Università di Genova
Facoltà di Ingegneria

Telematica

7. TCP/IP - IPv6

Prof. Raffaele Bolla

dist

IPv6

- L'uso del CIDR ha solo temporaneamente risolto (o 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)

7.2

IPv6

- Dare supporto a più tipi di servizi (non solo al *best effort*).
- 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**

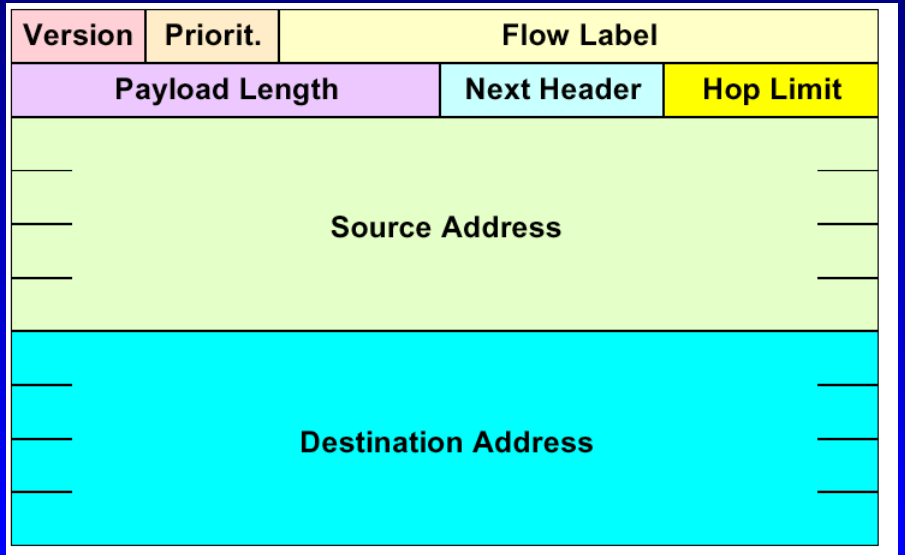
7.3

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 più tipi di servizi.

7.4

IPv6
Header



7.5

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).

7.6

IPv6
Header - Priority

Priorità crescente

CCT		Non CCT	
0	Non specificato Default	8	più scartabile (es. video alta qual.)
1	Di riempimento (es. news)	9	
2	Batch (es. email)	10	
3	Riservato	11	
4	Interattivo a bassa priorità (es. ftp, http)	12	
5	Riservato	13	
6	Interattivo ad alta priorità (es. Telnet, X)	14	
7	Di controllo (es. OSPF, SNMP)	15	meno scartabile (es. audio telefonico)

7.7

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.

7.8

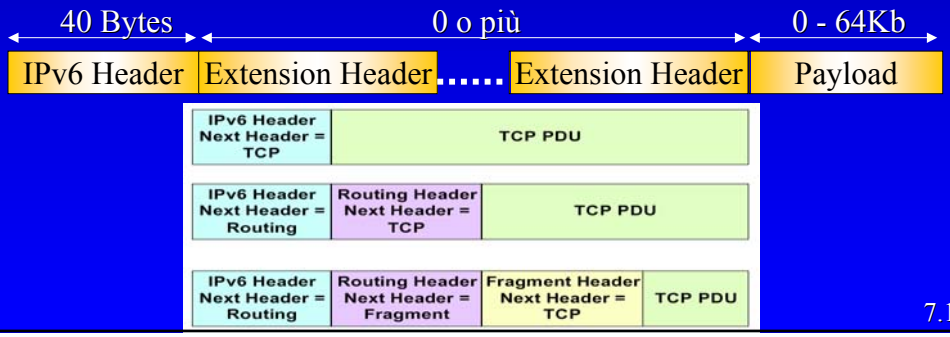
IPv6
Header

- ***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 +128 bits).

7.9

IPv6
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*.



7.10

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)

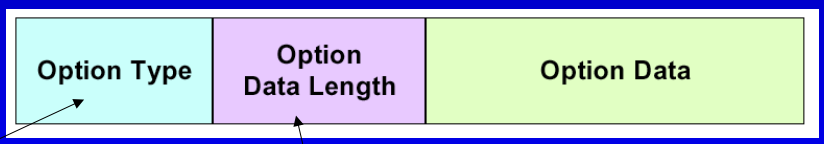
Gli *ExHeader* di IPv6 vanno inseriti (uno solo per tipo) ed elaborati nel seguente ordine:

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

7.11

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

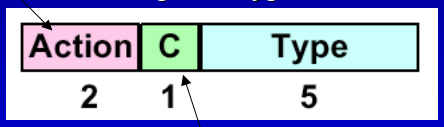
7.12

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

Option Type

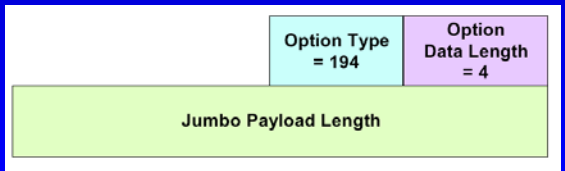


Specifica se l'opzione può (1) o non può (0) essere modificata lungo il percorso

7.13

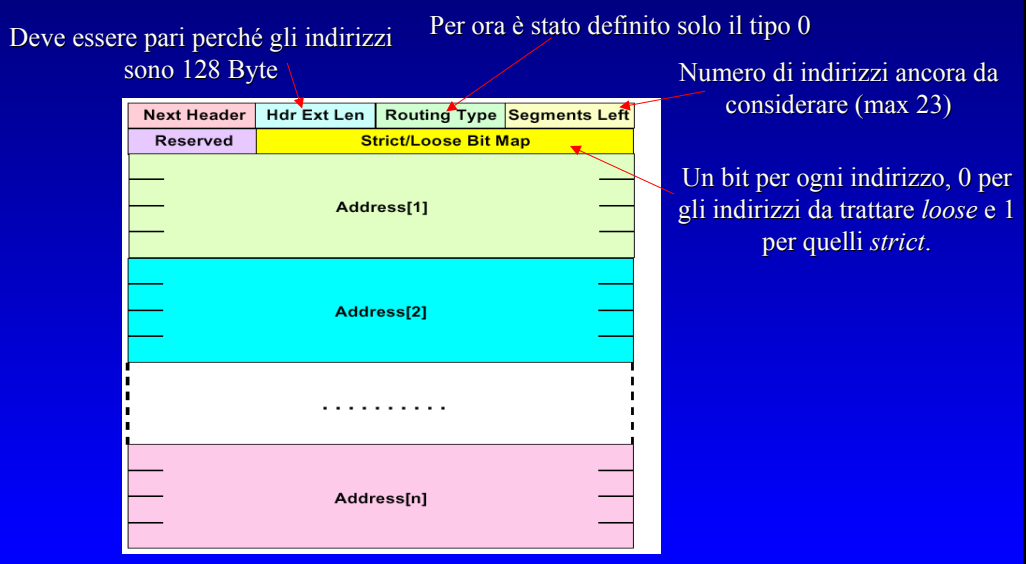
IPv6
Header - *Hop-by-Hop Header*

- 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.



7.14

IPv6
Header - *Routing Header*



7.15

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.

7.16

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 (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.

7.17

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.

7.18

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

7.19

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

7.20

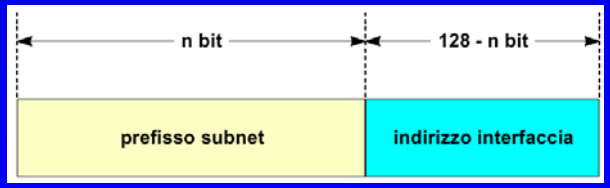
IPv6
Indirizzi

- 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

7.21

IPv6
Indirizzi (RFC 3513)

- 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
  - 11111110110111000000000010010001110000



7.22

IPv6
Indirizzi (RFC 3513)

Allocation	*Prefix*	*Fraction of Address Space*	
Reserved	0000 0000	1/256	
Unassigned	0000 0001	1/256	
Reserved for NSAP Allocation	0000 001	1/128	[RFC 1888]
Unassigned	0000 01	1/64	
Unassigned	0000 1	1/32	
Unassigned	0001	1/16	
Global Unicast	**001**	**1/8**	[RFC 3587]
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	
Link-Local Unicast Addresses	**1111 1110 10**	**1/1024**	
Site-Local Unicast Addresses	**1111 1110 11**	**1/1024**	
Multicast Addresses	**1111 1111**	**1/256**	

7.23

IPv6
Indirizzi *Reserved*

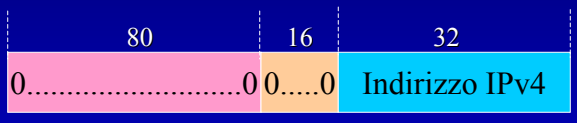
Tipo di indirizzo	*Prefisso binario*	*Notazione IPv6*
Unspecified	00...0 (128 bits)	::/128
Loopback	00…1 (128 bits)	::1/128
IPv4-compatible	00…0 (128 bits)	::/96
IPv4-mapped	0(80 bits)1(16 bits)	::FFFF:0:0/96

7.24

IPv6- Indirizzi
Reserved

Indirizzo IPv6 IPv4-compatible (::/96)

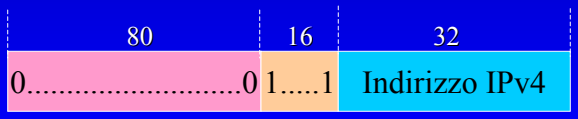
- L'indirizzo IPv4 usato deve essere un indirizzo unicast valido globalmente.
- Usato per fare tunnel di IPv6 su infrastruttura IPv4 (*automatic tunneling*)



Es. ::130.192.252.27

Indirizzo IPv6 IPv4-mapped (::FFFF:0:0/96)

- Usato per rappresentare indirizzi IPv4 nel formato nativo di IPv6.

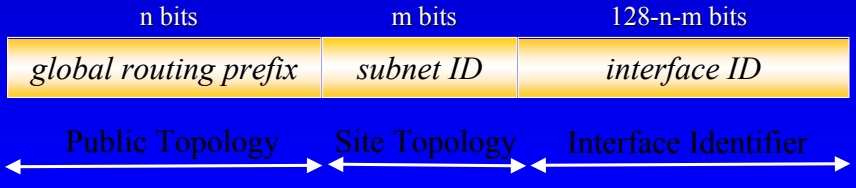


Es. ::FFFF:130.192.252.27

7.25

IPv6- Indirizzi *unicast*
Global Unicast Addresses – Formato generale

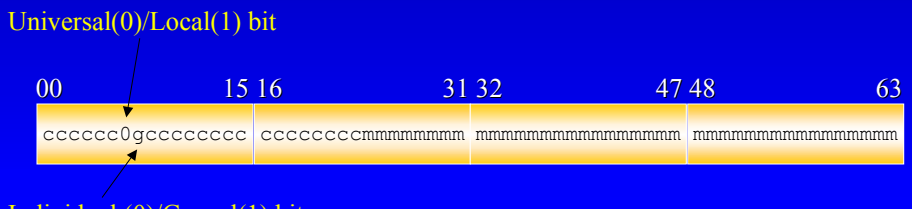
- **global routing prefix**: è un valore (tipicamente strutturato gerarchicamente) assegnato ad un singolo sito (definito come un insieme di sottoreti e *link*)
- **subnet id**: l'identificativo di una sottorete all'interno del sito; permette di organizzare in modo gerarchico il proprio sito.
- **interface ID**: l'identificativo della singola interfaccia all'interno della sottorete.



7.26

IPv6- Indirizzi *unicast*
Identificatori IEEE EUI-64

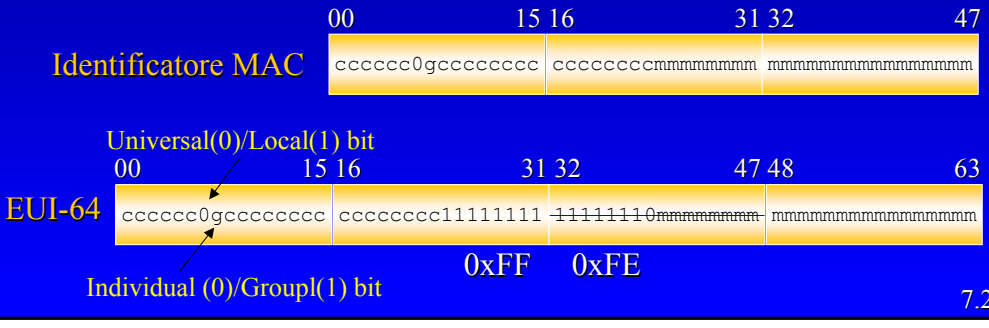
- L'IEEE EUI-64 (*Extended Unique Identifier*) è un identificatore a 64 bit assegnato dalla IEEE attraverso una apposita *Registration Authority*.
- I primi 24 bit (*company_id*, c) sono assegnati dall'IEEE alle aziende che ne fanno richiesta.
- I successivi 40 bit sono assegnati a discrezione della stessa azienda (m).
- Possono venire utilizzati per distinguere:
  - diversi protocolli o opzioni all'interno di uno standard;
  - diverse realizzazioni di dispositivi hardware.



7.27

IPv6- Indirizzi *unicast*
IEEE EUI-64 e indirizzi MAC (RFC 2464)

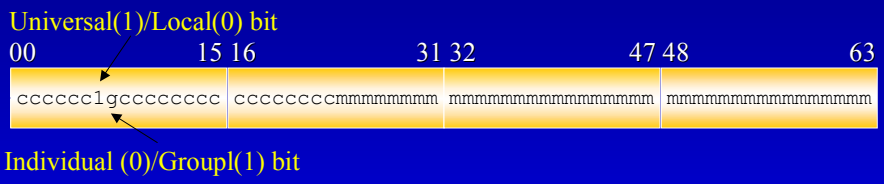
- Gli indirizzi MAC possono individuare univocamente una interfaccia di rete.
- L'IEEE prevede un meccanismo per derivare un identificativo EUI-64 a partire da un indirizzo MAC.
- Il procedimento consiste nell'inserire due ottetti, con valore 0xFF e 0xFE all'interno dell'indirizzo MAC stesso, tra l'identificativo dell'azienda (OUI, c) e l'identificativo del prodotto (*vendor supplied id*, m).



7.28

IPv6- Indirizzi *unicast*
Interface ID

- L'Interface ID consiste in un identificatore IEEE EUI-64 (*Extended Unique Identifier*) modificato.
- La modifica consiste nella diversa codifica del bit Universal/Local

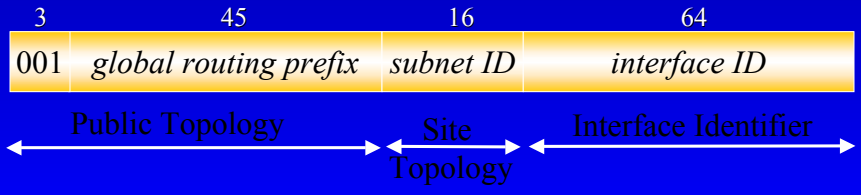


- In alcuni casi si preferisce usare identificatori locali dell'interfaccia (bit Universal/Local=0):
  - indirizzi mnemonici (es. server o router);
  - interfacce che non hanno indirizzi EUI-64 disponibili (es. link seriali o tunnel configurati); in questo caso si può utilizzare l'identificatore globale di un'altra interfaccia (o del nodo stesso, se disponibile) o un identificatore locale. L'identificativo scelto deve comunque essere univoco all'interno della specifica sottorete;
  - indirizzi link-local per interfacce virtuali IPv4 usate per fare tunneling di IPv6.

7.29

IPv6- Indirizzi *unicast*
Global Unicast Addresses - Assegnamento

- **L'IESG* a lo IAB° raccomandano di assegnare porzioni dello spazio di indirizzamento *global unicast* con il seguente criterio (RFC 3587):**
  - /48 in generale, ad eccezione di organizzazioni molto estese (per le quali si possono utilizzare allocazioni /47 o più prefissi /48);
  - /64 quando solo una sottorete è necessaria;
  - /128 quando è assolutamente certo che uno ed un solo dispositivo verrà connesso.
- Nel caso più frequente, il formato degli indirizzi assegnati dallo IANA risulterà:



* = Internet Engineering Steering Group
° = Internet Architecture Board
} Gruppi di lavoro interni all'IETF

7.30

IPv6- Indirizzi *unicast*
Global Unicast

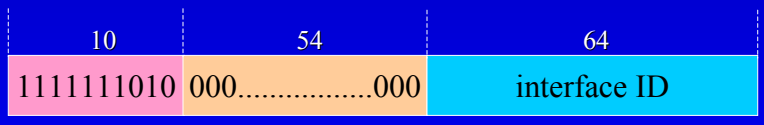
2000::/3 (001)

- 3ffe::/16 – ***6bone test address***
  - Sono stati i primi ad essere assegnati ed utilizzati;
  - una parte di essi non è globalmente unica: 3ffe:ffff::/32.
- 2002::/16 – ***6to4 tunneling***
  - Questi indirizzi sono riservati per un particolare meccanismo di *tunneling*, detto ***6to4*** e sono globali.
- 2001::/16 – ***hierarchical routing***
  - Assegnati agli ISP primari con prefisso /35;
  - gli ISP primari li riassegnano ad ISP secondari con prefisso /48.

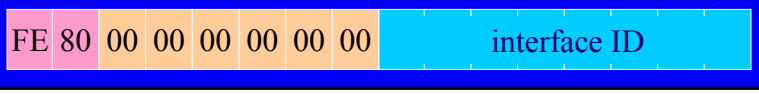
7.31

IPv6- Indirizzi *unicast*
Link Local

- Indirizzi "privati" (non annunciati dai *router*).
- Sono pensati per essere usati sul singolo *link* o comunque su reti non organizzate gerarchicamente.
- Possono venire utilizzate per:
  - configurazione automatica dell'indirizzo;
  - *neighbor discovery*;
  - in assenza di router sul link.



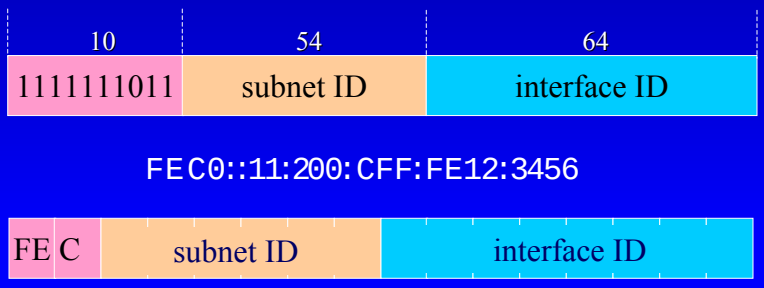
FE80::A00:2FF:FE12:3456



7.32

IPv6- Indirizzi *unicast*
Site Local

- Indirizzi "privati" (non annunciati).
- I *router* non propagano questi indirizzi fuori dal sito.
- Permettono la realizzazione di reti interne strutturate senza la necessità di avere un prefisso globale.
- Nel caso di siti con connessione globale, si ci aspetta che vengano utilizzati gli stessi *subnet ID* per gli indirizzi *site-local* e quelli globali (solo 16 bit).



7.33

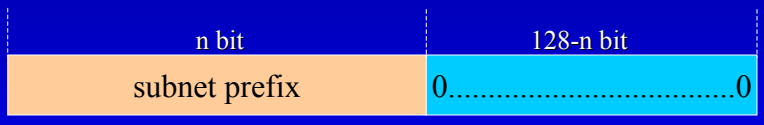
IPv6- Indirizzi
Anycast

- Un indirizzo *anycast* è globale ma viene assegnato a più interfacce.
- I pacchetti inviati con una destinazione *anycast* vengono inviati verso l'interfaccia con tale indirizzo più vicina.
- Gli indirizzi *anycast* sono assegnati all'interno dei *global unicast*:
  - risultano sintatticamente indistinguibili da quelli unicast;
  - il nodo a cui appartiene l'interfaccia deve essere esplicitamente configurato per riconoscere quell'indirizzo come *anycast*.
- Per ogni indirizzo *anycast* viene definito un prefisso **P** che identifica la regione in cui tutte le interfacce appartenenti a tale indirizzo risiedono:
  - all'interno della regione **P** le informazioni relative all'indirizzo *anycast* vengono mantenute come un elemento separato nelle *routing table;*
  - all'esterno di **P** l'indirizzo *anycast* è aggregato nella *routing entry* per il prefisso **P**.

7.34

IPv6- Indirizzi
Anycast

- 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 è stato definito solo l'indirizzo *anycast* per il *Subnet-Router*:



- Individua il *router* più vicino in una *subnet*.
- I pacchetti spediti a questo indirizzo saranno consegnati ad uno solo tra i router presenti sulla *subnet*.

7.35

IPv6- Indirizzi
Multicast

000T

T = 0 indirizzo permanente ("*well-known*")
T = 1 indirizzo temporaneo ("*transient*")

Usato per limitare la diffusione

0 reserved
1 ***interface-local scope***
2 ***link-local scope***
3 reserved
4 ***admin-local scope***
5 ***site-local scope***
6-7 (unassigned)
8 ***organization-local scope***
9-D (unassigned)
E ***global scope***
F reserved

7.36

IPv6- Indirizzi Multicast

- ***interface-local scope***: propaga il multicast solo su una interfaccia di un nodo ed è utile solo per trasmissioni *loopback* del *multicast*.
- ***link-local e site-local scope***: propagano il *multicast* nelle stesse regioni dei corrispondenti indirizzi *unicast*.
- ***admin-local scope***: è il più piccolo dominio di diffusione che può essere configurato amministrativamente (non derivabile automaticamente da connettività fisica o altre configurazioni indipendenti dal *multicast*).
- ***organization-local scope***: la diffusione del *multicast* avviene su più siti appartenenti alla stessa organizzazione.
- ***(unassigned)***: disponibili per la definizione di ulteriori regioni di *multicast*.

7.37

IPv6- Indirizzi *Multicast*

- Esempio dello *scope*: *Network Time Protocol* (NTP)
  - FF01::101 indica tutti i server NTP presenti sulla stessa interfaccia (cioè nodo) del mittente;
  - FF02::101 indica tutti i server NTP presenti sullo stesso link del mittente;
  - FF05::101 indica tutti i server NTP presenti sullo stesso sito del mittente;
  - FF0E::101 indica tutti i server NTP presenti sulla rete.
- Gli indirizzi non permanenti hanno significato solo entro un dato *scope*.
- Gli indirizzi *multicast* non possono essere usati come indirizzo sorgente in pacchetti IPv6.
- I nodi non devono generare pacchetti il cui scope sia 0 o F.

7.38

IPv6- Indirizzi
Multicast

- Indirizzo multicast di tutti i nodi:
  - FF01::1 (tutti i nodi su un'interfaccia)
  - FF02::1 (tutti i nodi su un link)
- Indirizzo multicast di tutti i router:
  - FF01::2 (tutti i router su un'interfaccia)
  - FF02::2 (tutti i router su un link)
  - FF05::2 (tutti i router su un sito)
- Indirizzo multicast *Solicited-Node*:
  - FF02::1:FFXX:XXXX
    - » XXXXXX sono i 24 bit meno significativi di un indirizzo IPv6 unicast o anycast;
    - » ogni nodo deve calcolare e configurare su ogni interfaccia un indirizzo *Solicited-Node* per ogni indirizzo *unicast* o *anycast* assegnato;
    - » utilizzato dal protocollo di *Neighbor Discovery* per limitare il numero di nodi interrogati.

7.39

IPv6
Indirizzi

- Quali indirizzi deve saper riconoscere un *host* come identificativo 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

7.40

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

7.41

IPv6
Transizione da IPv4 a IPv6

- La transizione da IPv4 ad IPv6 non può essere immediata:
  - aggiornamento protocolli software;
  - aggiornamento applicativi software (dns, web server, telnet, ftp, NAT, ssh, ecc.)
  - aggiornamento hardware;
  - tempo necessario per la riconfigurazione.
- La stesura graduale di una rete IPv6 presenta alcune problematiche:
  - coesistenza con IPv4:
    - » indirizzi multipli per un singolo *host*;
    - » *relay* verso *host* IPv4 non aggiornati a IPv6.
  - interconnessione di sistemi IPv6 non connessi a livello di linea:
    - » *tunneling* di IPv6 su infrastrutture IPv4 (assunta l'estensione globale di IPv4).

7.42

IPv6
Connessione di nodi IPv6

- La connessione di due nodi IPv6 può avvenire:
  - a livello di linea, se i due nodi sono collegati per es. da una linea seriale, una rete Ethernet o ATM;
  - tramite un tunnel su infrastruttura IPv4, dove i nodi non possono essere connessi direttamente da un livello di linea
    - » in questo caso spesso IPv4 è visto come un livello di linea virtuale.
- Esistono diverse metodologie per effettuare *tunneling* di IPv6 su un'infrastruttura IPv4:
  - *configured tunneling* (RFC 2893);
  - *automatic tunneling* (RFC 2893);
  - *6over4* (RFC 2529);
  - *6to4* (RFC 3056).

7.43

IPv6
Configured tunneling

- Il pacchetto IPv6 viene incapsulato in un pacchetto IPv4;
  - il numero di protocollo per IPv4 è 41.
- Il *tunnel* è configurato manualmente ed appare come un livello di linea (virtuale)
  - l'indirizzo IPv4 dell'host agli estremi del tunnel deve essere configurato manualmente;
  - i nodi agli estremi del tunnel devono possedere indirizzi link-local:
    - » sono formati dal prefisso FE80::/96 + l'indirizzo IPv4 come identificativo a 32 bit dell'interfaccia;
    - » servono ai protocolli di instradamento per identificare il next hop nelle tabelle di routing.
  - il tunnel può essere unidirezionale o bidirezionale.
- Solo i pacchetti provenienti da tunnel manualmente configurati sono accettati.

7.44

IPv6
Configured tunneling

- Host IPv6 per i quali non sono disponibili *router* direttamente raggiungibili a livello di linea possono utilizzare una *default route* su un *tunnel* configurato manualmente (***default configured tunnel***).
  - Questa soluzione si integra perfettamente con l'utilizzo di *tunnel* automatici.
- Il *default configured tunnel* di un host può essere essere terminato da un indirizzo *anycast* IPv4 appositamente allocato (***default configured tunnel with anycast address***); in questo caso il *default router* estrae i pacchetti IPv6 provenienti da qualsiasi sorgente IPv4 (se non esplicitamente limitato):
  - maggiore robustezza del *tunnel* (ci sono più alternative disponibili);
  - comportamento non corretto nel caso frammenti IPv4 di uno stesso pacchetto IPv6 siano consegnati a diversi *router*.

7.45

IPv6
Automatic tunneling

- L'indirizzo del terminatore remoto del tunnel è ricavato dall'indirizzo IPv4-compatibile della destinazione.
- Il *tunneling* automatico non può essere usato nei seguenti casi:
  - ***Router-to-Router***: il tunnel avviene tra due *router* in modo trasparente ai nodi sorgente-destinazione.
  - ***Host-to-Router***: la sorgente del pacchetto attiva il *tunnel* fino al primo *router*.

  ma solo nelle situazioni che prevedono l'invio del pacchetto alla destinazione finale, ossia nei casi
  - ***Host-to-Host:*** il tunnel viene stabilito direttamente tra sorgente-destinazione, senza router IPv6 intermedi.
  - ***Router-to-Host***: l'ultimo *router* IPv6 invia il pacchetto tramite tunnel alla destinazione finale.

7.46

IPv6
Automatic tunneling

- Gli indirizzi *IPv4-compatible* vengono assegnati solo a nodi che supportano il *tunneling* automatico.
- I meccanismi di incapsulamento ed estrazioni del pacchetto sono gli stessi del *tunnel* configurato.
- Una unica *entry* con prefisso ::/96 è necessaria per instradare tutti i pacchetti verso destinazioni *IPv4-compatible*
  - l'instradamento vero e proprio verso la destinazione avverrà all'interno della rete IPv4.
- Gli indirizzi IPv6 nativi non vengono mai instradati sul *tunnel*
  - per poter inviare pacchetti IPv6 nativi è necessario utilizzare un altro tipo di tunnel (es. *configured tunnel*).
- L'invio non deve avvenire per indirizzi IPv4 *multicast*, *broadcast*, non specificati o *loopback*.
  - il controllo sugli indirizzi avviene sia all'inizio che alla fine del tunnel.

7.47

IPv6
Automatic tunneling

- Il *tunneling* automatico è spesso utilizzato insieme al *tunneling* configurato:
  - il traffico verso *host IPv4-compatible* viene inviato tramite *tunnel* automatico;
  - il traffico IPv6 nativo viene instradato sul *tunnel* configurato (di solito un *default router* per IPv6).
- In questo caso si pone il problema di quale indirizzo utilizzare per la sorgente (*IPv4-compatible* o nativo IPv6)
  - la scelta influenza il meccanismo con il quale il traffico di risposta verrà inviato;
  - la tendenza è quella di favorire la simmetria del traffico:
    - » per invio verso destinazioni *IPv4-compatible* si utilizzerà l'indirizzo *IPv4-compatible* anche per la sorgente.
    - » per il traffico verso *host* IPv6 nativi si utilizzerà un indirizzo *global unicast*.

7.48

IPv6
6to4

- È stato pensato per offrire connettività IPv6 a siti o singoli *host* isolati.
- Richiede una configurazione minima.
- Il suo utilizzo implica:
  - una scelta particolare per gli indirizzi;
  - la configurazione dei soli *router* di bordo tra il dominio IPv6 e IPv4.
- Può essere utilizzato insieme ad altre tecniche di tunneling:
  - *automatic tunneling;*
  - *configured tunneling*.

7.49

IPv6
Indirizzi *6to4*

- Il sito deve avere un indirizzo IPv4 unico e globale (V4ADDR).
- Il prefisso 2002::/16 è stato permanentemente allocato dallo IANA per il meccanismo *6to4*.
- La struttura degli indirizzi *6to4* risulta la seguente:

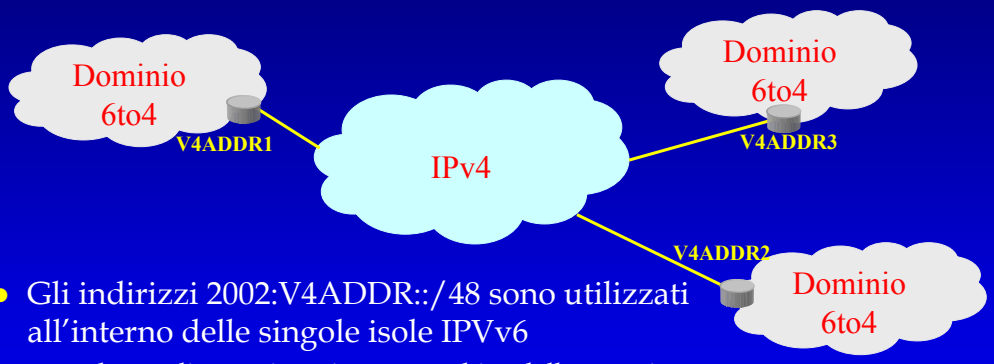
- Risulta avere un prefisso 2002:V4ADDR::/48, così come specificato nell'RFC 3587.
- L'indirizzo IPv6 così ricavato risulta univoco e globale; può essere utilizzato come un qualsiasi indirizzo IPv6.

7.50

IPv6
Indirizzi *6to4*

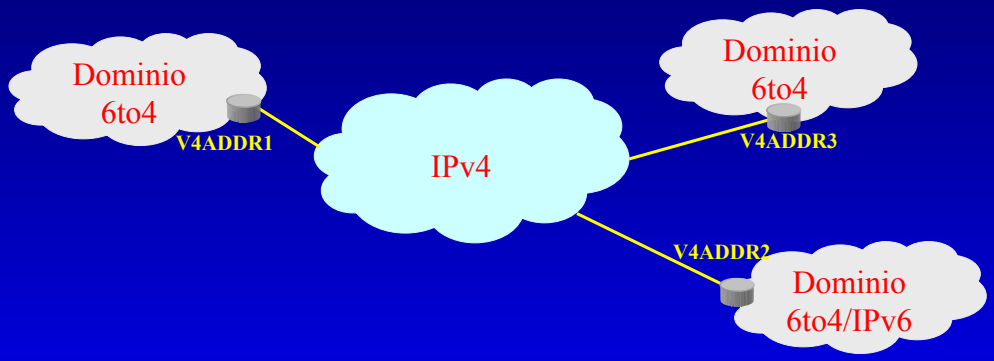
- L'indirizzo IPv4 del nodo di confine (6to4 *border router*) deve essere V4ADDR.
- Selezione dell'indirizzo da utilizzare per sorgente/destinazione:
  - se entrambi hanno un indirizzo 6to4 e solo uno dei 2 ha anche un indirizzo IPv6 nativo:
    - » si scegli l'indirizzo 6to4 per entrambi
  - nel caso siano disponibili indirizzi IPv6 nativi e 6to4 per entrambi gli host:
    - » si dovrebbe usare lo stesso tipo di indirizzo per entrambi;
    - » la scelta dovrebbe essere configurabile;
    - » il comportamento di default dovrebbe essere l'utilizzo degli indirizzi IPv6 nativi.
- L'indirizzo *link-local* non è richiesto; se serve può essere calcolato come per il *tunneling* configurato (RFC 2893). 7.51

IPv6
Scenario *6to4-only*



- Gli indirizzi 2002:V4ADDR::/48 sono utilizzati all'interno delle singole isole IPVv6
  - nel caso di organizzazione gerarchica della rete viene utilizzato internamente il normale routing IPv6.
- I pacchetti indirizzati ad indirizzi 2002::/16 vengono gestiti come non-locali ed inviati al *router 6to4* di bordo.
- Tra i diversi domini non è necessario il routing IPv6: la connettività tra le diverse isole è assicurata da IPv4. 7.52

IPv6
Scenario *6to4-mixed*



- In questo scenario uno o più siti hanno connettività IPv6 nativa (per es. attraverso tunnel configurati).
- I *router 6to4* con indirizzi IPv6 sono detti *relay router*.
- Il *relay router* può essere l'unico componente *6to4* nel dominio misto *6to4*/IPv6.

7.53

IPv6
Scenario *6to4-mixed*

- Politiche di routing:
  - interne ai domini 6to4,
    - » come specificate nel caso *6to4-only*
  - tra router *6to4* e *relay router*,
    - » nessun protocollo di routing, i domini 6to4 usano una default route IPv6 verso un relay router configurato manualmente
      - ✓ i relay router possono accettare traffico in ingresso solo da alcuni router 6to4.
    - » protocollo di routing IPv6 esterno (es. BGP4+, RFC 2283):
      - ✓ ogni relay router annuncia i prefissi IPv6 per i quali è disposto ad accettare traffico;
      - ✓ nessun router 6to4 deve propagare un prefisso 2002::/16;
      - ✓ il relay router coopera solo con i router 6to4 dei domini per i quali è disposto ad effettuare il relay.
  - tra router IPv6
    - » i router del dominio IPv6 nativo devono propagare solo il prefisso 2002::/16 (onde evitare la crescita delle RT);
    - » è compito dei gestori di rete filtrare opportunamente i prefissi 2002::/16 propagati da diversi relay router.

7.54

IPv6
Relay routers

- Internamente al proprio dominio IPv6:
  - partecipano all'instradamento;
    - » IPv6 nativo;
    - » 6to4.
  - propagano il prefisso 2002::/16.
- Esternamente al dominio IPv6:
  - partecipano all'instradamento IPv4;
  - se non si usa il BGP4+:
    - » accettano il traffico IPv6 proveniente dai *client 6to4* configurati;
  - se si usa il BGP4+:
    - » annunciano la propria rete IPv6 ai propri client 6to4;
    - » possono annunciare una *default route* IPv6, se collegati con la rete globale IPv6.
- Possono avere un indirizzo IPv4 *anycast* per raccogliere singoli utenti (es. dial-up).

7.55

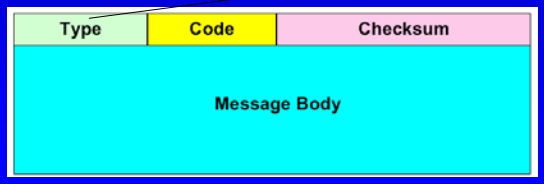
IPv6
ICMPv6

- 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*)

7.56

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

7.57

IPv6
ICMPv6

Destinazione non raggiungibile

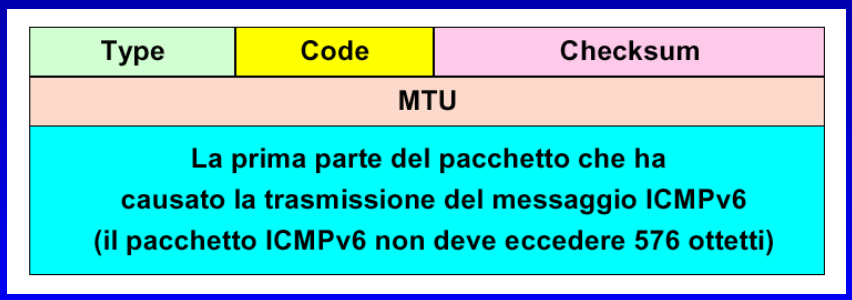
Type	Code	Checksum
Unused		
La prima parte del pacchetto che ha causato la trasmissione del messaggio ICMPv6 (il pacchetto ICMPv6 non deve eccedere 576 ottetti)		

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

7.58

IPv6
ICMPv6

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



7.59

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.

7.60

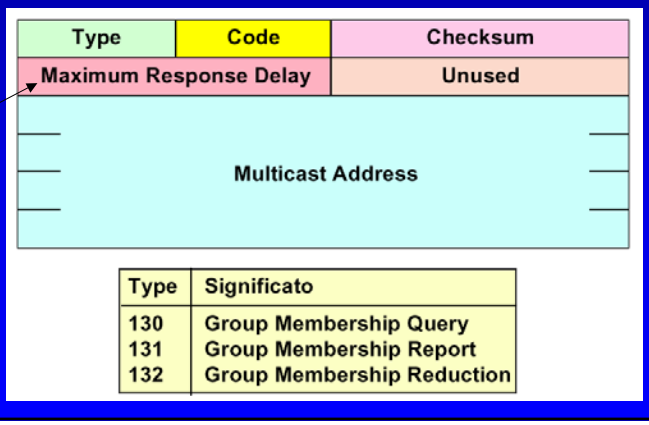
IPv6
ICMPv6

- 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

7.61

IPv6
ICMPv6

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



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

7.62

IPv6

ICMPv6

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

7.63

IPv6

ICMPv6 – *Neighbor Discovery*

- Il protocollo di *Neighbor Discovery* definisce 5 tipologie di pacchetti ICMP:
  - ***Router Advertisement***: utilizzati dai router per annunciare la loro presenza e comunicare alle stazioni diversi parametri (prefissi, *hop limit*, etc.);
  - ***Router Solicitation***: richiedono la generazione immediata di un pacchetto di *Router Advertisement*;
  - ***Neighbor Solicitation***: inviato per determinare l'indirizzo di livello 2 di una stazione o per verificarne il valore presente in cache;
  - ***Neighbor Advertisement***: in risposta al pacchetto di *Neighbor Solicitation* o per comunicare un cambiamento dell'indirizzo di linea;
  - ***Redirect***: usato dai *router* per informare gli *host* di un miglior *first hop* relativamente ad una destinazione.

7.64

IPv6

ICMPv6 – *Neighbor Discovery*

Router Advertisement

- Generati periodicamente o in risposta ad un *Router Solicitation*.
- Indirizzati al nodo richiedente o all'indirizzo *multicast* di tutti i nodi.
- Trasportano:
  - Cur Hop Limit (default Hop Limit da usare)
  - Flags (*Managed address configuration*, per utilizzare il meccanismo di autoconfigurazione *stateful* per l'indirizzo oltre al meccanismo stateless, *Other stateful configuration*, per la configurazione *stateful* di altri parametri diversi dall'indirizzo);
  - *Router Lifetime* (tempo di vita se usato come *default router*);
  - *Reachable Time* (tempo per cui si assume che un vicino sia raggiungibile dopo aver ricevuto una conferma di raggiungibilità);
  - *Source Link-layer address* (del *router*);
  - MTU (per le linee con MTU variabile);
  - *Prefix information*:
    - » determinazione prefissi *on-link*;
    - » configurazione *stateful* (DHCPv6) o *stateless* (autonomatic).

7.65

IPv6

ICMPv6 – *Neighbor Discovery*

Router Solicitation

- Generati dagli host per ottenere Router Advertisement rapidamente.
- Indirizzati in genere all'indirizzo multicast di tutti i router.
- Trasportano:
  - Source Link-layer address (del mittente).

7.66

IPv6

ICMPv6 – *Neighbor Discovery*

Neighbor Solicitation

- Generati dagli *host* per ottenere l'indirizzo di livello 2 di altri *host* sullo stesso *link*.
- Indirizzati:
  - al *solicited-address* nel caso di risoluzione di indirizzi;
  - all'indirizzo *unicast* del nodo nel caso di verifica di raggiungibilità di un nodo.
- Trasportano:
  - *Target address* (indirizzo da risolvere);
  - *Source Link-layer address* (del mittente).

7.67

IPv6

ICMPv6 – *Neighbor Discovery*

Neighbor Advertisement

- Generati dagli host
  - in risposta a *neighbor solicitation*;
  - *unsolicited*, per comunicare nuove informazioni rapidamente.
- Indirizzati:
  - al nodo che ha inviato il *solicited-address*;
  - all'indirizzo *multicast* di tutti i nodi nel caso sia *unsolicited*.
- Trasportano:
  - *Flags* (Router flag, se il mittente è un router, Solicited flag, per risposte a *neighbor solicitation*, Override flag, permette l'aggiornamento di informazioni già presenti in cache);
  - *Target address* (indirizzo richiesto da Neighbor Solicitation o interessato dall'aggiornamento per messaggi *unsolicited*);
  - *Target Link-layer address* (del mittente).

7.68

IPv6

ICMPv6 – *Neighbor Discovery*

Redirect

- I *router* inviano un pacchetto *Redirect* per informare un nodo di un miglior *first-hop*:
  - il first hop può coincidere con la stessa destinazione;
  - indipendentemente dal prefisso, il *first-hop* è sempre on-link
    - » a differenza di ICMP due destinazioni con prefissi diversi sulla stessa rete possono comunicare direttamente.
- Trasportano:
  - *Target Address* (indirizzo del *first-hop*, coincide con il destination address se la destinazione è on-link);
  - *Destination Address* (la destinazione per cui è valida il reindirizzamento);
  - *Target link-layer address* (indirizzo link layer per target);
  - *Redirected Header* (quanta più informazione possibile del pacchetto che ha generato la *redirect*, senza eccedere i 1280 bytes).

7.69

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*

7.70