Indirizzi fisici e indirizzi IP: protocollo ARP (1)

A ogni nodo di una rete TCP/IP viene assegnato un indirizzo logico IP che lo identifica univocamente sulla rete.

Affinché due host possano comunicare tra loro è necessario anche che le rispettive schede di rete siano capaci di localizzarsi reciprocamente.

Occorre che l'indirizzo fisico MAC del destinatario sia noto al mittente.

In IPv4 chi si occupa di mappare un indirizzo IP noto nel corrispondente indirizzo MAC sconosciuto, è il **protocollo ARP** (*Address Resolution Protocol*).

ARP (Address Resolution Protocol, RFC 826)

Il protocollo ***ARP*** serve per derivare, dall'indirizzo IP dell'host di destinazione, l'indirizzo di livello data link necessario per inviare il frame che incapsulerà il pacchetto destinato all'host di cui all'indirizzo IP.

Esso opera appoggiandosi direttamente sul livello data link e non su IP:

- viene inviata a tutte le stazioni della LAN, in data link broadcast, una richiesta del tipo: "chi ha l'indirizzo IP uguale a 190.3.6.7 ?"
- solo l'host che ha quell'indirizzo IP risponde, inserendo nella risposta il proprio indirizzo data link, salvando anch'esso l'informazione nella sua ARP cache;
- quando riceve la risposta, l'host la mantiene in memoria per un tempo stabilito dal sistema operativo.

Se l'indirizzo IP è relativo ad un'altra network:

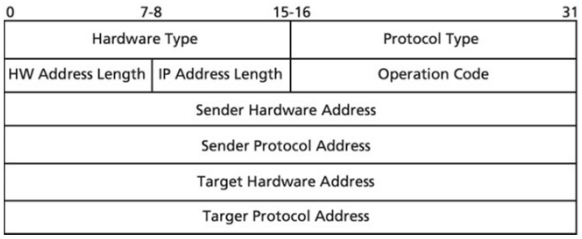
- alternativamente, si configura l'host impostando al suo interno l'indirizzo ethernet di default (quello del router) a cui mandare i pacchetti IP per le altre reti; anche in questo caso il router deve fare da tramite nella conversazione IP fra il mittente e il destinatario.

- la soluzione più semplice è mandare il pacchetto ARP come prima, configurando però il router in modo che risponda alle richieste ARP relative ad altre reti fornendo il proprio indirizzo ethernet(***proxy ARP***); il router farà poi da tramite nella conversazione IP fra il mittente e il destinatario, inviando di volta in volta all'uno i pacchetti IP che gli giungono dall'altro;

Indirizzi fisici e indirizzi IP: protocollo ARP (2)

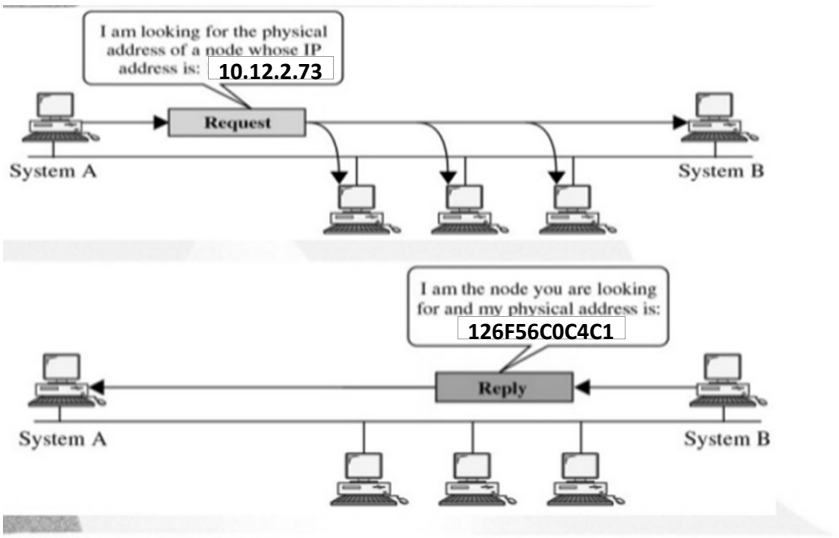
Un'implementazione TCP/IP usa di norma una **cache ARP**, dove ogni host mantiene e aggiorna una tabella con tutte le coppie IP-MAC a esso note.

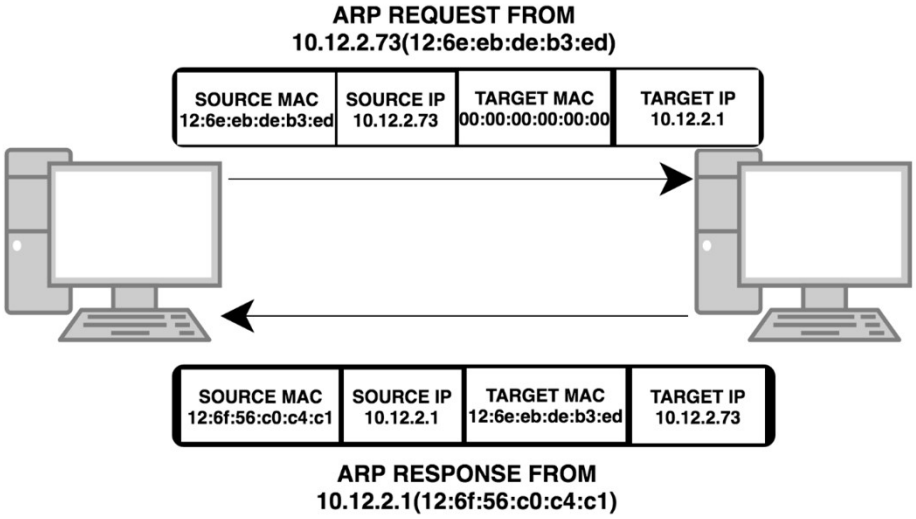
Il pacchetto ARP si divide in 9 campi.



0	7-8	15-16	31
Hardware Type		Protocol Type	
HW Address Length	IP Address Length	Operation Code	
Sender Hardware Address			
Sender Protocol Address			
Target Hardware Address			
Targer Protocol Address			

HRD (16bit): Tipo di hardware usato. Nel caso più comune si tratta di Ethernet e corrisponde al valore 1.
PRO (16bit): è il protocollo usato. IP ha valore 0x0800
HLN (8bit): Lunghezza in byte dell'indirizzo hardware (nel caso di Ethernet è il MAC address e vale 6).
PLN (8bit): Lunghezza in byte del protocollo usato (nel nostro caso, IPv4, è 4).
OP (16bit): Tipo di pacchetto. Nel nostro caso può essere REQUEST (1) o REPLY (2).
SHA (bytes): MAC address del mittente; ha una lunghezza che dipende da HLN.
SPA (bytes): Indirizzo IP del mittente; ha una lunghezza che dipende da PLN.
THA (bytes): MAC address del ricevente (dipende sempre da HLN).
TPA (bytes): Indirizzo IP del ricevente (dipende da PLN).





RARP (Reverse Address Resolution Protocol, RFC 903)

Il protocollo ***RARP*** risolve il problema inverso, cioè consente di trovare quale indirizzo IP corrisponda a un determinato indirizzo data link.

Il RARP consente anche ad un host di conoscere il proprio indirizzo IP all'accensione chiedendolo, in modalità broadcast agli altri host connessi alla rete. In genere la richiesta arriva ad un ***server RARP*** che contiene l'indirizzo di risposta nei propri file di configurazione.

Ormai è reso obsoleto dal BOOTP e dal suo successore DHCP più ricchi di funzionalità.

Vulnerabilità

L'ARP poisoning , (letteralmente avvelenamento dell'ARP) (detto anche **ARP spoofing**, letteralmente falsificazione dell'ARP) è una tecnica di hacking che consente ad un attacker, in una switched lan, di concretizzare un attacco di tipo *man in the middle* verso tutte le macchine che si trovano nello stesso segmento di rete quando queste operano a livello 3 cioè di internetworking con altre sottoreti scambiandosi traffico IP grazie al ricorso ad opportune manipolazioni tramite i protocolli di livello 2.
L'ARP poisoning consiste nell'inviare intenzionalmente e in modo forzato risposte ARP contenenti dati inesatti o, meglio, non corrispondenti a quelli reali.
In questo modo la tabella ARP (ARP entry cache) di un host conterrà dati alterati (da qui i termini poisoning, letteralmente avvelenamento e spoofing, raggiro).
Molto spesso lo scopo di questo tipo di attacco è quello di ridirezionare, in una rete commutata, i pacchetti destinati ad un host verso un altro al fine di leggere il contenuto di questi per catturare le password che in alcuni protocolli viaggiano in chiaro.

Vulnerabilità

Questo attacco si basa su una debolezza intrinseca nel protocollo ARP: la mancanza di un meccanismo di autenticazione.

Ethernet, il più diffuso standard per le reti locali, identifica gli host in base ad un indirizzo a 48 bit chiamato MAC, a differenza di Internet, dove ciascun host viene mappato grazie ai 32 bit del protocollo IP.

Il protocollo ARP si occupa di gestire l'associazione tra indirizzi IP e indirizzi MAC. Quest'associazione, in Ethernet, viene fatta prima di ogni tipo di comunicazione.

Un ipotetico host 192.168.1.1 che vuole comunicare con l'host 192.168.1.2 manderà una ARP request in broadcast con il proprio MAC il proprio indirizzo IP e l'indirizzo IP di destinazione; quando 192.168.1.2 riceverà l'ARP request risponderà con un'ARP reply destinato al MAC sorgente e contenente il proprio MAC. Per ottimizzare le prestazioni e limitare il traffico queste informazioni (associazione indirizzo IP/indirizzo MAC) vengono memorizzate nella tabella ARP (ARP cache) di ciascun host così che non sia necessario effettuare continue richieste per successivi eventuali indirizzamenti verso terminali host già noti. Per migliorare ancora di più le prestazioni quando si ricevono delle ARP reply (alcuni anche con le ARP request), anche se non sollecitate, gli host aggiornano le informazioni della propria ARP cache.

Vulnerabilità

Ora si analizzi il seguente scenario:

Attacker: IP = 192.168.1.2, MAC = 00:00:00:ZZ:ZZ:ZZ
John: IP = 192.168.1.13, MAC = 00:00:00:JJ:JJ:JJ
Linus: IP = 192.168.1.88, MAC = 00:00:00:LL:LL:LL
Le ARP cache di ciascun host prima dell'attacco saranno:

Per l'attacker:
192.168.1.13, MAC = 00:00:00:JJ:JJ:JJ
192.168.1.88, MAC = 00:00:00:LL:LL:LL
Per John:
192.168.1.2, MAC = 00:00:00:ZZ:ZZ:ZZ
192.168.1.88, MAC = 00:00:00:LL:LL:LL
Per Linus:
192.168.1.2, MAC = 00:00:00:ZZ:ZZ:ZZ
192.168.1.13, MAC = 00:00:00:JJ:JJ:JJ
Per realizzare l'ARP poisoning l'attacker invierà delle ARP reply opportunamente costruite/modificate: a John invierà una reply che ha come IP quello di Linus (192.168.1.88) ma come MAC il proprio (00:00:00:ZZ:ZZ:ZZ), a Linus invierà una reply con IP quello di John (192.168.1.13) e con MAC, anche questa volta, il proprio (00:00:00:ZZ:ZZ:ZZ). Per protrarre l'attacco è necessario inviare delle ARP reply ogni 10 secondi poiché spesso i sistemi operativi cancellano sistematicamente le voci dell'ARP cache dopo un certo periodo di

Vulnerabilità

Per realizzare l'ARP poisoning l'attacker invierà delle ARP reply :
a John invierà una reply che ha come IP quello di Linus (192.168.1.88) ma come MAC il proprio (00:00:00:ZZ:ZZ:ZZ),
a Linus invierà una reply con IP quello di John (192.168.1.13) e con MAC, anche questa volta, il proprio (00:00:00:ZZ:ZZ:ZZ).

Per protrarre l'attacco è necessario inviare delle ARP reply ogni 10 secondi poiché spesso i sistemi operativi cancellano sistematicamente le voci dell'ARP cache dopo un certo periodo di tempo.
Quindi dopo l'attacco le ARP cache di ciascun host saranno appunto avvelenate ovvero falsificate o corrotte:

Per l'attacker:
192.168.1.13, MAC = 00:00:00:JJ:JJ:JJ
192.168.1.88, MAC = 00:00:00:LL:LL:LL
Per John:
192.168.1.2, MAC = 00:00:00:ZZ:ZZ:ZZ
192.168.1.88, MAC = 00:00:00:ZZ:ZZ:ZZ
Per Linus:
192.168.1.2, MAC = 00:00:00:ZZ:ZZ:ZZ
192.168.1.13, MAC = 00:00:00:ZZ:ZZ:ZZ

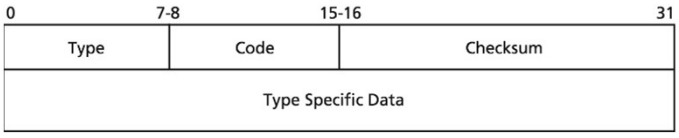
Vulnerabilità

Quando le due vittime, John e Linus, instaureranno una comunicazione tra loro, crederanno di comunicare reciprocamente, ma in realtà comunicheranno con l'attacker il quale, per mostrare trasparenza e regolarità nella comunicazione tra i due host e continuare quindi a sniffare il relativo traffico, inoltrerà il traffico proveniente da John verso Linus e viceversa il traffico proveniente da Linus verso John, realizzando così un MITM.

Il monitoring della rete con il protocollo ICMP (1)

L'**ICMP** (*Internet Control Message Protocol*) fornisce un meccanismo di monitoraggio della rete, utilizzato prevalentemente dai router o dagli host destinatari per segnalare agli host mittenti eventuali insuccessi nell'instradamento dei pacchetti.

Il pacchetto ICMP



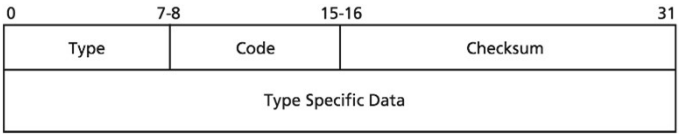
Il monitoring della rete con il protocollo ICMP (2)

ICMP consente ai router di scambiarsi informazioni di servizio (**messaggi router-to-router**) e di tenere sotto controllo le modalità con cui gli host generano pacchetti, inviando loro messaggi per rallentare o dirottare altrove un flusso di pacchetti (**messaggi router-to-host**).

Per quanto riguarda gli host, ICMP consente loro di scambiarsi informazioni di servizio (**messaggi host-to-host**) e di richiedere ai router informazioni utili sul funzionamento e la topologia della rete (**messaggi host-to-router**).

Due comandi molto utili per monitorare la rete sono **ping** e **tracert**.

Il monitoring della rete con il protocollo ICMP (3)



Il campo **Code** fornisce indicazioni aggiuntive non comprese nel campo Type.
Il campo **Checksum** contiene, come al solito, i bit per il controllo degli errori di trasmissione.
Il campo **Type Specific Data** contiene informazioni che dipendono dal tipo di servizio che l'ICMP sta offrendo. Per esempio le più comuni Echo Request/ Echo Reply comprendono un identificatore e un numero sequenziale che servono a identificare ciascuna richiesta di eco e ciascuna risposta.

Le principali funzioni che il protocollo ICMP può svolgere sono:

- fornire messaggi di eco per verificare la corretta configurazione di host sulla rete e che quindi una qualsiasi destinazione sia raggiungibile: Echo Request (type 8) del mittente, Echo Reply (type 0) del destinatario. Si realizza con il comando **ping** (di cui vedremo un esempio nel paragrafo di seguito);
- segnalare una destinazione non raggiungibile perché sconosciuta o perché un pacchetto è troppo grande ma non è consentito frammentarlo: Destination Unreachable (type 3);
- avvertire il mittente di rallentare l'invio dei pacchetti per problemi di congestione: Source Quench (type 4);
- reindirizzare il traffico per fornire un instradamento efficiente in caso di router congestionato da traffico eccessivo: Routing Redirect (type 5);
- avvertire il mittente che il tempo di vita di un suo pacchetto è scaduto (TTL = 0) e che quindi il pacchetto viene scartato: Time Exceeded (type 11);
- valutare le prestazioni di una rete misurando il tempo di attraversamento: Timestamp Request (type 13) del mittente, Timestamp Reply (type 14) del destinatario;
- rilevazione della lista dei nodi (router) attraversati da un pacchetto per giungere a destinazione: Traceroute (type 30). Si realizza con il comando **tracert** (anche di questo comando vedremo un esempio alla fine di questa lezione).

ICMP

Type 0 — Echo Reply
Type 1 — Unassigned
Type 2 — Unassigned
Type 3 — Destination Unreachable
Type 4 — Source Quench (Deprecated)
Type 5 — Redirect
Type 6 — Alternate Host Address (Deprecated)
Type 7 — Unassigned
Type 8 — Echo
Type 9 — Router Advertisement
Type 10 — Router Selection
Type 11 — Time Exceeded
Type 12 — Parameter Problem
Type 13 — Timestamp
Type 14 — Timestamp Reply
Type 15 — Information Request (Deprecated)
Type 16 — Information Reply (Deprecated)
Type 17 — Address Mask Request (Deprecated)
Type 18 — Address Mask Reply (Deprecated)
Type 19 — Reserved (for Security)
Types 20-29 — Reserved (for Robustness Experiment)

ICMP

Type 30 — Traceroute (Deprecated)
Type 31 — Datagram Conversion Error (Deprecated)
Type 32 — Mobile Host Redirect (Deprecated)
Type 33 — IPv6 Where-Are-You (Deprecated)
Type 34 — IPv6 I-Am-Here (Deprecated)
Type 35 — Mobile Registration Request (Deprecated)
Type 36 — Mobile Registration Reply (Deprecated)
Type 37 — Domain Name Request (Deprecated)
Type 38 — Domain Name Reply (Deprecated)
Type 39 — SKIP (Deprecated)
Type 40 — Photuris
Type 41 — ICMP messages utilized by experimental mobility protocols such as Seamoby
Type 42 — Extended Echo Request
Type 43 — Extended Echo Reply
Types 44-252 — Unassigned
Type 253 — RFC3692-style Experiment 1
Type 254 — RFC3692-style Experiment 2

Type 3 — Destination Unreachable CODES

0	Net Unreachable
1	Host Unreachable
2	Protocol Unreachable
3	Port Unreachable
4	Fragmentation Needed and Don't Fragment was Set
5	Source Route Failed
6	Destination Network Unknown
7	Destination Host Unknown
8	Source Host Isolated
9	Communication with Destination Network is Administratively Prohibited
10	Communication with Destination Host is Administratively Prohibited
11	Destination Network Unreachable for Type of Service
12	Destination Host Unreachable for Type of Service
13	Communication Administratively Prohibited
14	Host Precedence Violation
15	Precedence cutoff in effect

ICMP

https://www.iana.org/assignments/icmp-parameters/icmp-parameters.xhtml

PING

Se si verificano problemi di connettività, è possibile utilizzare il comando **ping** per controllare la raggiungibilità di un qualsiasi indirizzo IP e visualizzare i risultati ottenuti. Il comando ping indica se è stata restituita una risposta dalla destinazione e quanto tempo è trascorso prima di riceverla. Se si verifica un errore nella consegna, il comando ping visualizza un messaggio di errore.

È possibile utilizzare il comando ping per:

1. determinare se il protocollo TCP/IP è correttamente configurato sulla propria macchina;
2. determinare se è in esecuzione il router/gateway (rete locale raggiungibile);
3. determinare se un indirizzo Internet è raggiungibile (host remoto raggiungibile).

Utilizza pacchetti ICMP tipo:

8 Servizi Echo Request / 0 Echo Reply

TRACERT

Si occupa di ricavare il percorso seguito dai pacchetti sulle reti informatiche, ovvero l'indirizzo IP di ogni router attraversato per raggiungere il destinatario.

Il campo TTL

Questo comando sfrutta una particolare caratteristica del protocollo IP, ovvero il campo TTL (time to live) del messaggio. Questo campo specifica il numero degli apparati di rete che il pacchetto potrà attraversare prima di essere dichiarato scaduto.

Il funzionamento di questo campo è semplice: ogni router che riceve il pacchetto, prima di inviarlo nuovamente, diminuisce il campo di un'unità. Se si accorge che il campo ha assunto il valore 0, invia al mittente del pacchetto un messaggio di errore ICMP (Tipo 11) specificando l'indirizzo del router che l'ha generato.

TRACERT

Funzionamento

Invia un pacchetto al destinatario di cui si vuole ricavare il percorso con il campo TTL impostato ad 1. Il primo router che lo riceverà, constatando che il campo TTL ha raggiunto lo 0, invierà un errore al mittente (*ICMP 11 Time Exceeded*). L'applicazione memorizzerà l'indirizzo IP del primo router, quindi invierà un nuovo pacchetto con TTL impostato a 2. L'operazione verrà ripetuta finché il pacchetto non sarà arrivato al destinatario, che invierà un *ICMP 8 Echo Reply*.

Alla fine avrà ottenuto la lista degli indirizzi IP dei router su cui hanno transitato i pacchetti.

Calcolando anche il tempo trascorso tra l'invio di un pacchetto e l'arrivo del corrispondente messaggio di errore è possibile anche verificare la velocità di risposta dei router.

IP versione 6

Dopo un lungo lavoro, IETF ha approvato il successore di IP versione 4, cioè la versione 6 (***IPv6***, RFC 1883 - 1887).
I requisiti principali di progetto erano:

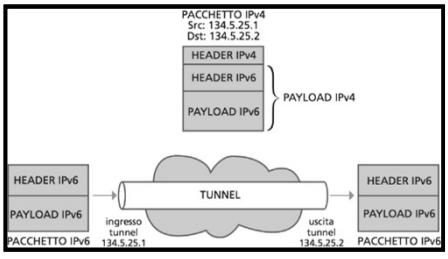
- aumentare il numero di indirizzi, ormai quasi esauriti;
- ottenere una maggiore efficienza nei router (tavole più piccole, routing più veloce);
- supportare meglio il traffico real time;
- offrire maggiore sicurezza ai dati riservati.

Le principali differenze rispetto alla versione 4 sono:

- indirizzi di 16 byte, il che significa disporre di 2^{128} indirizzi IP, e cioé $7*10^{23}$ indirizzi IP per metro quadro su tutto il nostro pianeta;
- header semplificato: 7 campi contro 13;
- funzioni di autentificazione e privacy, basate su crittografia;
- gestione della qualità di servizio attraverso un campo ***flow label***, che consente di istituire delle pseudoconnessioni con caratteristiche negoziate in anticipo.
- Maggiore efficienza eliminando il campo checksum per evitare il suo ricalcolo a ogni router.

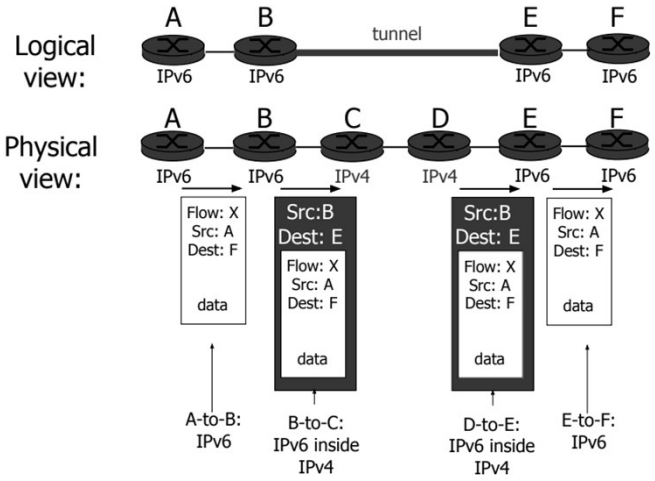
Ipv6 – Passaggio Ipv4 – Ipv6

Pur non essendo retro compatibili (**backward**), i due protocolli possono coesistere grazie alla creazione di un tunnel.



Ipv6 – Passaggio Ipv4 – Ipv6

I pacchetti IPv6 vengono trasportati come payload all'interno di datagrammi IPv4 tra router IPv4



Ipv6 - Tipi di indirizzi IPv6

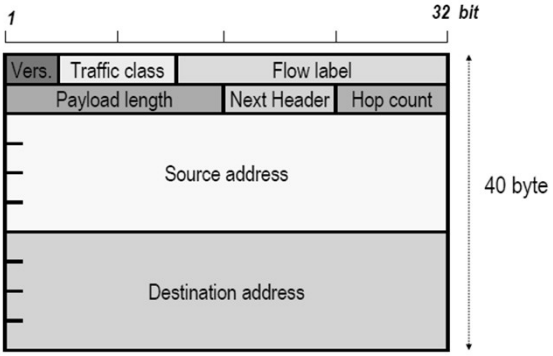
- **Unicast (*una sola destinazione*).** L'indirizzo unicast specifica un'interfaccia singola. Un pacchetto inviato ad una destinazione indirizzo unicast passa da un host ad un host di destinazione.

- **Multicast (*tutti quelli di un gruppo*)** L'indirizzo multicast specifica una serie di interfacce, possibilmente in più ubicazioni. Il prefisso utilizzato per un indirizzo multicast è ff. Se un pacchetto viene inviato ad un indirizzo multicast, una copia di tale pacchetto viene distribuita ad ogni membro del gruppo.

- **Anycast** Come gli indirizzi multicast identificano un gruppo di nodi. Diversamente dai multicast, un pacchetto destinato ad un indirizzo anycast verrà consegnato al nodo (appartenente al gruppo anycast) più vicino al nodo mittente (in base alle metriche presenti sul router). In questo modo risulta possibile identificare il più vicino router, DNS …

Ipv6 - Base Header IPv6

- **La dimensione del Base Header è fissa (40 byte)**



Ipv6 – Campi Header

Version 4 *Versione del Protocollo (6)*

Traffic Class 8 *Campo utilizzabile per distinguere diversi tipi di traffico nelle reti, consento di assegnare una priorità.*

Flow Label 20 *Usata dal mittente per etichettare una sequenza di pacchetti come se fossero nello stesso flusso che dovrebbe essere trattato uniformemente dai router.*

Payload Length 16 *Lunghezza del payload del pacchetto in byte (eccetto gli header)*

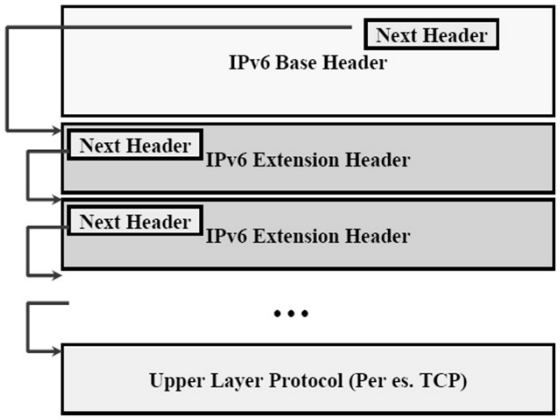
Next Header 8 *Identifica il tipo di header che segue il basic header (può essere di livello superiore come TCP o un extension header)*

Hop Limit 8 *Stessa funzione del TTL(Time To Live) di IPv4*

Source Address 128 *Indirizzo di sorgente*

Destination Addr 128 *Indirizzo di destinazione*

Next Header



Next Header

- Differenze con l'header IPv4
- Checksum: rimossa completamente per ridurre il tempo di processamento nei router ad ogni hop
- Options: sono previste, ma non nell'header. E' possibile prevederle fuori dall'header utilizzando il campo "Next Header"

Ipv6 – Hop by hop option header

Contiene informazioni che devono essere elaborate da ogni router della rete attraversata dal pacchetto.

Può indicare il pacchetto jumbo, cioè superiore a 65536 byte

Ipv6 – Destination option header

- Questo header serve ad indicare informazioni aggiuntive (**opzioni**) sul destinatario.

- Se le *destination options* sono solo per l'utente finale, questo extension header è l'ultimo. Se sono invece dirette ad un router intermedio, tale opzione è usata in unione con l'opzione *routing header* e precede quest'ultima.

- Si possono inserire due destination options per distinguere le informazioni dirette ai router intermedi da quelle dirette all'utente finale.

Ipv6 - Routing Header

Il Routing Header è utilizzato da una sorgente IPv6 per specificare una lista di uno o più nodi intermedi (router) che devono essere attraversati da un pacchetto nella sua strada verso la destinazione

Ipv6 – Fragment Extension Headers

Serve a gestire la frammentazione.
In IPv6 solo il mittente può frammentare un pacchetto, a differenza che in IPv4 in cui può essere un router lungo il cammino.

Questo serve a ridurre l'overhead dovuto a tale operazione nei router

Per poter capire se la frammentazione è necessaria o meno, il trasmettitore deve conoscere la massima MTU del path (la ottiene mediante i messaggi di MTU Path discovery di ICMPv6)

Nel caso in cui una router cambi, anche la MTU può ridursi, rendendo necessaria la frammentazione. In tal caso ICMPv6 è stato esteso per far sì che il router possa segnalare al trasmettitore la necessità di frammentare.

Ipv6 - Extension Headers

Authentication Header

Questo header assicura che il datagramma sia autentico e cioè che non sia stato alterato durante il transito in rete e sia stato emesso effettivamente dal mittente indicato nel datagramma. IPv4 non fornisce invece questa garanzia.

Encrypted Security Payload

Serve per crittografare il payload (altro pacchetto IP o livelli superiori)

Ipv6

- Si scrivono in esadecimale come 8 numeri naturali 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