

IP versione 6

# IPv6

- L'uso del CIDR ha solo temporaneamente risolto (attenuato) i problemi legati allo spazio di indirizzamento ed alle tabelle di *routing*.
- Per cui già nel 1990 è iniziata la fase di standardizzazione di una nuova versione di IP, che dovesse avere i seguenti requisiti
  - Supportare miliardi di utenti (anche presupponendo un inefficiente uso dello spazio di indirizzamento).
  - Ridurre, o comunque mantenere piccole le RT
  - Semplificare il protocollo
  - Migliorare la sicurezza (sia autenticazione, sia protezione del dato)

# IPv6

- Dare supporto a tipi di servizi diversi
- Agevolare il multicast
- Permettere lo spostamento dell'host mantenendo lo stesso indirizzo
- Semplificare evoluzioni future
- Permettere la co-esistenza con IPv4 per lungo tempo.
- La scelta fatta fra diverse proposte è stata
- Simple Internet Protocol Plus (SIPP)
- IPv6

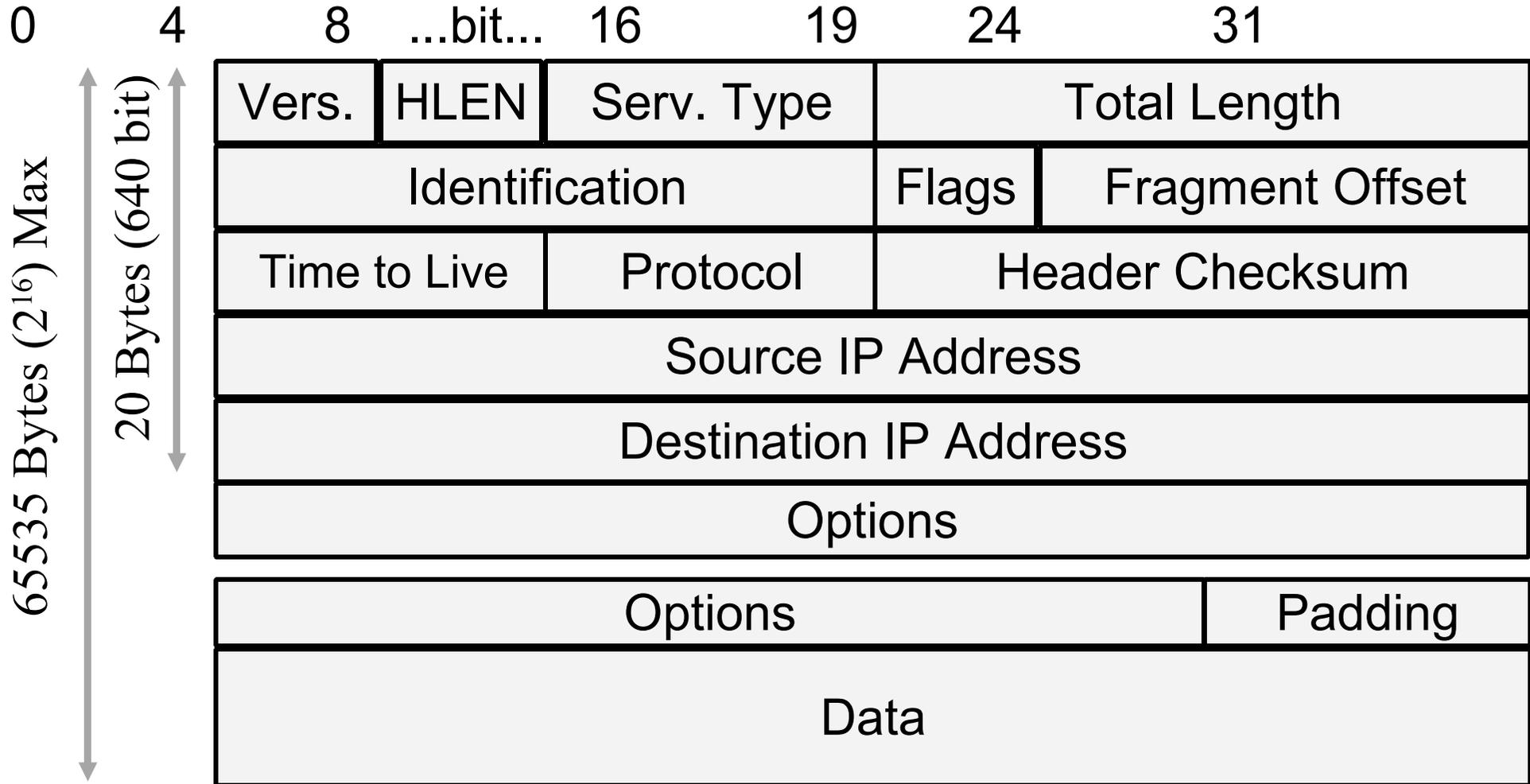
# IPv6

- Gli elementi distintivi principali del nuovo standard sono
  - Non richiede sostanziali modifiche allo standard precedente
  - Gli indirizzi sono significativamente più lunghi
  - L'*header* è più semplice (7 campi invece di 14)
  - Le opzioni sono gestite meglio (anche per permettere una più veloce commutazione dei pacchetti).
  - Maggiore sicurezza
  - Supporto per servizi di tipo diverso.

# IPv6: Header

Version	Priorit.	Flow Label	
Payload Length		Next Header	Hop Limit
Source Address			
Destination Address			

# Il pacchetto IPv4



# IPv6: Header

- **Versione** (4 bit): il valore è 6, anche se in fase di transizione è stato suggerito (per velocizzare) di inserire l'informazione nel livello 2 come si trattasse di due protocolli diversi;
- **Priorità** (o *Traffic Class*, 4 bit): la sorgente dichiara tramite questo campo il trattamento che il pacchetto deve subire. Si distingue inizialmente fra:
  - ***Congestion Controlled Traffic*** (CCT): ossia il traffico su cui viene effettuato un controllo di congestione ed un recupero dell'errore (tutto il traffico dati in genere).
  - **Non- CCT**: i traffici che generano flussi di dati per lo più continui che necessitano di un ritardo ridotto (voce - video).

# IPv6: Header - Priority

Priorità crescente

## CCT

- |   |  |
|---|--|
| 0 | Non specificato Default                      |
| 1 | Di riempimento (es. news)                    |
| 2 | Batch (es. email)                            |
| 3 | Riservato                                    |
| 4 | Interattivo a bassa priorità (es. ftp, http) |
| 5 | Riservato                                    |
| 6 | Interattivo ad alta priorità (es. Telnet, X) |
| 7 | Di controllo (es. OSPF, SNMP)                |

## Non CCT

- |    |  |
|----|--|
| 8  | ammesse perdite più elevate<br>(es. video alta qual.)  |
| 9  |  |
| 10 |  |
| 11 |  |
| 12 |  |
| 13 |  |
| 14 |  |
| 15 | ammesse perdite meno elevate<br>(es. audio telefonico) |

# IPv6: Header - Flow Label

- Questo campo individua dei flussi, ossia sequenze di pacchetti emessi dalla stessa sorgente per lo stesso servizio.
- Questa informazione dovrebbe permettere ai *router* di negoziare un trattamento particolare per alcuni flussi di dati.
- Le regole con cui trattare il campo sono:
  - Gli *host/router* che non gestiscono flussi devono lasciare il campo invariato nel *forwarding*, o metterlo a zero se sono origine del pacchetto.
  - Tutti i pacchetti generati dalla stessa sorgente con lo stesso numero di flusso (diverso da zero) devono avere gli stessi indirizzi di destinazione, sorgente e *Hop by Hop Option Header* (se presente) e *Routing Header* (se presente).
  - Gli ID di un flusso vanno scelti casualmente, con distribuzione uniforme da 1 a  $2^{20}-1$  (per rendere efficienti le tabelle di *hash*), con la restrizione che una sorgente non possa riutilizzare numeri che sta già usando per altri flussi attivi.

# IPv6: Header

- ***Payload Length*** (16 bits): lunghezza della parte dati del datagram in ottetti (a differenza dell'IPv4 non comprende l'intestazione).  
La parte fissa dell'header è lunga 40 ottetti (contro i 20 dell'IPv4).
- ***Next Header*** (8 bits)
- ***Hop Limits*** (8 bits): Viene decrementato di 1 ogni nodo attraversato (non si tiene più conto del tempo di attesa).
- Indirizzo di sorgente e di destinazione  
(128 bits + 128 bits).

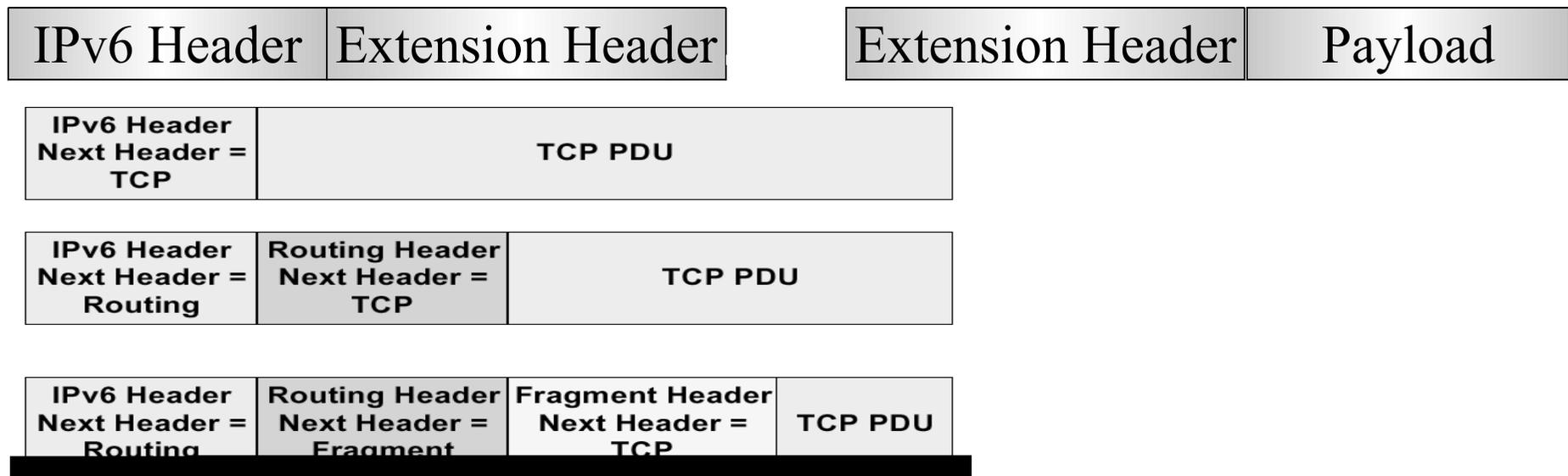
# Header - Next Header

- Il campo *Next header* identifica il successivo *header* che può essere un altro protocollo trasportato (e quindi essere contenuto nel *payload* e da elaborare solo alla destinazione) oppure degli *header* aggiuntivi (*Extension Header*) di IPv6. Gli *header* aggiuntivi contengono a loro volta il campo *next header* che permette di creare una catena di *ExHeader*.

40 Bytes

0 o più

0 - 64Kb



# IPv6: Header - Next Header

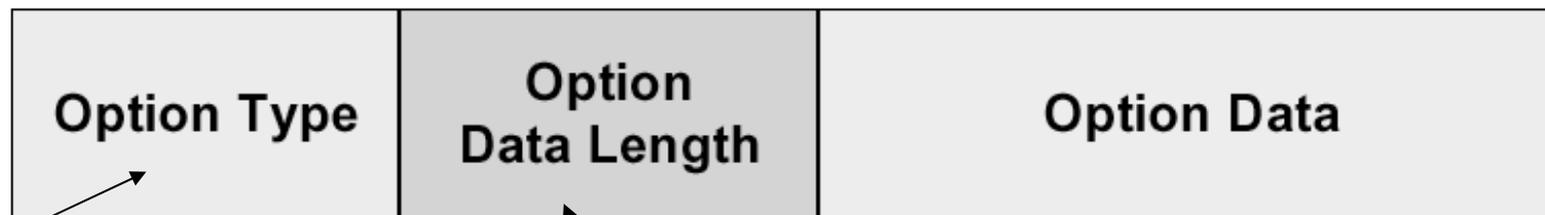
- ➔ **0 HBH Hop by Hop option (IPv6)**
  - 1 ICMP Internet Control Message (IPv4)
  - 2 IGMP Internet Group Management (IPv4)
  - 3 GGP Gateway-to-Gateway
  - 4 IP IP in IP (IPv4 encapsulation)
  - 6 TCP Transmission Control
  - 17 UDP User Datagram
  - 29 TP4 ISO Transport class 4
- ➔ **43 RH Routing Header (IPv6)**
- ➔ **44 FH Fragment Header**
  - 45 IDRP Interdomain Routing
- ➔ **50 ESP Encrypted Security Payload**
- ➔ **51 AH Authentication Header**
  - 58 ICMP Internet Control Message (IPv6)
  - 59 Null No next header (IPv6)
- ➔ **60 DOH Destination Option Header**
  - 80 ISO-IP ISO 8473 CLNP
  - 88 IGRP Interior Gateway Routing
  - 89 OSPF Open Shortest Path First (IPv6)

Gli *ExHeader* di IPv6 vanno inseriti (uno solo per tipo) ed elaborati nel seguente ordine:

- *Hop-by-Hop Header*
- *Routing Header*
- *Fragment Header*
- *Authentication Header*
- *Encapsulating Security Payload Header*
- *Destination Options Header*

# IPv6: Header - *Hop-by-Hop Header*

- Trasporta informazioni che devono essere elaborate in ogni nodo di transito. I campi di cui è composto sono:
  - *Next Header* (8 bit)
  - *Header Extension Length* (8 bit): in numero di blocchi da 64 bit esclusi i primi 64.
  - Opzioni: ogni opzione è codificata con tre campi:

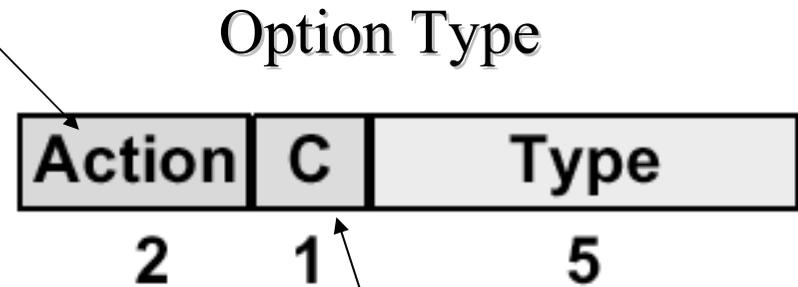


Tipo di Opzione  
Lunghezza del campo *Option Data* in ottetti

# IPv6: Header - *Hop-by-Hop Header*

Specifica cosa fare se non si riconosce l'opzione:

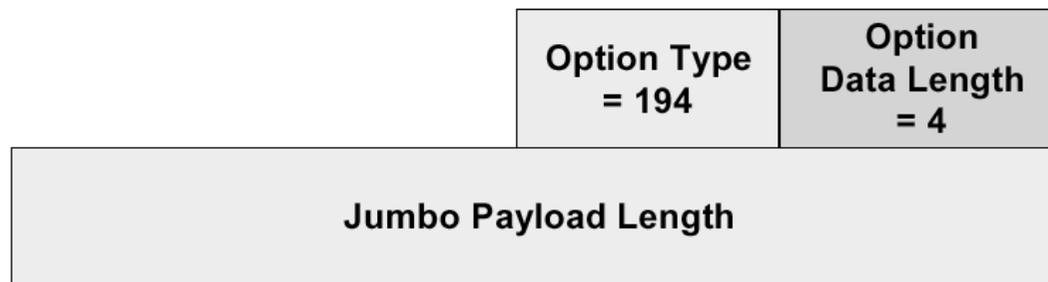
- 00 si ignora quella sconosciuta e si continua a elaborare la successiva
- 01 si scarta il pacchetto
- 10 si scarta il pacchetto e si notifica al mittente tramite ICMP anche con destinazione multicast
- 11 si scarta il pacchetto e si notifica al mittente tramite ICMP solo con destinazione unicast



Specifica se l'opzione può (1) o non può (0) essere modificata lungo il percorso

# IPv6: Header - *Hop-by-Hop Header*

- Attualmente sono state definite solo 3 opzioni:
  - Pad1 (*Option Type* = 0) non ha i campi lunghezza e dati e rappresenta solo un riempimento di un byte.
  - PadN (*Option Type* = 1), ha tutti campi, e serve per realizzare riempimenti da 2 a N bytes.
  - *Jumbo Payload*: il campo JPL indica la lunghezza del datagram in ottetti, escluso l'*header* IP ma compreso HbHH. La lunghezza deve essere più di 64Kb, e deve avere un allineamento di  $4n+2$ .

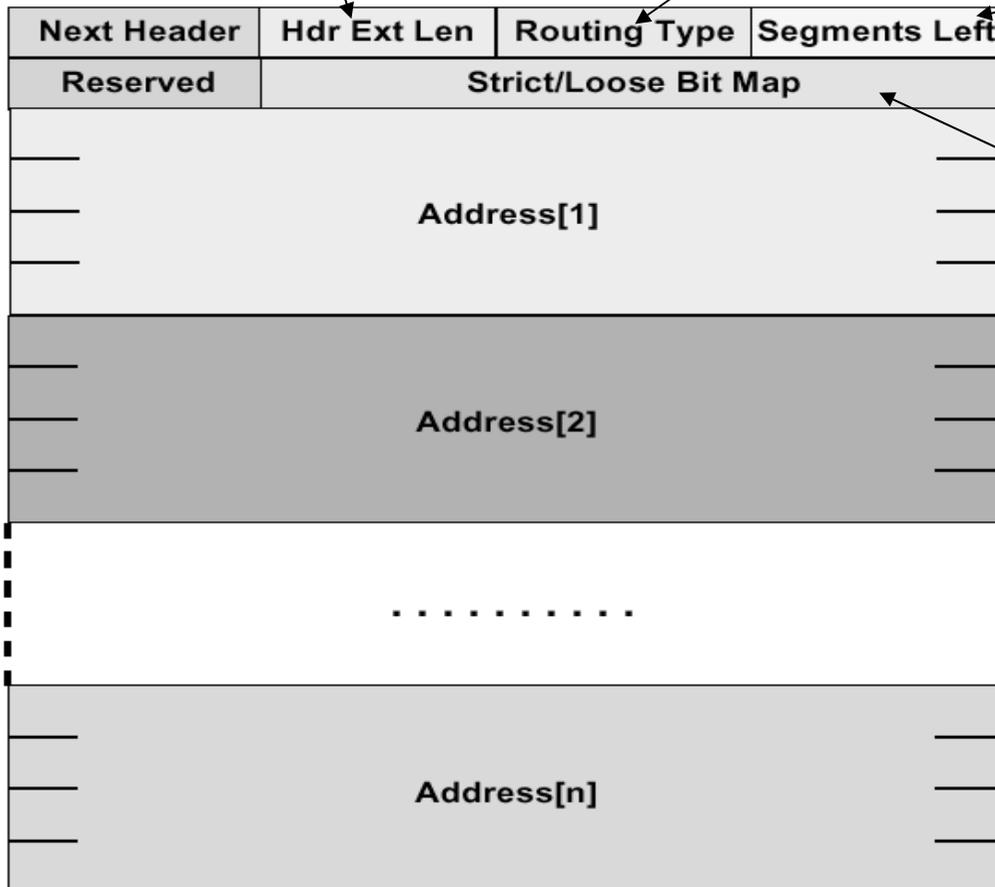


# IPv6: Header - *Routing Header*

Deve essere pari perché gli indirizzi sono 128 Byte

Per ora è stato definito solo il tipo 0

Numero di indirizzi ancora da considerare (max 23)



Un bit per ogni indirizzo, 0 per gli indirizzi da trattare *loose* e 1 per quelli *strict*.

# IPv6: Header - *Routing Header*

- Permette di realizzare un *Source Routing*
- L'indirizzo inserito nel campo di destinazione del *Header IPv6* non è la destinazione finale ma la successiva da raggiungere nell'elenco, così che ogni nodo intermedio non debba elaborare il campo opzionale.
- Si osservi che IPv6 richiede che le risposte ai pacchetti contenenti un RH debbano utilizzare lo stesso percorso all'indietro. Questo fornisce un potente mezzo per stabilire vincoli di instradamento a priori.

# IPv6: Header - *Fragment Header*

- Il processo di frammentazione è diverso in IPv6 rispetto ad IPv4. In IPv6 solo la sorgente può frammentare il *datagram*, l'eventuale frammentazione dipende dalla *Maximum Transfer Unit* (MTU) che la sorgente dovrebbe poter verificare sul percorso verso la destinazione. Altrimenti dovrebbe ipotizzare la MTU più piccola di 576 ottetti.
- Il *datagram* è diviso in una parte non frammentabile (composta dall'*header* originale e da ExHeader HbHH e RH che vanno duplicati in ogni frammento) e una frammentabile che contiene il resto.
- Nell'*header* si trovano i campi: ***Fragment offset*** (13 bit) in numero di 64 bit, ***MFlag*** (1 ci sono ancora seg., 0 se è l'ultimo), ***Identification*** (32 bits): deve essere unico per una coppia di indirizzi sorgente -destinazione.

# IPv6: Header

- Se si confronta l'*header* IPv4 e IPv6 si notano alcune differenze sostanziali (a prescindere dagli indirizzi):
  - Il campo HL non c'è più perché in IPv6 la lunghezza dell'*header* è fissa
  - Il campo *Protocol* è sostituito da *NextHeader*
  - Tutti i campi legati alla frammentazione non ci sono più.
  - Il campo *checksum* è stato eliminato per velocizzare il trattamento del pacchetto.

# IPv6: Indirizzi

- 128 bit
  - $2^{128}$  indirizzi
  - circa  $10^{38}$  indirizzi
  - Più precisamente
    - 340.282.366.920.938.463.463.374.607.431.768.211.456 indirizzi
- Alcune stime:
  - superficie della terra 511.263.971.197.990 mq
  - 655.570.793.348.866.943.898.599 indirizzi IPv6 per mq

# IPv6: Indirizzi

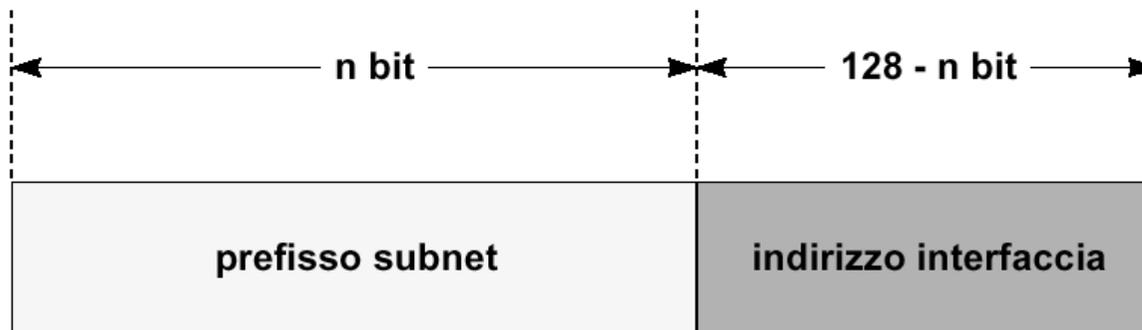
- Tre tipi di indirizzo:
  - *Unicast*
    - indirizzi verso singole stazioni
  - *Anycast*
    - Identifica un insieme di interfacce, ma un pacchetto con questo indirizzo deve raggiungerne una sola, ma una qualsiasi, in genere la più “vicina” (usato per servizi)
  - *Multicast*
    - indirizzi di gruppi di stazioni
- Non viene più utilizzato il *Broadcast*
- Gli indirizzi sono associati alle interfacce
- Possibilità di avere più indirizzi per ogni interfaccia

# IPv6: Indirizzi

- Si scrivono in esadecimale come 8 gruppi di 4 cifre separati da “:”
  - FEDC:BA98:0876:45FA:0562:CDAF:3DAF:BB01
  - 1080:0000:0000:0007:0200:A00C:3423
- Esistono delle semplificazioni:
  - si possono omettere gli zero iniziali  
1080:0:0:7:200:A00C:3423
  - Si possono sostituire gruppi di zero con “::”
  - 1080::7:200:A00C:3423
- Gli indirizzi di compatibilità IPv4 si scrivono:
  - 0:0:0:0:0:0:A00:1
  - ::A00:1
  - ::10.0.0.1

# IPv6: Indirizzi

- Scompare il concetto di *Netmask*
- Viene sostituito da quello di “*Prefix*”
- Il *prefix* si indica aggiungendo ad un indirizzo “/N”, dove N è la lunghezza in bit del *prefix*
- Esempio:
  - FEDC:0123:8700::/36 indica il prefisso
  - 1111111011011100000000001001000111000



# IPv6: Indirizzi

<u>Allocation</u>	<u>Prefix</u>	<u>Fraction of Address Space</u>	
Reserved	0000 0000	1/256	
Unassigned	0000 0001	1/256	
Reserved for NSAP Allocation	0000 001	1/128	
Reserved for IPX Allocation	0000 010	1/128	
Unassigned	0000 011	1/128	
Unassigned	0000 1	1/32	
Unassigned	0001	1/16	
<b>Aggregatable Global Unicast Addresses</b>	<b>001</b>	<b>1/8</b>	←
Unassigned	010	1/8	
Unassigned	011	1/8	
Unassigned	100	1/8	
Unassigned	101	1/8	
Unassigned	110	1/8	
Unassigned	1110	1/16	
Unassigned	1111 0	1/32	
Unassigned	1111 10	1/64	
Unassigned	1111 110	1/128	
Unassigned	1111 1110 0	1/512	
<b>Link-Local Unicast Addresses</b>	<b>1111 1110 10</b>	<b>1/1024</b>	←
<b>Site-Local Unicast Addresses</b>	<b>1111 1110 11</b>	<b>1/1024</b>	←
<b>Multicast Addresses</b>	<b>1111 1111</b>	<b>1/256</b>	←

## IPv6- Indirizzi *unicast*

# *Aggregatable Global Unicast Addresses*

- FP *Format Prefix (001)*
- TLA-ID *Top-Level Aggregation Identifier*
- RES *Reserved (per usi futuri)*
- NLA-ID *Next-Level Aggregation Identifier*
- SL- ID *Site-Level Aggregation Identifier*
- INTERFACE-ID *Interface Identifier*



Public Topology

Site  
Topology

Interface  
Identifier

# IPv6- Indirizzi *unicast*

## AGUA - TLA-ID

- Il *Top-Level Aggregation Identifier* identifica gli ISP principali che forniscono il servizio di connettività della rete.
- I *router* che operano a questo livello della gerarchia devono avere una riga per ogni TLA nella RT.
- 13 bit permettono 8.192 ( $2^{13}$ ) diversi ISP. Attualmente ci sono *router* che operano con tabelle con più di 50.000 elementi, ma l'IETF ha deciso di ridurre la dimensione delle RT dei *router* all'apice della gerarchia.
- E' previsto l'eventuale allargamento di questa parte dell'indirizzo sia tramite i *Reserved* bit sia allocando un altro FP.

IPv6- Indirizzi *unicast*

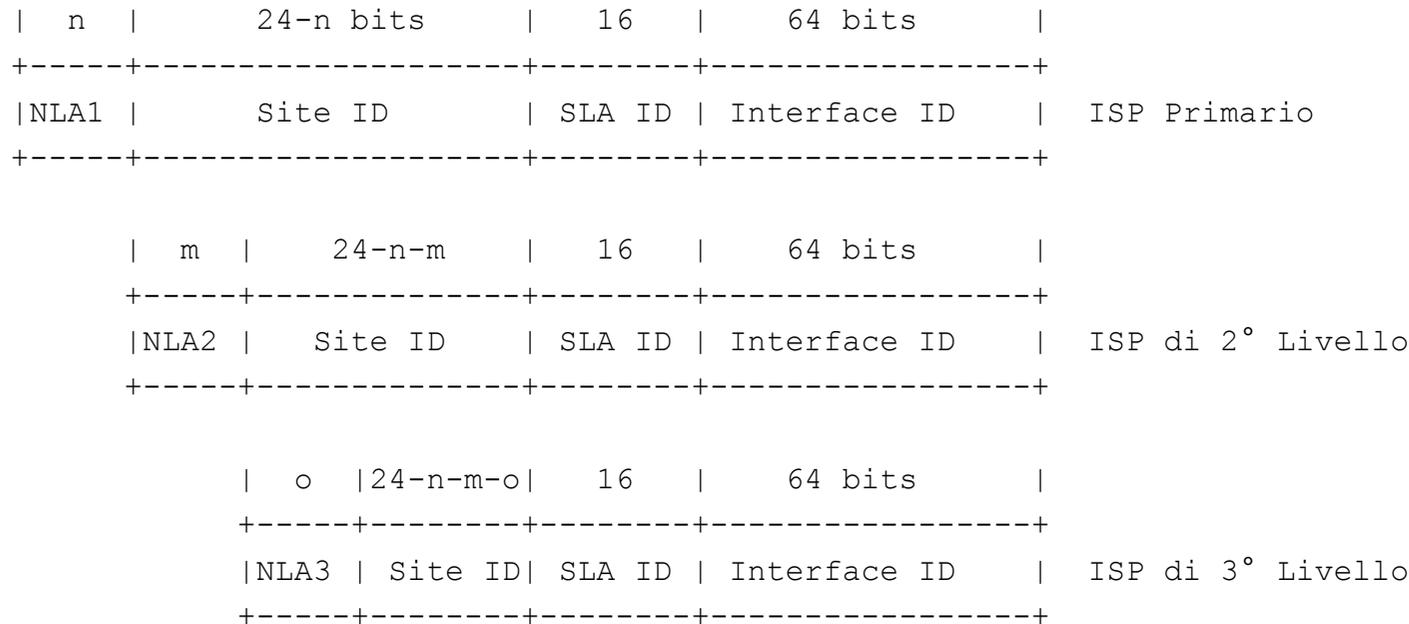
## AGUA - RES

- I *Reserved* bit devono essere posti a 0.
- Sono pensati per permettere, in relazione ai bisogni che dovessero presentarsi, sia l'eventuale espansione (a destra) del campo TLA-ID sia di quello NLA-ID (a sinistra).

# IPv6- Indirizzi *unicast*

## AGUA - NLA-ID

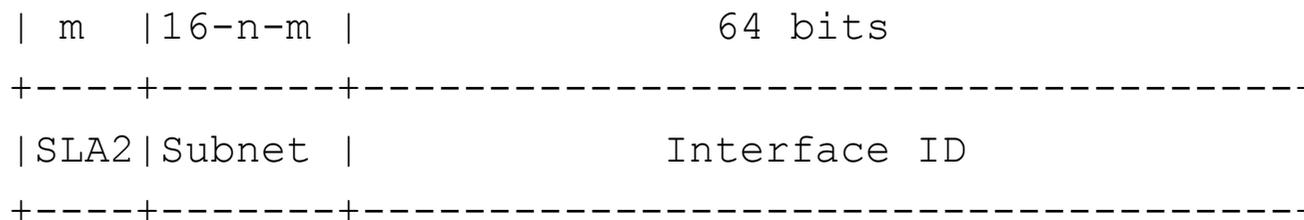
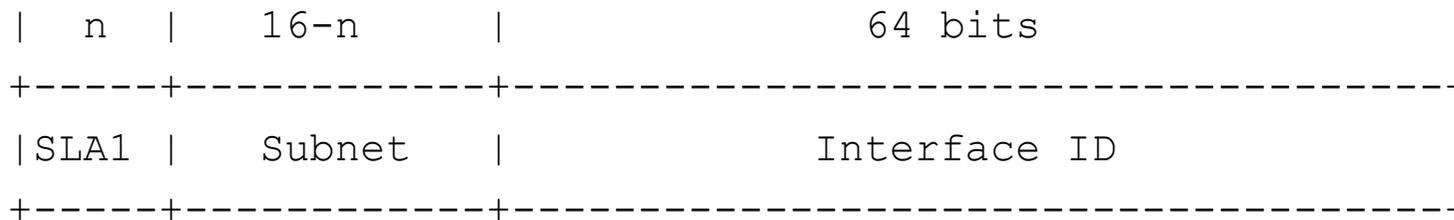
- Il *Next-Level Aggregation Identifier* è usato dall'ISP per organizzare la propria rete interna e può eventualmente essere a sua volta strutturato in modo gerarchico e sue parti assegnate ad ISP secondari.



# IPv6- Indirizzi *unicast*

## AGUA - SL-ID

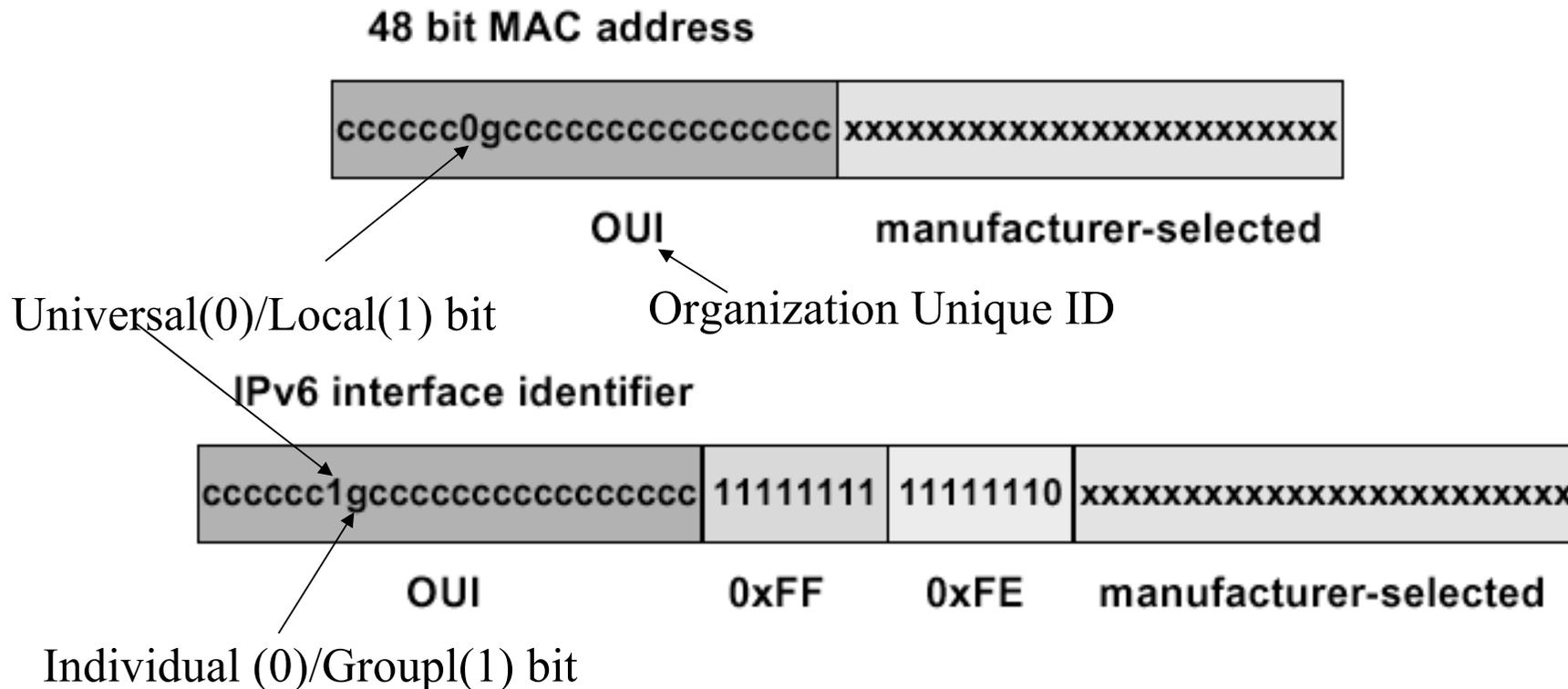
- Il *Site-Level Aggregation Identifier* viene assegnato dall'utente (identificato da un NLA -ID) che può mantenere una gestione dei propri indirizzi di tipo "piatto" (flat), oppure a sua volta gestire delle gerarchie per ridurre le proprie tabelle di routing.
- Lo spazio di indirizzamento è grande (come una classe B IPv4), l'organizzazione che avesse necessità ancora superiori può chiedere ulteriori siti (NLA ID).



# IPv6- Indirizzi *unicast*

## AGUA - Interface-ID

- L'Interface ID viene ricavato usando gli indirizzi di livello 2. Per esempio nel caso di indirizzo MAC:



# IPv6- Indirizzi *unicast* AGUA

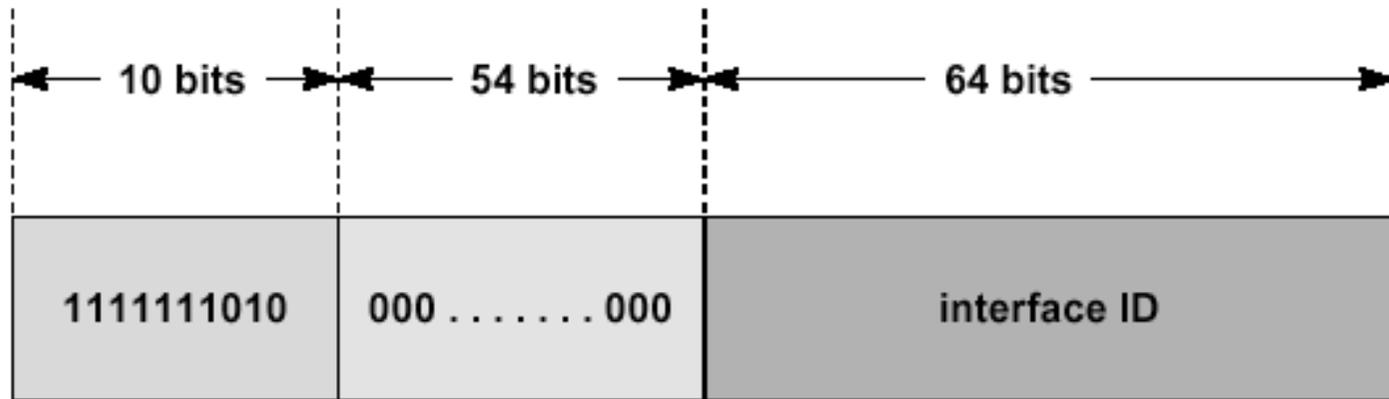


**3F15 : F0 : 839A : 300 : 260 : 8CFF:FE15:6B60**

Bit U/I=1

# IPv6- Indirizzi *unicast* *Link Local*

- Indirizzi “privati” (non annunciati dai router) pensati per piccole reti, autoconfiguranti, prive di router (interni).

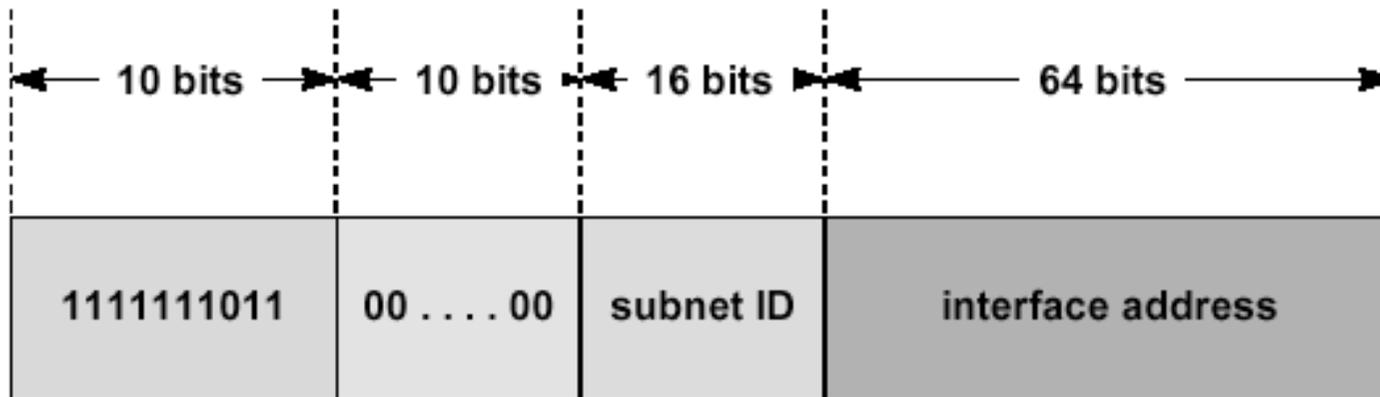


FE80::A00:2FF:FE12:3456



# IPv6- Indirizzi *unicast* *Site Local*

- Gli indirizzi *Site Local* sono privati (non annunciati) e permettono la realizzazione di reti interne strutturate



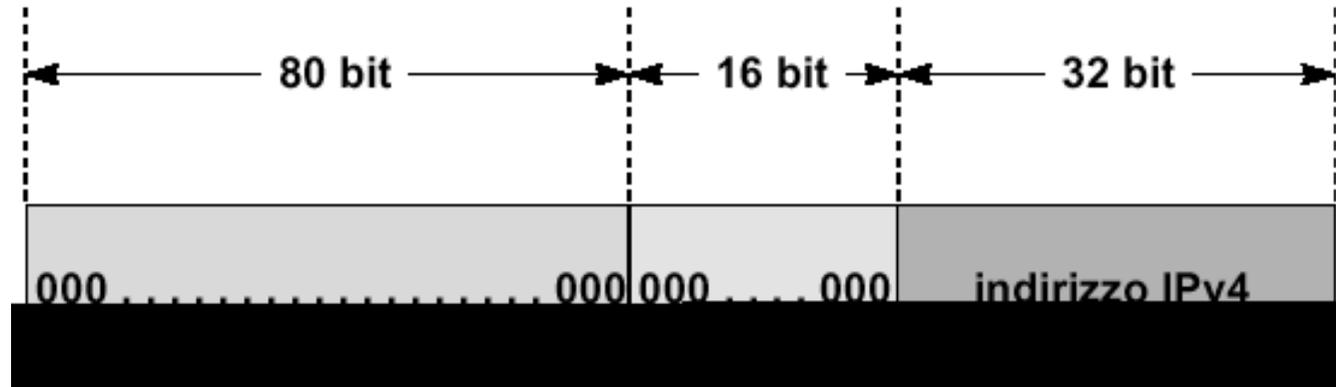
**FEC0::11:200:CFF:FE12:3456**



# IPv6- Indirizzi *Reserved*

## Indirizzo IPv6 compatibile IPv4

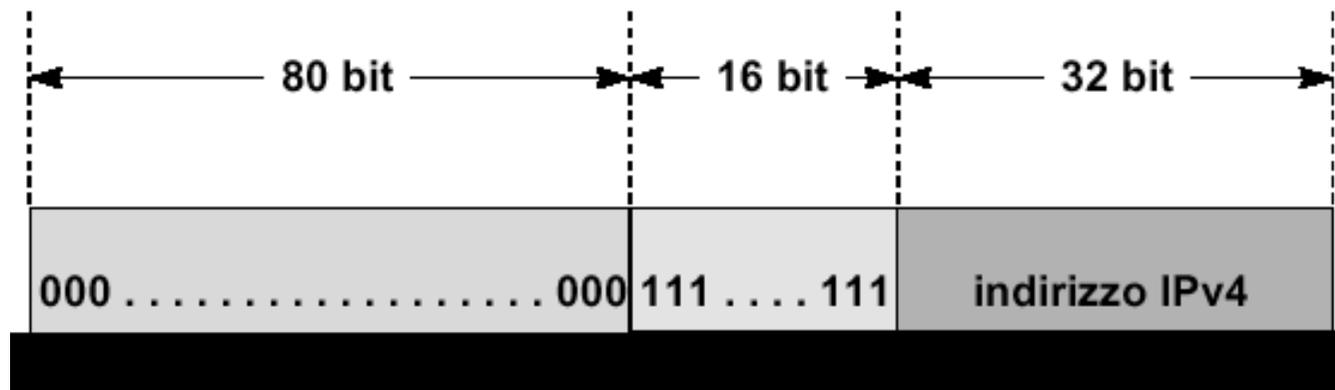
Usato per fare *tunneling*  
di IPv6 su infrastrutture  
IPv4



Es. **::130.192.252.27**

## Indirizzo IPv6 ricavato da un indirizzo IPv4

Usato per fare tunneling  
di IPv4 su infrastrutture  
IPv6



Es. **::FFFF:130.192.252.27**

# IPv6- Indirizzi *unicast* *Reserved*

- Indirizzi riservati particolari sono:
  - :: (significa nessun indirizzo)
  - ::1 (*loopback*)

## IPv6- Indirizzi

### *Anycast*

- Identifica il *server* più vicino al mittente che fornisce un dato servizio.
- Per il momento sono state definite alcune regole:
  - Non può essere usato come indirizzo di sorgente
  - Non può essere assegnato a *host* ma solo a *router*
- Per ora ne è stato definito solo uno:



- Che individua il *router* più vicino in una *subnet*

# IPv6- Indirizzi

## *Multicast*



000T

T = 0 indirizzo permanente ("*well-known*")

T = 1 indirizzo temporaneo ("*transient*")

Usato per limitare la diffusione

0 reserved

**1** *node-local scope*

**2** *link-local scope*

3-4 (unassigned)

**5** *site-local scope*

6-7 (unassigned)

**8** *organization-local scope*

9-D (unassigned)

**E** *global scope*

F reserved

# IPv6- Indirizzi

## *Multicast*

- Esempio dello *scope*: *Network Time Protocol* (NTP)
  - FF01::43 indica tutti i server NTP presenti sullo stesso nodo del mittente;
  - FF02::43 indica tutti i server NTP presenti sullo stesso link del mittente;
  - FF05::43 indica tutti i server NTP presenti sullo stesso sito del mittente;
  - FF0E::43 indica tutti i server NTP presenti sulla rete.

# IPv6- Indirizzi

## *Multicast*

- Alcuni degli indirizzi permanenti sono:
  - Tutti i nodi su un link
    - FF02:0000:0000:0000:0000:0000:0000:0001
  - Tutti i router su un link
    - FF02:0000:0000:0000:0000:0000:0000:0002
  - Tutti i server DHCP su un link
    - FF02:0000:0000:0000:0000:0000:0000:000C
  - Solicited Node Multicast Address (usato dal protocollo di Neighbor Discovery)
    - FF02:0000:0000:0000:0000:0001:xxxx:yyyy
  - dove xxxx:yyyy sono i 32 bit meno significativi di un indirizzo IPv6 unicast o anycast

# IPv6

## Indirizzi

- Quali indirizzi deve saper riconoscere un *host* come identificatori di se stesso?
  - Il suo indirizzo *Link Local* per ogni interfaccia
  - Gli indirizzi *unicast* assegnati alle interfacce
  - L'indirizzo di *loopback*
  - Il *multicast address* permanente che identifica tutti i nodi
  - I *multicast address* di *Neighbor Discovery* associati a tutti gli indirizzi unicast e anycast assegnati alle interfacce
  - I *multicast address* dei gruppi cui il nodo appartiene

# IPv6

## Indirizzi

- Quali indirizzi deve saper riconoscere un *router* come identificatori di se stesso?
  - Il suo indirizzo *Link Local* per ogni interfaccia
  - Gli indirizzi *unicast* assegnati alle interfacce
  - L'indirizzo di *loopback*
  - Il *Subnet Router anycast address* per tutti i *link* su cui ha interfacce
  - Gli altri indirizzi *anycast* assegnati alle interfacce
  - Il *multicast address* permanente di tutti i nodi
  - Il *multicast address* permanente di tutti i *router*
  - I *multicast address* di *Neighbor Discovery* associati a tutti gli indirizzi *unicast* e *anycast*
  - I *multicast address* dei gruppi cui il nodo appartiene

IPv6

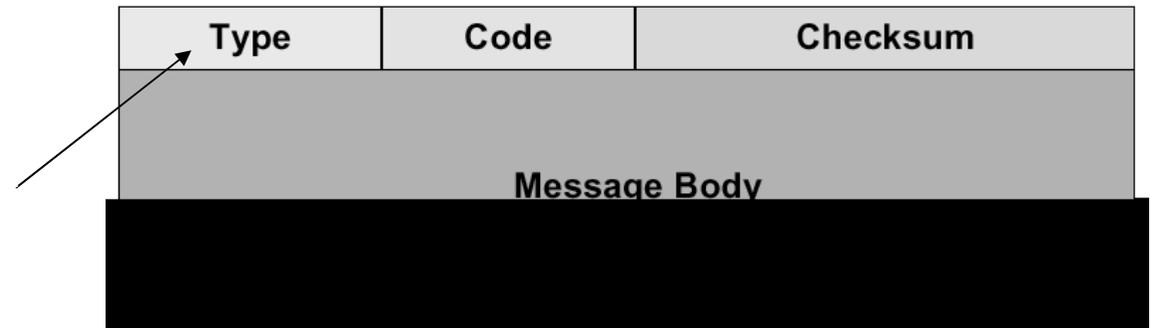
# ICMPv6

- L' *Internet Control Message Protocol v6 (ICMPv6)* ha tre impieghi principali
  - Diagnostica
  - *Neighbor Discovery*
  - Gestione dei gruppi multicast
- Riunisce al suo interno le funzionalità che in IPv4 erano suddivise tra:
  - ICMP
  - ARP (*Address Resolution Protocol*)
  - IGMP (*Internet Group Membership Protocol*)

# IPv6: ICMPv6

- Il messaggio ICMPv6 è trasportato in un pacchetto IPv6 ed è indicato dal valore 58 nel campo *Next Header*

- 1 Destination Unreachable
- 2 Packet too big
- 3 Time exceeded
- 4 Parameter Problem
- 128 Echo Request
- 129 Echo Reply
- 130 Group Membership Query
- 131 Group Membership Report
- 132 Group Membership Termination
- 133 Router Solicitation
- 134 Router Advertisement
- 135 Neighbor Solicitation
- 136 Neighbor Advertisement
- 137 Redirect



# IPv6: ICMPv6

## Destinazione non raggiungibile

Type	Code	Checksum
<b>Unused</b>		
<b>La prima parte del pacchetto che ha causato la trasmissione del messaggio ICMPv6 (il pacchetto ICMPv6 non deve eccedere 576 ottetti)</b>		

Code	Significato
0	No route to destination
1	Communication with destination admin. Prohibited
2	Not a neighbor
3	Address unreachable
4	Port unreachable

# IPv6: ICMPv6

**Pacchetto troppo grande**  
(ossia ha ecceduto la MTU in un qualche tratto del percorso)

Type	Code	Checksum
<b>MTU</b>		
<b>La prima parte del pacchetto che ha causato la trasmissione del messaggio ICMPv6 (il pacchetto ICMPv6 non deve eccedere 576 ottetti)</b>		

# IPv6: ICMPv6

- La precedente segnalazione di ICMPv6 viene usata dal *Path MTU Discovery*, che è un protocollo che permette la ricerca della dimensione ottimale del pacchetto per aumentare il Throughput
- Assume inizialmente come Path MTU il valore dell'MTU del primo link
  - ICMP notifica Path MTU errate
  - Memorizza le informazioni sul Path MTU
  - Cancellazione delle informazioni obsolete

# IPv6: ICMPv6

- Altre segnalazioni di errore sono fornite tramite:
  - *Time exceeded*: superato l'*Hop Limit*
  - *Parameter Problem* : problemi legati agli *header*
- *Echo Request* ed *Echo Reply* hanno sostanzialmente lo stesso uso di ICMP e sono messaggi di diagnostica

# IPv6: ICMPv6

- *Group Membership*, in sostanza ingloba le funzionalità di IGMP in ICMPv6

Type	Code	Checksum
Maximum Response Delay		Unused
Multicast Address		

Max tempo di attesa  
di una risposta alla  
*query* in ms

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

# IPv6: ICMPv6

- In IPv6 ARP scompare sostituito dalle nuove funzionalità di ICMP:
  - *Router e Prefix Discovery*
  - *Neighbor Discovery*
  - *Neighbor Unreachability Detection*
  - *Address Resolution*
  - *Next-Hop Determination*
  - *Duplicate Address Detection*

# IPv6: ICMPv6

## Router/Prefix Discovery

- *Router Advertisement* generati dai *router*:
  - *solicited*: in risposta a *Router Solicitation* da *host*
  - *unsolicited*: periodici
- Trasportano
  - indirizzo *link-local* e parametri del router
  - prefissi
- Prefissi hanno 2 scopi:
  - *Stateless Address Autoconfiguration*
  - determinazione nodi *on/off link*

# IPv6: ICMPv6

## ***Address Resolution***

- Una stazione che debba trasmettere un pacchetto verifica se l'indirizzo è locale (confronto con un *address prefix*) o remoto:
- Se è locale:
  - determina l'indirizzo tramite una *Neighbor Solicitation*
- Se è remoto:
  - sceglie un *router* tra quelli imparati tramite un *Router Advertisement*

# IPv6: ICMPv6

## ***Redirect***

- *Router* generano pacchetto di *Redirect* per informare un *host* di un miglior *first-hop*
- Il *first-hop* è sempre *on-link*, indipendentemente dal prefisso
- Quindi, a differenza di ICMP, la *redirect* permette di far comunicare direttamente due *host* con prefissi diversi ma connessi alla stessa rete fisica.