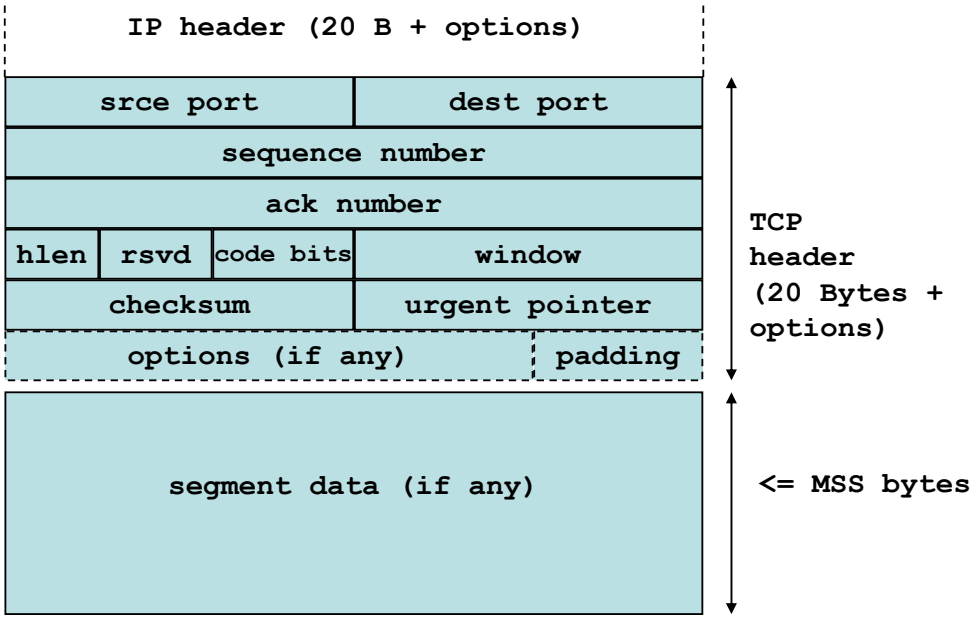
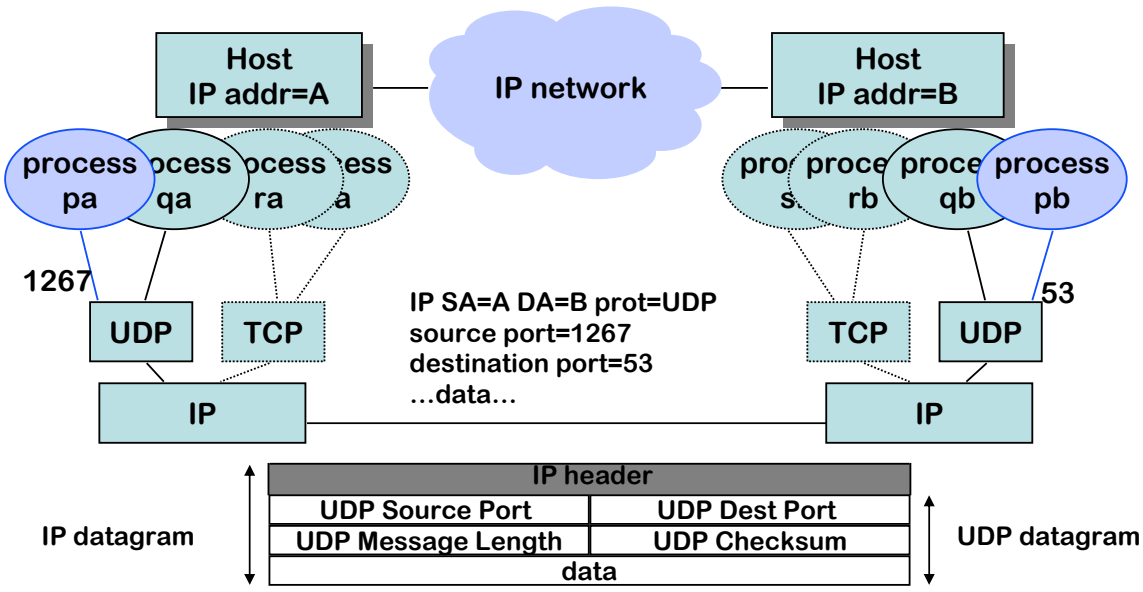
TCP Header

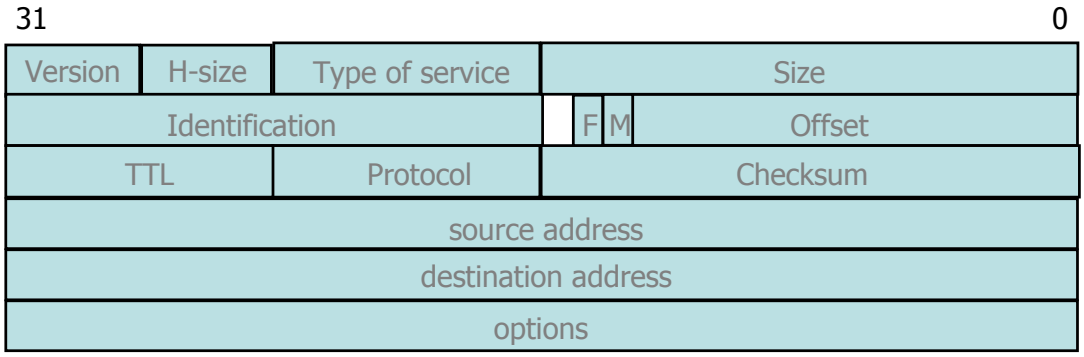


code bit	meaning
urg	urgent ptr is valid
ack	ack field is valid
psh	this seg requests a push
rst	reset the connection
syn	connection setup
fin	sender has reached end of byte stream

UDP Uses Port Numbers



IP header



- Transmitted "big-endian" - bit 31 first
  - Version is always 4 (IPv6 uses a different packet format)
  - Header size
    - options - variable size
    - in 32 bit words

IP header

- Type of service
  - Previously used to encode priority;
  - now used by DiffServ (Differentiated Services)
  - 1 byte codepoint determining QoS class
    - Expedited Forwarding (EF) - minimize delay and jitter
    - Assured Forwarding (AF) - four classes and three drop-precedences (12 codepoints)
  - Used only in corporate networks
- Packet size
  - in bytes including header

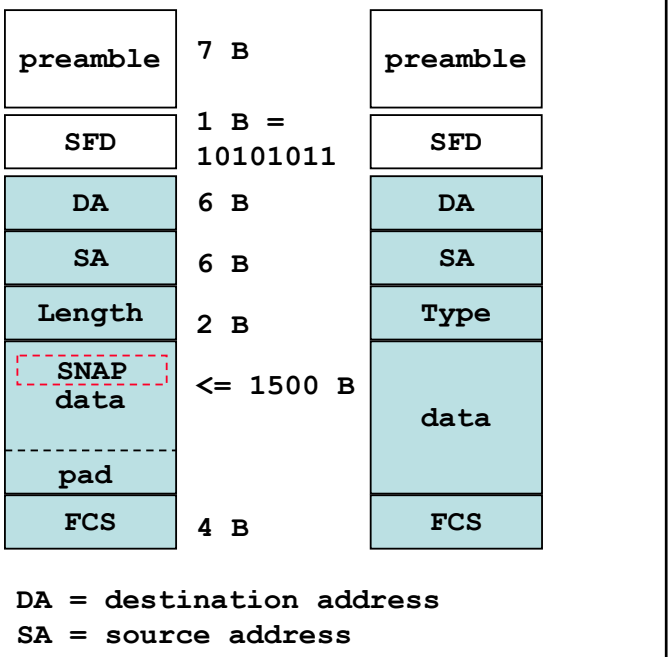
  ≤ 64 Kbytes; limited in practice by link-level MTU (Maximum Transmission Unit)
  - every subnet should forward packets of 576 = 512 + 64 bytes
- Id
  - unique identifier for re-assembling
- Flags
  - M : more ; set in fragments
  - F : prohibits fragmentation
- Offset
  - position of a fragment in multiples of 8 bytes
- TTL (Time-to-live)
  - in seconds
  - now: number of hops
  - router : --, if 0, drop (send ICMP packet to source)
- Protocol
  - identifier of protocol (1 - ICMP, 6 - TCP, 17 - UDP)
- Checksum
  - only on the header

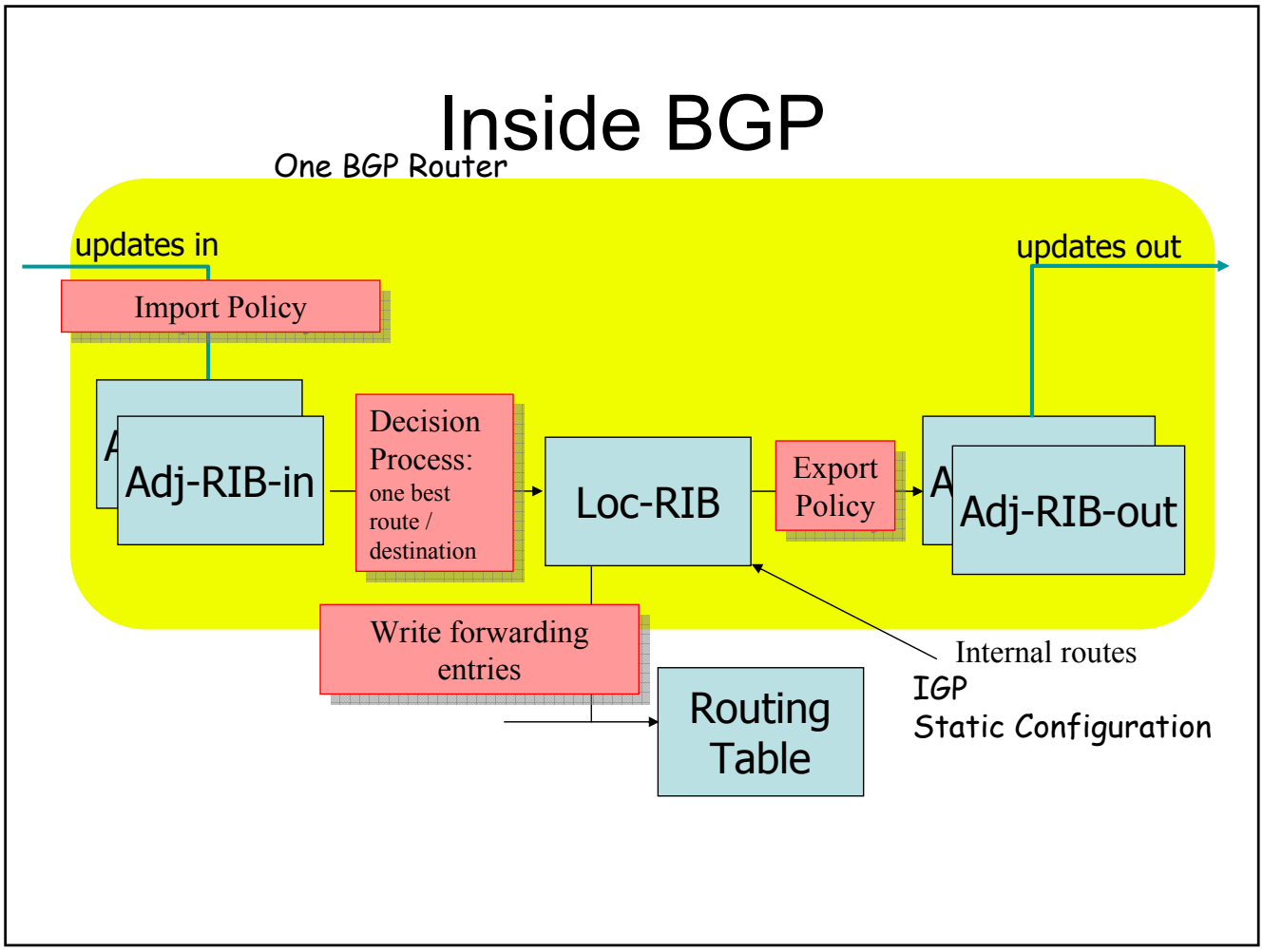
Ethernet / IEEE 802.3 Frame format

- **Ethernet =**
  CSMA/CD with exponential backoff as shown earlier.
- Ethernet PDU is called packet or more often *frame*

Ethernet history

1980 : Ethernet V1.0 (Digital, Intel, Xerox)
1982 : Ethernet V2.0
1985 : IEEE 802.3 standard
small differences in both specifications; adapters today support both
1995 : IEEE 802.3 100Mb/s standard




Inside BGP
One BGP Router
updates in
updates out
Import Policy
Adj-RIB-in
Decision Process: one best route / destination
Loc-RIB
Export Policy
Adj-RIB-out
Write forwarding entries
Routing Table
Internal routes
IGP
Static Configuration

BGP: Routes, RIBs, Routing Table

- The records sent in BGP messages are called **"Routes".** Routes + their attributes are stored in the Adj-RIB-in, Loc-RIB, Adj-RIB-out.
  A route is made of:
  - destination (subnetwork prefix)
  - path to the destination (AS-PATH)
  - Attributes
    - Well-known Mandatory
      - ORIGIN (route learnt from IGP, BGP or static)
      - AS-PATH
      - NEXT-HOP
    - Well-known Discretionary
      - LOCAL-PREF (see later)
      - ATOMIC-AGGREGATE (= route cannot be dis-aggregated)
    - Optional Transitive
      - MULTI-EXIT-DISC (MED)(see later)
      - AGGREGATOR (who aggregated this route)
    - Optional Nontransitive
      - WEIGHT (see later)
- In addition, like any IP host or router, a BGP router also has a **Routing Table** = IP forwarding table
    - Used for packet forwarding, in real time
    - This is not the same as a RIB, we will see the interaction later

BGP: The Decision Process

- The **decision process** decides which route is selected; its output is written into Loc-RIB
- At most one best route to exactly the same prefix is chosen
  - Only one route to 2.2/16 can be chosen
  - But there can be different routes to 2.2.2/24 and 2.2/16
- Routes are compared against each other using the following sequence of criteria, until only one route remains:
  0. Highest weight (Cisco proprietary)
  1. Highest LOCAL-PREF
  2. Shortest AS-PATH
  3. Lowest MED, if taken seriously by this network
  4. E-BGP > I-BGP
  5. Shortest path to NEXT-HOP, according to IGP
  6. Lowest BGP identifier

Dijkstra's Shortest Path Algorithm

- The nodes are `0...N` and the algorithm computes best paths from node `0`
- `c(i,j)` is the cost of `(i,j)`,
- `pred(i)` is the predecessor of node `i` on the tree `M` being built
- `m(j)` is the distance from node `0` to node `j`.

```
m(0) = 0; M = {0};
for k=1 to N {
   find (i0, j0) that minimizes m(i) + c(i,j),
                     with i in M, j not in M
   m(j0) = m(i0) + c(i0, j0)
   pred(j0) = i0
   M = M ∪ {j0}
}
```

- like Bellman-Ford, works for any min-plus algebra

The Centralized Bellman-Ford Algorithm

- **What:** Given a directed graph with links costs A(i,j), computes the best path from i to j for any couple (i,j).
  - We assume A(i, j) > 0 and A(i,j) = ∞ when i and j are not connected.
- **How**: Take for example j=1 and let p(i) be the cost of the best path from i to 1.
  - Define $p^k(i)$ as the cost of the best path from i to 1 in at most k hops. Let $p^0(1) = 0$, $p^0(i) = \infty$ for $i \neq 1$.
  - (Bellman Ford, BF1)

$$p^0(1) = 0,\ p^0(i) = \infty \text{ for } i \neq 1$$

for $k = 1, 2, \ldots$ do
$$p^k(i) = \min_{j \neq i} \left[A(i,j) + p^{k-1}(j) \right] \text{ for } i \neq 1$$
$$p^k(1) = 0$$
until $p^k = p^{k-1}$

- **Theorem**
  1. If the network is fully connected, the algorithm stops at the latest for k=n and then $p^k(i)=p(i)$ for all i
  2. The shortest path from $i \neq 1$ to 1 is defined by $pred(i) = \text{Argmin}_{j \neq i} [A(i,j) + p(j)]$.

  Idea of Proof: $p^k(i)$ is the distance from i to 1 in at most k hops.
  Comment: recursion is equivalent to : $p^k(i) = \min\{ \min_{j \neq i,\ j \neq 1} [A(i,j) + p^{k-1}(j)] , A(i,1) \}$

Distributed Bellman Ford

- BF1 can be used in a centralized algorithm to compute p(i) i.e. find the spanning tree. However, this is not its main interest, because there is a better algorithm (Dijkstra) that can be used in a centralized method
- But: it can be distributed, as follows.

Distributed Bellman-Ford Algorithm v1, BFD1

every node, say i, maintains an estimate q(i) of the distance p(i) to some fixed node 1;
initial conditions are arbitrary but q(1)=0 at all steps
from time to time, i sends the new value q(i) to all its neighbours
when node i receives a value $q(j_0)$ from *any neighbour* j_0, it sets $q(j_0)$ to the received value and updates q(i) by recomputing

eq (1) $q(i) := \min_{j \text{ neighbour}} (A(i,j)+q(j))$

if eq (1) causes q(i) to be modified, pred(i) is set to a value of j that achieves the min

Distributed Bellman-Ford, cont'd

- Requires only to remember the best neighbour (pred(i))

Distributed Bellman-Ford Algorithm, version 2 BFD2

every node, say i, maintains an estimate q(i) of the distance p(i) to some fixed node 1; initial conditions are arbitrary but q(1)=0 at all steps

from time to time, i sends its value q(i) to all its neighbours

when node i receives a value $q(j_0)$ from *any neighbour* j_0, it sets $q(j_0)$ to the received value and updates q(i) by recomputing

eq (2) **if** j_0 == pred(i)
then $q(i) := A(i,j_0)+q(j_0)$
else $q(i) := \min \{ A(i,j_0) + q(j_0), q(i) \}$

if eq (2) causes q(i) to be modified, pred(i) is set to j_0

Fairness of TCP

- A: TCP tends to distribute rate so as to maximize utility of source given by

$$\frac{\sqrt{2}}{\tau_i} \arctan \frac{x_i \tau_i}{\sqrt{2}}$$

with x_i = rate, τ_i = RTT for source i

13

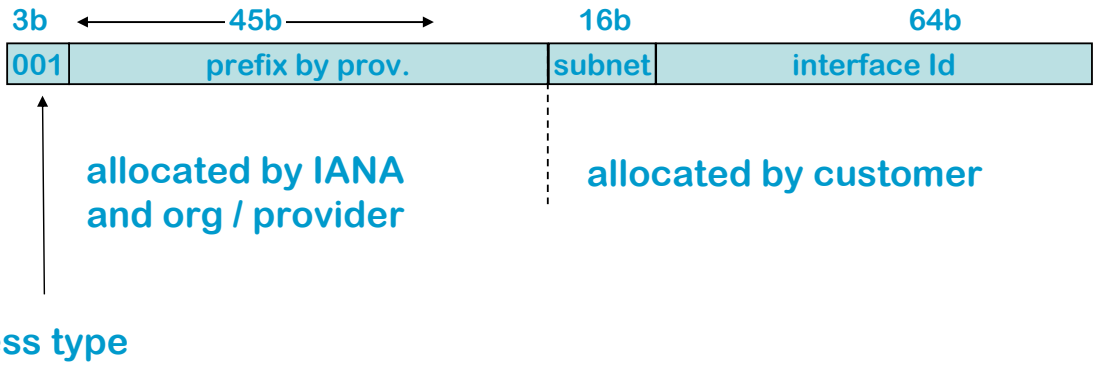
TCP Loss - Throughput Formula

$$\theta = \frac{LC}{T\sqrt{q}}$$

- TCP connection with
  - RTT *T*
  - segment size *L*
  - average packet loss ratio *q*
  - constant *C* = 1.22
- Transmission time negligible compared to RTT, losses are rare, time spent in Slow Start and Fast Recovery negligible

14

IPv6 Addresses



15

IPv6 Addresses: Notation

- IPv6 address is 16B = 128 bits
- Notations: 1 *piece* = 16 bits = [0-4]hexa digits; pieces separated by ":"
  :: replaces any number of 0s; appears only once in address
- Examples
  2001:80b2:9c26:0:800:2078:30f9
  permanent IPv6 address (allocated 2001 and later)
  2002:80b2:9c26:0:800:2078:30f9
  6to4 IPv6 address of dual stack host with IPv4 address 128.178.156.38 and MAC address 08:00:20:78:30:f9
  0:0:0:0:0:FFFF:128.178.156.38
  IPv4 mapped address (IPv4 only host)
  ::FFFF:80b2:9c26
  same as previous
  FF02::43
  all NTP servers on this LAN
  0:0:0:0:0:0:0:0 = :: = unspecified address (absence of address)
- hosts may have several addresses
- addresses are: unicast, anycast or multicast
- url with IPv6 address: use square brackets
  http://[2001:80b2:9c26:0:800:2078:30f9]/index.html

16

From RFC4291, Feb 2006

Address type	Binary prefix	IPv6 notation
Unspecified	00…0 (128 bits)	::/128
Loopback	00…1 (128 bits)	::1/128
Multicast	11111111	FF00::/8
Link-Local unicast	1111111010	FE80::/10
Global Unicast	(everything else)	

17

INTERNET PROTOCOL VERSION 6 ADDRESS SPACE (IANA)

[last updated 27 February 2006]

IPv6 Prefix	Allocation	Reference	Note
0000::/8	Reserved by IETF	[RFC3513]	[1] [5]
0100::/8	Reserved by IETF	[RFC3513]	
0200::/7	Reserved by IETF	[RFC4048]	[2]
0400::/6	Reserved by IETF	[RFC3513]	
0800::/5	Reserved by IETF	[RFC3513]	
1000::/4	Reserved by IETF	[RFC3513]	
2000::/3	Global Unicast	[RFC3513]	[3]
4000::/3	Reserved by IETF	[RFC3513]	
6000::/3	Reserved by IETF	[RFC3513]	
8000::/3	Reserved by IETF	[RFC3513]	
A000::/3	Reserved by IETF	[RFC3513]	
C000::/3	Reserved by IETF	[RFC3513]	
E000::/4	Reserved by IETF	[RFC3513]	
F000::/5	Reserved by IETF	[RFC3513]	
F800::/6	Reserved by IETF	[RFC3513]	
FC00::/7	Unique Local Unicast	[RFC4193]	
FE00::/9	Reserved by IETF	[RFC3513]	
FE80::/10	Link Local Unicast	[RFC3513]	
FEC0::/10	Reserved by IETF	[RFC3879]	[4]
FF00::/8	Multicast	[RFC3513]	

[0] The IPv6 address management function was formally delegated to IANA in December 1995 [RFC1881].

[1] The "unspecified address", the "loopback address", and the IPv6 Addresses with Embedded IPv4 Addresses are assigned out of the 0000::/8 address block.

[2] 0200::/7 was previously defined as an OSI NSAP-mapped prefix set [RFC-gray-rfc1888bis-03.txt]. This definition has been deprecated as of December 2004 [RFC4048].

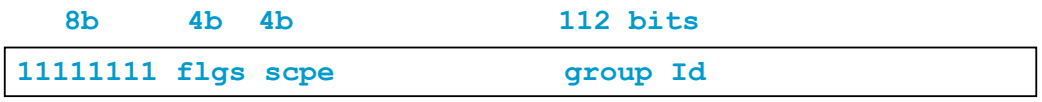
[3] The IPv6 Unicast space encompasses the entire IPv6 address range with the exception of FF00::/8. [RFC3513] IANA unicast address assignments are currently limited to the IPv6 unicast address range of 2000::/3. IANA assignments from this block are registered in the IANA registry: iana-ipv6-unicast-address-assignments.

[4] FEC0::/10 was previously defined as a Site-Local scoped address prefix. This definition has been deprecated as of September 2004 [RFC3879].

[5] 0000::/96 was previously defined as the "IPv4-compatible IPv6 address" prefix. This definition has been deprecated by [RFC4291].

18

IPv6 Multicast Addresses



flgs: *(flags)*=000T T=0: well-known T=1: transient
scpe: *(scope)*
0: reserved 1: node local 2:link local 5: site local
8: org local E: global F: reserved
examples: FF01::43 = all NTP servers on this node
FF02::43 = all NTP servers on this link
FF05::43 = all NTP servers on this site
FF0E::43 = all NTP servers in the Internet

reserved addresses:
FF0x::1 all nodes in the scope (x=1, 2)
FF0x::2 all routers in the scope (x=1, 2)
FF02::1:0 all DHCP servers/relay on this link

solicited node multicast:
FF02::1:XXXX:XXXX
where XXXX:XXXX= lowest order 32 bits of unicast addr.

19

IPv6 Packet Format

IPv6 Header

Viagénie

- IPv4 header = 20 bytes without options

Ver.	header	TOS	total length
identification		flag	fragment offset
TTL	Protocol	Checksum	
32 bits Source Address			
32 bits Destination Address			

removed

changed

IPv6 Header

Viagénie

- IPv6 header = 40 bytes without extentions

Ver.	TrafficClass	Flow Label	
Payload Length		Next Header	Hop Limit
128 bits Source Address			
128 bits Destination Address			

20

6to4 Addresses

- Introduced to support *automatic* tunnels, i.e. without configuration of encapsulator/decapsulator pairs
- Definition: *6to4 address*
  - To any valid IPv4 address *n* we associate the IPv6 prefix 2002:n / 48

    example: the 6to4 address prefix that corresponds to 128.178.156.38 is 2002: 80b2:9c26

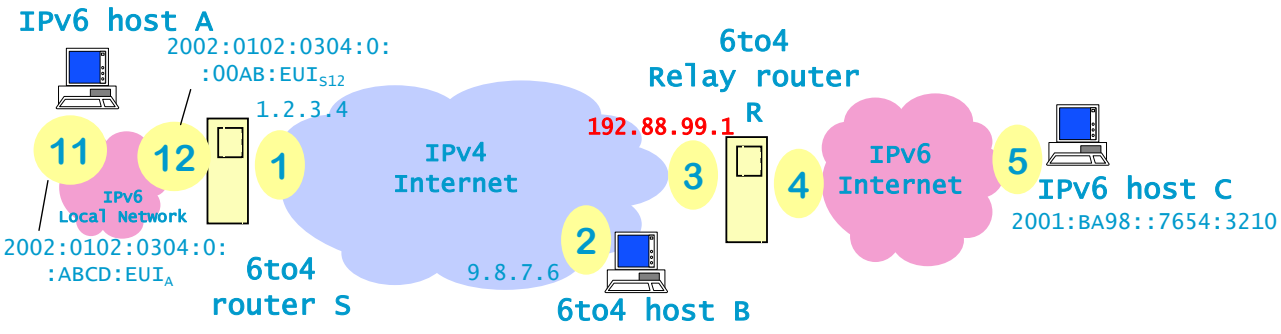
  - An IPv6 address that starts with 2002:… is called a 6to4 address
  - The bits 17 to 48 of a 6to4 address are the corresponding IPv4 address

  - 2002::/16 is the prefix reserved for 6to4 addresses
- A 6to4 host or router is one that is dual stack and uses 6to4 as IPv6 address
- In addition, the IPv4 address 192.88.99.1 is reserved for use in the context of 6to4 addresses (see next slides)

21

6to4 Relay Router and the 192.88.99.1 Anycast Address

- R is a “6to4 relay router”: has both 6to4 interfaces and is both on the IPv4 and IPv6 internets
- All of R’s interfaces on the IPv4 internet have an IPv4 address plus the address 192.88.99.1
- This is a reserved *anycast* address.
  - It is a normal IPv4 address, but there can be several machines with this same address, as there are several relay routers on the Internet.
  - This does not matter: routing protocols continue to work even if we inject the same address at different points – it happens all the time with addresses learnt by BGP.



22