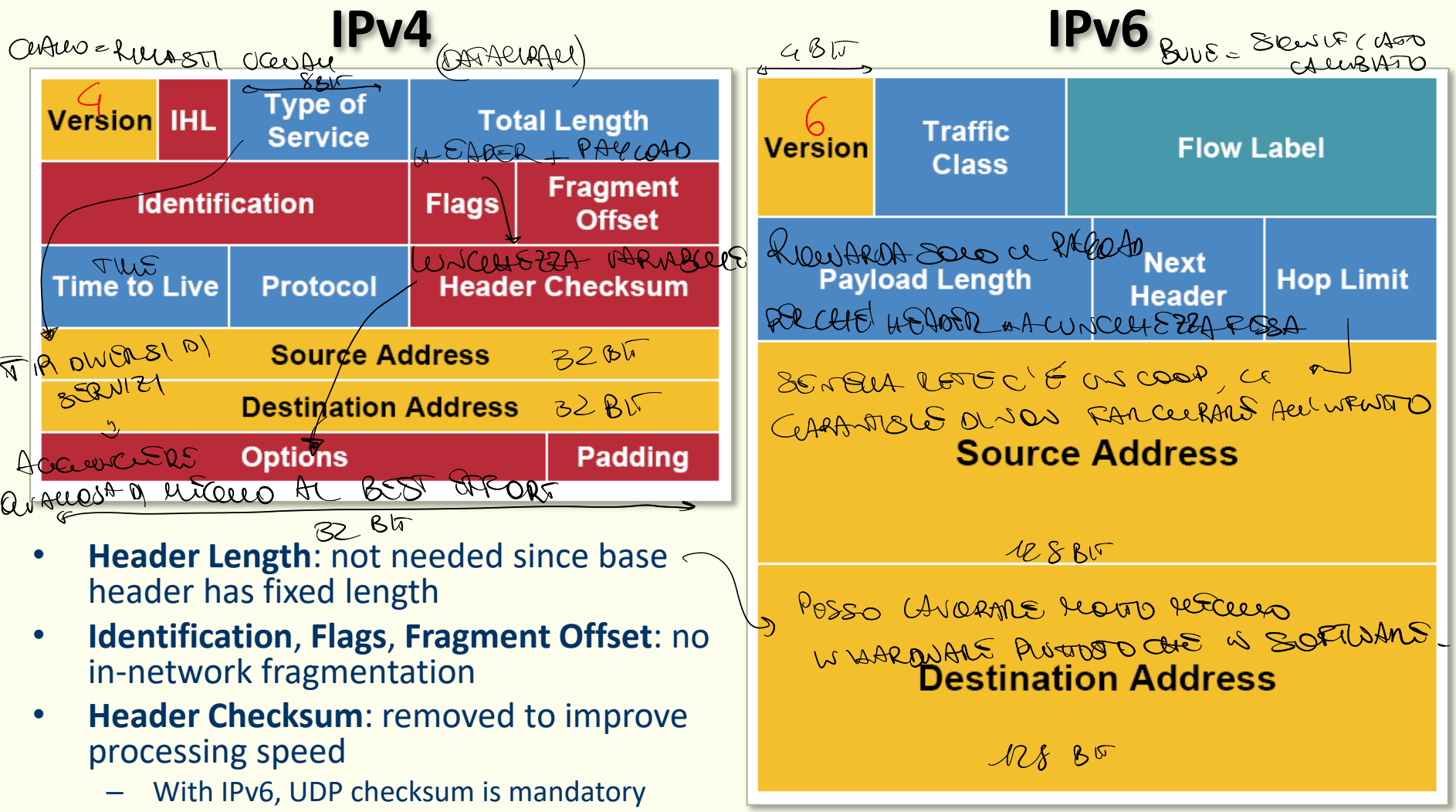
IPv6

Header structure

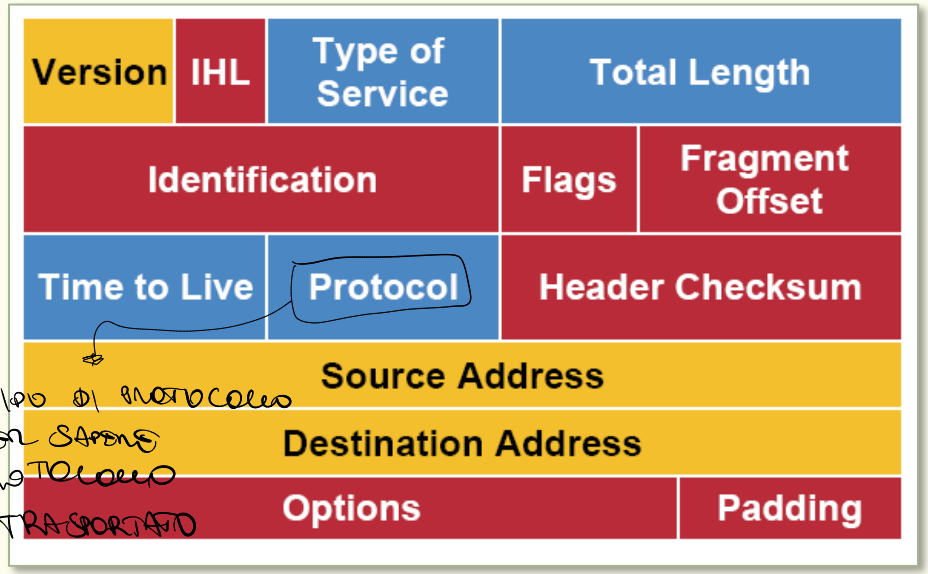
IPv4 vs. IPv6 headers



- **Header Length**: not needed since base header has fixed length
- **Identification, Flags, Fragment Offset**: no in-network fragmentation
- **Header Checksum**: removed to improve processing speed
  - With IPv6, UDP checksum is mandatory

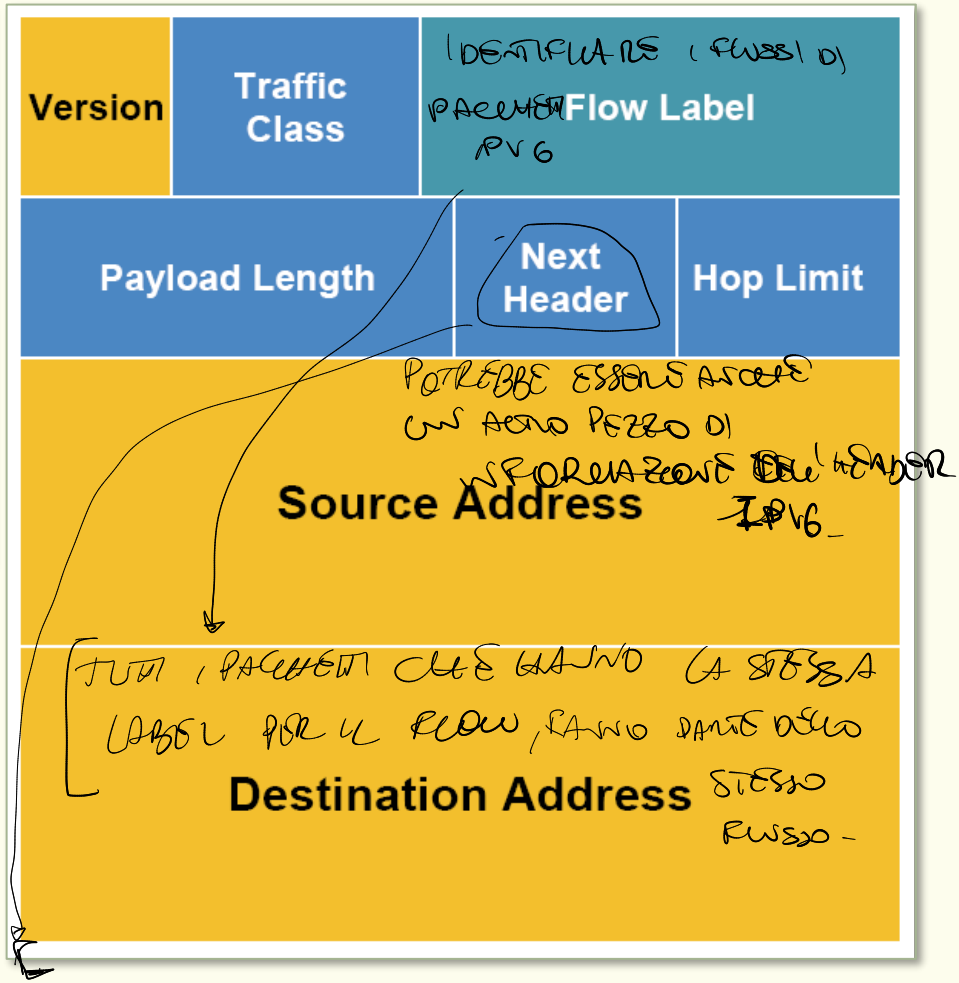
IPv4 vs. IPv6 headers

IPv4

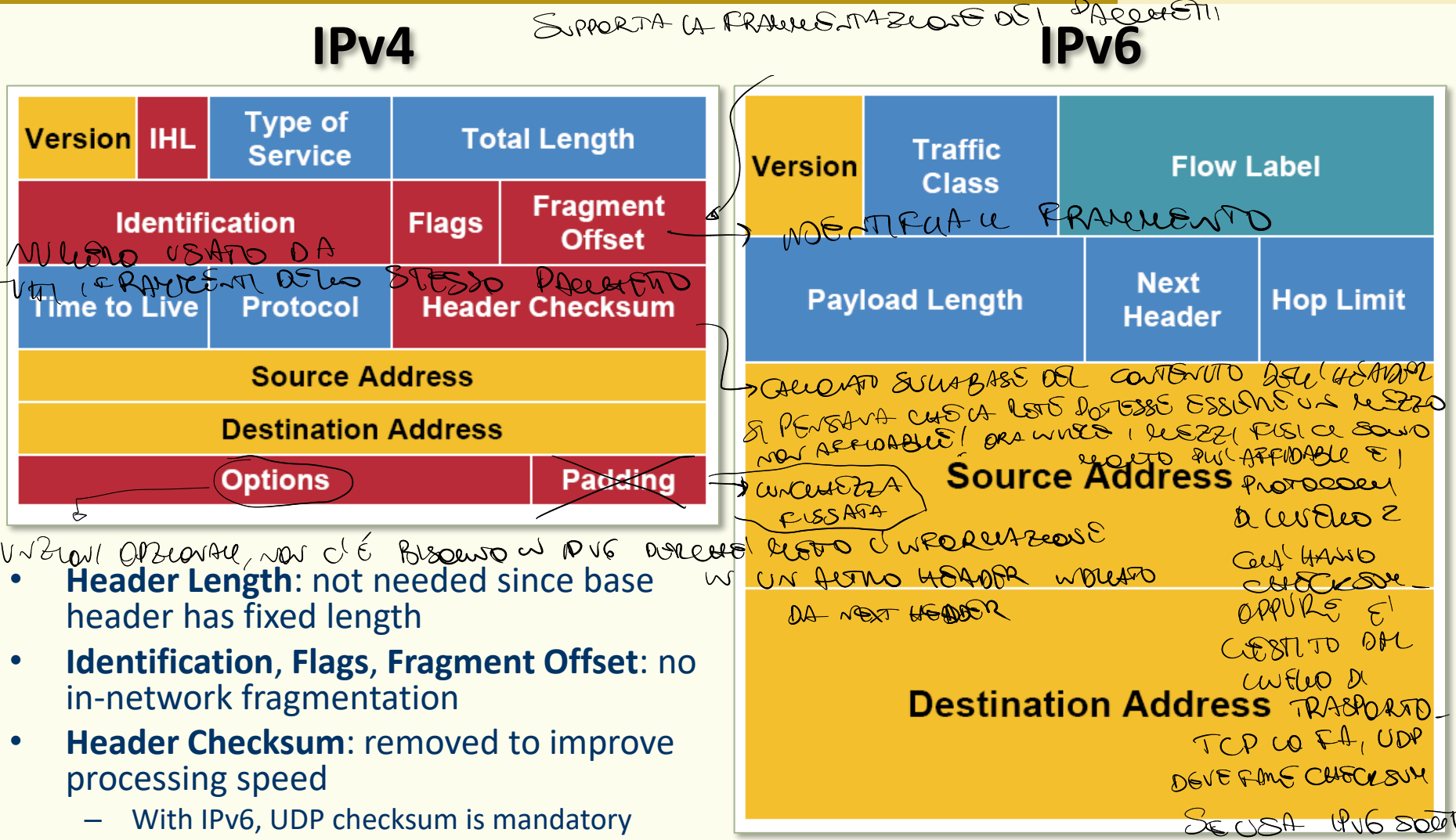


- **Header Length**: not needed since base header has fixed length
- **Identification**, **Flags**, **Fragment Offset**: no in-network fragmentation
- **Header Checksum**: removed to improve processing speed
  - With IPv6, UDP checksum is mandatory

IPv6

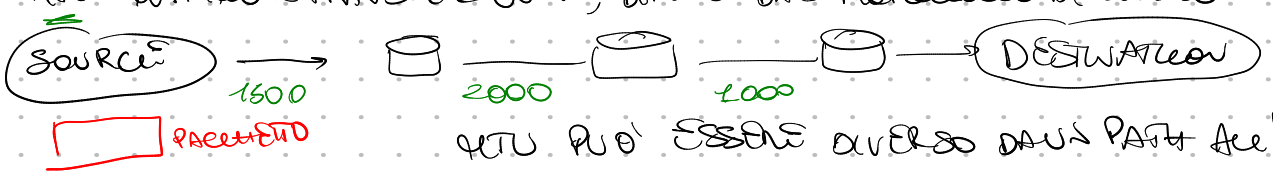


IPv4 vs. IPv6 headers



- **Header Length**: not needed since base header has fixed length
- **Identification**, **Flags**, **Fragment Offset**: no in-network fragmentation
- **Header Checksum**: removed to improve processing speed
  - With IPv6, UDP checksum is mandatory

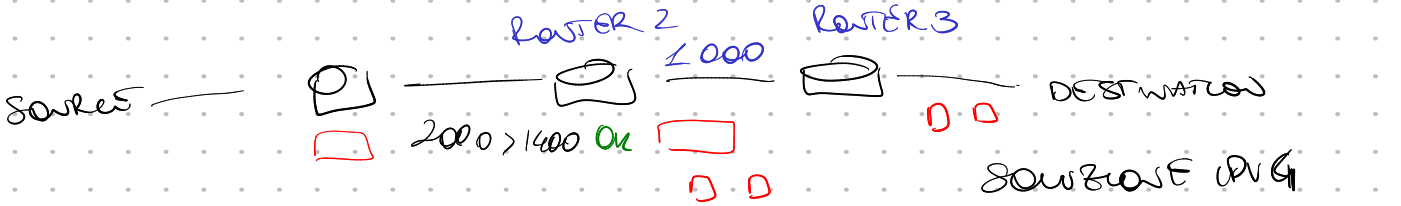
MTU = MAXIMUM TRANSFER UNIT, DIPENDE DAL PROTOCOLLO DI LIVELLO INFERIORE



MTU PUÒ ESSERE DIVERSO DA UN PATH ALL'ALTRO

MANDO IL PACCHETTO AL NEXT HOP, L'UNICO DA CUI CONOSCO MTU -

SE IL PACCHETTO È DI 1400 BYTES => SIZEOF(PACKET) < MTU OK!



SORGENTE IPV4

MTU (ROUTER 2 - ROUTER 3) < SIZEOF(PACKET) => FRAMMENTAZIONE

LA DESTINAZIONE RICEVERÀ I FRAMMENTI E SI OCCUPERÀ DI RICOSTRUIRE IL PACCHETTO

DEVO ESSERE IN GRADO DI RIPRENDERE L'HEADER

FATTO DAL ROUTER 2 IN MODO TRASPARENTE RISPETTO ALLA DESTINAZIONE.

IN IPV6 NON È POSSIBILE FARE LA FRAMMENTAZIONE.

OGGI SI FA FAST FORWARDING => SPOSTO TUTTO QUELLO CHE POSSO IN HARDWARE

LA FRAMMENTAZIONE NON PUÒ ESSERE FATTA IN HARDWARE

IN IPV6 SE SIZEOF(PACKET) > MTU => IL PACCHETTO VIENE SCARTATO

LA DIMENSIONE DEL PACCHETTO VIENE DECISA DALLA SORGENTE

PATH MTU DISCOVERY => DISPONIBILE ANCHE IN IPV4, MA NON VENIVA USATO - QUANDO UN ROUTER RICEVE UN PACCHETTO TROPPO GRANDE E LO SCARTA RISPONDE CON UN PACCHETTO ICMP DI ERRORE CON INCLUSO L'MTU CORRETTO

QUANDO CON QUESTO MTU ARRIVA FRAMMENTO ALLA SORGENTE

PROBLEMI SOLUZIONE IPV6;

1) LATENCY → DEVO ASPETTARE IL RITORNO DEL PACCHETTO ICMP

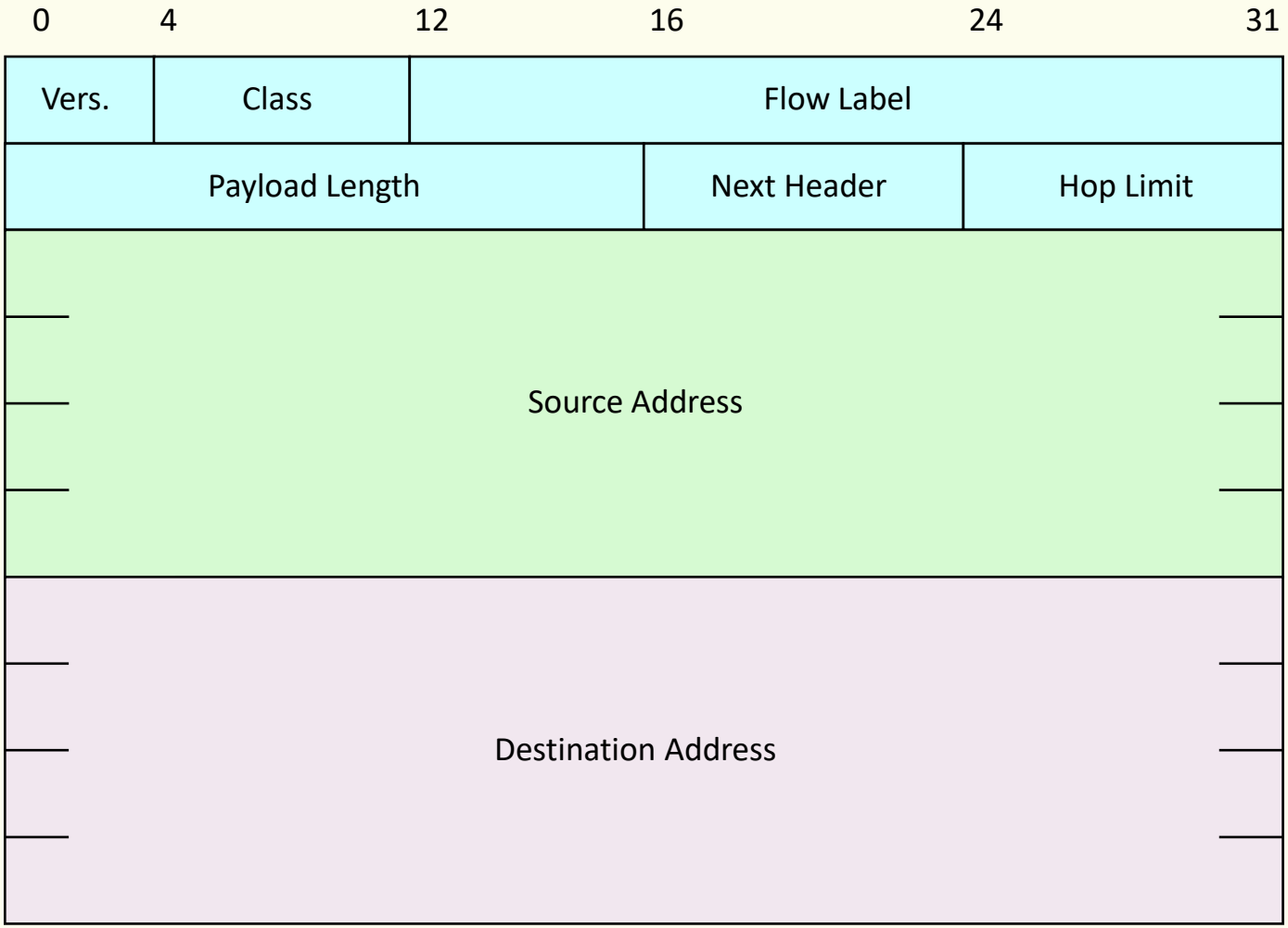
2) SCALABILITY → SE LO DEVO FARE PER OGNI CONN È INFATTIBILE

COME POSSO RISOLVERE?

POSSO USARE UN DEFAULT → SE VUOI ESSERE ON CONN IPV6 NON PUOI AVERE MTU MINORI DI

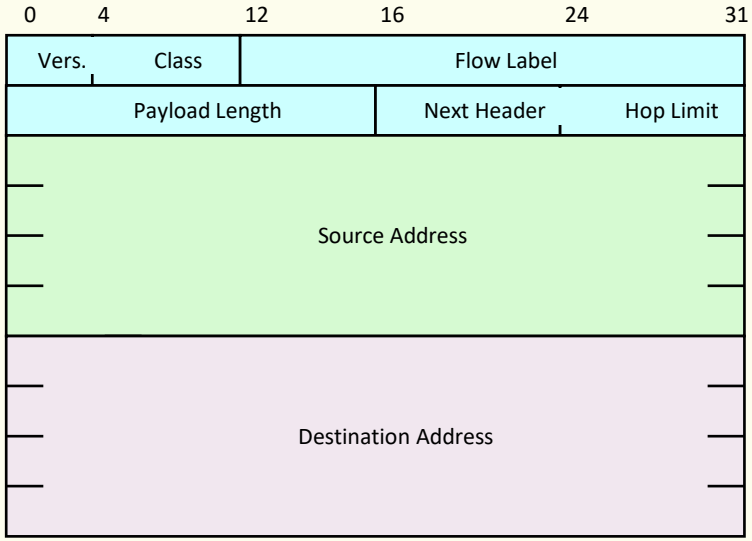
1280

IPv6 base header (40 bytes)



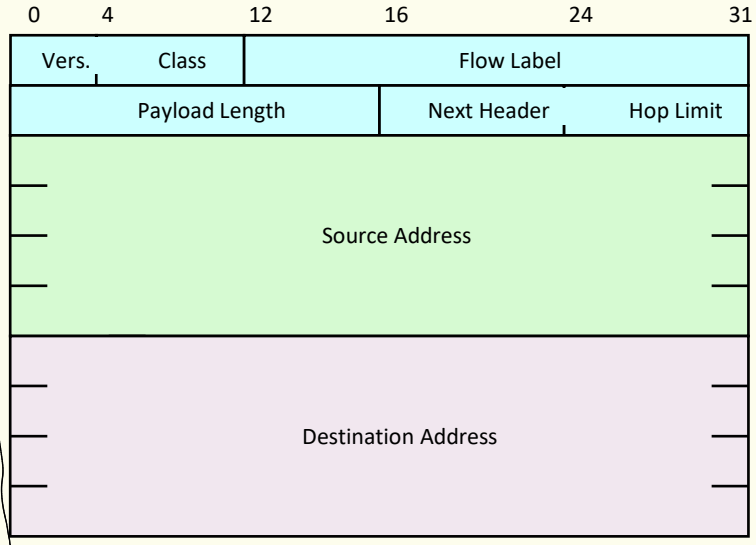
IPv6 base header

- Version (4 bits)
  - 6
- Traffic Class (1 byte)
  - Substitutes the ToS field in IPv4
  - Can be used to give priority to certain packets within a flow, or it can be used to give priority to datagrams from certain applications (for example, ICMP packets) over datagrams from other applications

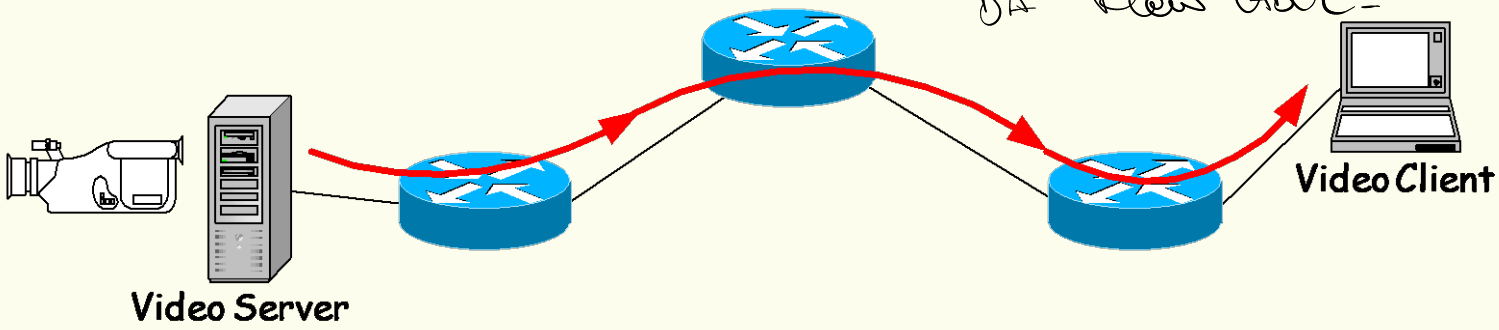


IPv6 base header

- Flow label (20 bits)
  - randomly generated
  - distinguishes packets that require the same treat-ment in order to facilitate the handling of real-time traffic

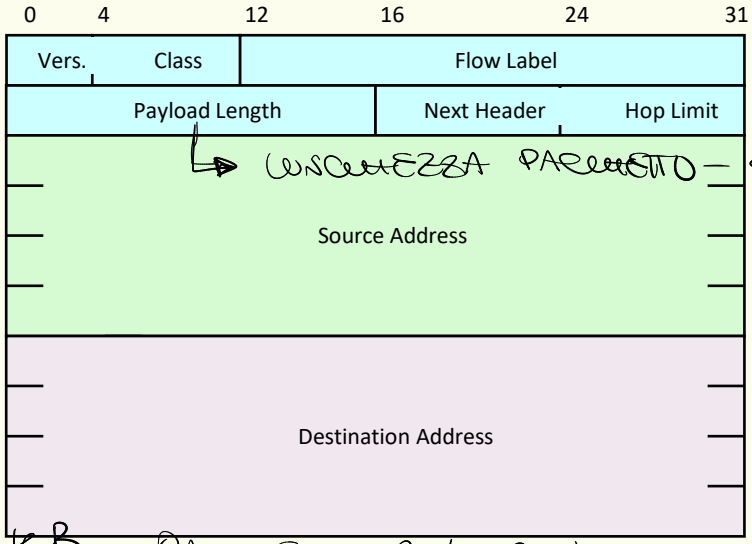


SE DEVE PRENDERE UNA DECISIONE BASATA NON SOLO SU DESTINAZIONE MA ANCHE DA FLOW LABEL.



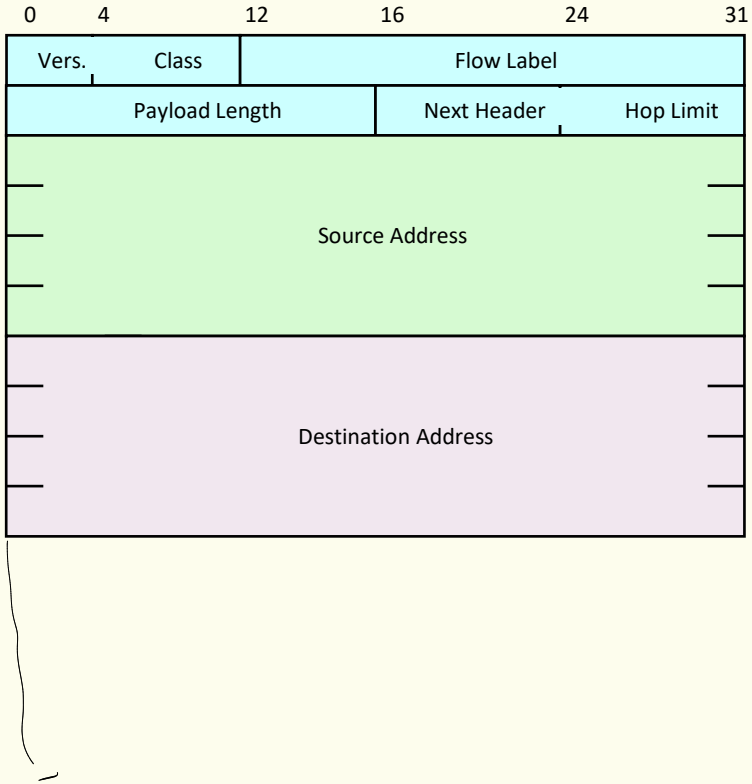
IPv6 base header

- Payload length (2 bytes)
  - Does not include the header length (as in IPv4)
  - Extension headers are considered part of the payload
- Hop limit (1 byte)
  - Analogous to TTL field in IPv4, but no more expressed in seconds



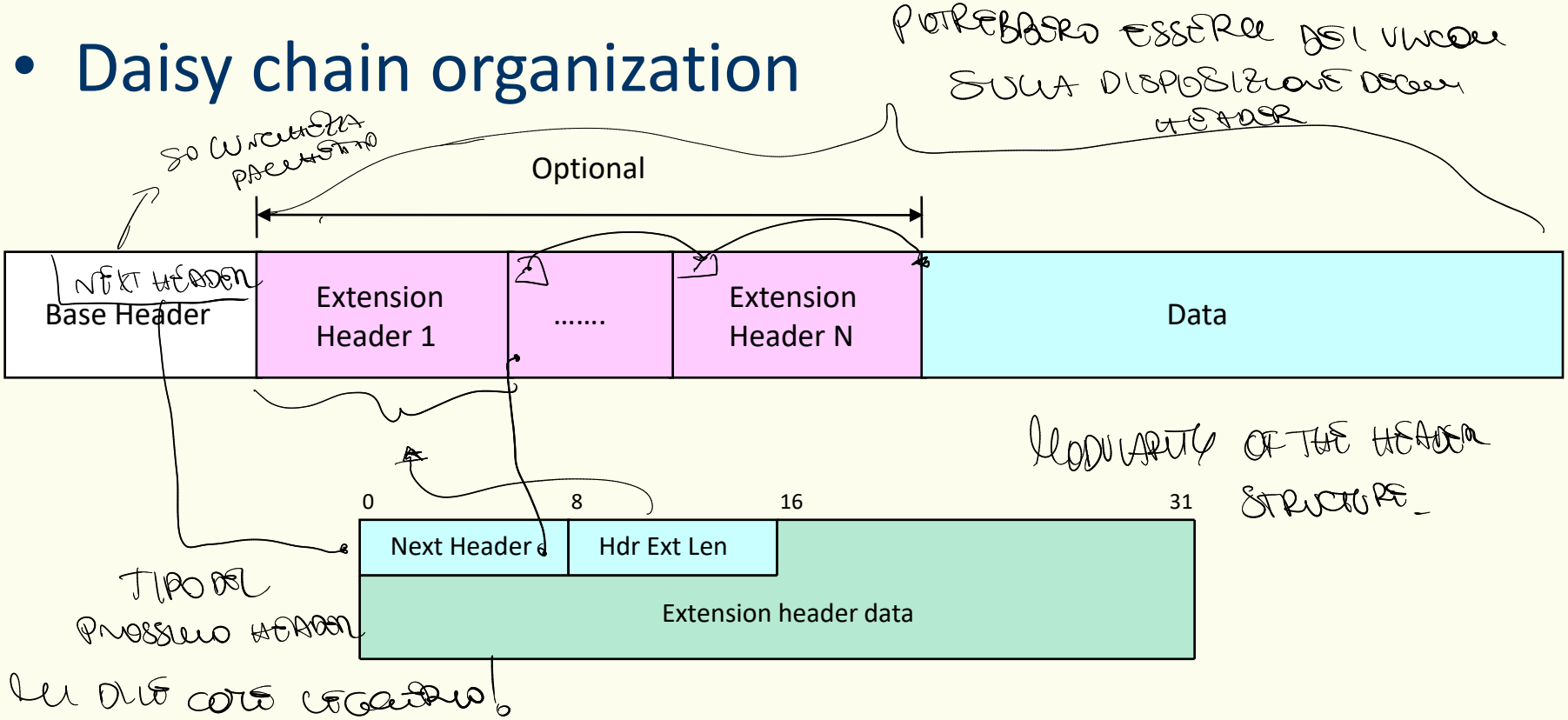
IPv6 base header

- Next header (1 byte)
  - Resembles the Protocol Type in IPv4, but …
  - It is much more, reflecting the new organization of IPv6 packets



IPv6 header structure

- One base header (40 bytes), plus
- Zero or more extension headers (variable size)
- Daisy chain organization



IPv6 header structure

- Values in the Next Header field (base and extension headers)

Values	Meaning
0	Hop-by-Hop Options header
4	IPv4
6	Transmission Control Protocol
17	User Datagram Protocol
41	IPv6
43	Routing header
44	Fragment header
46	Resource Reservation Protocol
50	Encapsulating Security Payload header
51	Authentication header
58	Internet Control Message Protocol v6
59	No Next header
60	Destination Options header
135	Mobility Header (Mobile IPv6)
…	…

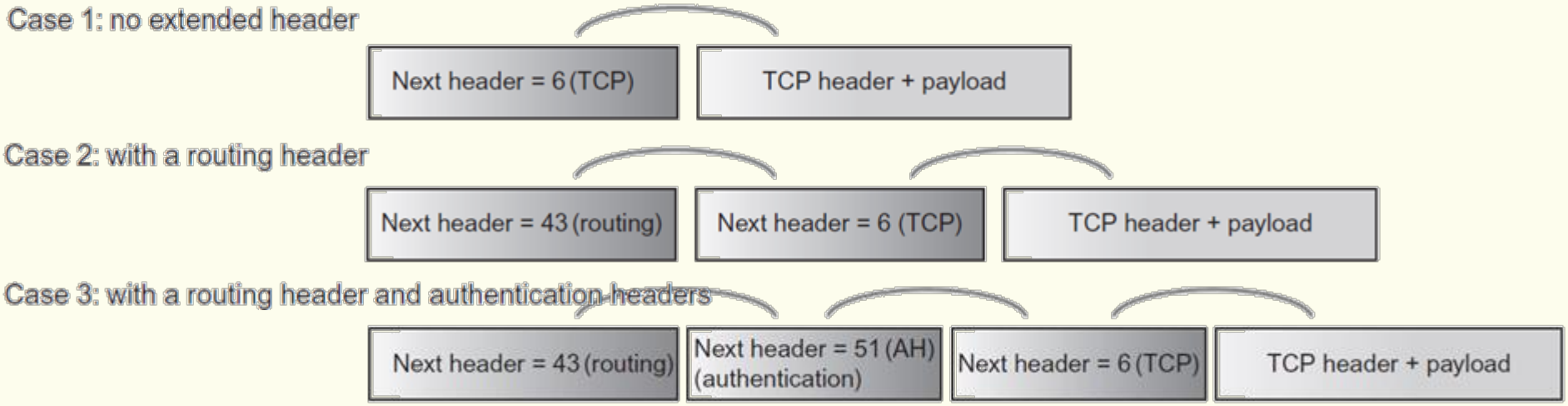
AUMENTA FLESSIBILITÀ, SE VOGLIO AGGIUNGERE UN'ESTENSIONE BASTA DEFINIRE UN NUOVO HEADER E ASSEGNARGLI UN CODICE.

AL POSTO DI IPSEC

IPv6 extension headers

- Inserted only when needed
- Processed in the order they appear by the node identified by the DA (with one exception)
  - A specific order *is recommended*
- Open to further extensions

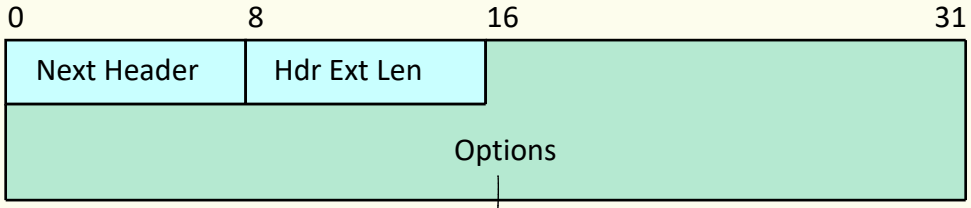
Value	Extension Header
0	Hop-by-Hop Options Header
43	Routing Header
44	Fragment Header
50	ESP Header
51	Authentication Header
59	No Next Header
60	Destination Options Header



Hop-by-Hop Options header

- Carries optional information that must be examined by every node along the path
- Format
  - Next header (1 byte)
  - Header Extension Length (1 byte)
    - Header length in eight-byte units (minus one)
  - Options
    - One or more

SE PRESENTE, TIPICAMENTE E' LA PRIMA!

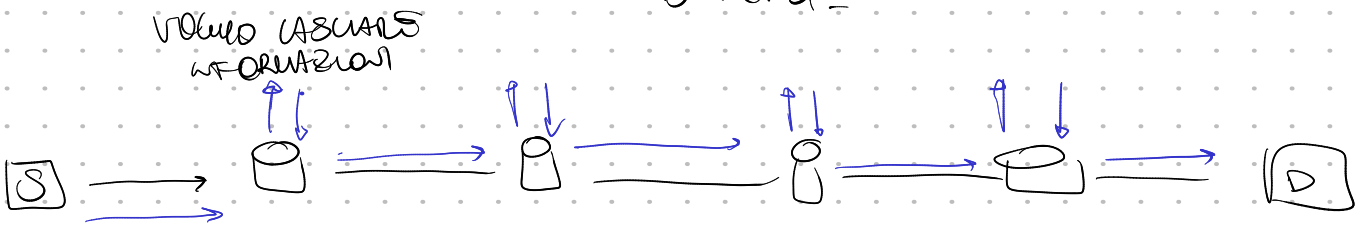


UNA INFORMAZIONE SPECIFICA PER OGNI ROUTER NEL PATH

D	S

SE ABBIAMO BISOGNO DI QUALCHE SERVIZIO CHE DISCRIMINA NEI RIGUARDI DEI ROUTER ATTRAVERSATI?

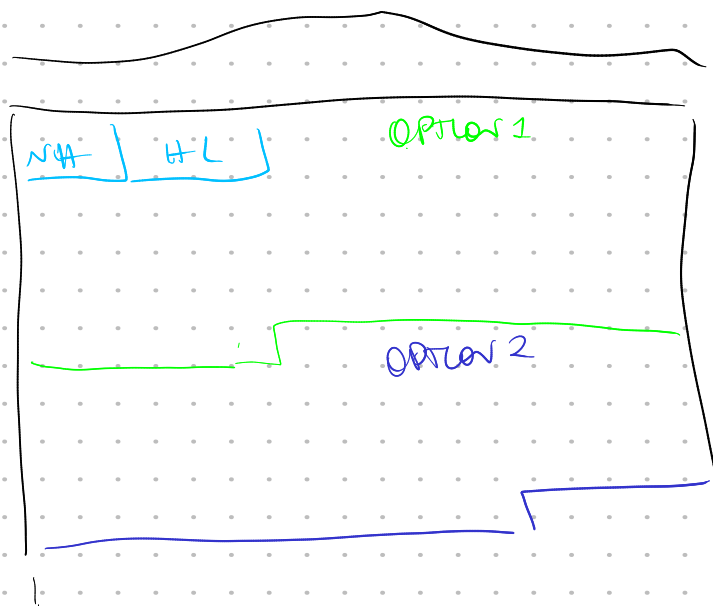
EQUIVALENTE 2 PACCHETTI CON D E S UGUALI POSSONO ATTRAVERSARE PATH DIVERSI_



MODI:

1. FILTRARE I PACCHETTI NEI ROUTER E QUANDO TROVO UN PACCHETTO CON UN DATO HEADER LO PROCESSO_
   NON SO QUALI ROUTER SARANNO QUELLI INTERESSATI_
2. SE IL DESTINATARIO È L'UNICO IP DEL ROUTER IL PACCHETTO È PROCESSATO DA QUEST'ULTIMO
   MA NO VOGLIO CHE D SIA L'UNICO DEL DESTINATARIO
3. PROGETTO UN EXTENSION HEADER CON UN CODICE CHE DÀ AL ROUTER DI EFFETTUARE PROCESSAZIONI AD OGNI HOP_

HEADER: 8 BYTES

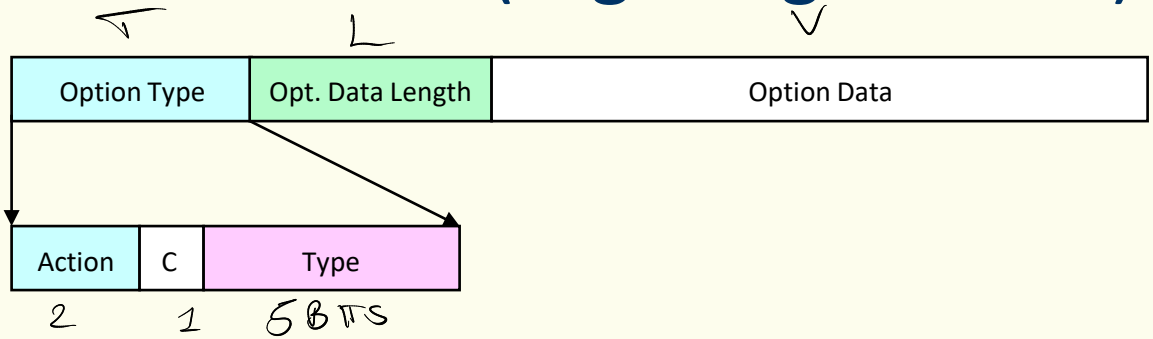


CODIFICATO CON TLV

TAG LENGTH VALUE

Hop-by-Hop Options header

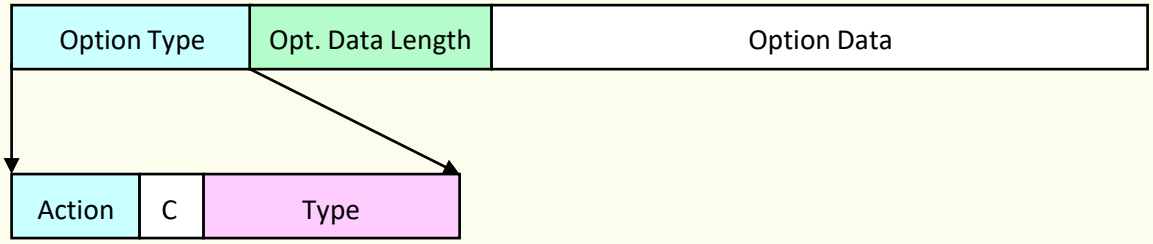
- Options field format (Tag-Length-Value)



  - Action (2 bits): what to do if the option is not recognized
    - 00: Skip and continue processing
    - 01: Discard the packet
    - 10: Discard the packet and send an ICMP Parameter Problem message to the packet's Source address
    - 11: Discard the packet and send ICMP Parameter Problem message to the packet's Source address only if the destination is not a multicast address

Hop-by-Hop Options header

- Options field format (Tag-Length-Value)



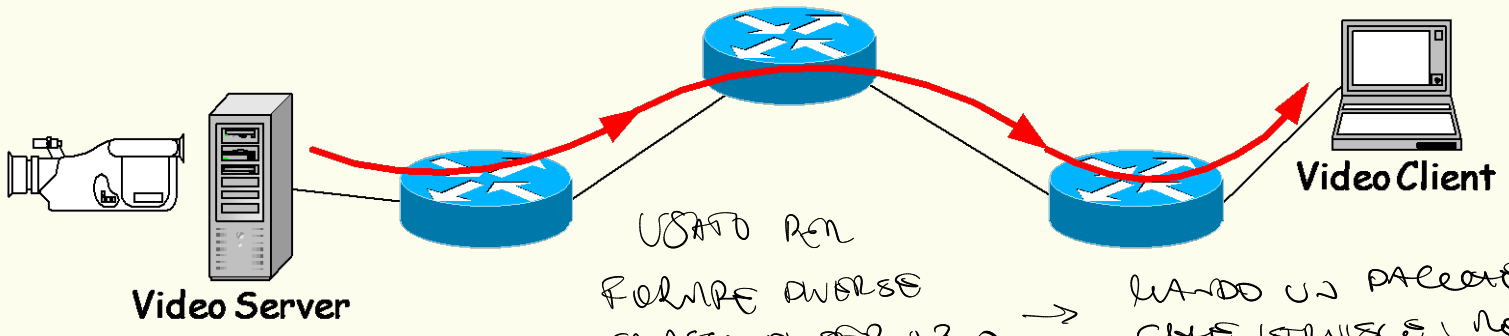
  - C (1 bit)
    - 1: the option information can change en route
    - 0: the option information does not change en route
  - Type
    - Jumbo Payload
    - Router Alert

Hop-by-Hop Options header

- Jumbo Payload (Type=194)
  - Used to send very large packets whose length cannot be encoded on 16 bits only (>64KB)
    - when used, the IPv6 payload length field is set to zero
  - Packet length encoded with 32-bits
    - supports the transmission of packets that are between 65,536 and 4,294,967,295 bytes (4GB)
- Compromise between the initial design of IPv6 and special networking requirements

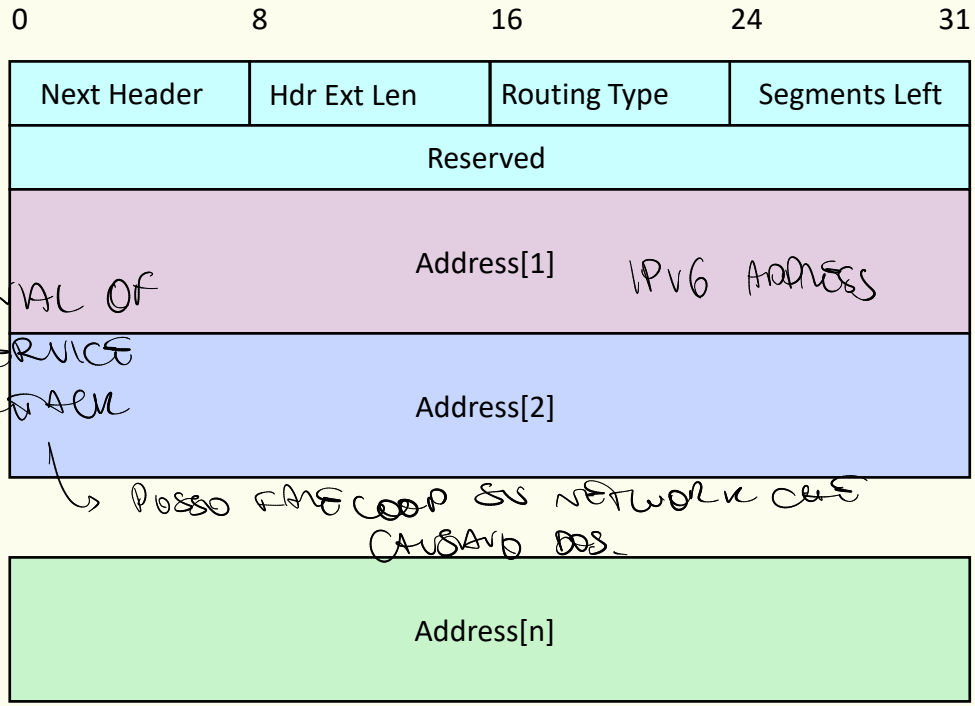
Hop-by-Hop Options header

- Router Alert (Type=5)
  - indicates to a router on the forwarding path that the packet contains important information to be processed by the router
  - Example: RSVP uses control packets containing information that needs to be interpreted or updated by routers along the path



Routing header

- Gives a list of intermediate nodes to be visited on the path to the destination
  - Routing Type
    - **0: default**
    - 2: Mobile IPv6
    - 3: RPL
  - Segments Left
    - Nodes left to be visited
  - Address (RT=0)

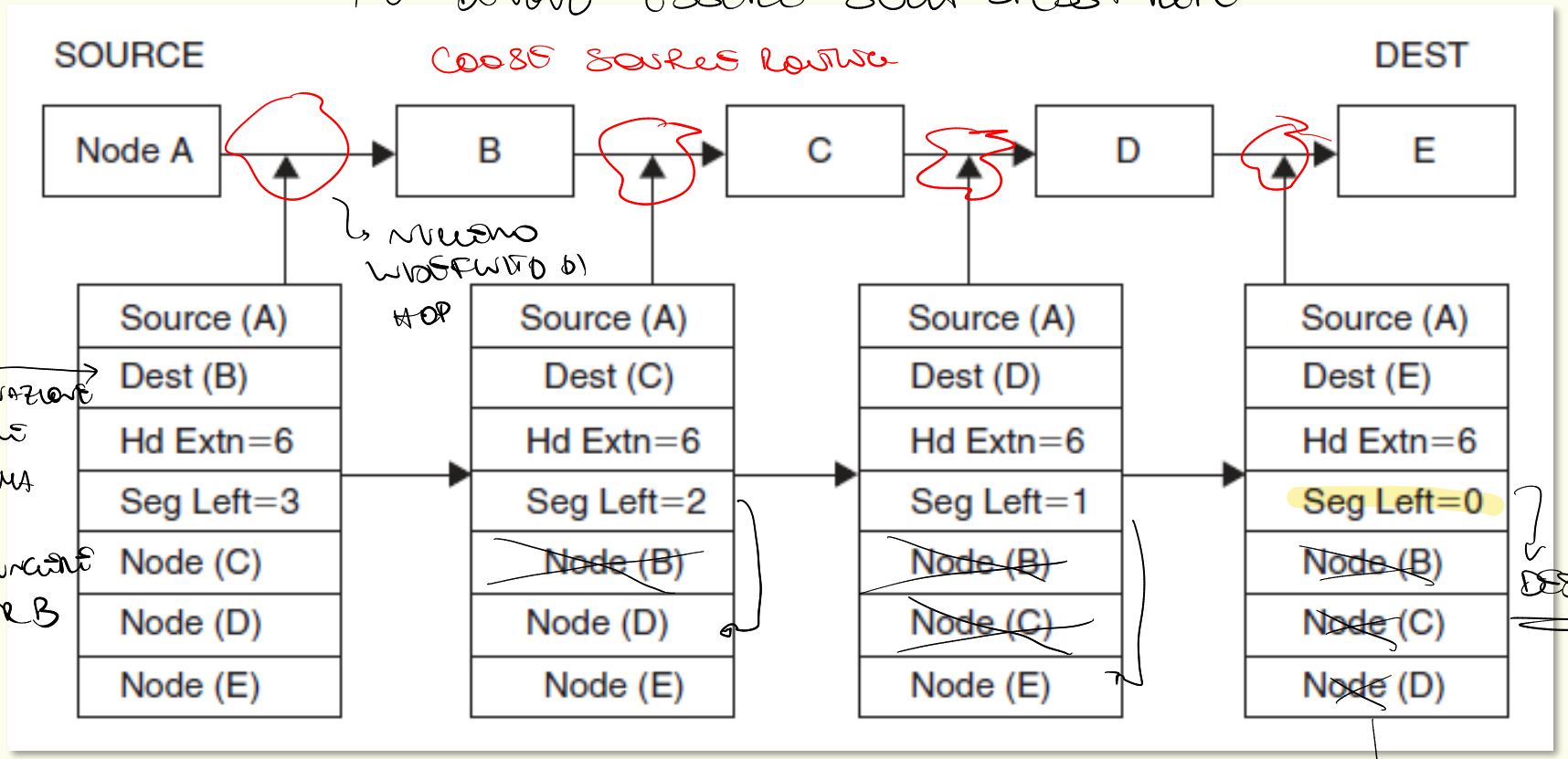


SEGMENT ROUTINES

CONTROLLO COSA SUCCEDE AD UN PACCHETTO SECONDO
PER SECONDO. SI STANNO STUDIANDO ESTENSIONI
PER IPv6.

Routing header

- Example

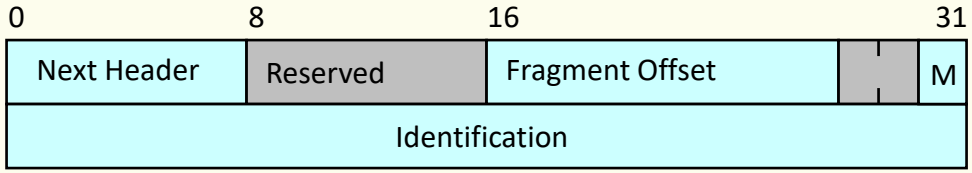


Fragment header

- Unlike IPv4, routers do not fragment IPv6 packets
  - Fragmentation occurs only at the source host sending the packet
  - The destination host only handles reassembly
  - IPv6 packets larger than the MTU of the forwarding link are discarded by the router
- IPv6 hosts use a Path MTU discovery procedure
- The IPv6 minimum MTU size is 1280 bytes!!!

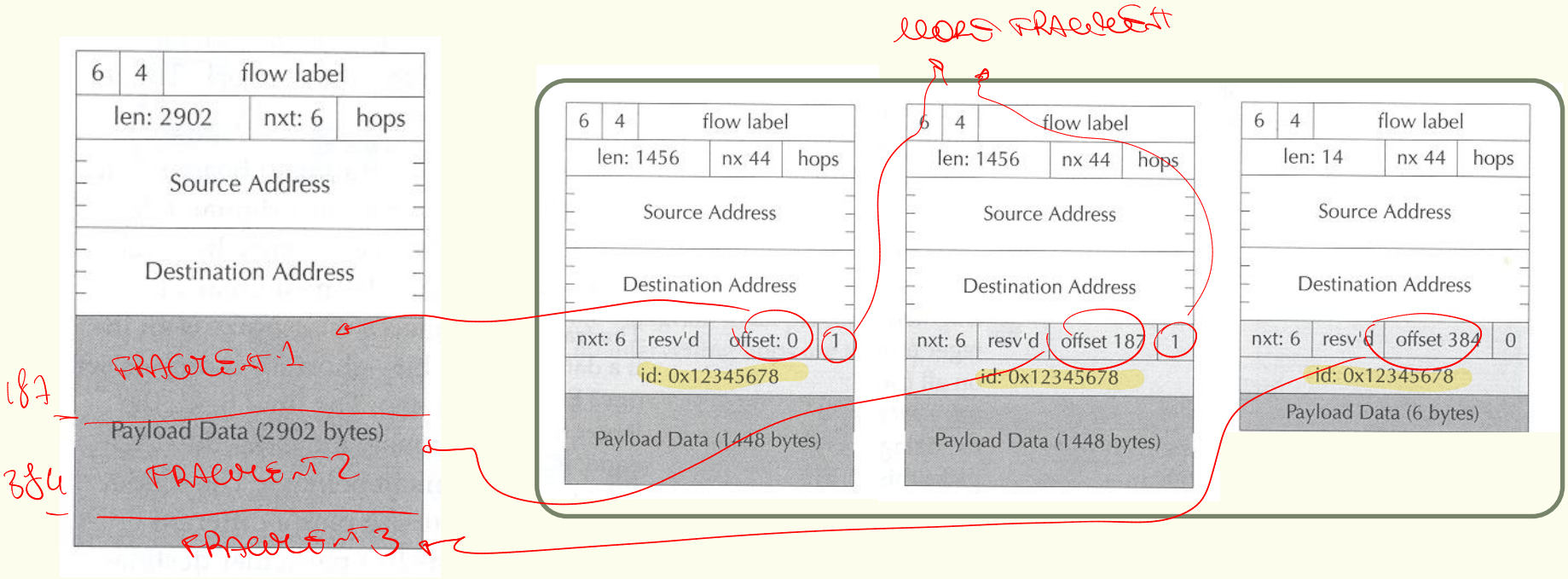
Fragment header

- Fragment Offset (13 bits)
  - The offset in 8-byte units of the data in this packet relative to the start of the data in the original packet
- M-Flag (1 bit)
  - 1: more fragments
  - 0: last fragment
- Identification (4 bytes)
  - Generated by the source host in order to identify all packets belonging to the original packet



Fragment header

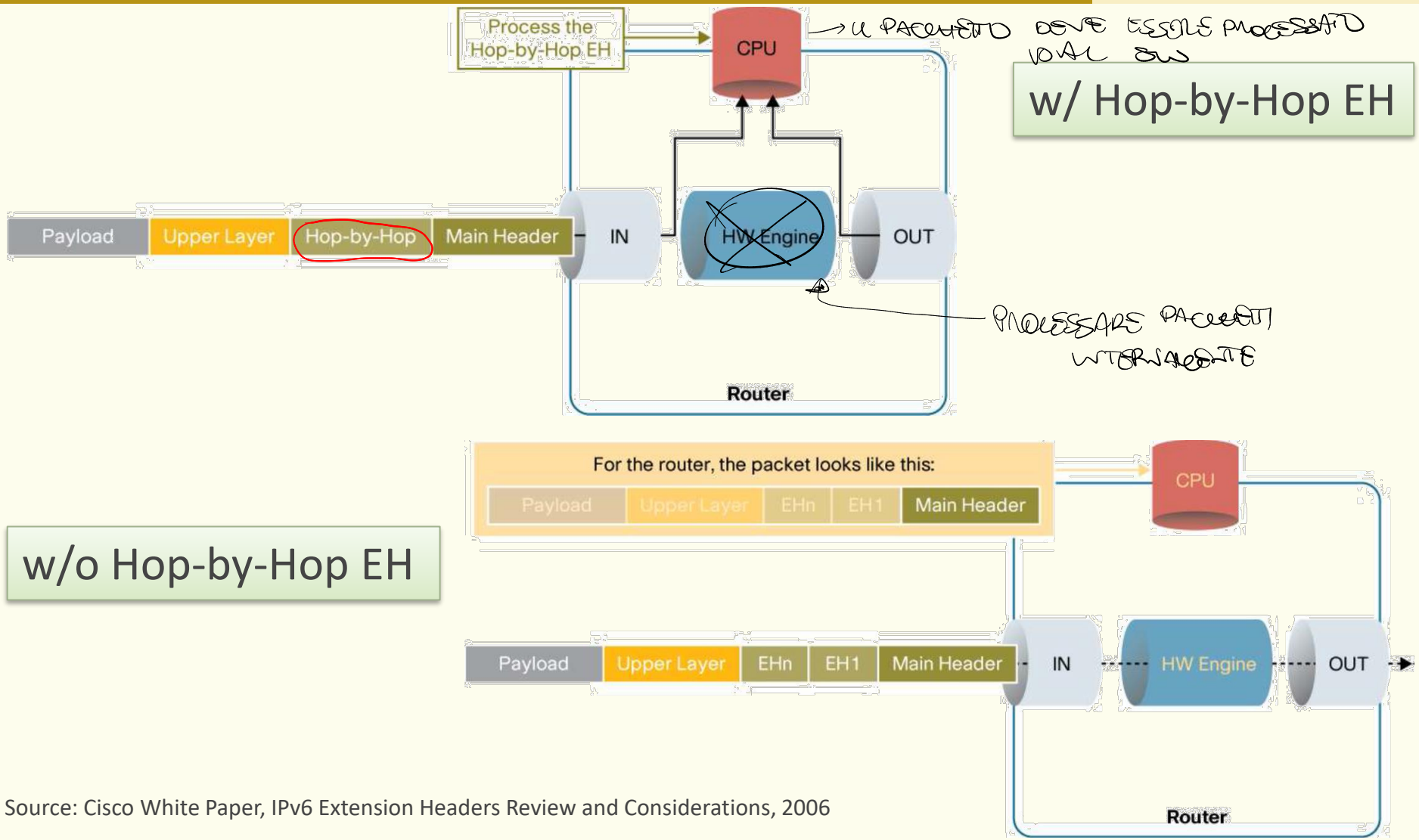
- Example
  - One packet fragmented into three fragments



Other extension headers

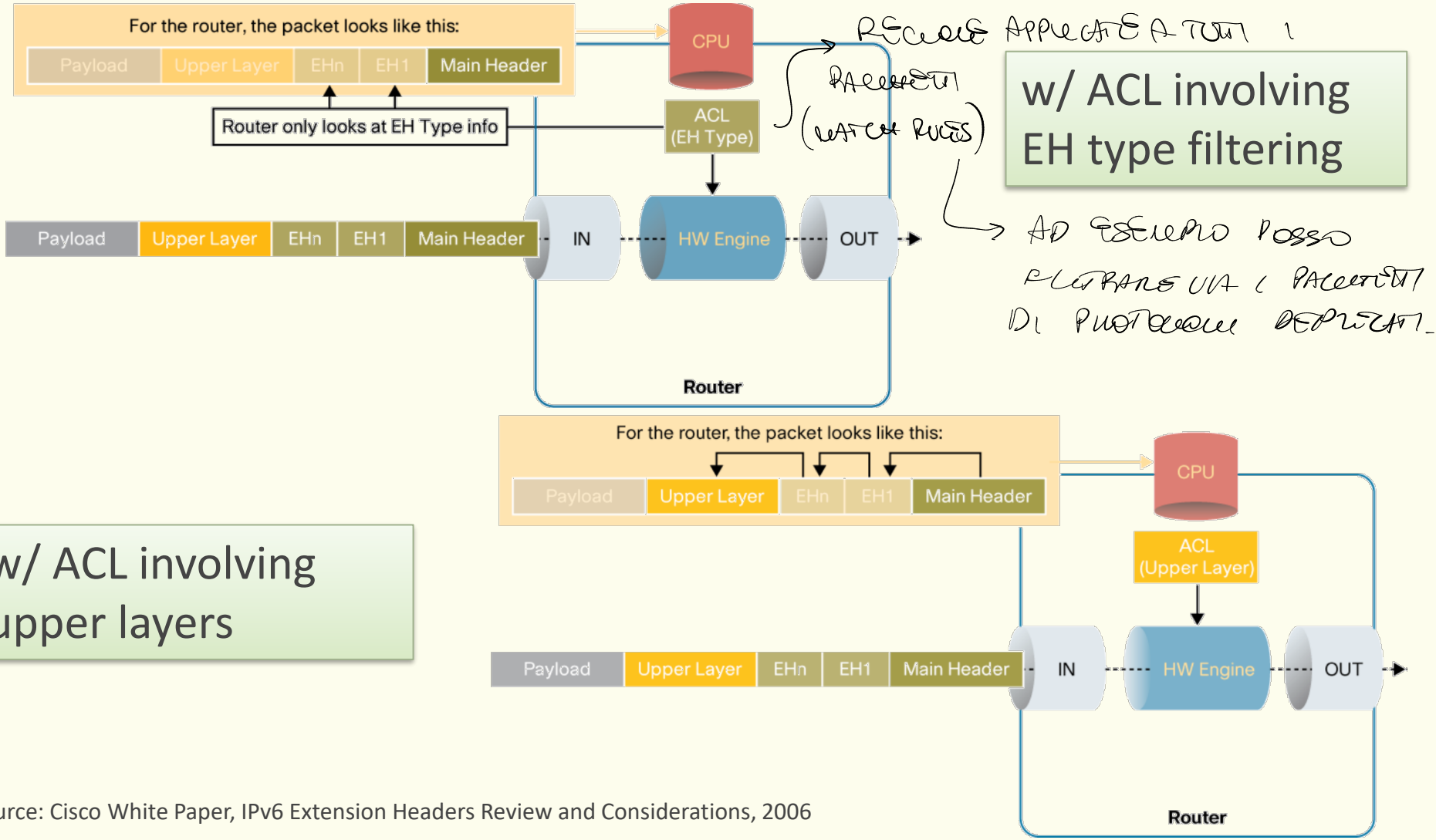
- Authentication Header
- Encapsulating Payload Security Header
  - IPv6 security support (IPsec)
- No Next Header
  - No payload
- Destination Options Header
  - Same as Hop-by-Hop, but options to be processed only at the destination
  - Used by Mobile IPv6

IPv6 EH processing



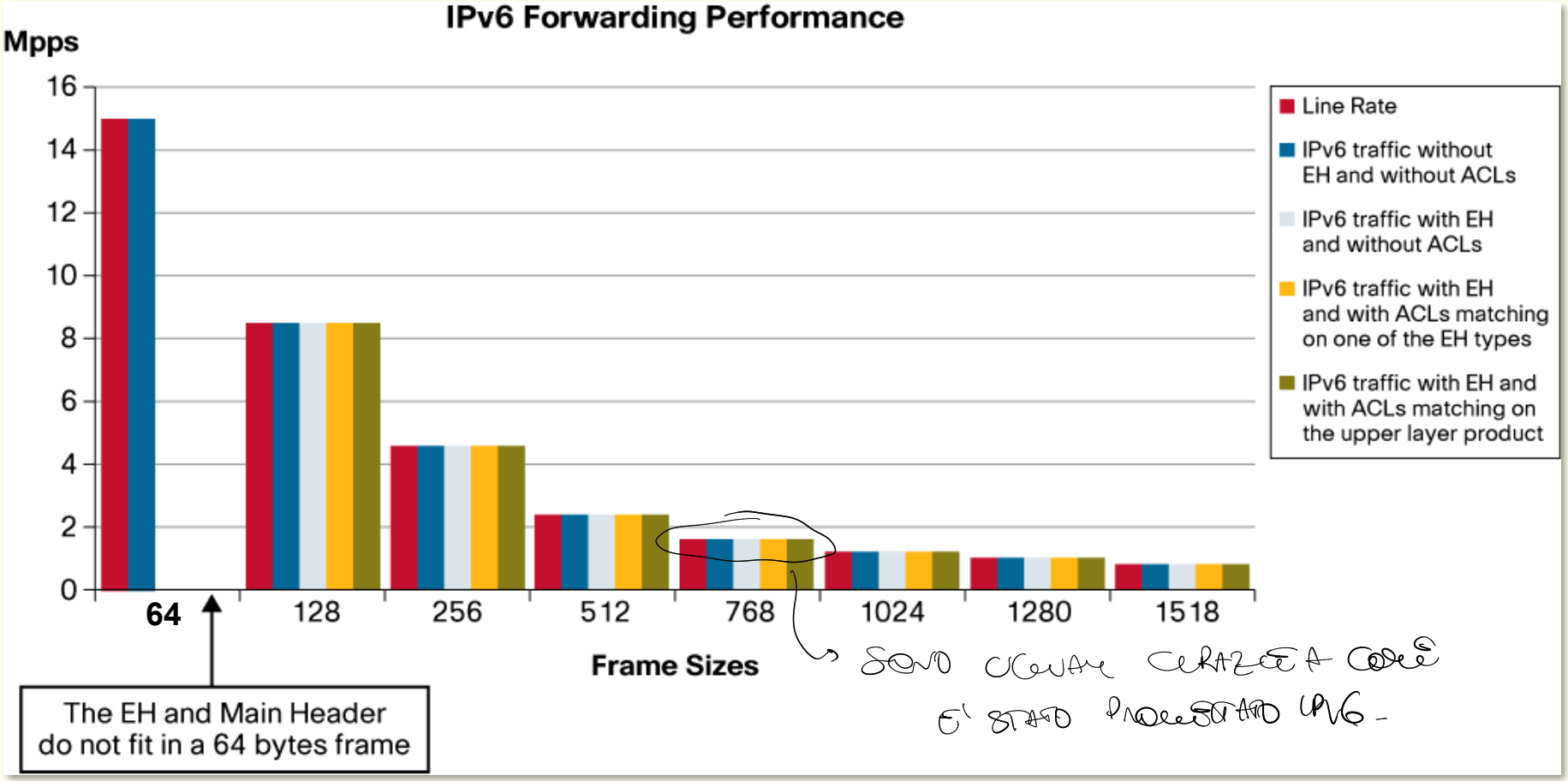
Source: Cisco White Paper, IPv6 Extension Headers Review and Considerations, 2006

IPv6 EH processing



Source: Cisco White Paper, IPv6 Extension Headers Review and Considerations, 2006

IPv6 EH processing



Source: Cisco White Paper, IPv6 Extension Headers Review and Considerations, 2006

References

- S. Hagen. **IPv6 essentials**. 3/ed. O'Reilly, 2014

- RFC 8200, “Internet Protocol, Version 6 (IPv6) Specification,” 2017
- RFC 2675, “IPv6 Jumbograms,” 1999
- RFC 2711, “IPv6 Router Alert Option,” 1999