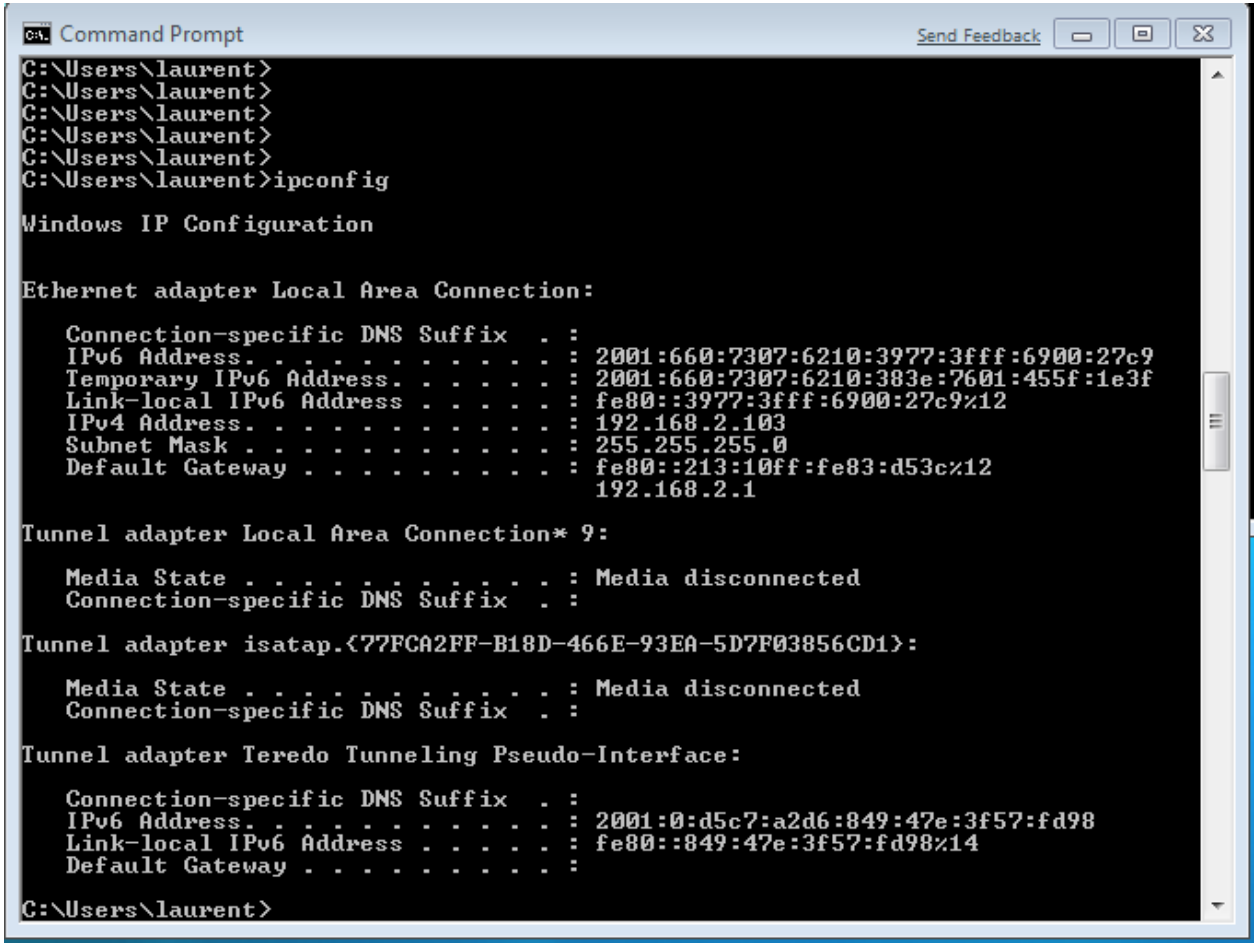


- There is no conflicts if IID are manually numbered: 1, 2, 3, ...

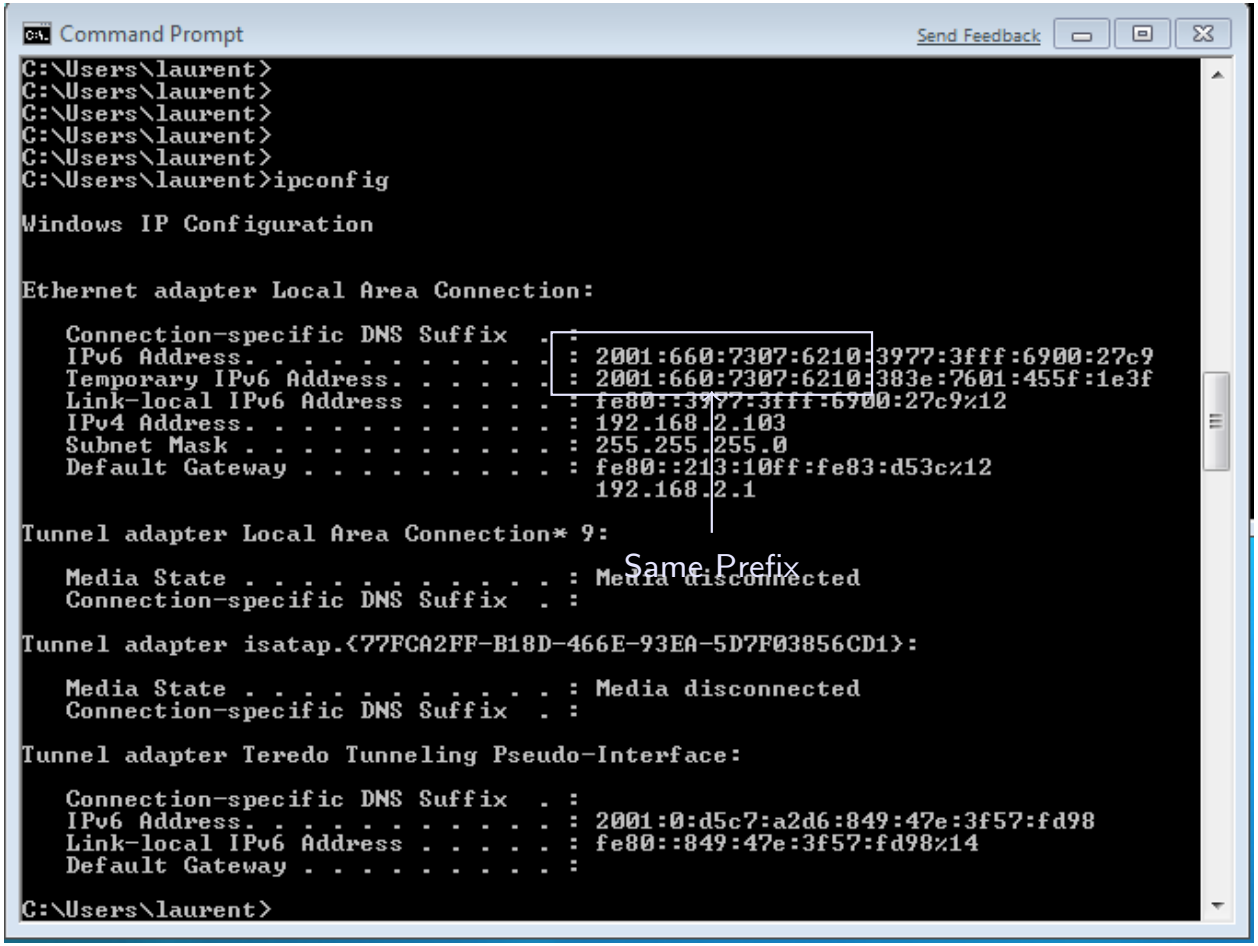
Example : Mac / Unix


```
%ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
        inet 127.0.0.1 netmask 0xff000000
en1: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500

        inet6 fe80::216:cbff:febe:16b3%en1 prefixlen 64 scopeid 0x5

        inet 192.168.2.5 netmask 0xffffff00 broadcast 192.168.2.255
        inet6 2001:660:7307:6031:216:cbff:febe:16b3 prefixlen 64
        autoconf

        ether 00:16:cb:be:16:b3
        media: autoselect status: active
        supported media: autoselect
```

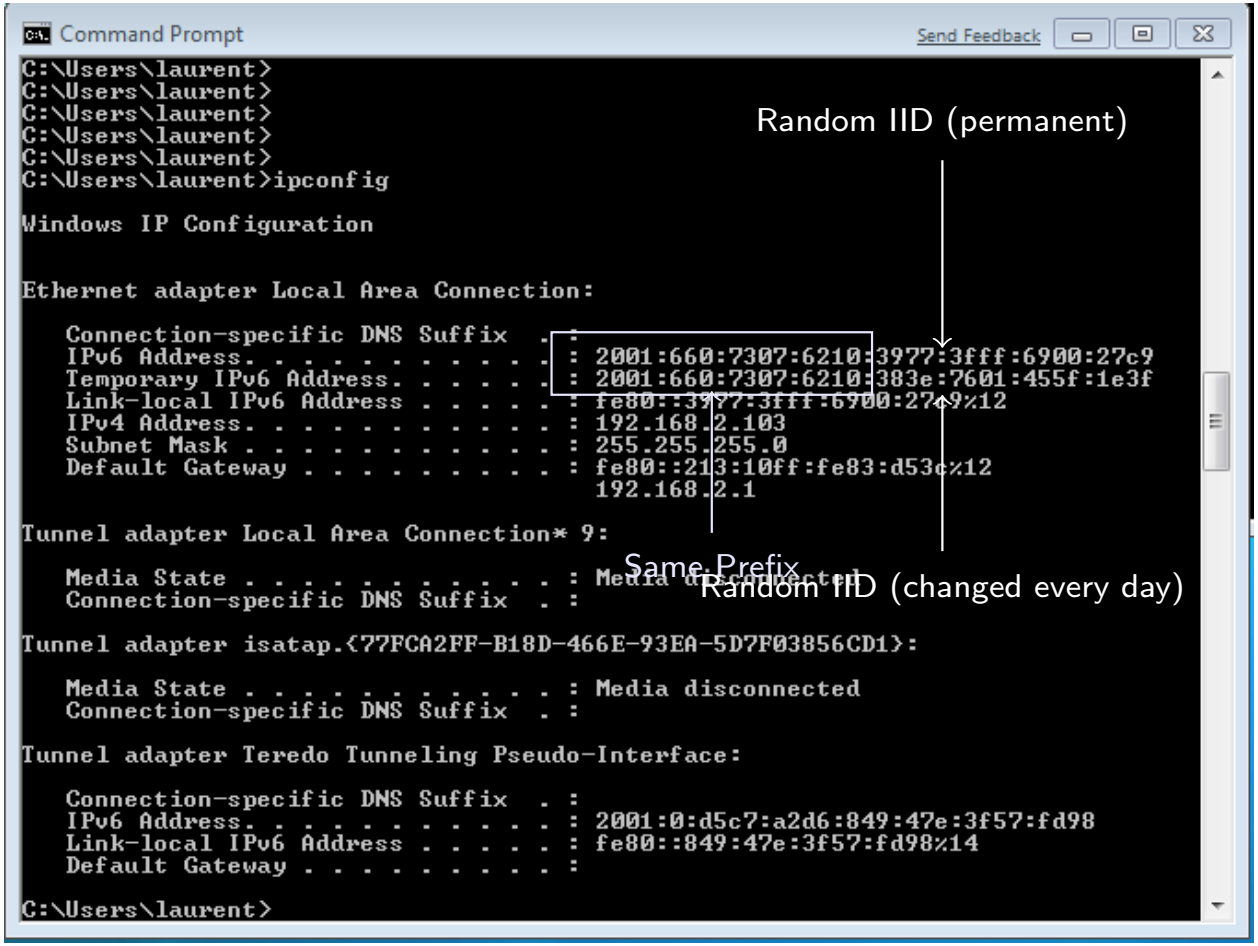
Windows 7



Windows 7



Windows 7



Addresses

Kind of addresses

Link-Local Scoped Addresses

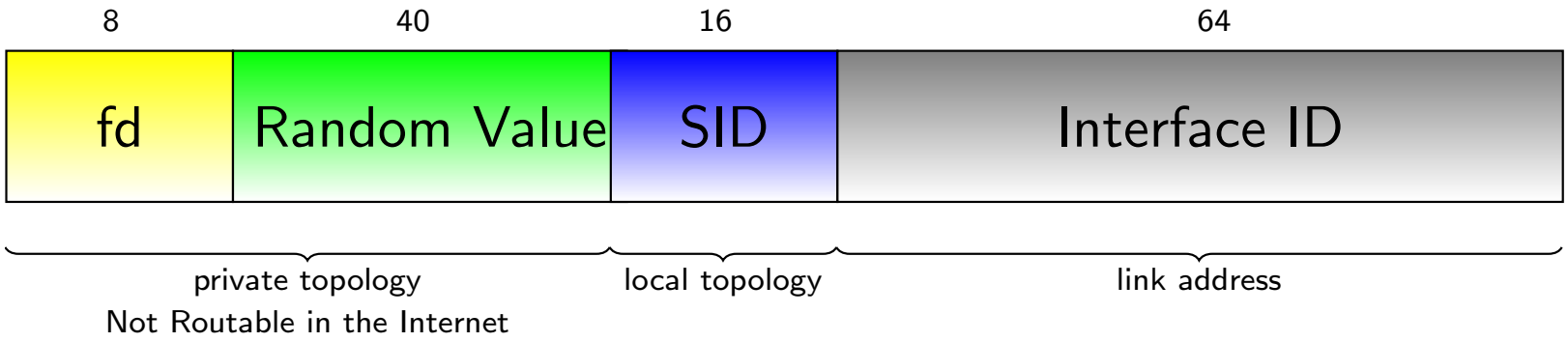
- Global Address, the prefix designates the exit interface
- Link-Local address, the prefix is always fe80::/10
  - The exit interface is not defined
  - A %iface, can be added at the end of the address to avoid ambiguity
- Example:

```
Routing tables

Internet6:

Destination                    Gateway                        Flags      Netif Expire
default                        fe80::213:c4ff:fe69:5f49%en0   UGSc         en0
```

Other kind of addresses : ULA (RFC 4193)

- Equivalent to the private addresses in IPv4
- But try to avoid same prefixes on two different sites:
  - avoid renumbering if two company merge
  - avoid ambiguities when VPN are used
- These prefixes are not routable on the Internet

Unique Local IPv6 Unicast Addresses:



http://www.sixxs.net/tools/grh/ula/ to create your own ULA prefix.

Generic Format:

- T (Transient) 0: well known address - 1: temporary address
- P (Prefix) 1 : assigned from a network prefix (T must be set to 1)
- R (Rendez Vous Point) 1: contains the RP address (P & T set to 1)
- Scope :
  - 1 - interface-local
  - 2 - link-local
  - 3 - reserved
  - 4 - admin-local
  - 5 - site-local
  - 8 - organisation-local
  - e - global
  - f - reserved

Some Well Known Multicast Addresses


```
ff02:0:0:0:0:0:0:1 All Nodes Address (link-local scop
ff02:0:0:0:0:0:0:2 All Routers Address
ff02:0:0:0:0:0:0:5 OSPFIGP
ff02:0:0:0:0:0:0:6 OSPFIGP Designated Routers
ff02:0:0:0:0:0:0:9 RIP Routers
ff02:0:0:0:0:0:0:fb mDNSv6
ff02:0:0:0:0:0:1:2 All-dhcp-agents
ff02:0:0:0:0:1:ffxx:xxxx Solicited-Node Address
ff05:0:0:0:0:0:1:3 All-dhcp-servers (site-local scope
```

http://www.iana.org/assignments/ipv6-multicast-addresses

Some Well Known Multicast Addresses


```
ff02:0:0:0:0:0:0:1 All Nodes Address (link-local scop
ff02:0:0:0:0:0:0:2 All Routers Address
ff02:0:0:0:0:0:0:5 OSPFIGP
ff02:0:0:0:0:0:0:6 OSPFIGP Designated Routers
ff02:0:0:0:0:0:0:9 RIP Routers
ff02:0:0:0:0:0:0:fb mDNSv6
ff02:0:0:0:0:0:1:2 All-dhcp-agents
ff02:0:0:0:0:1:ffxx:xxxx Solicited-Node Address
ff05:0:0:0:0:0:1:3 All-dhcp-servers (site-local scope
```

http://www.iana.org/assignments/ipv6-multicast-addresses

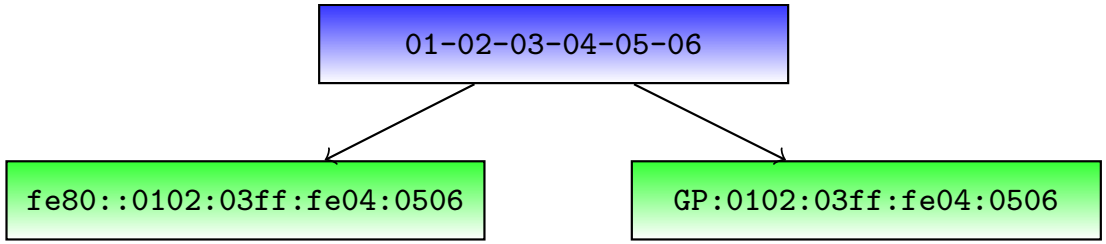
Solicited Multicast Addresses

- Derive a Multicast Address from a Unicast Address
  - Widely used for stateless auto-configuration
  - Avoid the use of broadcast

01-02-03-04-05-06

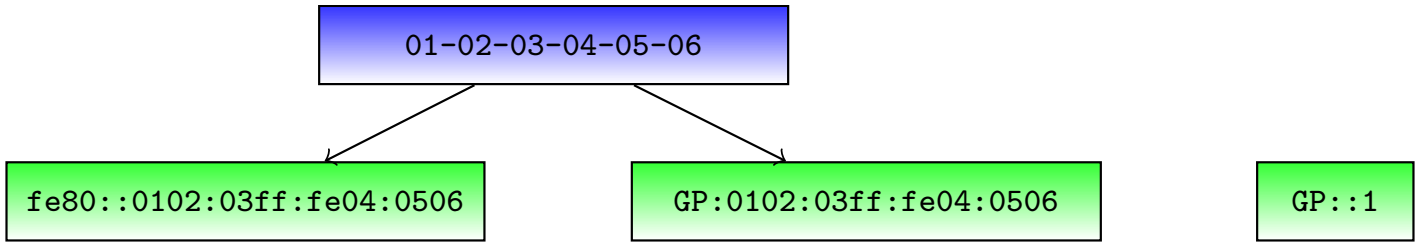
Solicited Multicast Addresses

- Derive a Multicast Address from a Unicast Address
  - Widely used for stateless auto-configuration
  - Avoid the use of broadcast



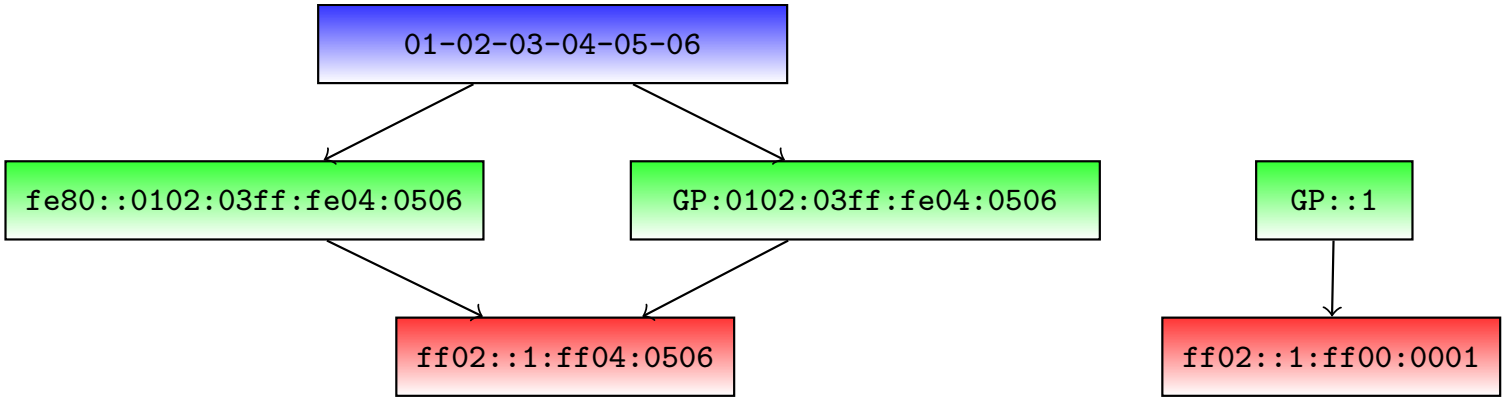
Solicited Multicast Addresses

- Derive a Multicast Address from a Unicast Address
  - Widely used for stateless auto-configuration
  - Avoid the use of broadcast



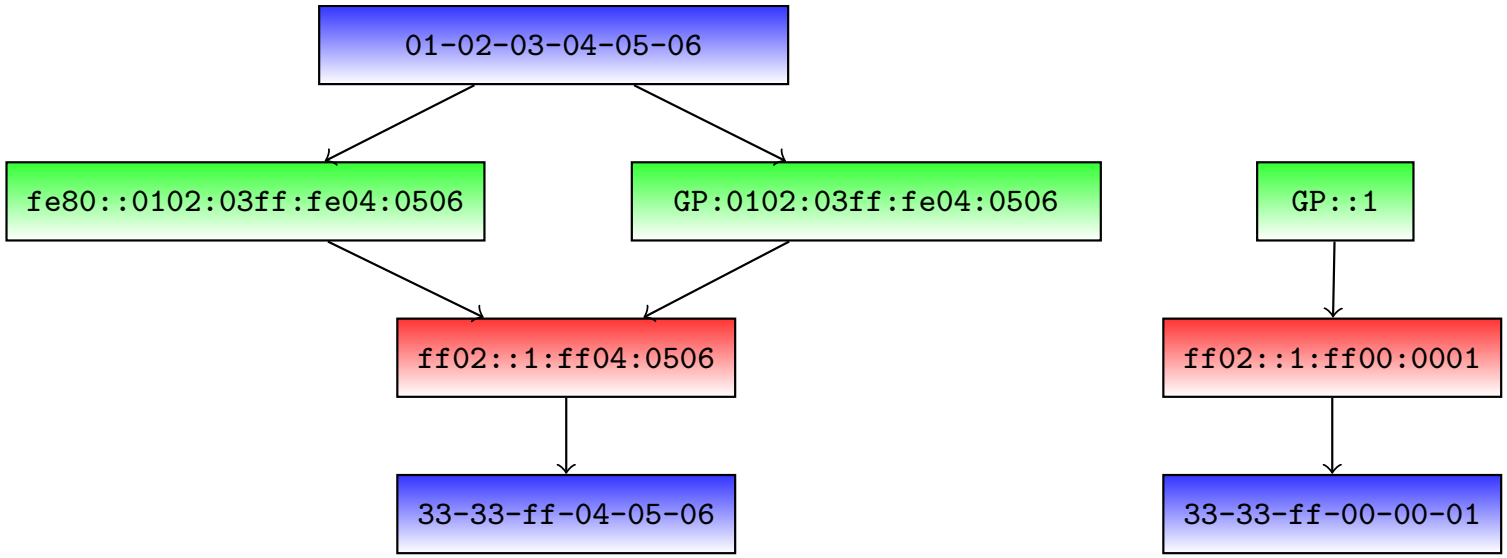
Solicited Multicast Addresses

- Derive a Multicast Address from a Unicast Address
  - Widely used for stateless auto-configuration
  - Avoid the use of broadcast



Solicited Multicast Addresses

- Derive a Multicast Address from a Unicast Address
  - Widely used for stateless auto-configuration
  - Avoid the use of broadcast



Example


```
Vlan5 is up, line protocol is up
IPv6 is enabled, link-local address is fe80::203:fdff:fed6:d400
Description: reseau C5
Global unicast address(es):
  2001:660:7301:1:203:fdff:fed6:d400, subnet is 2001:660:7301:1::/64

Joined group address(es):
  ff02::1  <- All nodes
  ff02::2  <- All routers
  ff02::9  <- RIP
  ff02::1:ffd6:d400  <- Solicited Multicast
```

Example


```
Vlan5 is up, line protocol is up
IPv6 is enabled, link-local address is fe80::203:fdff:fed6:d400
Description: reseau C5
Global unicast address(es):
  2001:660:7301:1:203:fdff:fed6:d400, subnet is 2001:660:7301:1::/64

Joined group address(es):
  ff02::1  <- All nodes
  ff02::2  <- All routers
  ff02::9  <- RIP
  ff02::1:ffd6:d400  <- Solicited Multicast
```

Example


```
Vlan5 is up, line protocol is up
IPv6 is enabled, link-local address is fe80::203:fdff:fed6:d400
Description: reseau C5
Global unicast address(es):
  2001:660:7301:1:203:fdff:fed6:d400, subnet is 2001:660:7301:1::/64

Joined group address(es):
  ff02::1  <- All nodes
  ff02::2  <- All routers
  ff02::9  <- RIP
  ff02::1:ffd6:d400  <- Solicited Multicast
```

Example


```
Vlan5 is up, line protocol is up
IPv6 is enabled, link-local address is fe80::203:fdff:fed6:d400
Description: reseau C5
Global unicast address(es):
  2001:660:7301:1:203:fdff:fed6:d400, subnet is 2001:660:7301:1::/64

Joined group address(es):
  ff02::1  <- All nodes
  ff02::2  <- All routers
  ff02::9  <- RIP
  ff02::1:ffd6:d400  <- Solicited Multicast
```

Example


```
Vlan5 is up, line protocol is up
IPv6 is enabled, link-local address is fe80::203:fdff:fed6:d400
Description: reseau C5
Global unicast address(es):
  2001:660:7301:1:203:fdff:fed6:d400, subnet is 2001:660:7301:1::/64

Joined group address(es):
  ff02::1  <- All nodes
  ff02::2  <- All routers
  ff02::9  <- RIP
  ff02::1:ffd6:d400  <- Solicited Multicast
```

Protocol

IPv6 Header

IPv6 Packet : Simpler

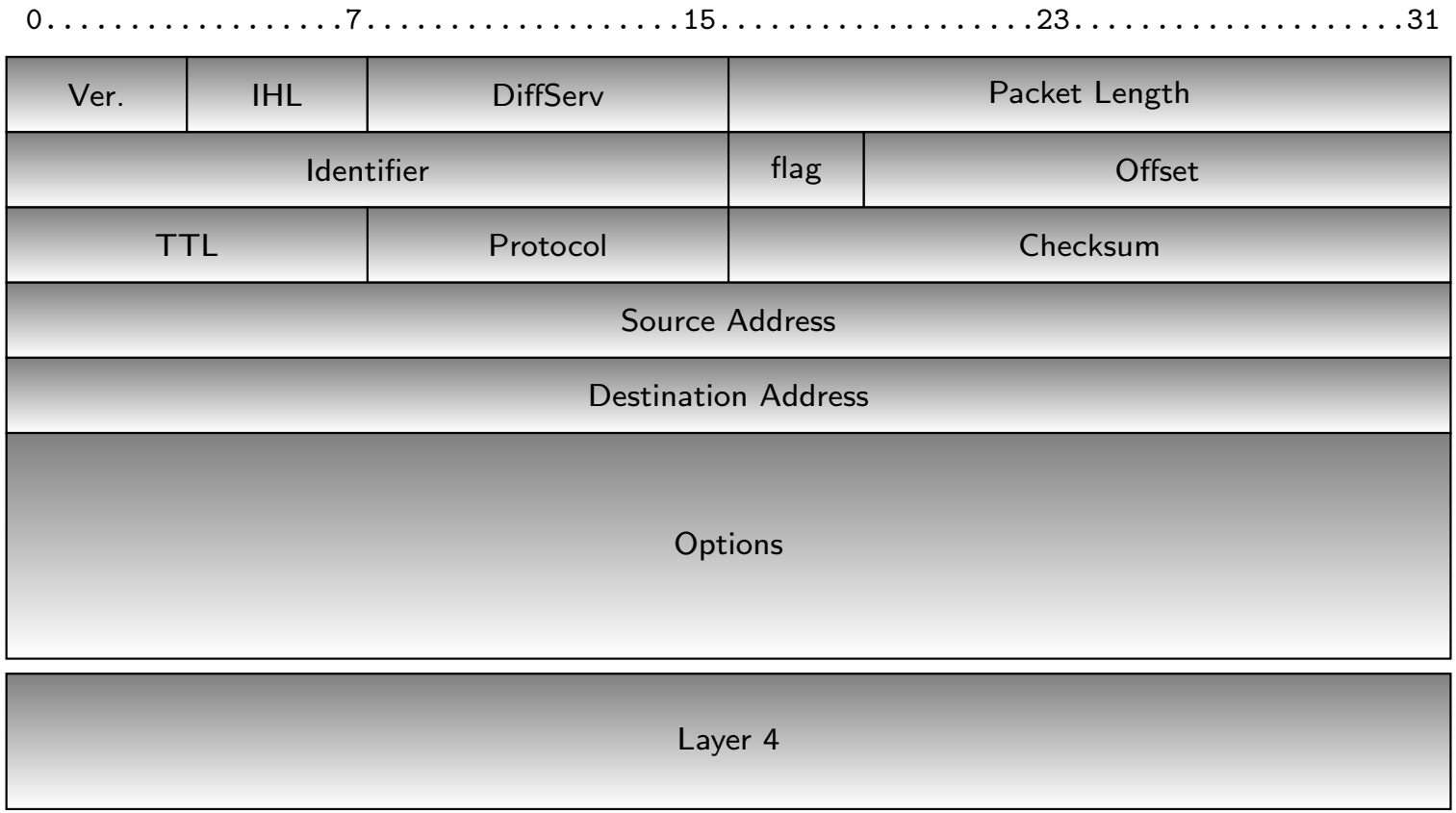
Definition

- IPv6 header follows the same IPv4 principle:
  - fixed address size ... but 4 times larger
  - alignment on 64 bit words (instead of 32)
- Features not used in IPv4 are removed
- Minimum MTU 1280 Bytes
  - If L2 cannot carry 1280 Bytes, then add an adaptation layer such as AAL5 for ATM or 6LoWPAN (RFC 4944) for IEEE 802.15.4.

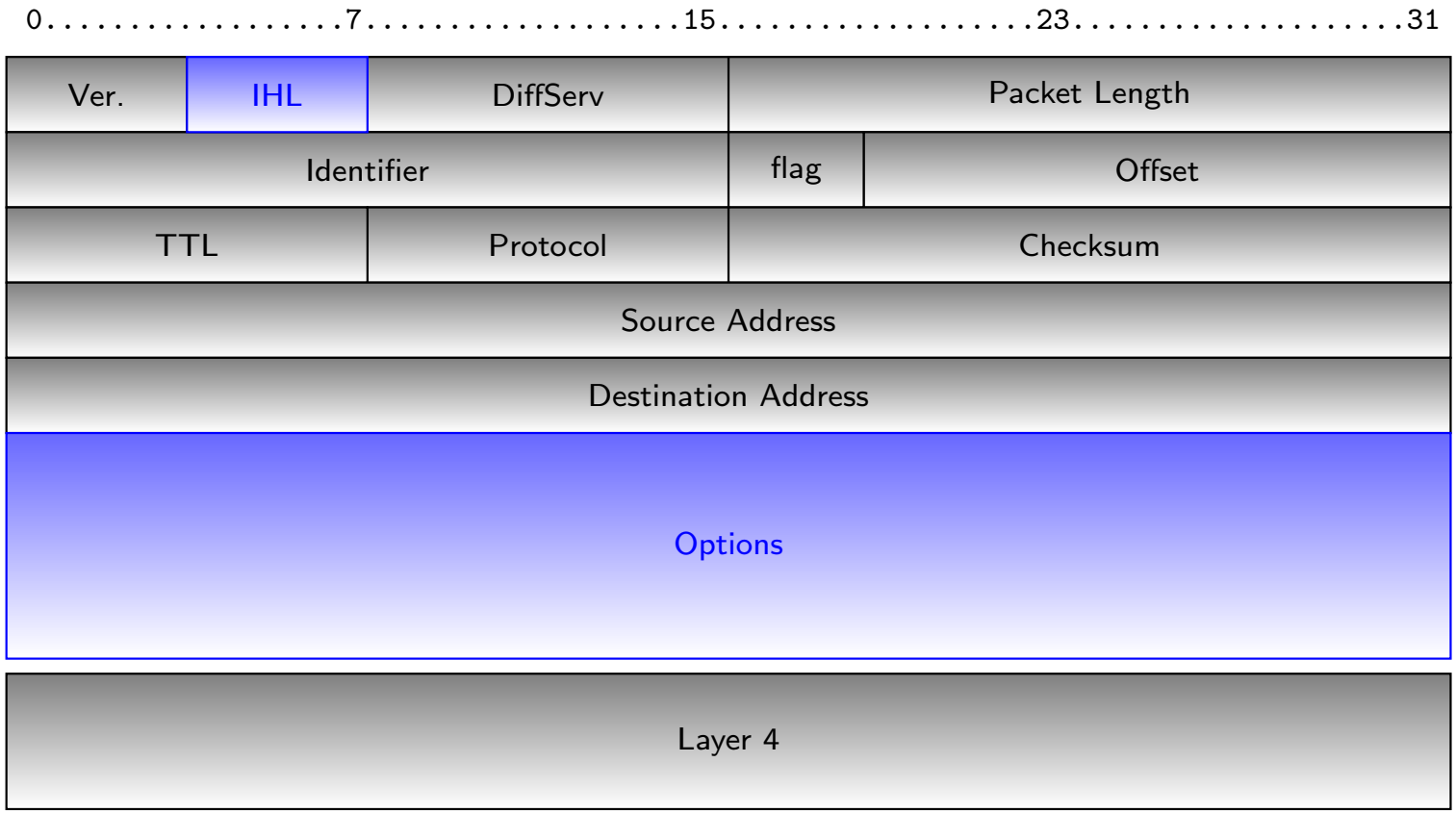
Goal :

- Forward packet as fast as possible
- Less processing in routers
- More features at both ends

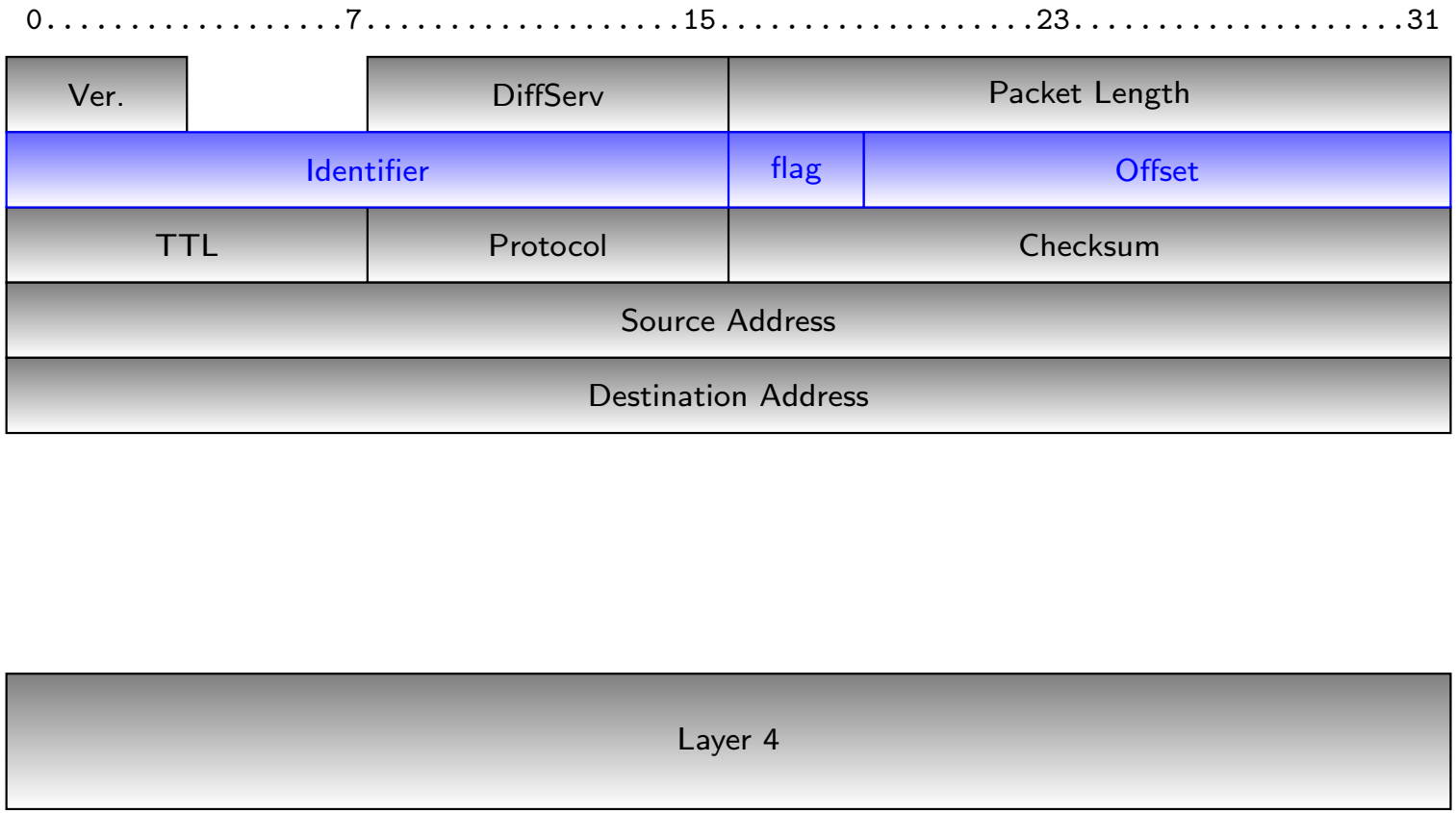
IPv4 Header



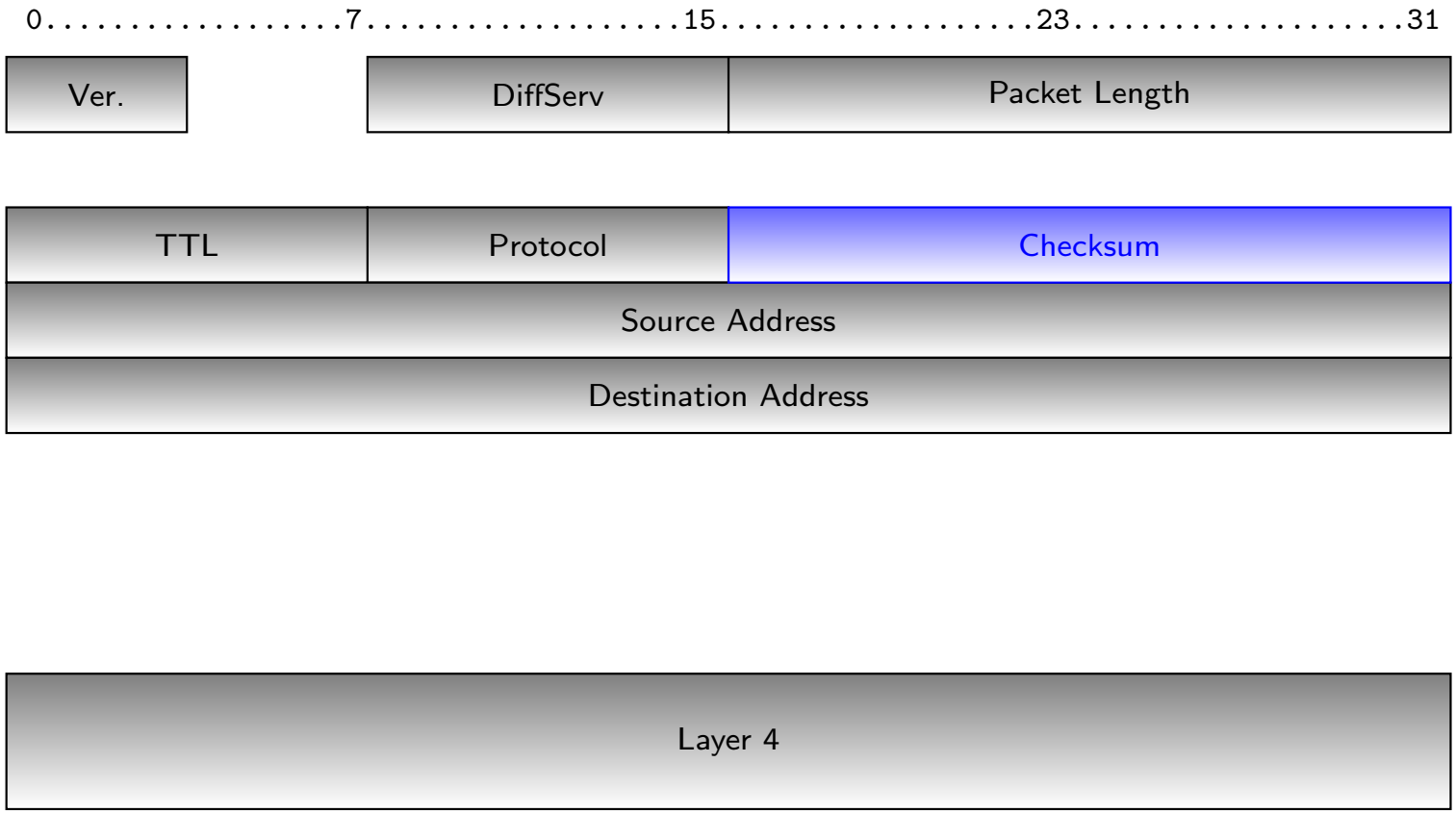
IPv4 Header



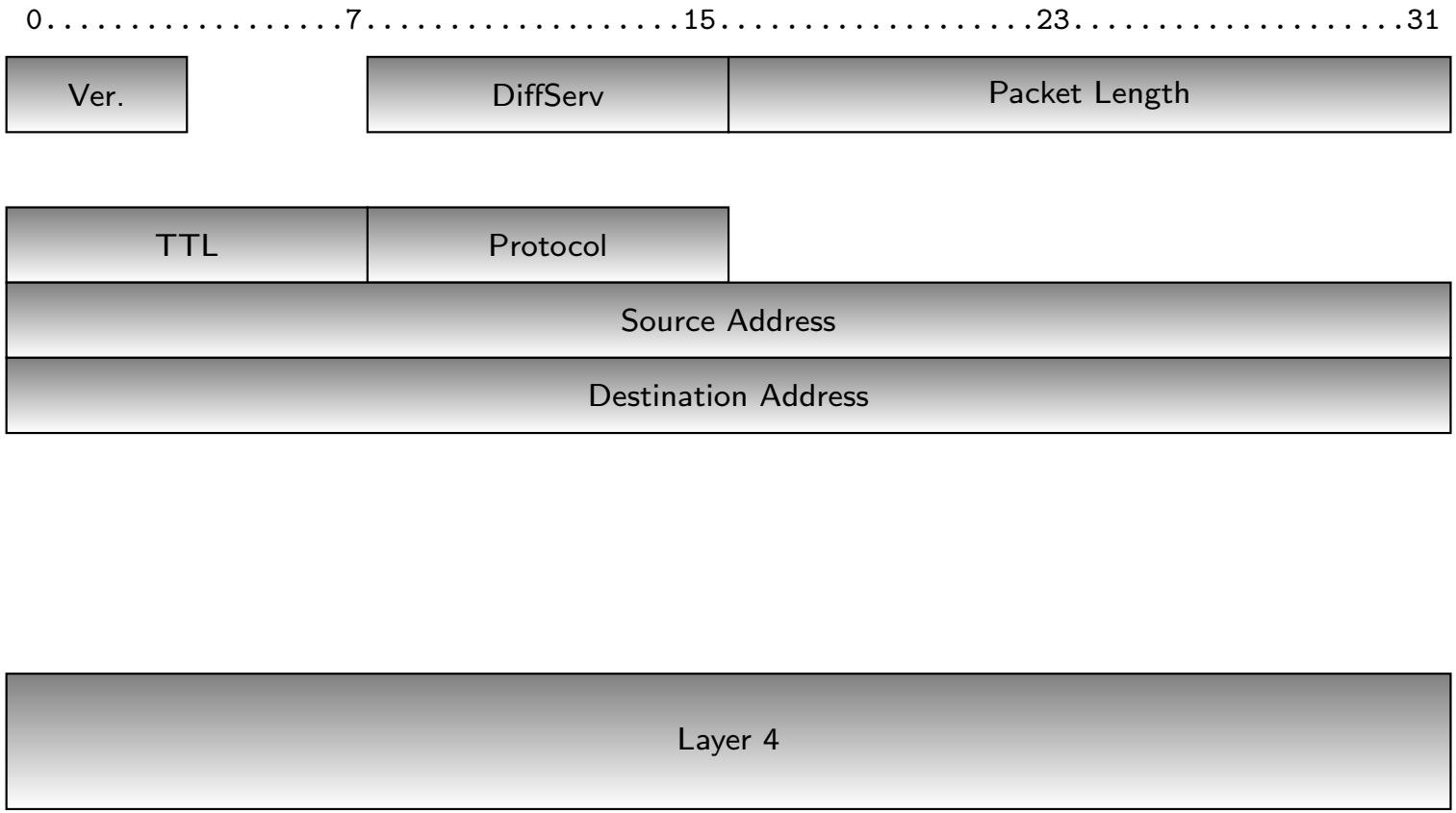
IPv4 Header



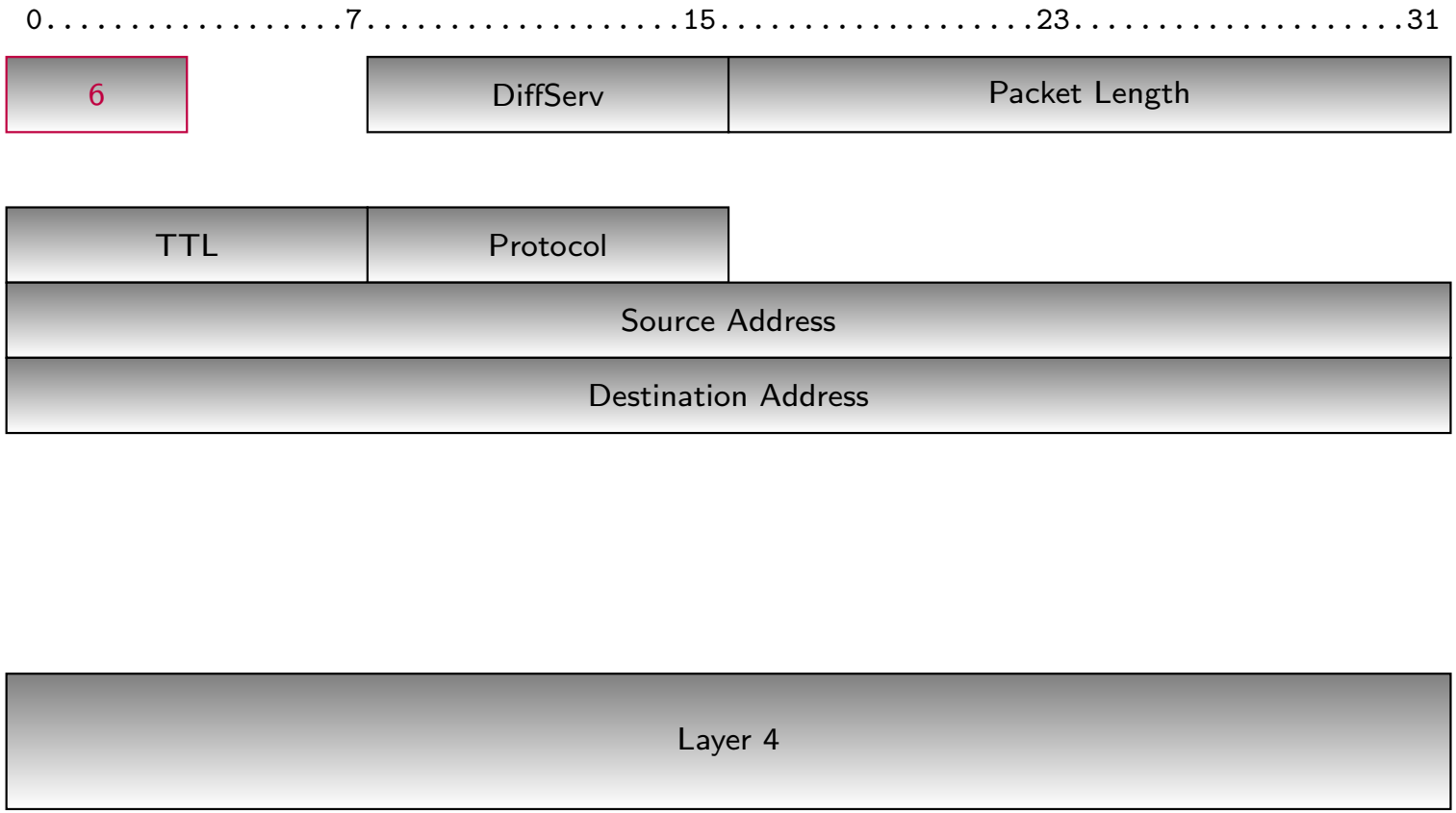
IPv4 Header



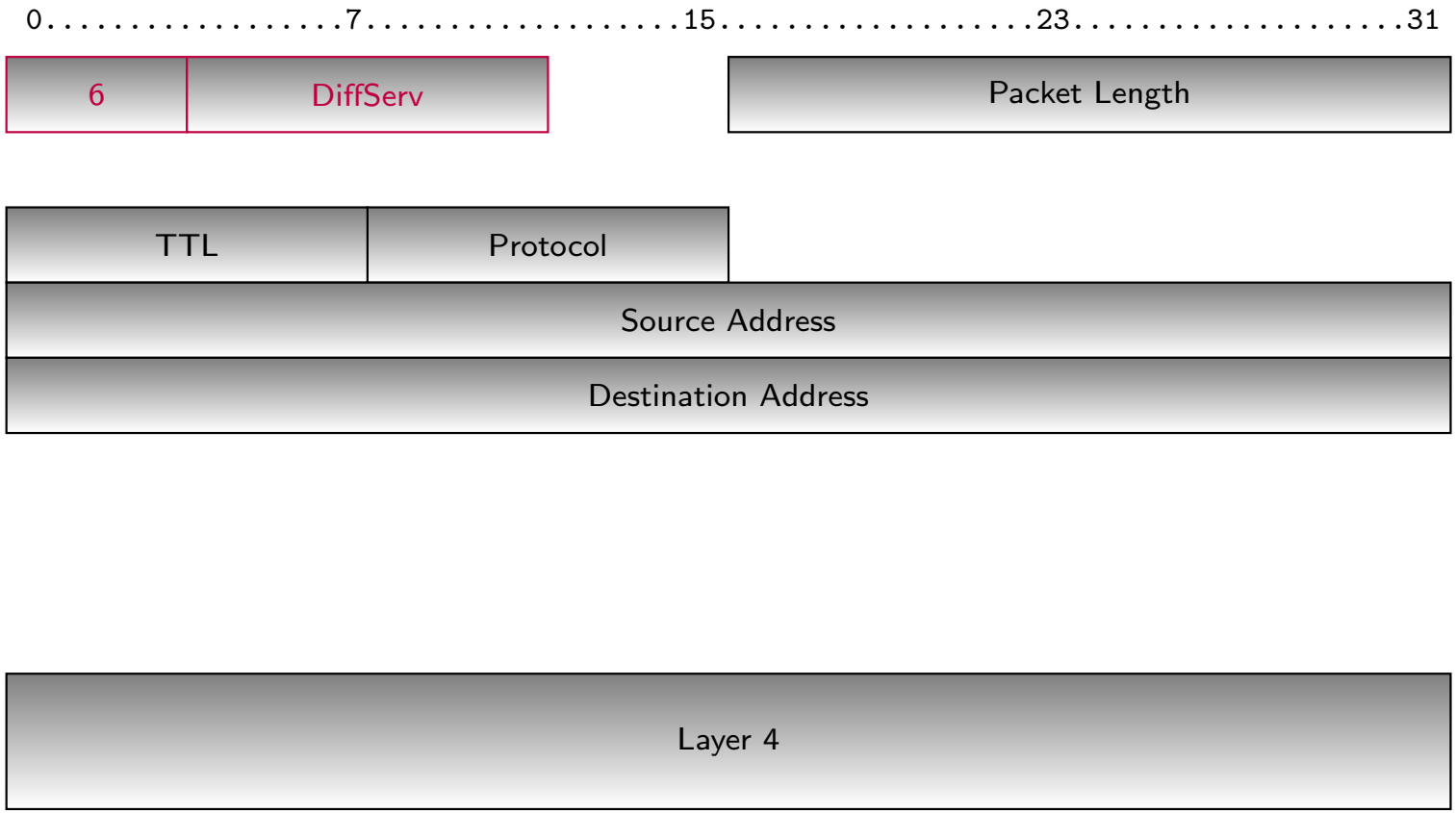
IPv4 Header



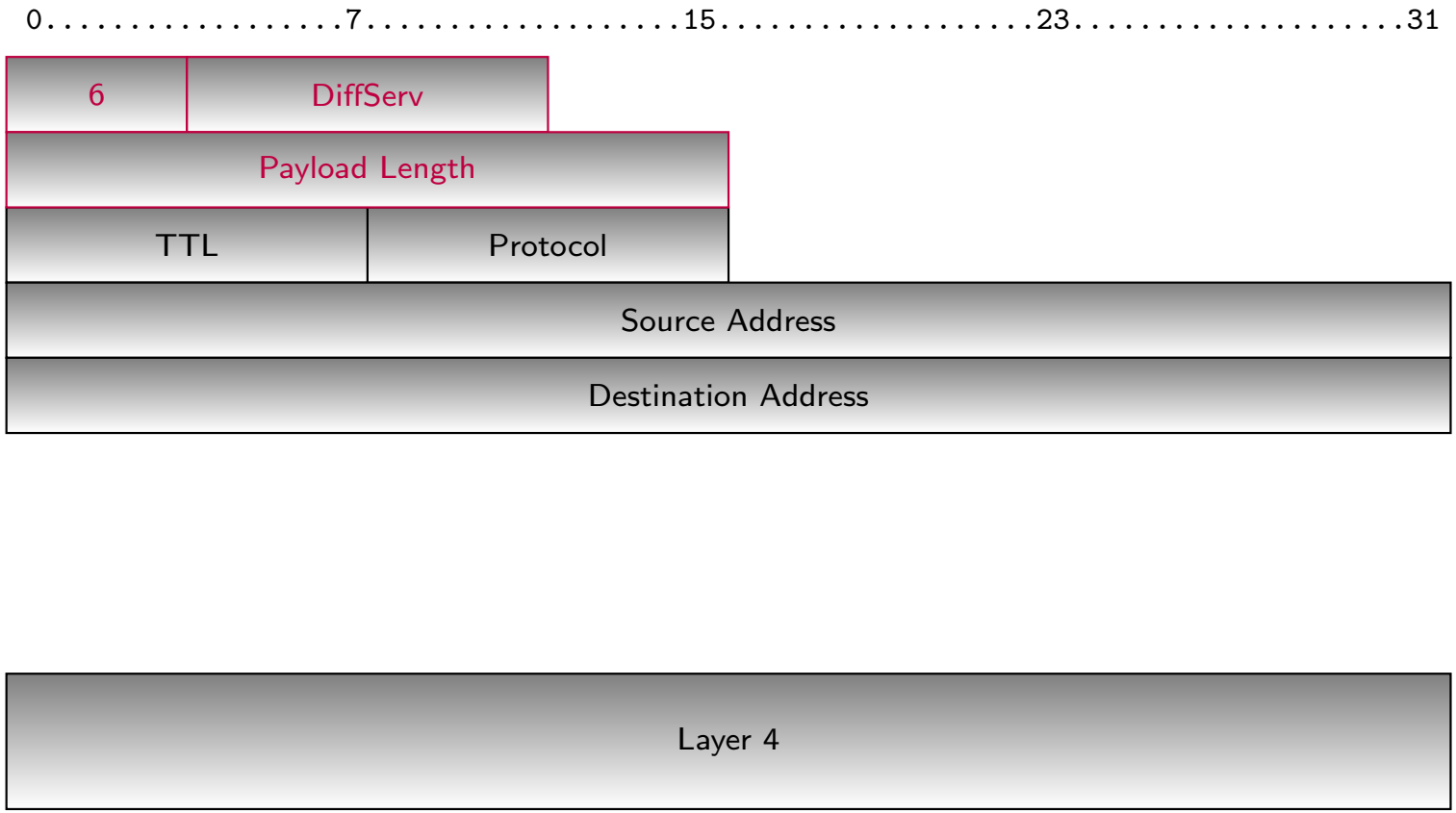
IPv4 Header



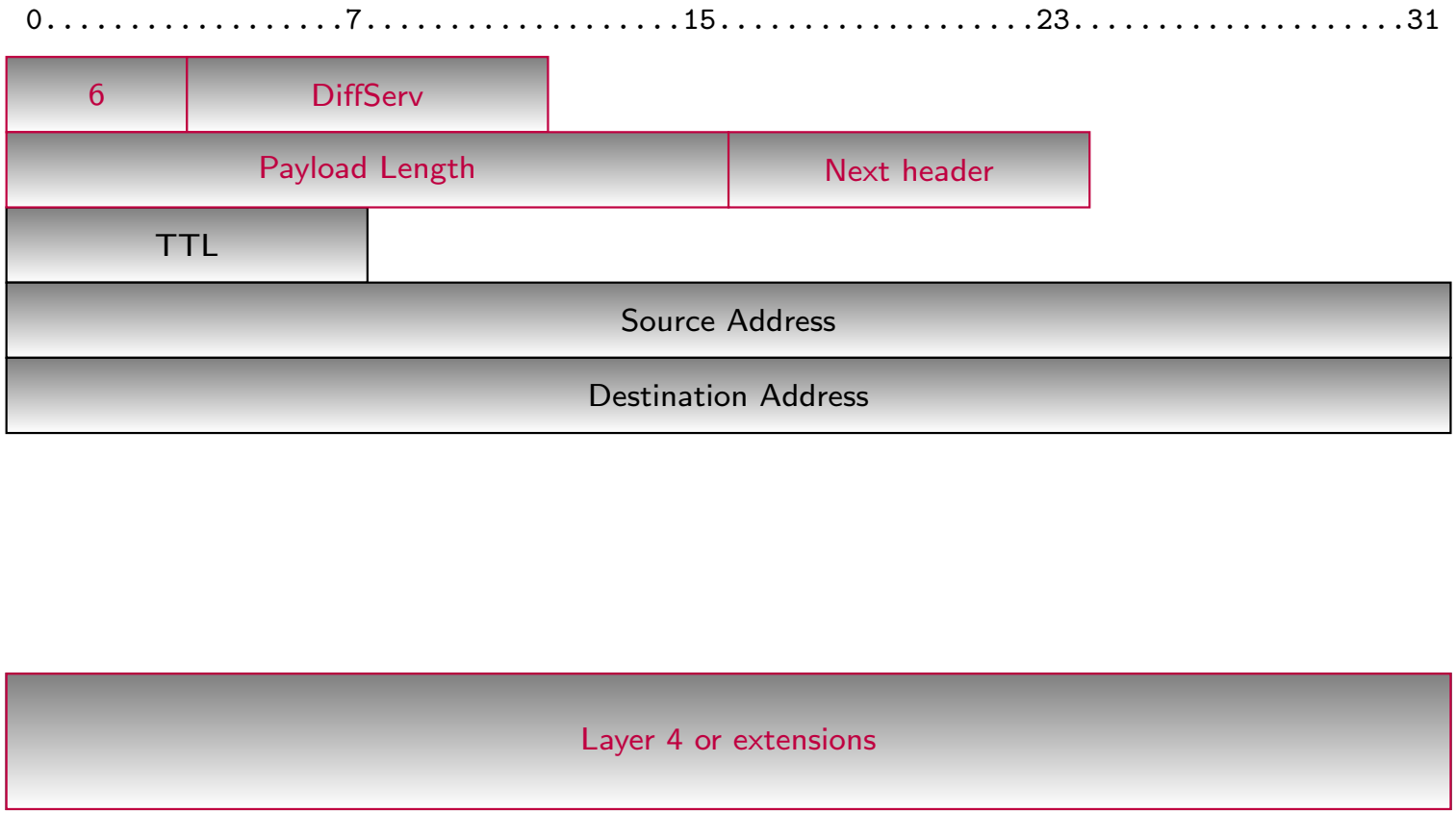
IPv4 Header



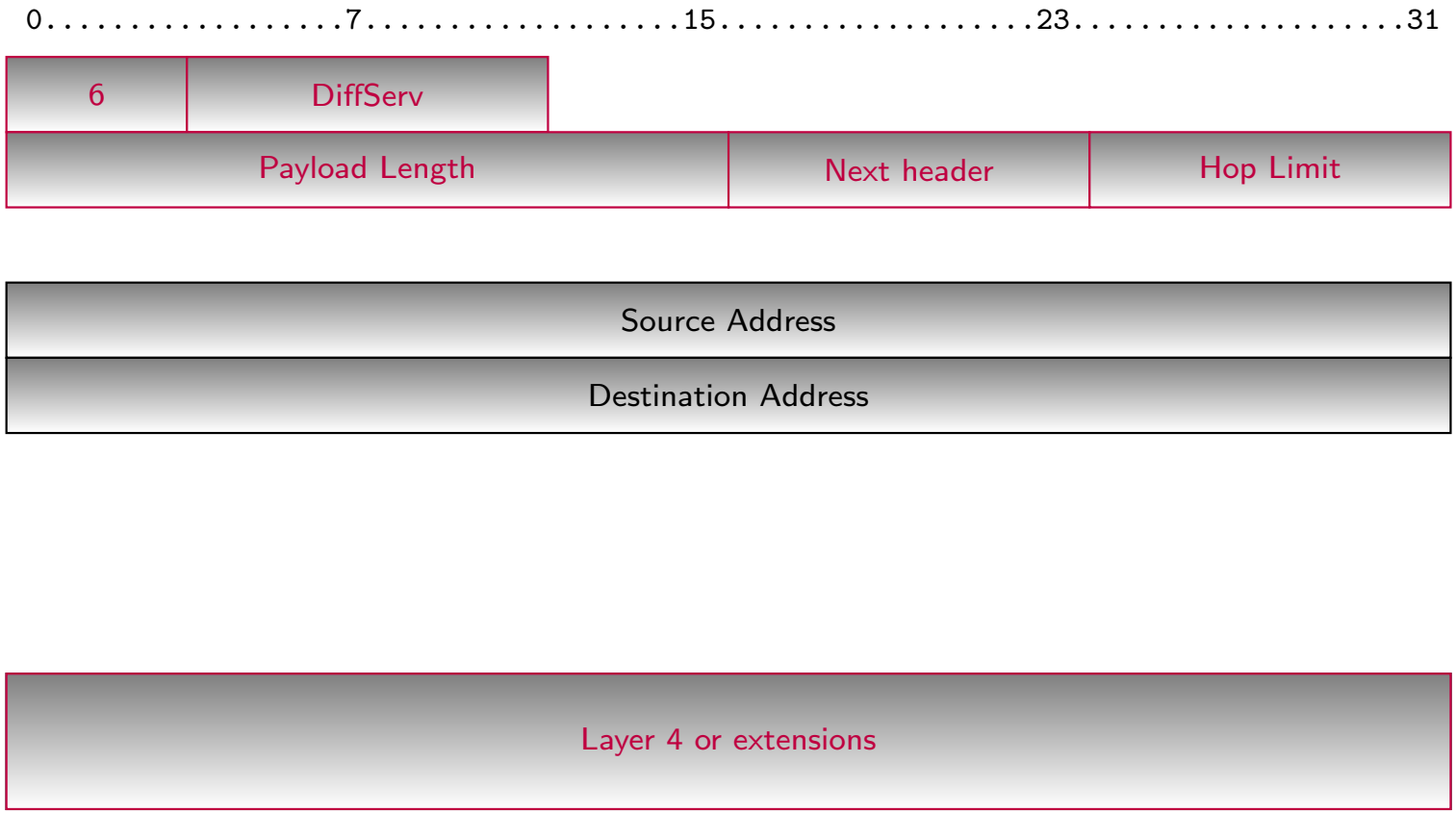
IPv4 Header



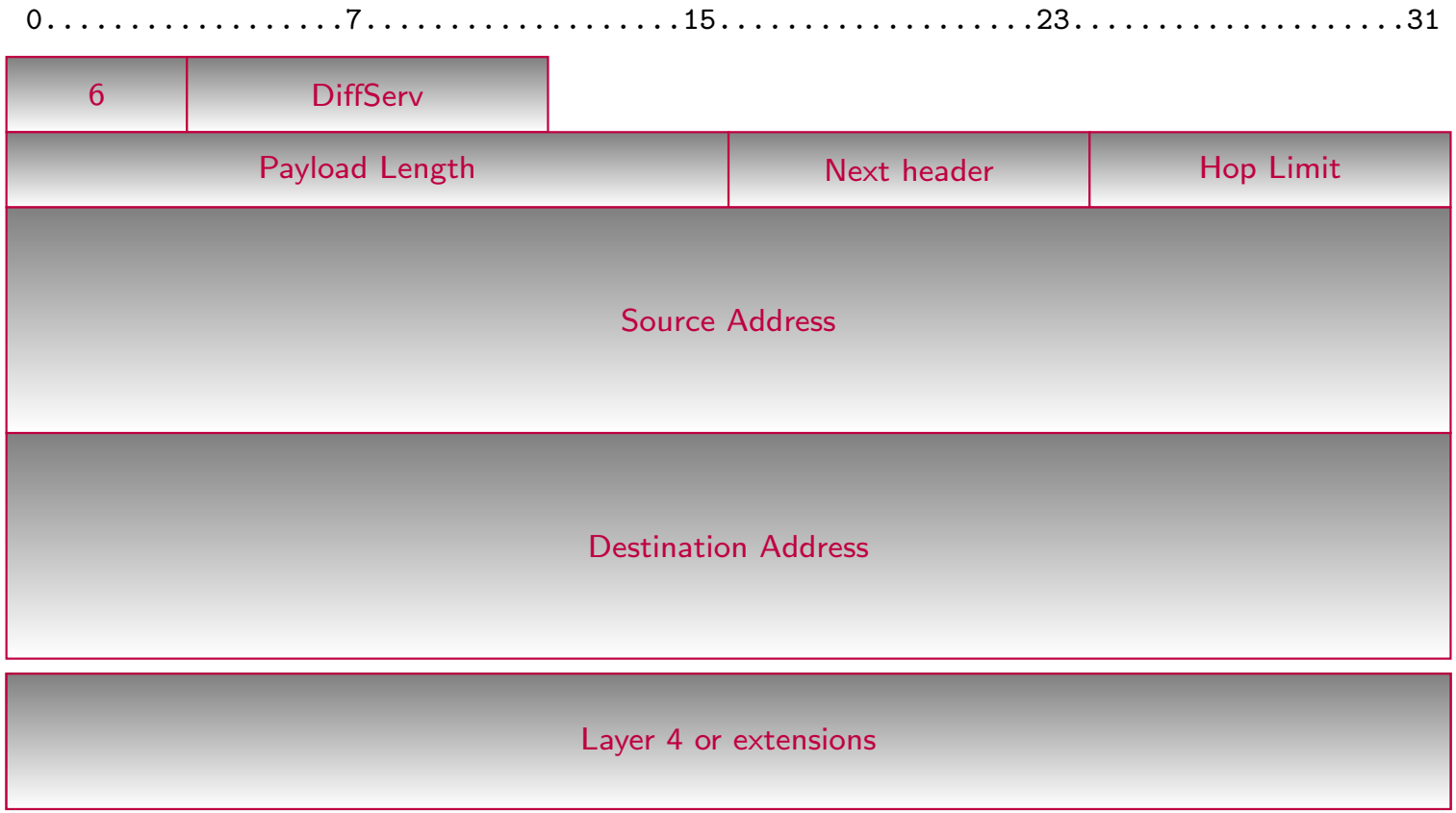
IPv4 Header



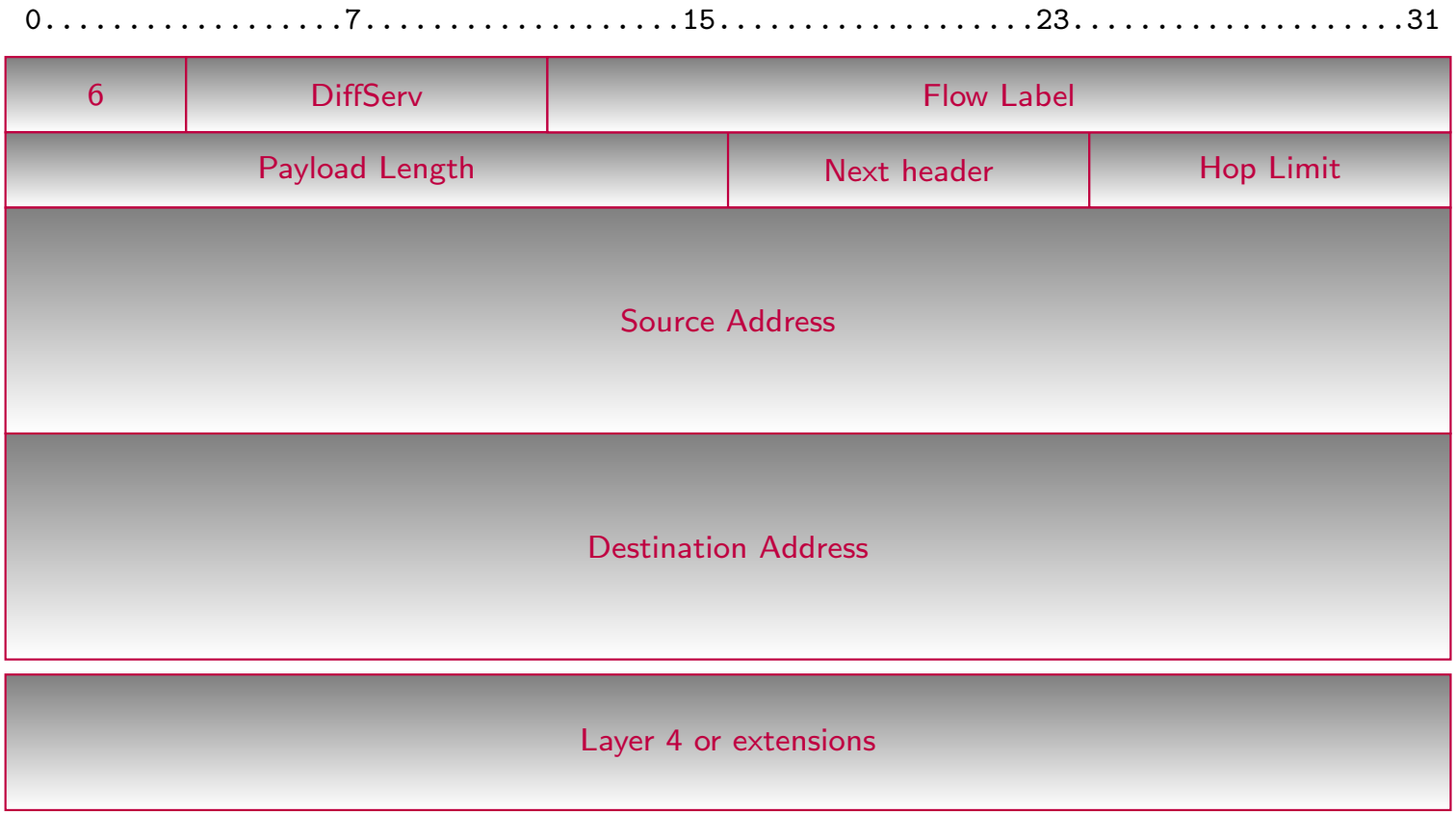
IPv4 Header



IPv4 Header



IPv6 Header

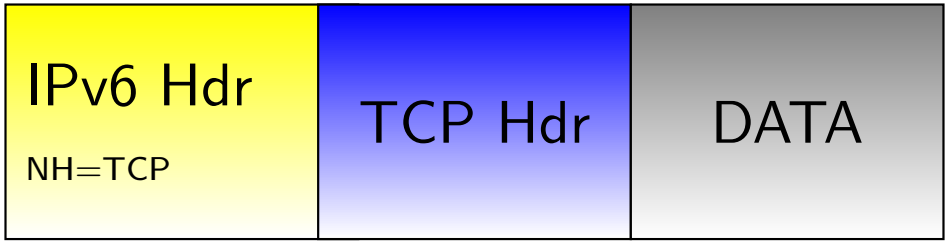


Protocol
IPv6 Extensions

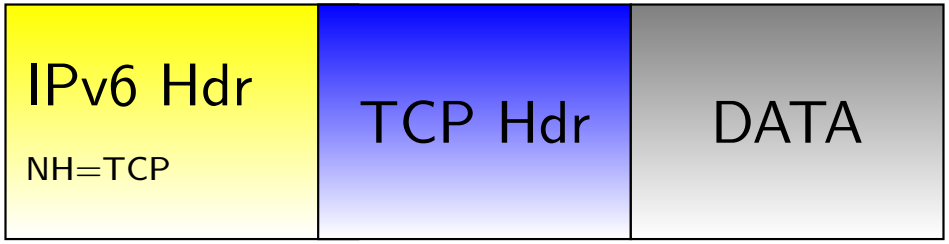
Extensions

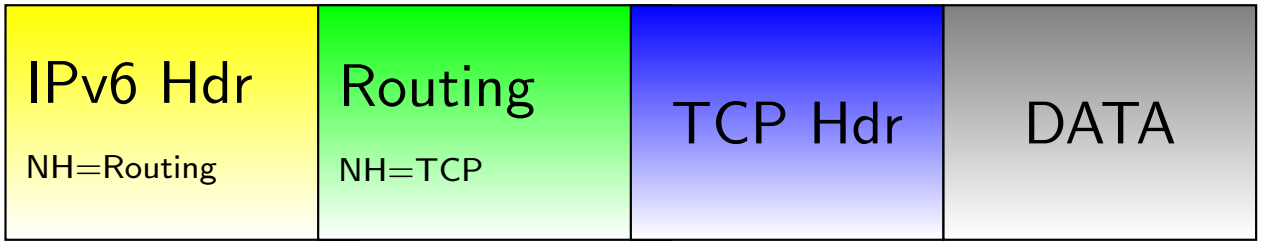
- Seen as a L4 protocol
- Processed only by destination
  - Except Hop-by-Hop processed by every router
  - Equivalent of option field in IPv4
- No size limitation
- Several extensions can be linked to reach L4 protocol
- Processed only by destination
  - Destination (mobility)
  - Routing (loose source routing, mobility)
  - Fragmentation
  - Authentication (AH)
  - Security (ESP)

Extensions in packets

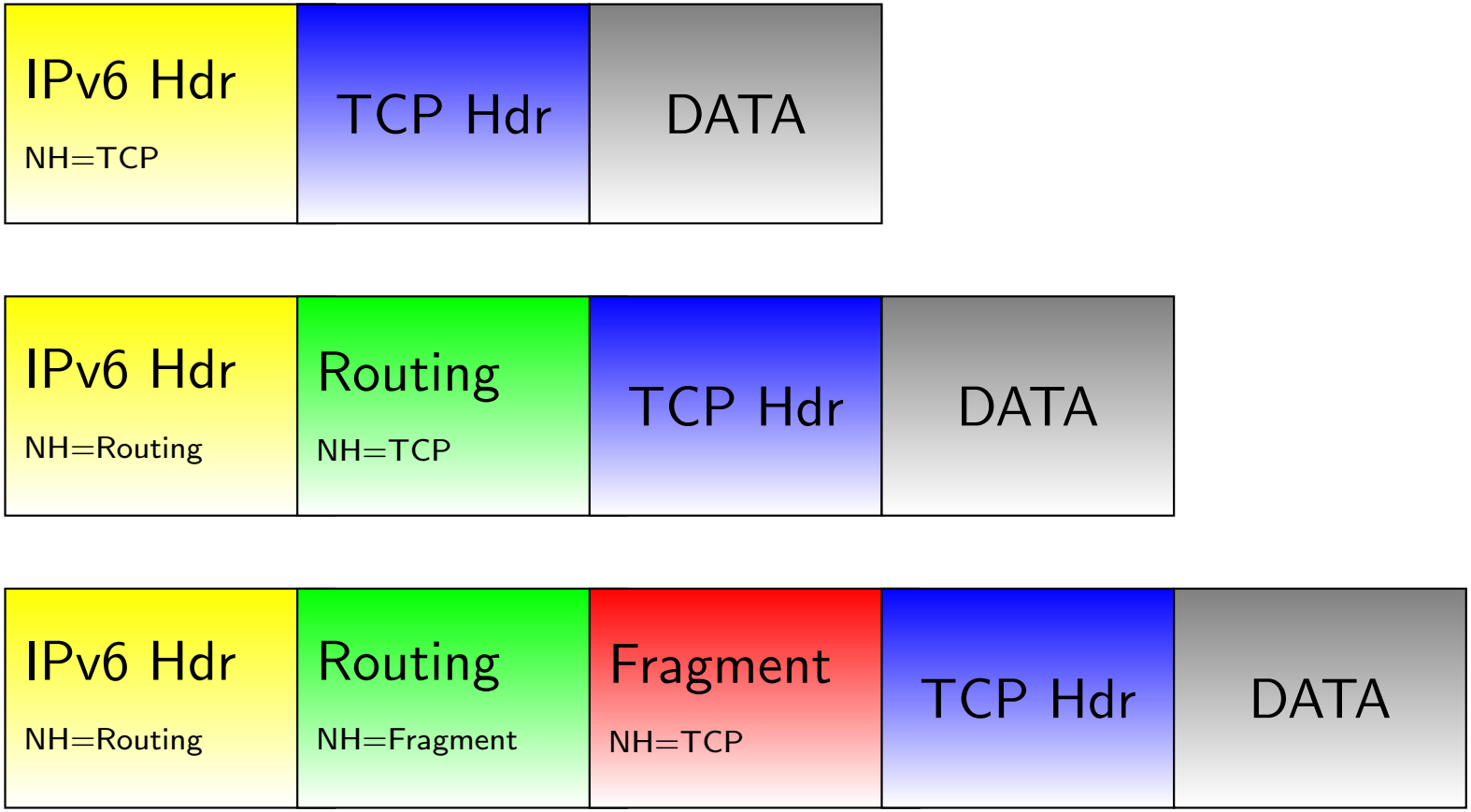


Extensions in packets





Extensions in packets

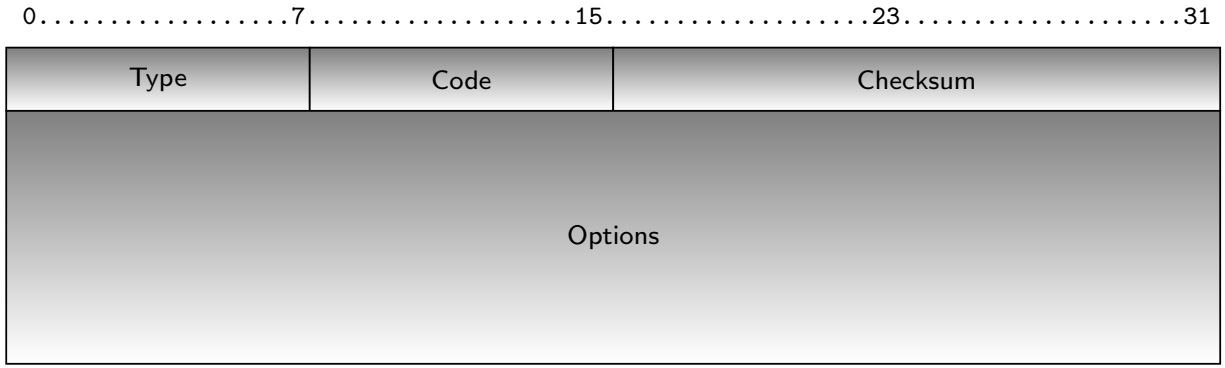


Protocol
ICMPv6

ICMPv6

- ICMPv6 is different from ICMP for IPv4 (RFC 4443)
  - IPv6 (or extension): 58
- Features are extended and better organized
- Never filter ICMPv6 messages blindly, be careful to what you do (see RFC 4890)

Format :



Precision

type code nature of the message ICMPv6
code specifies the cause of the message ICMPv6
mandatory *checksum* used to verify the integrity of ICMP packet

ICMPv6 : Two Functions

- Error occurs during forwarding ($value < 128$)

1	Destination Unreachable
2	Packet Too Big
3	Time Exceeded
4	Parameter Problem

- Management Applications ($value > 128$)

128	Echo Request
129	Echo Reply

130	Group Membership Query
131	Group Membership Report
132	Group Membership Reduction

133	Router Solicitation
134	Router Advertissement
135	Neighbor Solicitation
136	Neighbor Advertissement
137	Redirect

Associated Protocols & Mechanisms
Neighbor Discovery

Neighbor Discovery (RFC 4861)

- IPv6 nodes sharing the same physical medium (link) use Neighbor Discovery (ND) to:
  - determine link-layer addresses of their neighbors
    - IPv4 : ARP
  - Address auto-configuration
    - Layer 3 parameters: IPv6 address, default route, MTU and Hop Limit
    - Only for hosts !
    - IPv4 : impossible, mandate a centralized DHCP server
  - Duplicate Address Detection (DAD)
    - IPv4 : gratuitous ARP
  - maintain neighbors reachability information (NUD)
- Mainly uses multicast addresses but also takes into account NBMA Networks (eg., ATM)
- Protocol packets are transported/encapsulated by/in ICMPv6 messages:
  - Router Solicitation: 133 ; Router Advertisement: 134 ; Neighbor Solicitation: 135 ; Neighbor Advertisement: 136 ; Redirect: 137

Neighbor Discovery (RFC 4861)

- IPv6 nodes sharing the same physical medium (link) use Neighbor Discovery (ND) to:
  - determine link-layer addresses of their neighbors
    - IPv4 : ARP
  - Address auto-configuration
    - Layer 3 parameters: IPv6 address, default route, MTU and Hop Limit
    - Only for hosts !
    - IPv4 : impossible, mandate a centralized DHCP server
  - Duplicate Address Detection (DAD)
    - IPv4 : gratuitous ARP
  - maintain neighbors reachability information (NUD)
- Mainly uses multicast addresses but also takes into account NBMA Networks (eg., ATM)
- Protocol packets are transported/encapsulated by/in ICMPv6 messages:
  - Router Solicitation: 133 ; Router Advertisement: 134 ; Neighbor Solicitation: 135 ; Neighbor Advertisement: 136 ; Redirect: 137

Neighbor Discovery (RFC 4861)

- IPv6 nodes sharing the same physical medium (link) use Neighbor Discovery (ND) to:
  - determine link-layer addresses of their neighbors
    - IPv4 : ARP
  - Address auto-configuration
    - Layer 3 parameters: IPv6 address, default route, MTU and Hop Limit
    - Only for hosts !
    - IPv4 : impossible, mandate a centralized DHCP server
  - Duplicate Address Detection (DAD)
    - IPv4 : gratuitous ARP
  - maintain neighbors reachability information (NUD)
- Mainly uses multicast addresses but also takes into account NBMA Networks (eg., ATM)
- Protocol packets are transported/encapsulated by/in ICMPv6 messages:
  - Router Solicitation: 133 ; Router Advertisement: 134 ; Neighbor Solicitation: 135 ; Neighbor Advertisement: 136 ; Redirect: 137

Neighbor Discovery (RFC 4861)

- IPv6 nodes sharing the same physical medium (link) use Neighbor Discovery (ND) to:
  - determine link-layer addresses of their neighbors
    - IPv4 : ARP
  - Address auto-configuration
    - Layer 3 parameters: IPv6 address, default route, MTU and Hop Limit
    - Only for hosts !
    - IPv4 : impossible, mandate a centralized DHCP server
  - Duplicate Address Detection (DAD)
    - IPv4 : gratuitous ARP
  - maintain neighbors reachability information (NUD)
- Mainly uses multicast addresses but also takes into account NBMA Networks (eg., ATM)
- Protocol packets are transported/encapsulated by/in ICMPv6 messages:
  - Router Solicitation: 133 ; Router Advertisement: 134 ; Neighbor Solicitation: 135 ; Neighbor Advertisement: 136 ; Redirect: 137

Neighbor Discovery (RFC 4861)

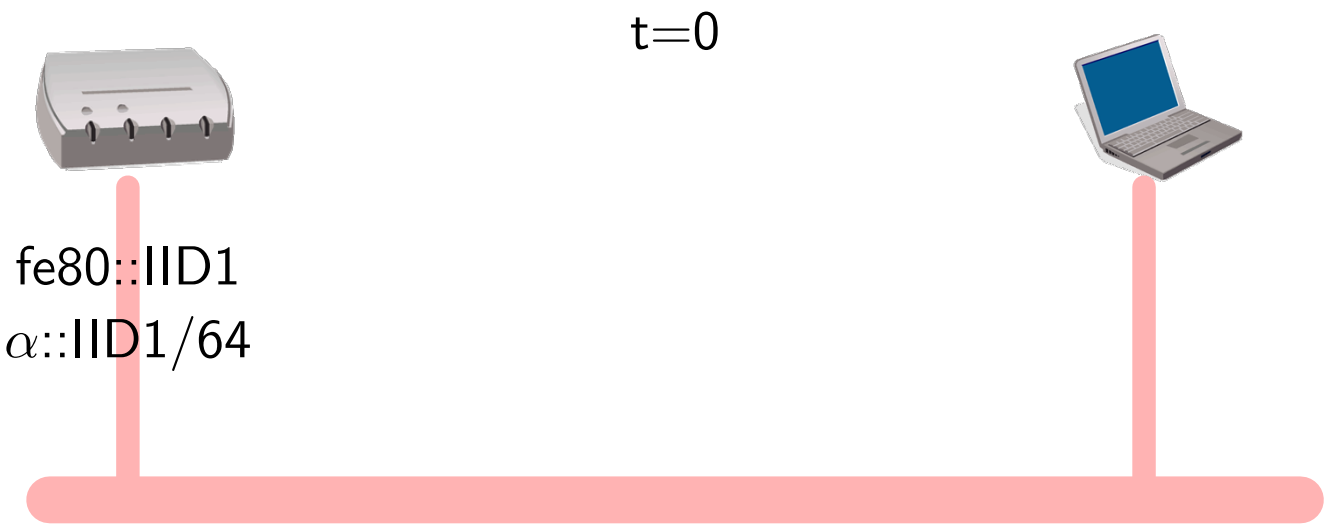
- IPv6 nodes sharing the same physical medium (link) use Neighbor Discovery (ND) to:
  - determine link-layer addresses of their neighbors
    - IPv4 : ARP
  - Address auto-configuration
    - Layer 3 parameters: IPv6 address, default route, MTU and Hop Limit
    - Only for hosts !
    - IPv4 : impossible, mandate a centralized DHCP server
  - Duplicate Address Detection (DAD)
    - IPv4 : gratuitous ARP
  - maintain neighbors reachability information (NUD)
- Mainly uses multicast addresses but also takes into account NBMA Networks (eg., ATM)
- Protocol packets are transported/encapsulated by/in ICMPv6 messages:
  - Router Solicitation: 133 ; Router Advertisement: 134 ; Neighbor Solicitation: 135 ; Neighbor Advertisement: 136 ; Redirect: 137

Neighbor Discovery (RFC 4861)

- IPv6 nodes sharing the same physical medium (link) use Neighbor Discovery (ND) to:
  - determine link-layer addresses of their neighbors
    - IPv4 : ARP
  - Address auto-configuration
    - Layer 3 parameters: IPv6 address, default route, MTU and Hop Limit
    - Only for hosts !
    - IPv4 : impossible, mandate a centralized DHCP server
  - Duplicate Address Detection (DAD)
    - IPv4 : gratuitous ARP
  - maintain neighbors reachability information (NUD)
- Mainly uses multicast addresses but also takes into account NBMA Networks (eg., ATM)
- Protocol packets are transported/encapsulated by/in ICMPv6 messages:
  - Router Solicitation: 133 ; Router Advertisement: 134 ; Neighbor Solicitation: 135 ; Neighbor Advertisement: 136 ; Redirect: 137

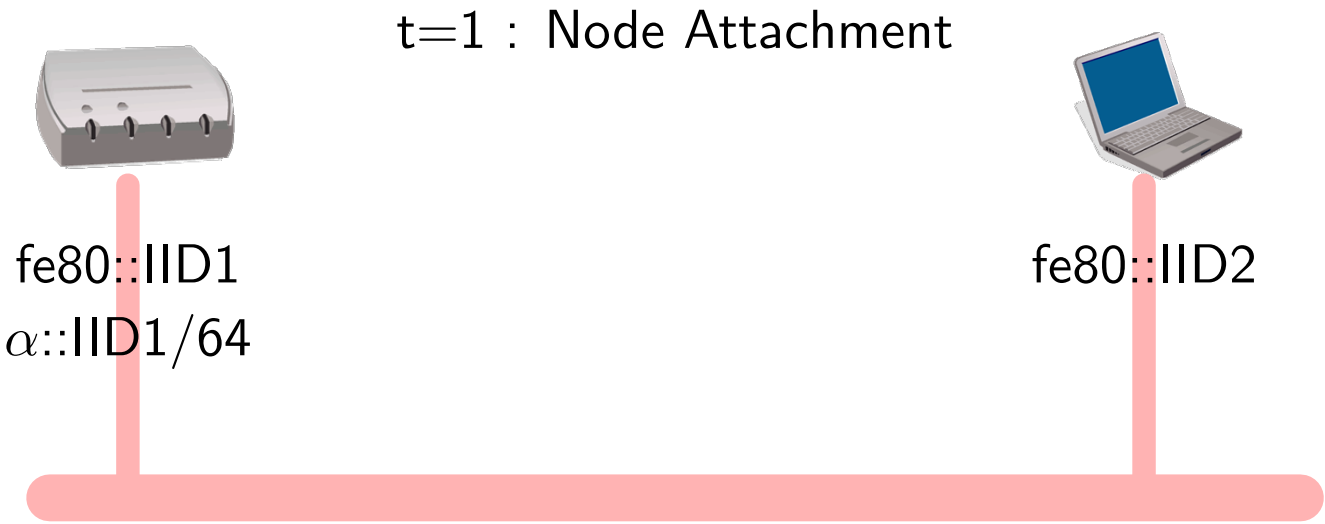
Stateless Auto-configuration: Basic Principles

Stateless Auto-configuration: Basic Principles



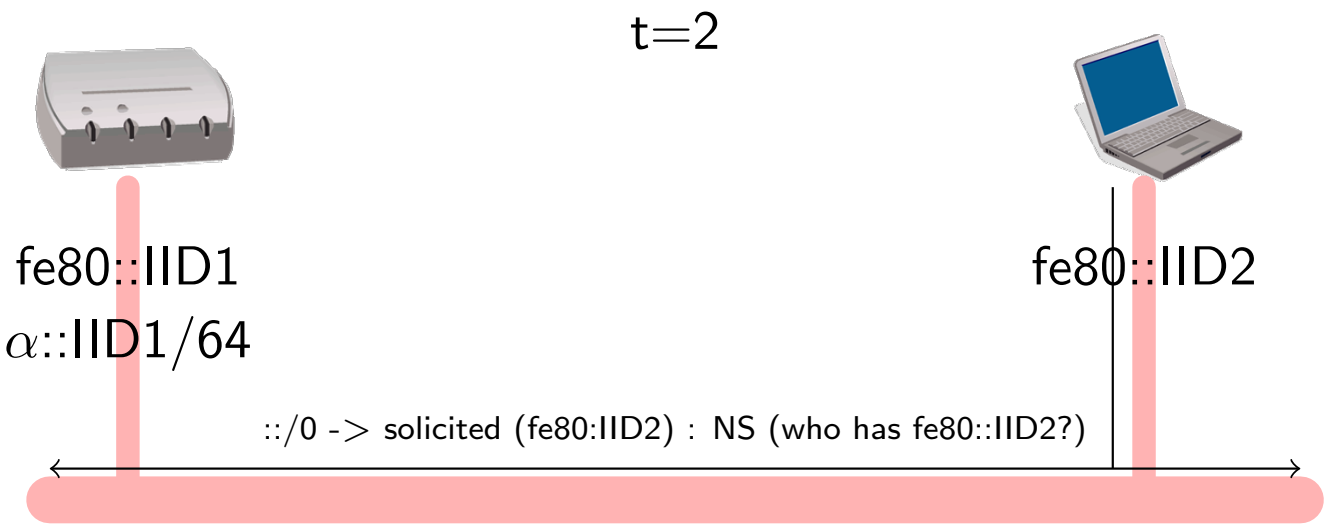
Time t=0: Router is configured with a link-local address and manually configured with a global address (α::/64 is given by the network administrator)

Stateless Auto-configuration: Basic Principles



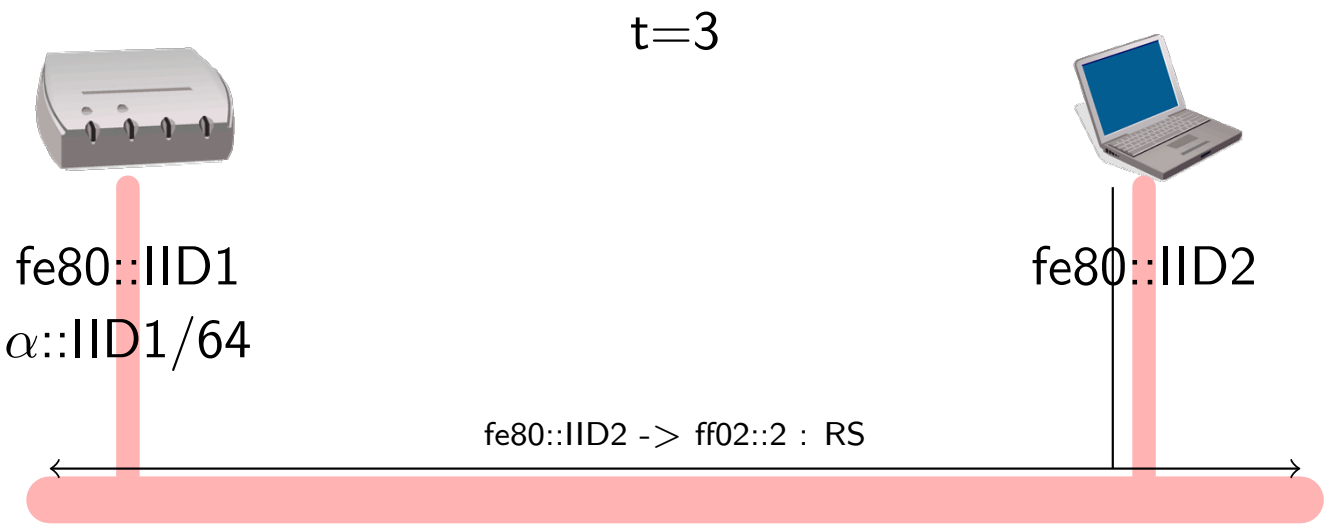
Host constructs its link-local address based on the interface MAC address

Stateless Auto-configuration: Basic Principles



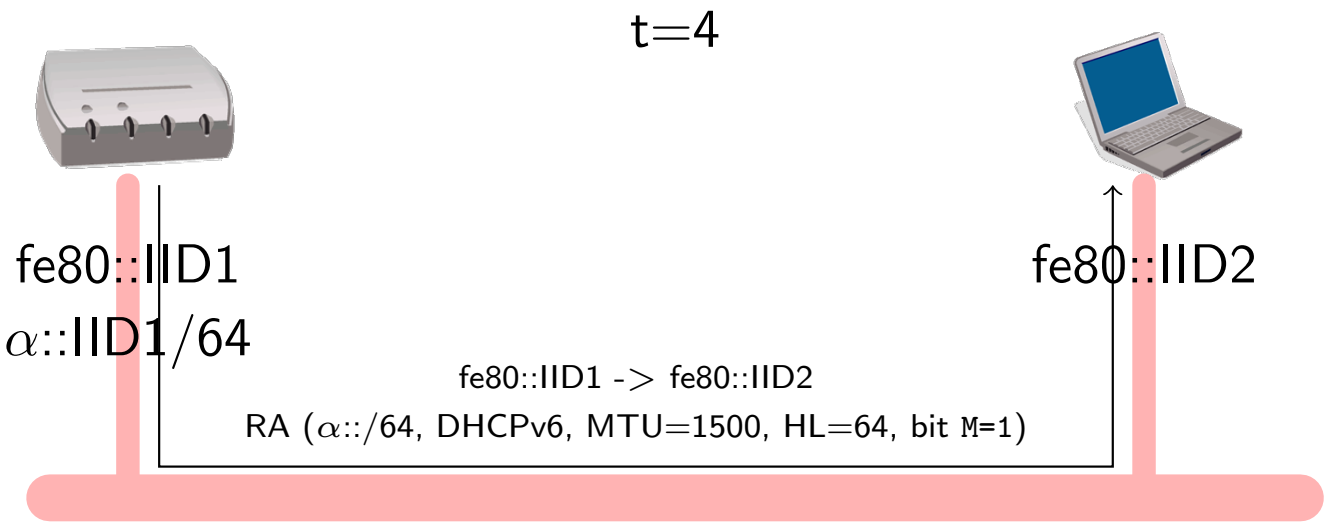
Host does a DAD (i.e. sends a Neighbor Solicitation to query resolution of its own address (tentative): no answers means no other host has this value).

Stateless Auto-configuration: Basic Principles



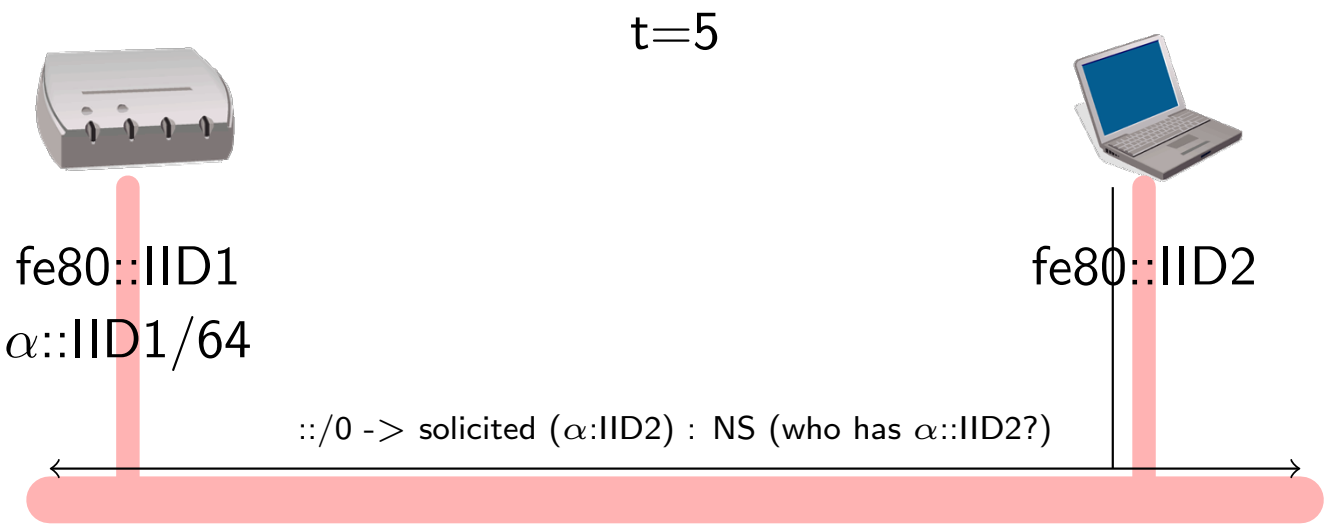
Host sends a Router Solicitation to the Link-Local All-Routers Multicast group using the newly link-local configured address

Stateless Auto-configuration: Basic Principles



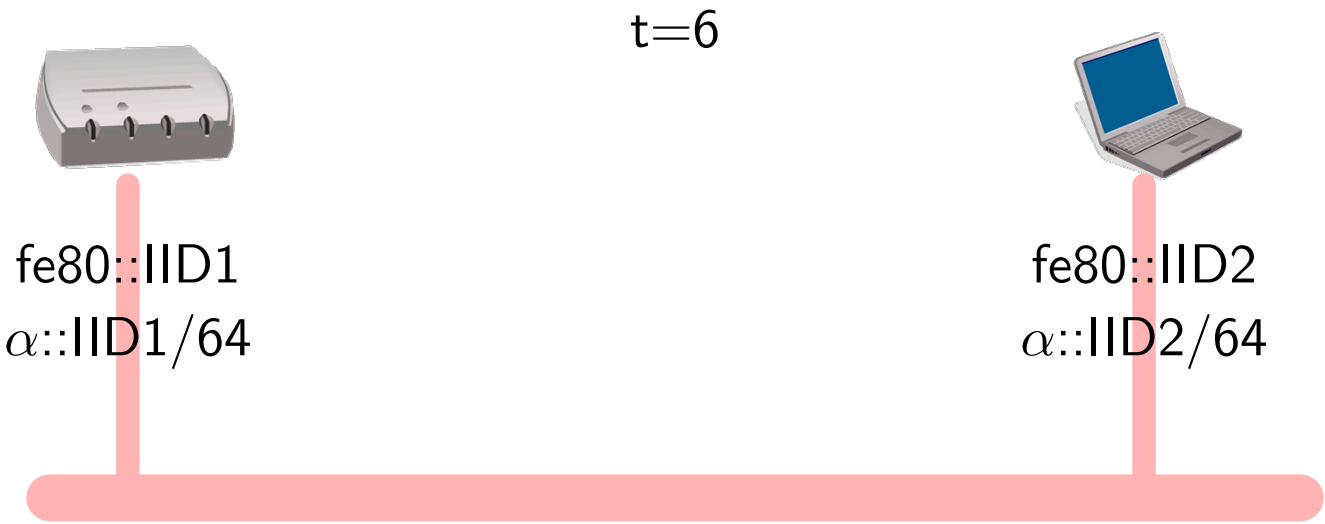
Router directly answers the host using Link-local addresses. The answer may contain a/several prefix(es). Router can also mandate hosts to use DHCPv6 to obtain prefixes (statefull auto-configuration) and/or other parameters (DNS servers...): Bit $\mathtt{M} = 1$.

Stateless Auto-configuration: Basic Principles



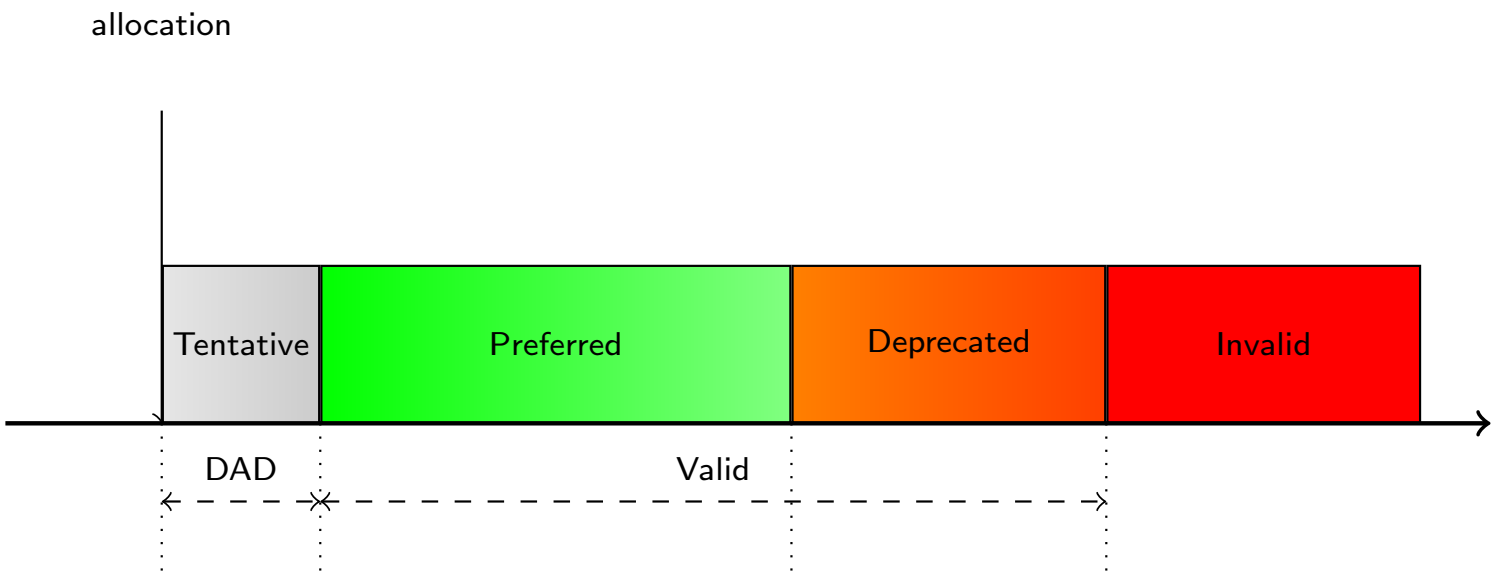
Host does a DAD (i.e. sends a Neighbor Solicitation to query resolution of its own global address: no answers means no other host as this value).

Stateless Auto-configuration: Basic Principles



Host sets the global address and takes answering router as the default router.

Address Lifetime



Router Configuration Example


```
interface Vlan5
 description reseau C5
 ip address 192.108.119.190 255.255.255.128
...
 ipv6 address 2001:660:7301:1::/64 eui-64
 ipv6 enable
 ipv6 nd ra-interval 10
 ipv6 nd prefix-advertisement 2001:660:7301:1::/64 2592000\
 604800 onlink autoconfig
```

Router Configuration Example


```
interface Vlan5
 description reseau C5
 ip address 192.108.119.190 255.255.255.128
...
 ipv6 address 2001:660:7301:1::/64 eui-64
 ipv6 enable
 ipv6 nd ra-interval 10
 ipv6 nd prefix-advertisement 2001:660:7301:1::/64 2592000\
 604800 onlink autoconfig
```

Router Configuration Example


```
interface Vlan5
 description reseau C5
 ip address 192.108.119.190 255.255.255.128
...
 ipv6 address 2001:660:7301:1::/64 eui-64
 ipv6 enable
 ipv6 nd ra-interval 10
 ipv6 nd prefix-advertisement 2001:660:7301:1::/64 2592000\
 604800 onlink autoconfig
```

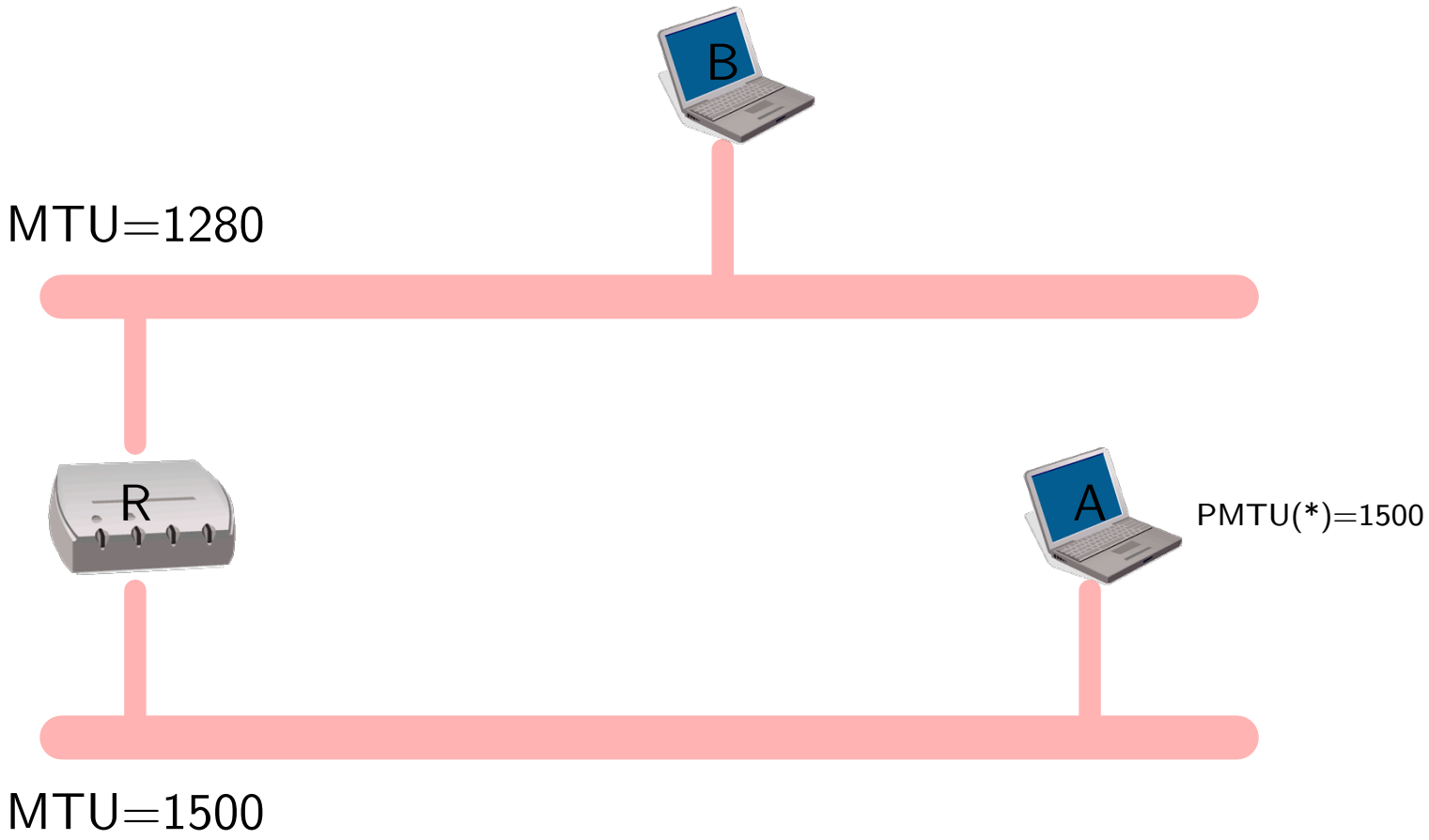
Router Configuration Example


```
interface Vlan5
 description reseau C5
 ip address 192.108.119.190 255.255.255.128
...
 ipv6 address 2001:660:7301:1::/64 eui-64
 ipv6 enable
 ipv6 nd ra-interval 10
 ipv6 nd prefix-advertisement 2001:660:7301:1::/64 2592000\
 604800 onlink autoconfig
```

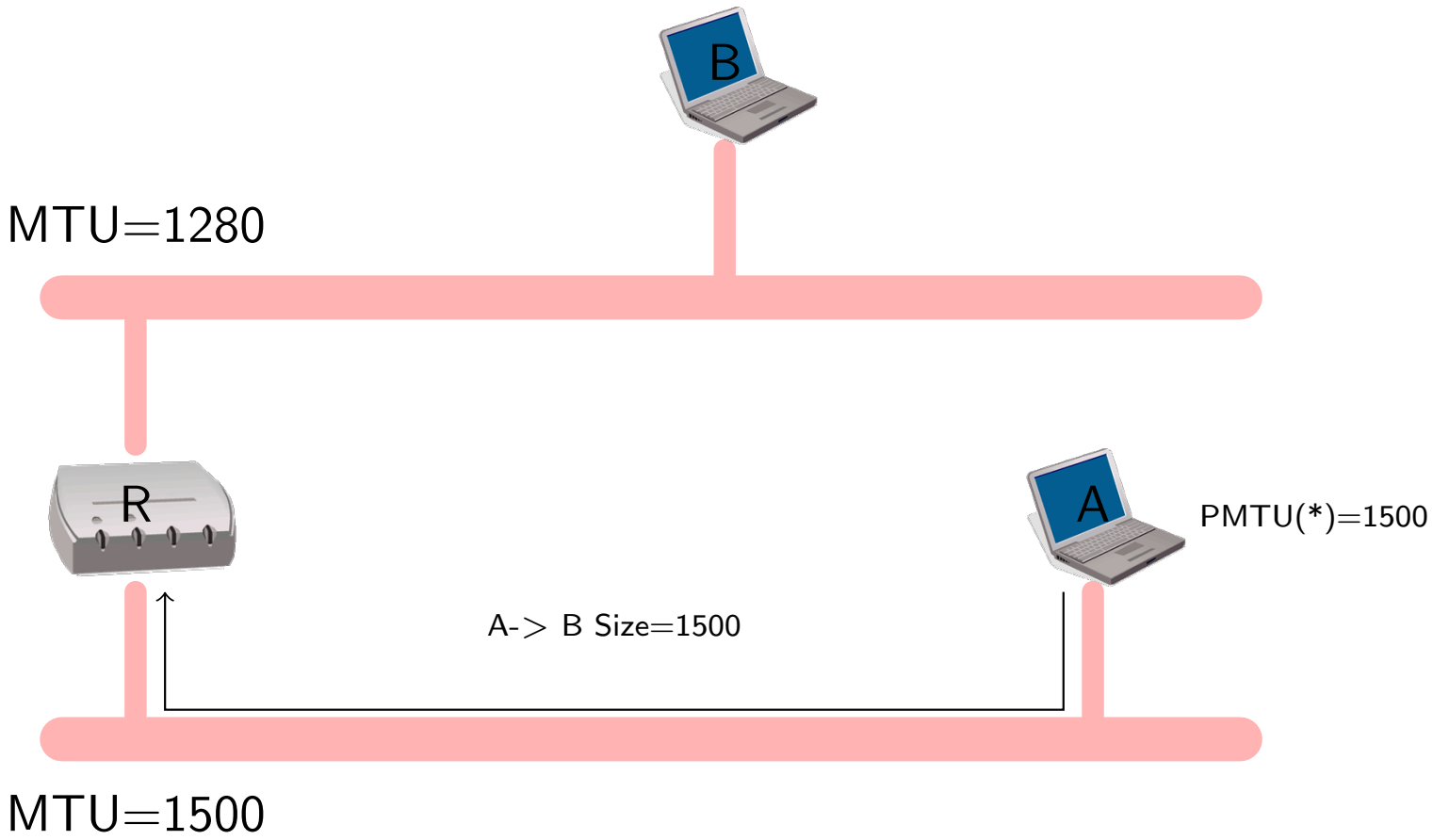
Associated Protocols & Mechanisms

Path MTU discovery

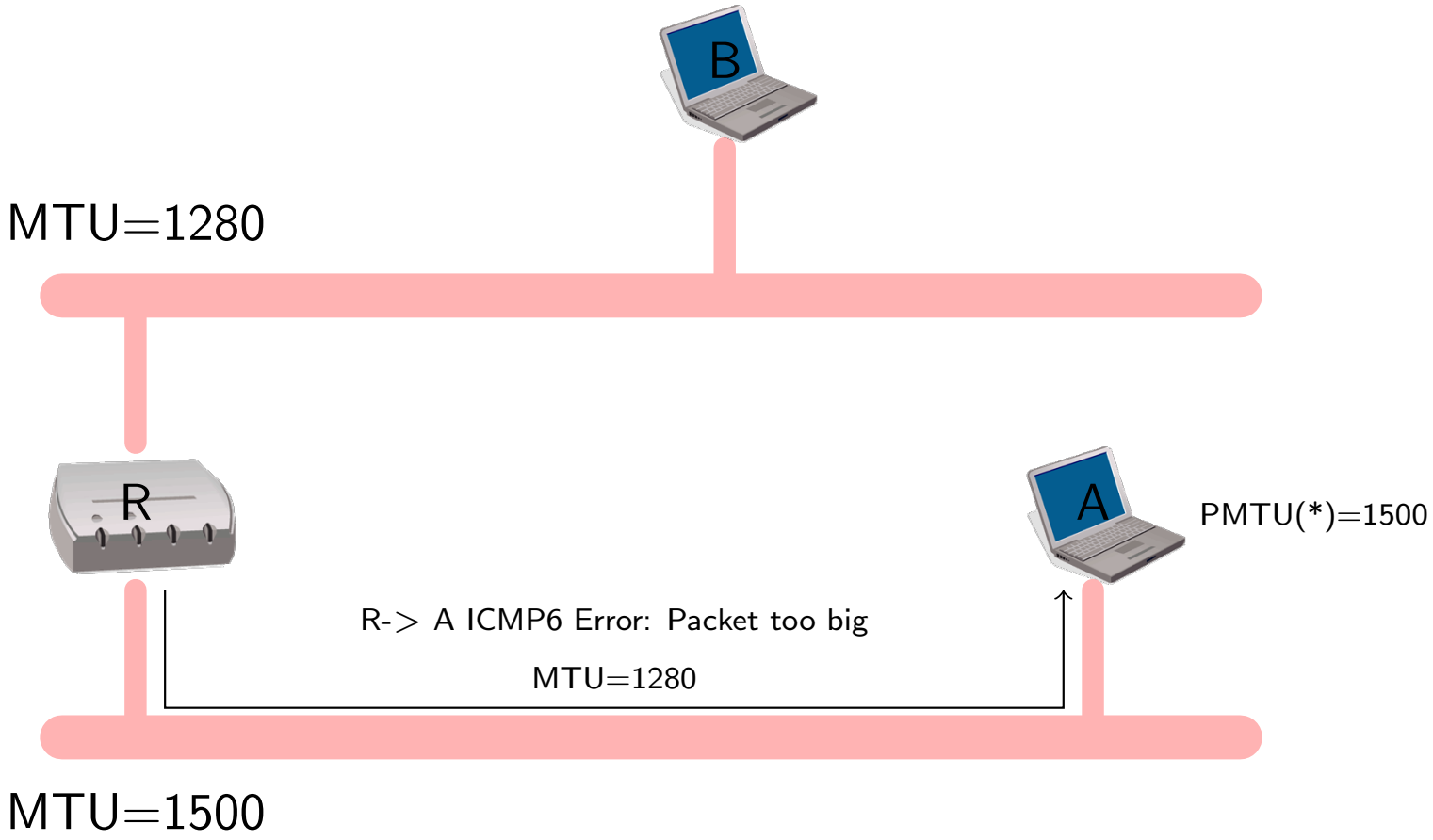
Path MTU discovery for IPv6 (RFC 1981)



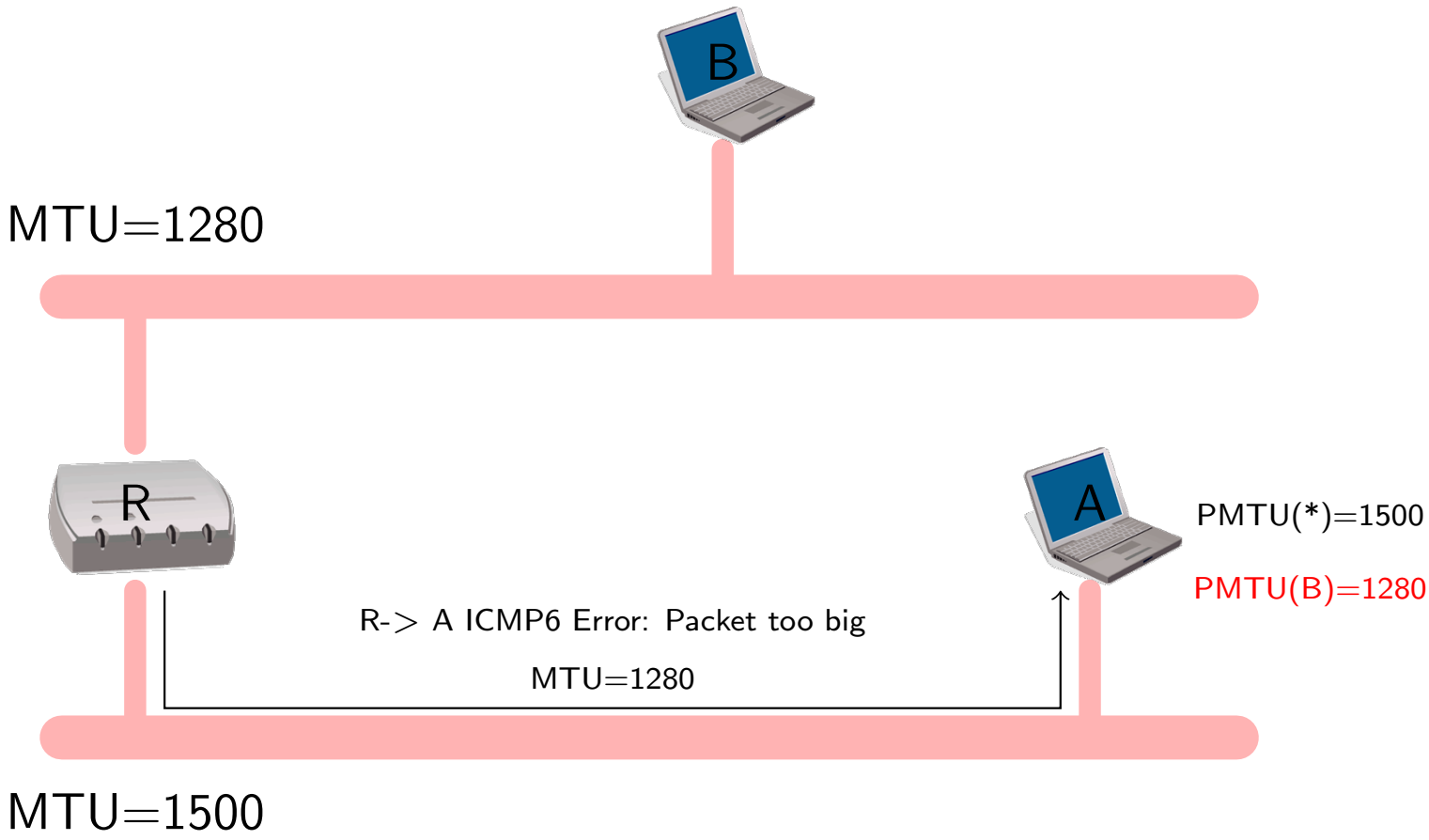
Path MTU discovery for IPv6 (RFC 1981)



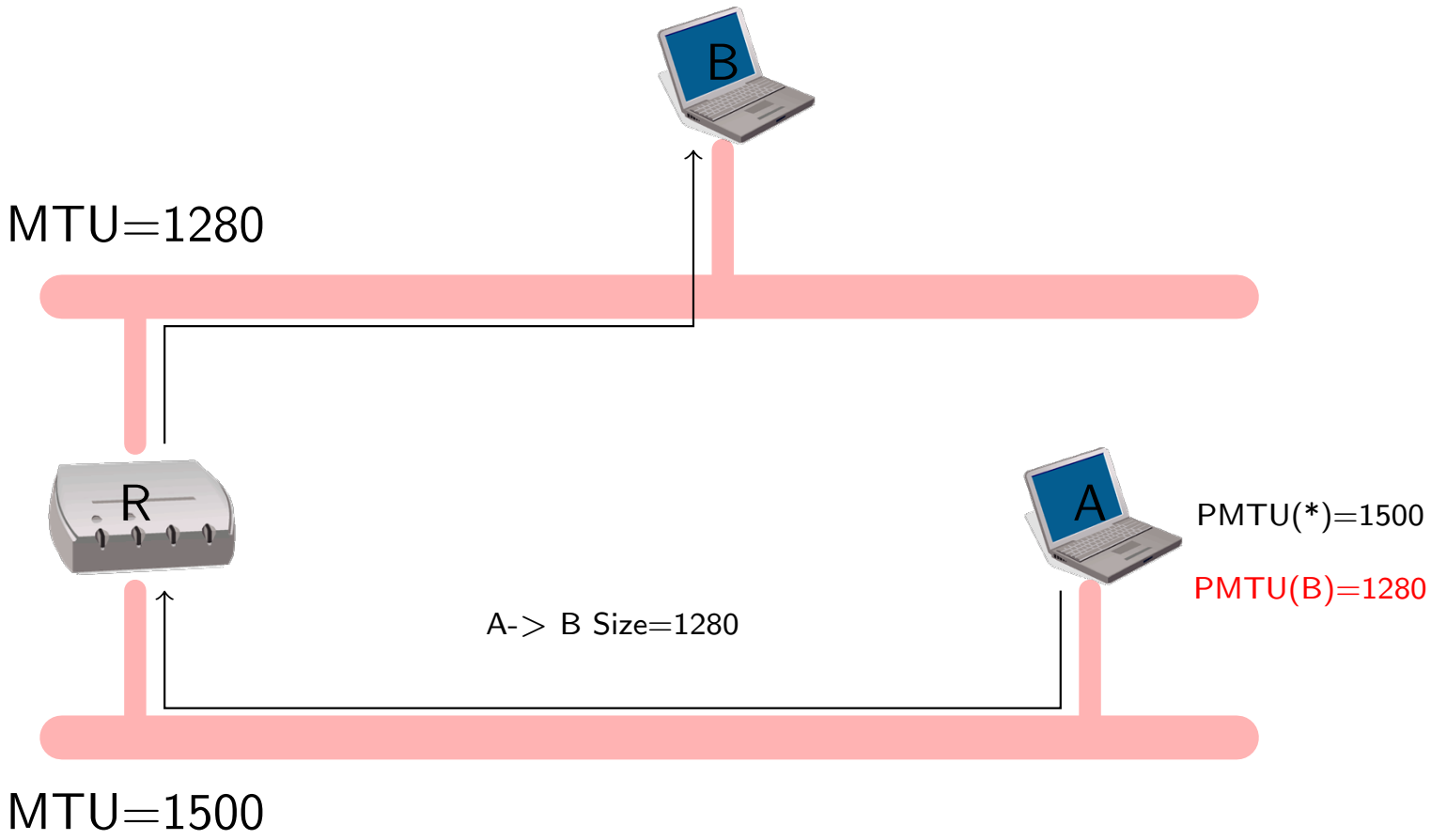
Path MTU discovery for IPv6 (RFC 1981)



Path MTU discovery for IPv6 (RFC 1981)

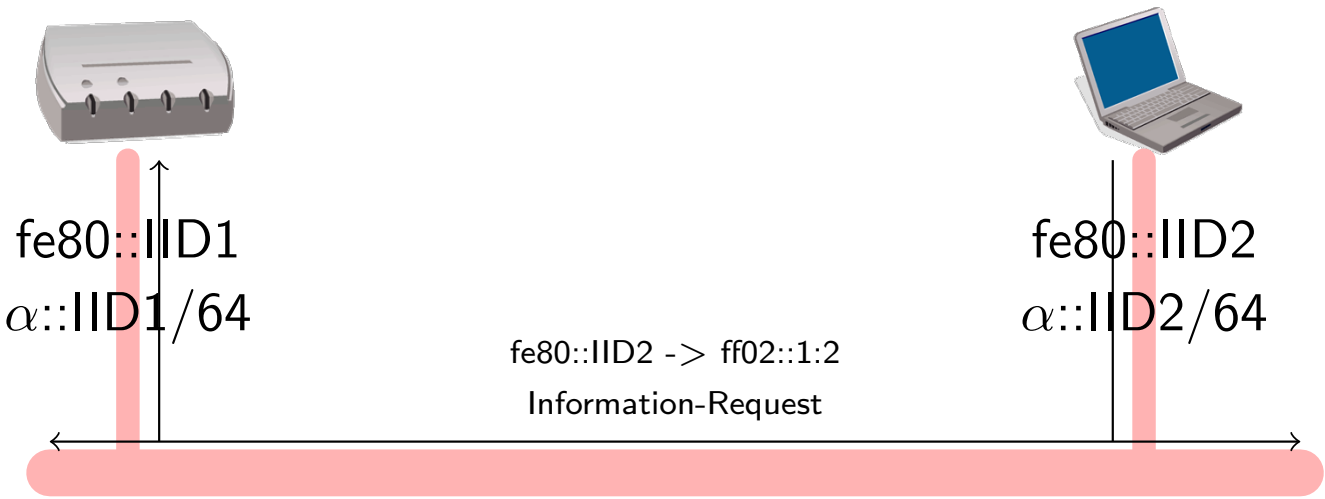


Path MTU discovery for IPv6 (RFC 1981)



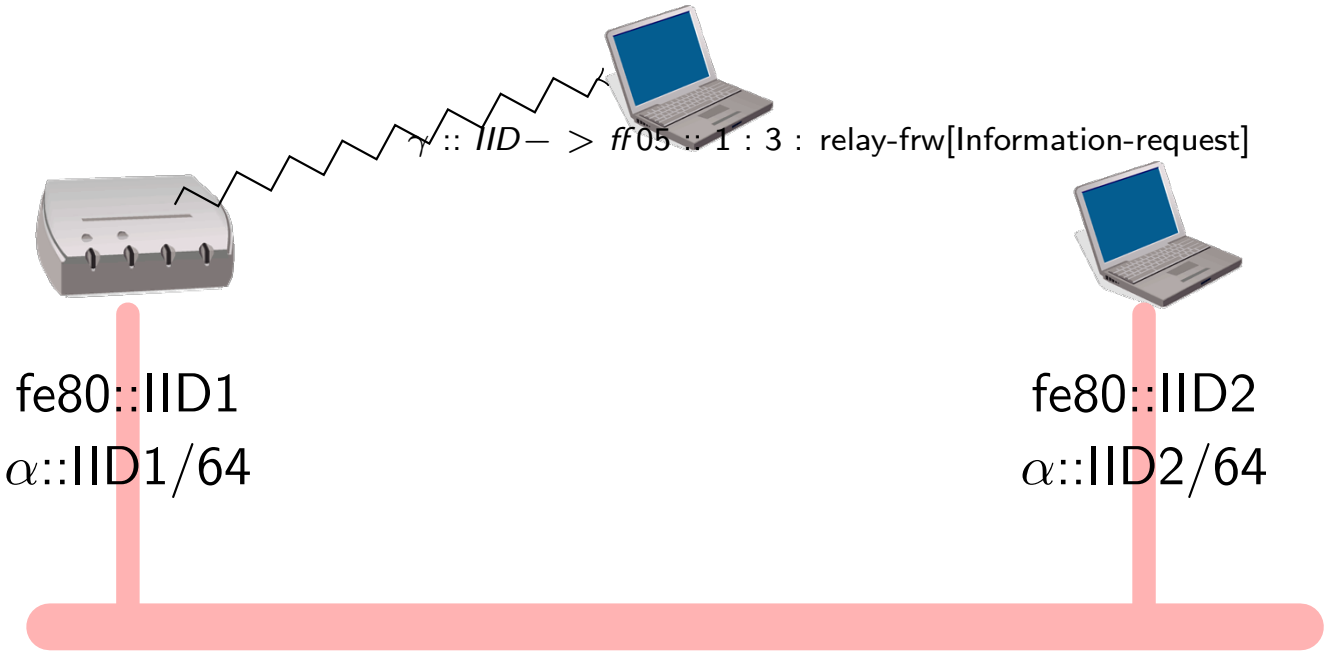
Associated Protocols & Mechanisms
DHCPv6

Stateless DHCPv6 (RFC 3736): With static parameters



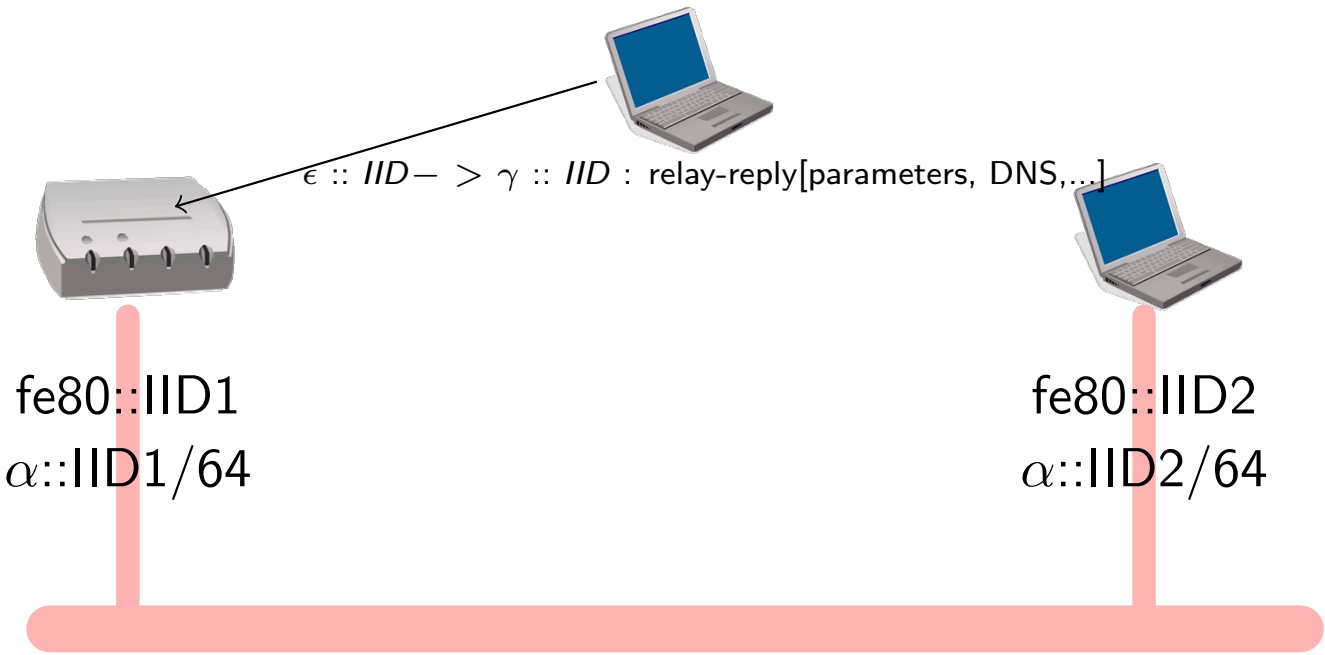
Host needs only static parameters (DNS, NTP,...). It sends an Information-Request message to All_DHCP_Agents multicast group. The scope of this address is link-local.

Stateless DHCPv6 (RFC 3736): With static parameters



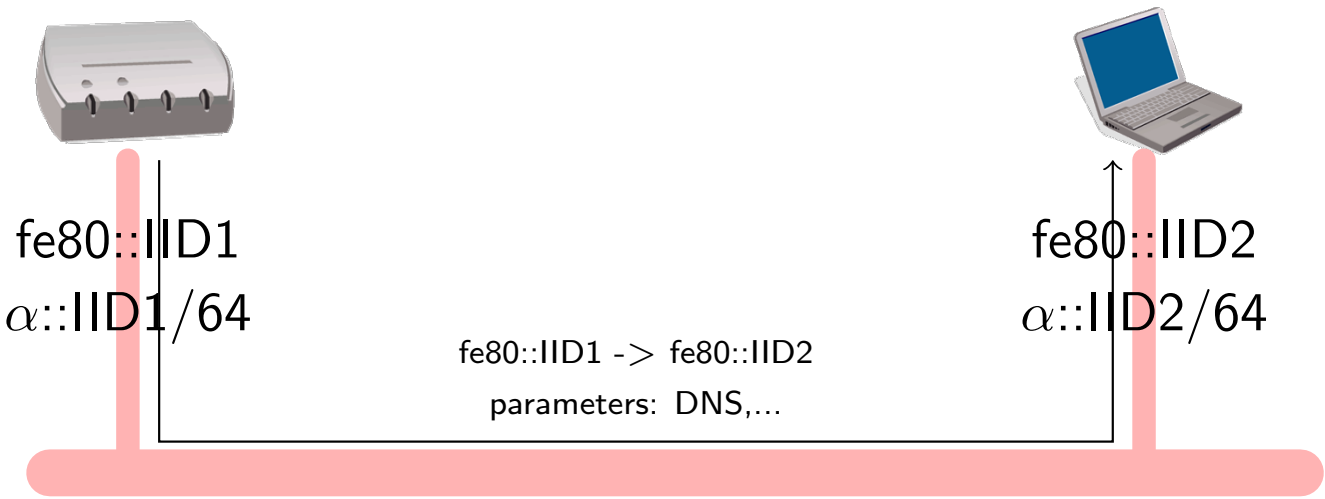
A relay (generally the router) encapsulates the request into a *Forward message and sends it either to the All_DHCP_Servers site-local multicast group or to a list of pre-defined unicast addresses.*

Stateless DHCPv6 (RFC 3736): With static parameters



The server responds to the relay

Stateless DHCPv6 (RFC 3736): With static parameters

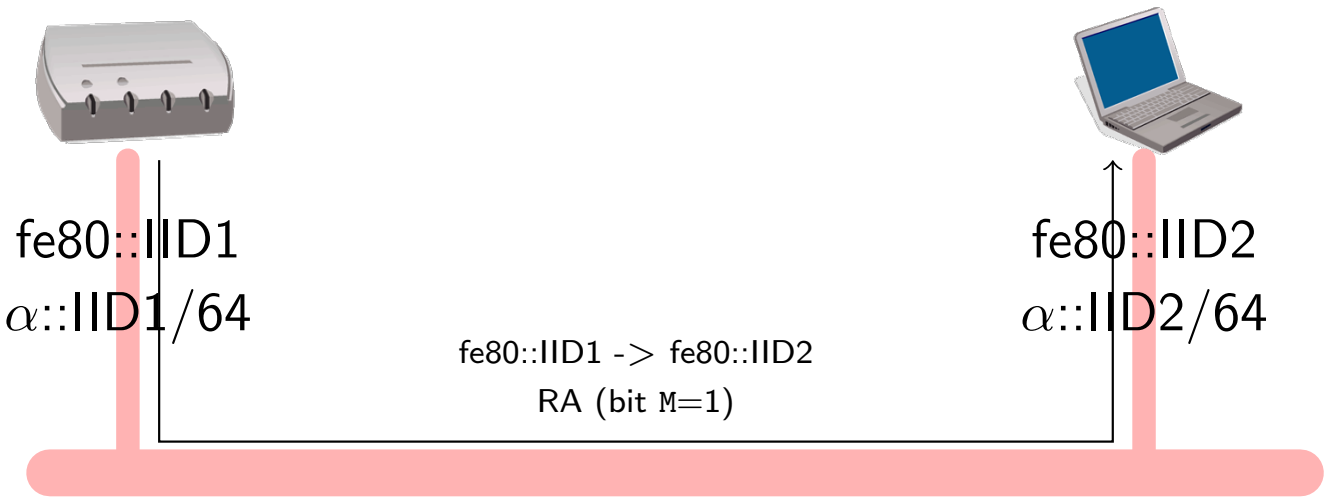


The router extracts information from the message to create answer and sends information to the host

Stateless DHCPv6 (RFC 3736): With static parameters

Host is now configured to resolve domain names through the DNS

DHCPv6 : Stateful Auto-Configuration

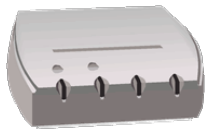


Router responds to RS with a RA message with bit `M` set to 1. Host should request its IPv6 address from a DHCPv6 server.

DHCPv6 : Prefix Delegation

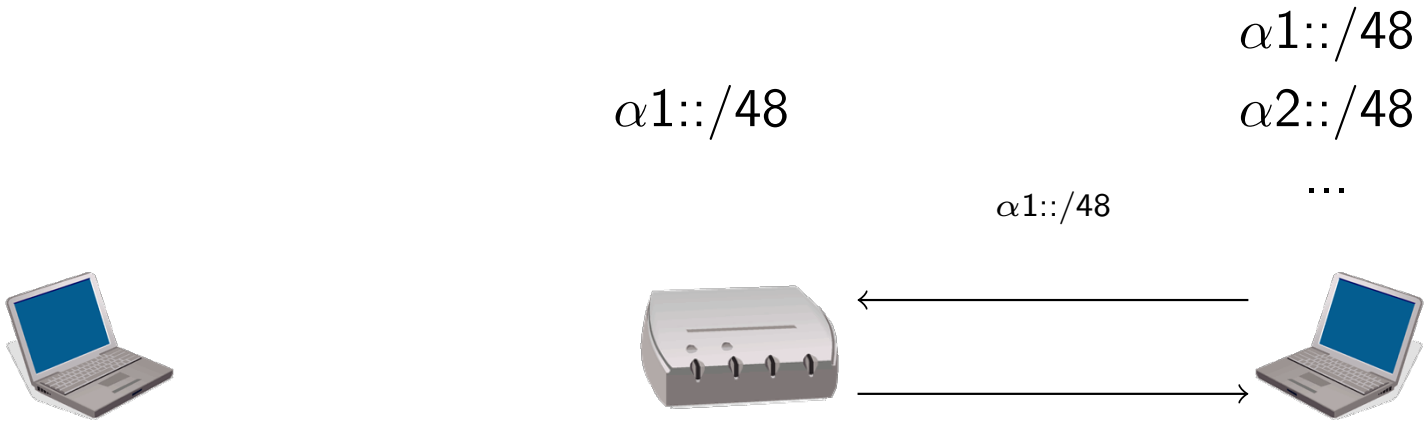
- Dynamic configuration for routers
- ISP solution to delegate prefixes over the network

$\alpha 1::/48$
$\alpha 2::/48$
...

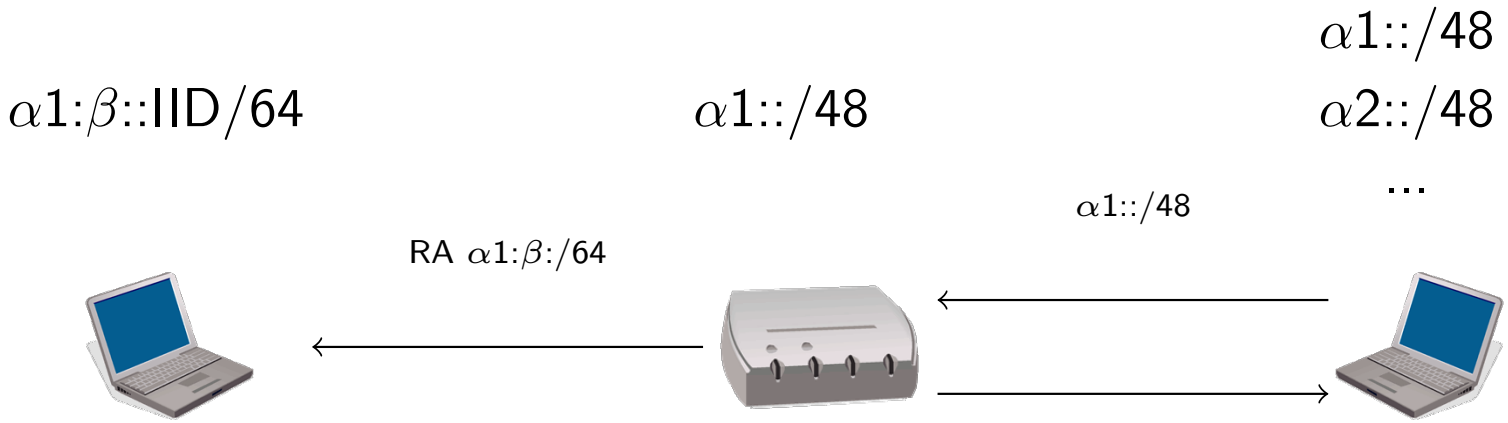

DHCPv6 : Prefix Delegation

- Dynamic configuration for routers
- ISP solution to delegate prefixes over the network



DHCPv6 : Prefix Delegation

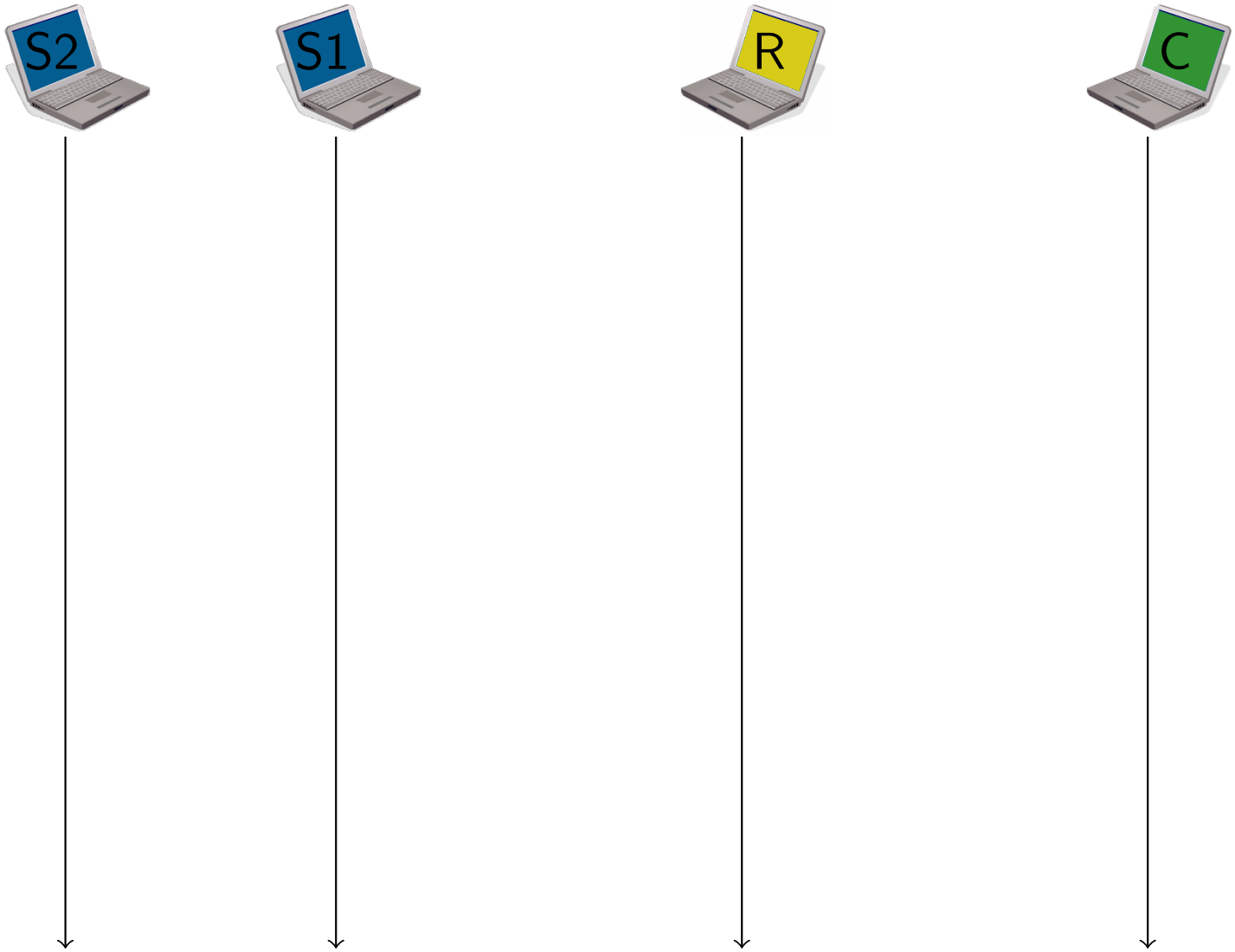
- Dynamic configuration for routers
- ISP solution to delegate prefixes over the network



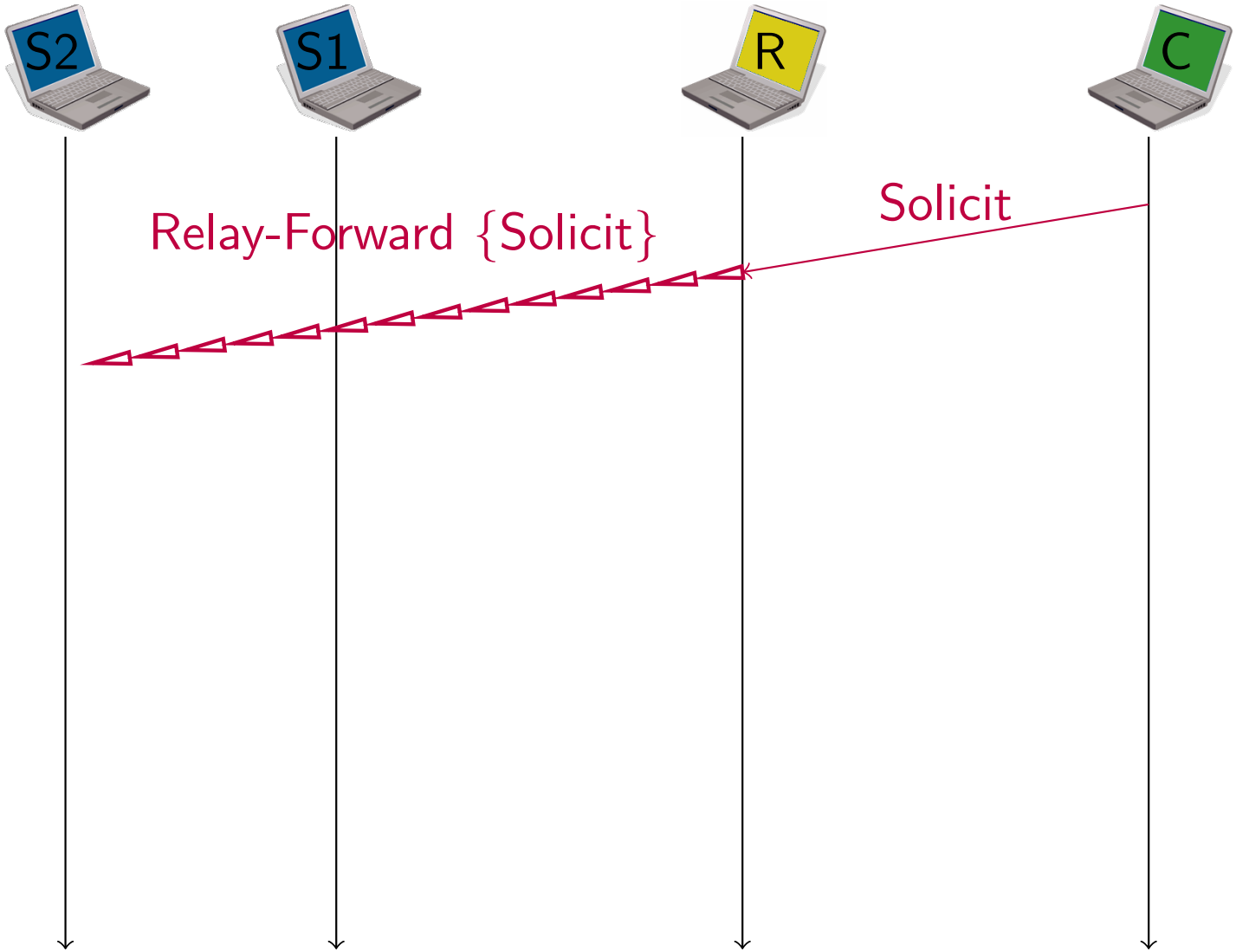
DHCPv6 Full Features

- For address or prefix allocation information from only one DHCPv6 must be taken into account. Four message exchange :
  - Solicit : send by clients to locate servers
  - Advertise : send by servers to indicate services available
  - Request : send by client to a specific server (could be through relays)
  - Reply : send by server with parameters requested
- Addresses or Prefixes are allocated for certain period of time
  - Renew : Send by the client tells the server to extend lifetime
  - Rebind : If no answer from renew, the client use rebind to extend lifetime of addresses and update other configuration parameters
  - Reconfigure : Server informs availability of new or update information. Clients can send renew or Information-request
  - Release : Send by the client tells the server the client does not need any longer addresses or prefixes.
  - Decline : to inform server that allocated addresses are already in use on the link

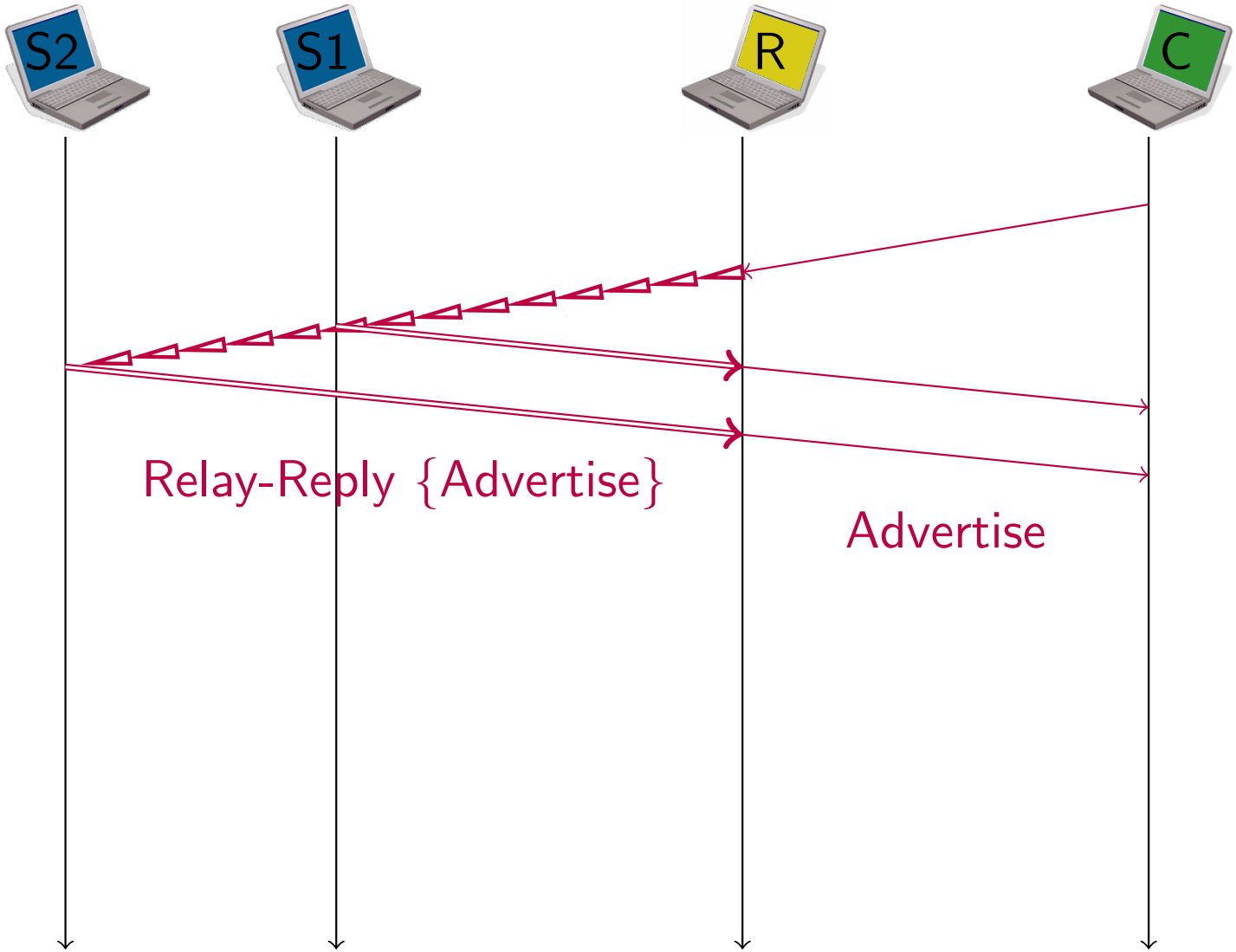
DHCPv6 Scenarii



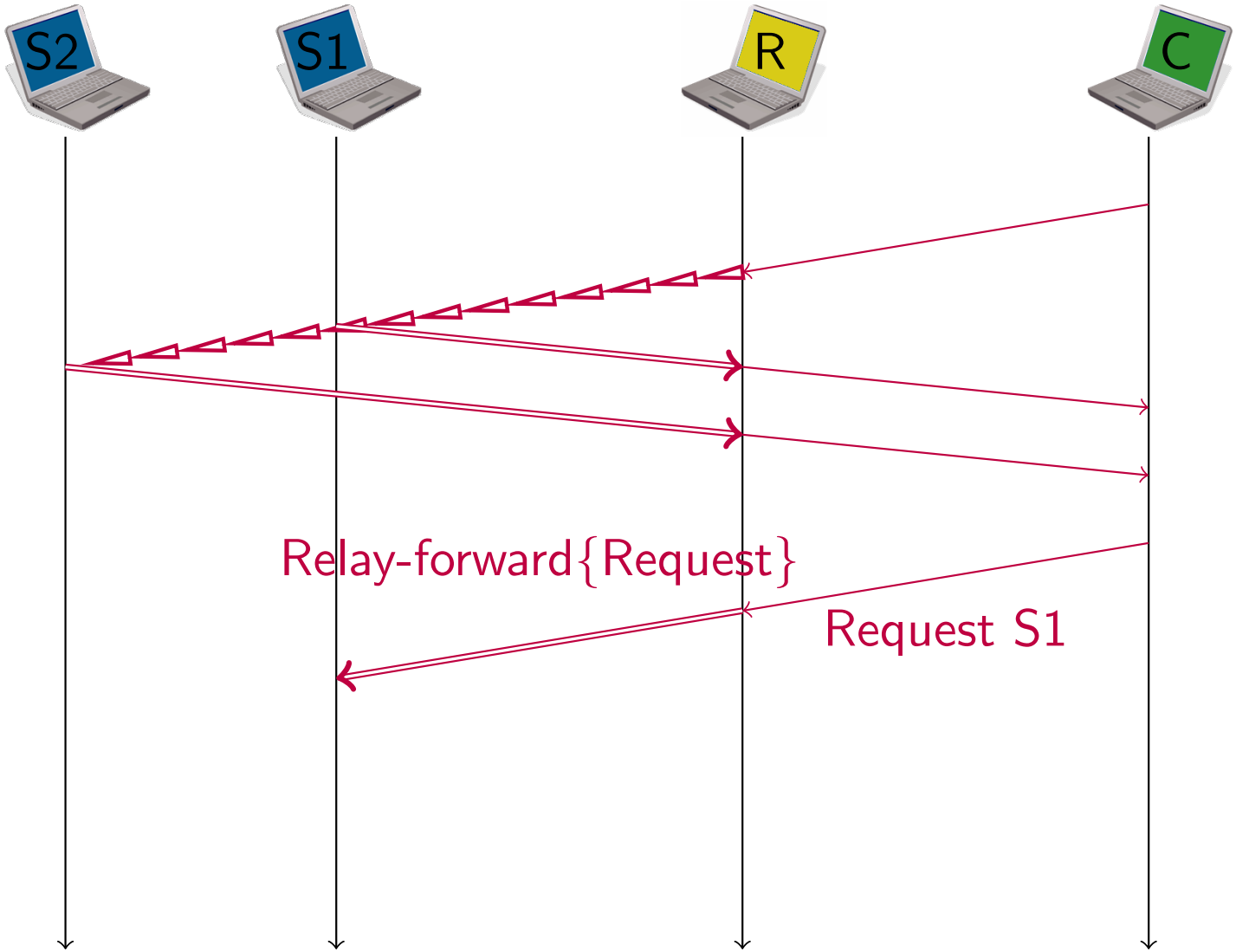
DHCPv6 Scenarii



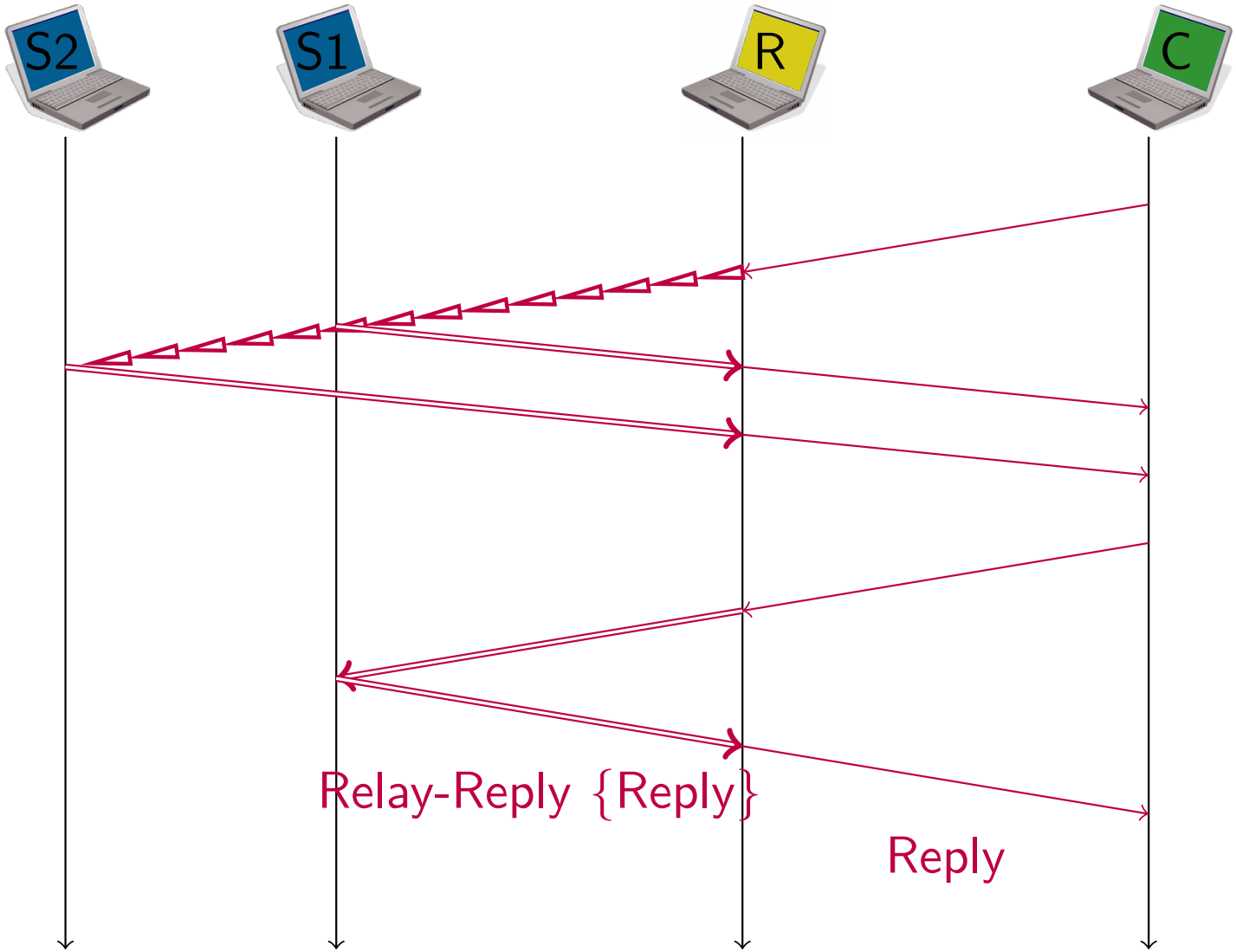
DHCPv6 Scenarii



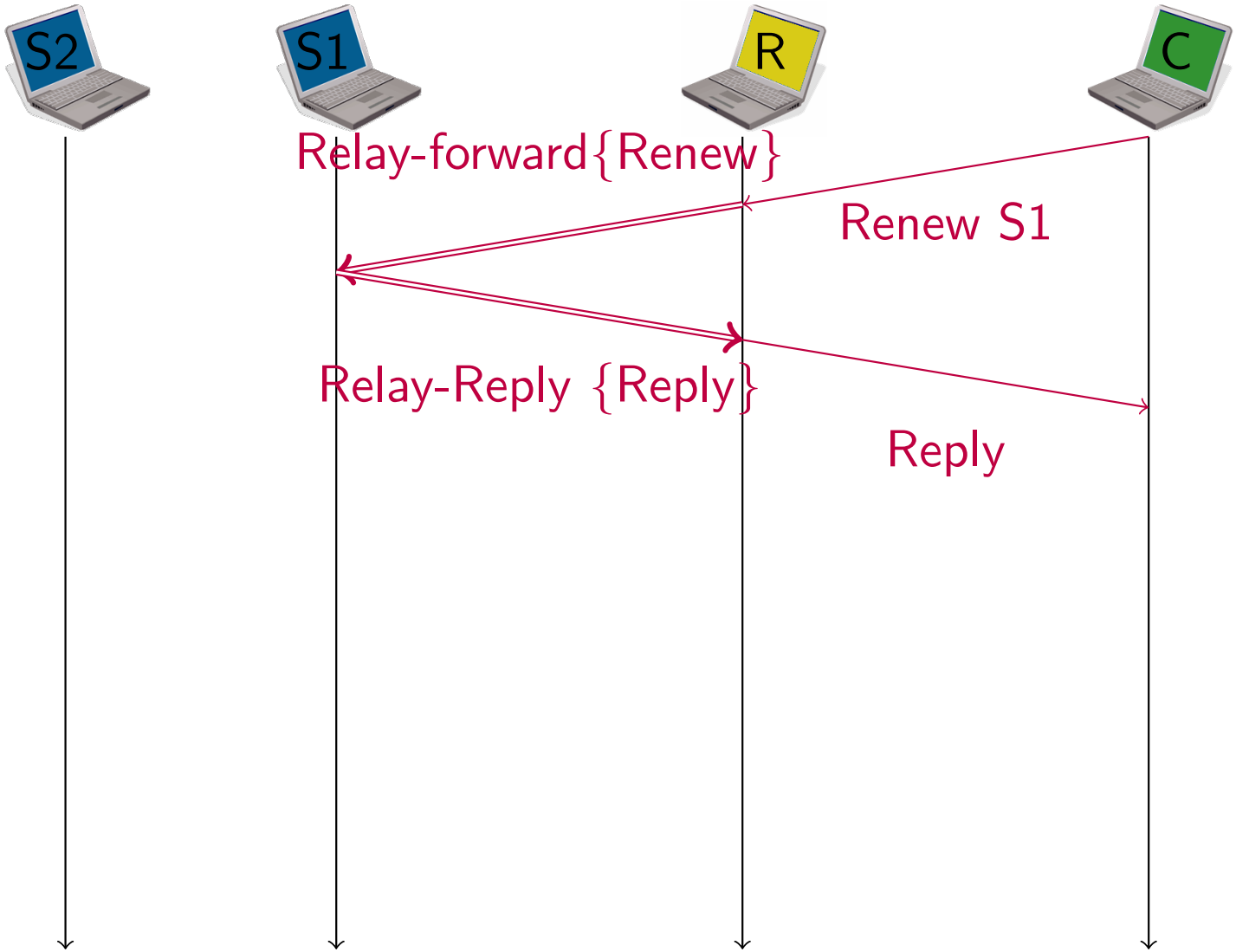
DHCPv6 Scenarii



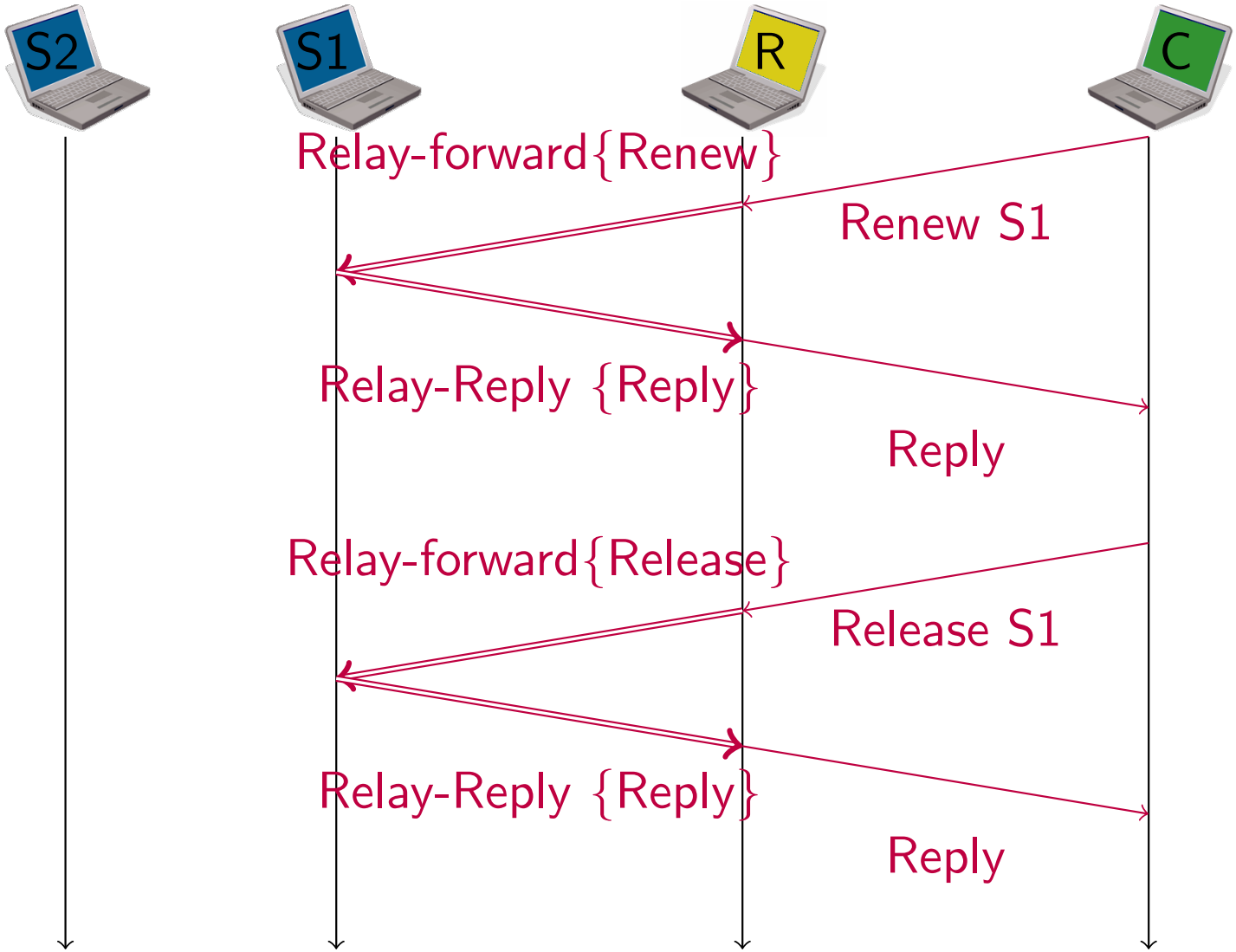
DHCPv6 Scenarii



DHCPv6 Scenarii



DHCPv6 Scenarii



DHCPv6 Identifiers

- DHCPv6 defines several stable identifiers
- After a reboot, the host can get the same information.
- DUID (DHCPv6 Unique IDentifier) :
  - Identify the client
  - Variable length:
    - Link-layer address plus time
    - Vendor-assigned unique ID based on Enterprise Number
    - Link-layer address
- For instance:

```
>od -x /var/db/dhcp6c_duid
0000000 000e 0100 0100 5d0a 5233 0400 9e76 0467
```

DHCPv6 Identifier : IA and IA_PD

- IA and IA_PD are used to link Request and Reply
  - IA is used for Address Allocation and is linked to an Interface
  - IA_PD is used for Prefix Delegation and can be shared among interfaces
- They must be stable (e.g. defined in the configuration file)

Associated Protocols & Mechanisms

Stateless vs Stateful

Auto-configuration: Stateless vs. Stateful

Stateless

Pro:

- Reduce manual configuration
- No server, no state (the router provides all information)

Cons:

- Non-obvious addresses
- No control on addresses on the LAN

Stateful (DHCPv6)

Pro:

- Control of addresses on the LAN
- Control of address format

Cons:

- Requires an extra server
- Still needs RA mechanism
- Clients to be deployed

- Stateless: Typically, for Plug-and-Play networks (Home Network)
- Stateful: Typically, for administrated networks (enterprise, institution)

IPv6 & DNS

Reminder: The two faces of the DNS

The DNS seen as a TCP/IP application

- The service is accessible in either transport modes (UDP/TCP) and over either IP versions (v4/v6)
- If IPv6 transport is not supported yet, then it's highly time!
- *Caution: Information given over either IP version MUST BE CONSISTENT!*

The DNS seen as a database

- Stores different types of resource records (RR), including those related to IPv4 and IPv6 addresses: SOA, NS, A, AAAA, MX, PTR, TXT
- IPv6 nodes & services become visible as soon as their related resources are published in the DNS database
- *Caution: DNS database is IP transport version agnostic!*

DNS Extensions for IPv6 Support (RFC 3596)

Forward lookup ('Name → IPv6 Address')

- A new Resource Record (RR) : **AAAA**
- The "AAAA" RR is for IPv6 what the "A" RR is for IPv4

Example:

```
www.afnic.fr.          IN        A        192.134.4.20
                       IN        AAAA     2001:660:3003:2::4:20
```

Reverse lookup ('IPv6 Address → Name')

- A new and dedicated reverse tree: **ip6.arpa**
- The IPv6 equivalent to the IPv4 dedicated in-addr.arpa tree
- PTRs labels follow a nibble-boundary (4 bits)

Example:

```
0.2.0.0.4.0.0.0.0.0.0.0.0.0.0.0.2.0.0.0.3.0.0.3.0.6.6.0.1.0.0.2.ip6.arpa. PTR www.afnic.fr.
```

Recursive Name Servers Information Discovery

A Stub Resolver needs a Recursive Name Server **address** to which it sends **name resolution** queries

In the IPv4 world, this DNS information is:

- Either configured manually in the stub resolver (e.g. `/etc/resolv.conf for Unix stations`)
- Or discovered via DHCPv4

In the IPv6 world: RFC 4339 (IPv6 Host Configuration of DNS Server Information Approaches)

- Via stateful DHCPv6: RFC 3315
- Via stateless DHCPv6: RFC 3736, "DHCPv6-light"
- RA-based: RFC 6106 ("IPv6 Router Advertisement Options for DNS Configuration", obsoletes RFC 5006)
- Manual configuration as for IPv4
- If IPv4 is supported, than run a DHCPv4 client

DNSv6 Operational Requirements, Recommendations & Issues

RFC 3901: "DNS IPv6 Transport Operational Guidelines"

- For DNS service continuity across a mixture of v4/v6 networks: Recursive Name Servers SHOULD be dual-stack → Use dual-stack forwarders if necessary
- DNS zones SHOULD be served by at least one v4-reachable Authoritative Name Server → Avoid v6-only servers

Bear in mind

- During the long v4-v6 transition period: some systems will stay v4-only, others will be dual-stack and others v6-only

RFC 4472 "Operational Considerations and Issues with IPv6", among others:

- Misbehavior of some DNS servers and Load-balancers
- Handling special (e.g. limited-scope) IPv6-addresses (published vs reachable)
- Service name *vs* Node name
- IPv6 and Dynamic DNS Update (RFC 2136)