

# Appunti Reti2

Brendon Mendicino

December 10, 2022

# Contents

<b>1</b>	<b>Introduzione</b>	<b>4</b>
<b>2</b>	<b>Multicast</b>	<b>4</b>
<b>3</b>	<b>IPv6</b>	<b>5</b>
3.1	Tipologie di Indirizzi . . . . .	5
3.1.1	Indirizzi Multicast . . . . .	5
3.1.2	Indirizzi Global Unicast . . . . .	6
3.1.3	Link Local/Site Local . . . . .	7
3.1.4	Unique Local Address . . . . .	7
3.1.5	IPv4 Embedded Address . . . . .	8
3.1.6	Loopback Address . . . . .	8
3.1.7	Unspecified Address . . . . .	8
3.1.8	Indirizzi Anycast . . . . .	8
3.2	Protocolli IPv6 . . . . .	9
3.2.1	IP . . . . .	9
3.3	Interfaccia con i livelli più bassi . . . . .	10
3.3.1	IPv6 Multicast Transmission . . . . .	10
3.3.2	Neighbor Discovery . . . . .	10
3.4	ICMPv6 . . . . .	11
3.4.1	Formato del messaggio . . . . .	12
3.4.2	Error Message . . . . .	12
3.4.3	Neighbor Solicitation . . . . .	13
3.4.4	Neighbor Advertisement . . . . .	13
3.4.5	Group Management . . . . .	13
3.5	Configurazione . . . . .	14
3.5.1	Interface ID . . . . .	14
3.5.2	Address Prefix . . . . .	16
3.5.3	Duplicate Address Detection (DAD) . . . . .	18
3.6	Stateless Config . . . . .	18
3.7	Stateful Config . . . . .	18
3.8	Scope . . . . .	19
<b>4</b>	<b>IPv6 Transitioning</b>	<b>20</b>
4.1	Isolated IPv6 networks . . . . .	21
<b>5</b>	<b>Wireless and Cellular Networks</b>	<b>27</b>
5.1	Wireless LAN . . . . .	27
5.2	CSMA/CA . . . . .	28
5.3	Cellular Networks . . . . .	29
5.4	Basi Procedures . . . . .	32

5.5	Evoluzione delle reti cellulari . . . . .	32
5.6	GSM . . . . .	33
5.7	LTE . . . . .	35
<b>6</b>	<b>5G</b>	<b>38</b>
<b>7</b>	<b>VPN (Virtual Private Network)</b>	<b>38</b>
7.1	Deployment Model . . . . .	39
7.2	Livelli delle VPN . . . . .	40
7.3	Protocolli per VPN . . . . .	41
7.3.1	GRE . . . . .	41
7.4	L2TP . . . . .	42
7.4.1	IPsec . . . . .	43
7.4.2	SSL . . . . .	43
<b>8</b>	<b>Routing</b>	<b>45</b>
8.1	Centralized . . . . .	45
8.2	Isolated . . . . .	45
8.3	Distributed . . . . .	45
8.4	Internet Routing Architecture . . . . .	48
<b>9</b>	<b>Multi-Protocol Label Switching (MPLS)</b>	<b>49</b>
9.1	Shim Header . . . . .	50
9.2	Forwarding Equivalence Class (FEC) . . . . .	50
9.3	VPN via MPLS . . . . .	51
9.4	6BE . . . . .	51
<b>10</b>	<b>Reti ottiche</b>	<b>52</b>
10.1	Piano di Controllo . . . . .	52
<b>11</b>	<b>Qualità del servizio</b>	<b>53</b>

## 1 Introduzione

...

## 2 Multicast

Gli indirizzi che identificano dei gruppi multicast sono quelli di tipo D, iniziano con 1110 (224.0.0.0 – 239.255.255.255).

Si prendono i 23 bit bassi dell'IP e vengono assegnati ai 23 bit bassi dell'indirizzo MAC multicast, questa operazione si chiama di **join** ad un gruppo multicast. Questo approccio potrebbe portare a dei conflitti, la probabilità che una collisione avvenga è molto bassa ma non è zero, i conflitti comportano ricevere il traffico di un altro gruppo multicast anche se non abbiamo fatto un join con quello.

Esempio: viene usato l'indirizzo 224.0.0.0 come un gruppo multicast, gli host che vogliono connettersi a questo gruppo dovranno fare il join, e quindi impostare la loro scheda di rete (solitamente una scheda di rete virtuale, in cui viene aggiunto il MAC multicast) con il MAC multicast, in questo caso gli ultimi 23 bit saranno a 0.

## 3 IPv6

Gli indirizzi ipv6 sono rappresentati su 128, quindi si hanno  $2^{128}$  combinazioni. Per rappresentarli si divide l'indirizzo in 8 gruppi di 2 byte, separati da un ":". Ci sono delle strategie per rendere più leggibile l'indirizzo:

1. gli zeri in fronte possono essere omessi;
2. gli zeri ("::") possono essere sostituiti con un ":" solo una volta;

Per rappresentare l'indirizzo di rete si usano 64 bit. Il concetto di aggregazione gerarchico viene mantenuto del prefix length e dalla netmask, dunque il prefix viene usato per il subnetting.

I principi di assegnamento sono:

- **subnetwork**: set of host with the same prefix;
- **link**: physical network;
- **on-link**: comunicazioni tra host con lo stesso prefisso;
- **off-link**: comunicazioni tra host con prefisso diverso;

### 3.1 Tipologie di Indirizzi

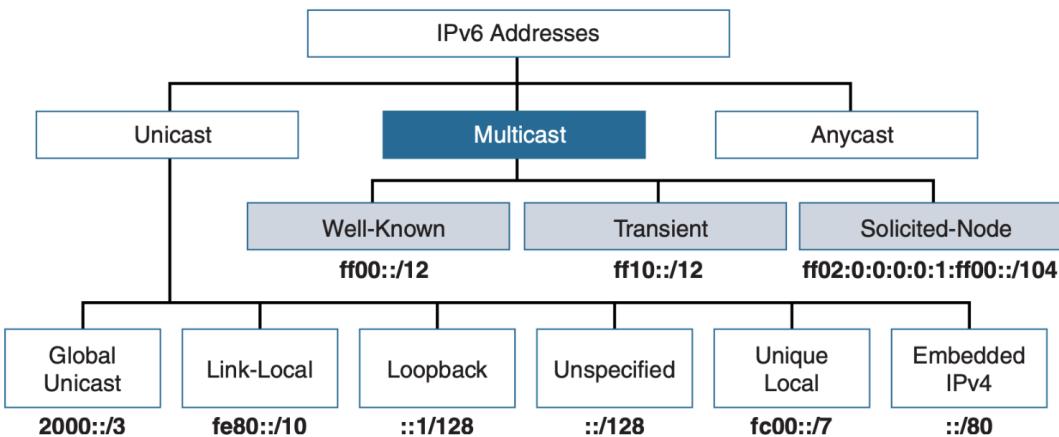


Figure 1: Ipv6 Spazio Di Indirizzamento

#### 3.1.1 Indirizzi Multicast

Il multicast ha una rappresentazione simile ad IPv4, infatti hanno un range di FF00::/8 (1111 1111 ...), che si dividono in tre sottocategorie:

- **well-known multicast** FF00::/12 (1111 1111 0000 ...): questo range di indirizzi è assegnato dalla IANA, utilizzato dagli ISP per scopi di comunicazione;
- **transient** FF10::/12 (1111 1111 0001 ...): assegnati dinamicamente;
- **solicited-node multicast** FF02:0:0:0:0:1:FF00::/104: simile al broadcast address in ARP;

Un indirizzo multicast è formato da:

- I primi 8 bit mi identificano un indirizzo multicast, tutti settati ad 1;
- 4 bit assegnati a dei flag: l'unico utilizzabile è il campo T che specifica se l'indirizzo è permanente (0), ovvero assegnato dalla IANA, oppure non-permanente (1), gli altri campi non hanno un'assegnazione;
- 4 bit per stabilire lo **scope**, definisce il range di indirizzi multicast;
- gli ultimi 112 rappresentano il **group id**;

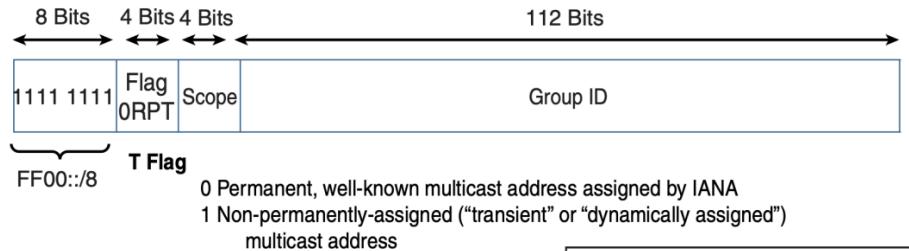


Figure 2: Indirizzo Multicast

### 3.1.2 Indirizzi Global Unicast

Sono l'equivalente degli indirizzi pubblici IPv4. Quando un nuovo host si collega alla rete, sa automaticamente il suo indirizzo, infatti gli indirizzi unicast sono plug and play. Gli indirizzi sono composti da tutto la spazio 2000::/3, l'indirizzo si divide in:

- 3 bit: 001;
- n bit: global routing prefix;
- m bit: subnet ID;
- $128 - m - n - 3$  bit: interface ID;

Il prefisso moderno è stato assegnato formalmente da entità multi-livello:

- 3 bit: 001;

- 13 bit: TLA ID, Top Level Authority (Large ISP);
- 32 bit: NLA ID, Next Level Authority (Organizzazione);
- 16 bit: SLA ID, Subnet Level Authority;
- 64 bit: Interface ID;

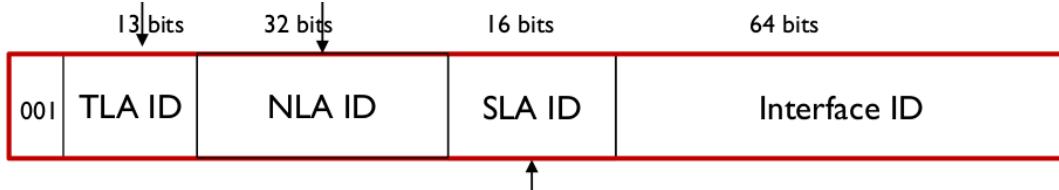


Figure 3: Indirizzo Global Unicast

### 3.1.3 Link Local/Site Local

Gli indirizzi link/site local sono assegnati automaticamente, fanno parte del gruppo fe80::/9 (1111 1110 1...), e si distinguono in:

- **link-local** fe80::/10 (1111 1110 10...): vengono generati automaticamente, ogni host in una rete ipv6 deve avere un link-local address utilizzato per comunicazioni di **neighbor discovery**;
- **site-local**: fec0::/10 (1111 1110 11...), indirizzi deprecati, utilizzati per assegnare degli indirizzi privati univoci;

### 3.1.4 Unique Local Address

Gli ULA sono univoci rimanendo comunque privati, quindi non devono essere esposti alla rete. Gli Unique Local sono stati pensati per accedere a macchine che non devono essere esposte alla rete pubblica. Gli Unique Local usano un range è di fc00::/7. La particolarità è che l'ottavo bit è il **local flag** (L), se questo bit è settato ad 1 l'indirizzo è assegnato localmente, se invece è a 0 potrebbe essere assegnato in futuro. I successivi 40 bit sono assegnati casualmente, per cercare di mantenere l'univocità.



Figure 4: Indirizzo Unique Local

### 3.1.5 IPv4 Embedded Address

Sono usati per rappresentare gli indirizzi ipv4 sugli indirizzi ipv6 e vengono messi nello spazio ::ff:0:0/80. Sono composti da:

- i primi 80 bit a 0;
- 16 bit a 1;
- gli ultimi 32 bit rappresentano l'indirizzo ipv4;

### 3.1.6 Loopback Address

L'indirizzo ::1 ha lo stesso scopo dell'indirizzo di loopback 127.0.0.1 di ipv4.

### 3.1.7 Unspecified Address

È un indirizzo unicast non specificato ::0, anche in questo caso il suo comportamento è lo stesso di ipv4.

### 3.1.8 Indirizzi Anycast

Ho degli indirizzi assegnati a dei nodi nelle reti e quando mando un pacchetto voglio che esso arrivi ad uno di essi (inizialmente pensato per i DNS server). Gli indirizzi anycast non sono utilizzati.

## 3.2 Protocolli IPv6

In ipv6 alcuni protocolli sono stati integrati o rimossi rispetto ad ipv4, infatti ARP ed IGMP sono stati integrati in ICMP, la maggior parte degli altri protocollli è stata fatta una modifica per supportare gli indirizzi a 128 ma le modifiche rimangono minime.

### 3.2.1 IP

L'header in ipv6 contiene meno dati che in ipv4, il motivo è che le informazioni sono presenti nel padding, tutto ciò è possibile grazie al campo next header. Si crea così una catena di header. Se non ho bisogno di estensioni allora il campo next header punterà all'header tcp.

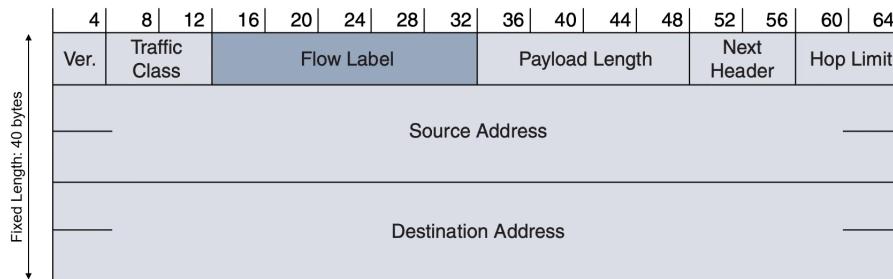


Figure 5: Ipv6 Header

I campi rimasti sono:

- ver: la versione del protocollo;
- traffic class: definisce delle classi di pacchetti per determinare delle priorità tra i traffici degli utenti;
- flow label: etichetta associata al flusso, sarà possibile fare routing grazie a questo campo;
- payload lenght: lunghezza del payload;
- hop limit: time to live del ipv4;
- source address: indirizzo sorgente;
- destinatoin address: indirizzo destinazione;

**Headers extensions** Il protocollo utilizza dei codici per specificare quale sarà il campo dell'prossima estensione.

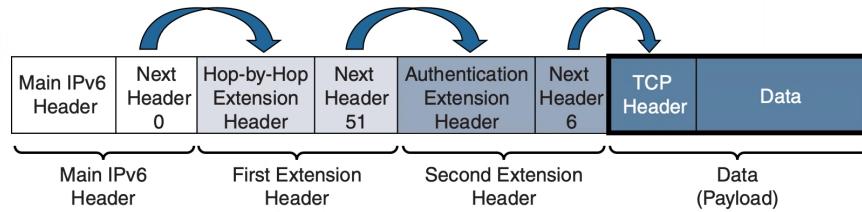


Figure 6: Header Chaining

I diversi extension header hanno tutti gli stessi attributi:

- next header code;
- lunghezza;
- extension data: dati relativi all'extension header;

Un'estensione molto usata è la **Routing Extension Header**, dove viene specificata una lista di indirizzi su cui il pacchetto deve passare.

### 3.3 Interfaccia con i livelli più bassi

Nel livello 3 si è deciso di differenziare il protocollo ipv4 e ipv6 (approccio dual stack), infatti nel livello 2 esiste un campo che specifica il protocollo a livello superiore. Il type per ipv6 è 86DD.

#### 3.3.1 IPv6 Multicast Transmission

Quando si deve mandare un messaggio in multicast con ipv6 si utilizza una tecnica di mapping per creare un indirizzo MAC multicast appositamente per questo scambio di messaggi, il motivo è che si cerca di evitare l'utilizzo di un MAC broadcast address. I 32 bit bassi dell'indirizzo ipv6 multicast vengono mappati ai 32 bit bassi dell'indirizzo MAC, questo mapping viene specificato quando i primi 2 byte del MAC sono settati a 33:33, dunque un indirizzo MAC mappato sarà del tipo 33:33:xx:xx:xx:xx. Quando si manda un pacchetto all'indirizzo di broadcast ff0c::89:aabb:ccdd l'indirizzo MAC corrispondente sarà 33:33:aa:bb:cc:dd.

#### 3.3.2 Neighbor Discovery

Il protocollo ARP sarà sostituito dalla nuova versione del protocollo ICMPv6. Quando si vuole mandare un pacchetto ad un host nella stessa sottorete e non si è a conoscenza del MAC, si utilizza il protocollo di neighbor discovery.

Il meccanismo di neighbor discovery avviene nel seguente modo: partendo da un indirizzo unicast si prendono i 24 bit bassi dell'indirizzo e si crea un indirizzo multicast solicited-node con i 24 bit bassi corrispondenti a quelli dell'unicast, questo permette quando si fa il mapping, di avere degli indirizzi MAC multicast che iniziano con 33:33:ff, grazie a come sono composti gli indirizzi solicited-node. In questo modo viene limitato il numero di pacchetti mandati nelle reti, la probabilità che il pacchetto venga spedito all'host di destinazione è molto probabile, invece in ARP si mandava un pacchetto in broadcast a tutti gli host della sottorete.

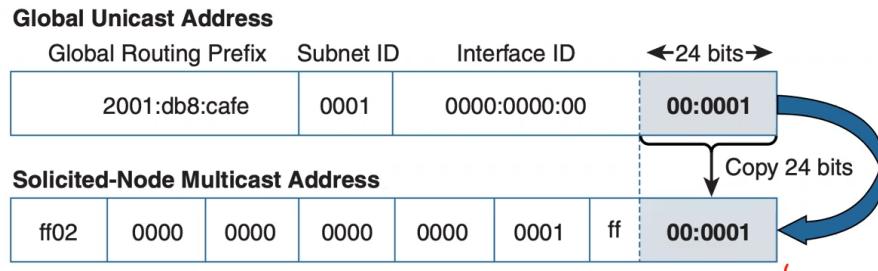


Figure 7: Solicited Node Multicast Address

ARP:

- manda un trama in broadcast;

Multicast:

- manda a tutti;
- manda solo a chi fa parte di un gruppo: il MAC di multicast permette di mandare messaggi solo a chi appartiene a chi potenzialmente fa parte di quel gruppo;

### 3.4 ICMPv6

Il protocollo mantiene le opzioni di quello usato in ipv4 aggiungendo delle funzionalità per sostituire ARP e IGMP, ICMP è usato per:

- diagnostica;
- neighbor discovery;
- multicast group;
- issue notification;

### 3.4.1 Formato del messaggio

ICMP è incapsulato nel pacchetto ip.

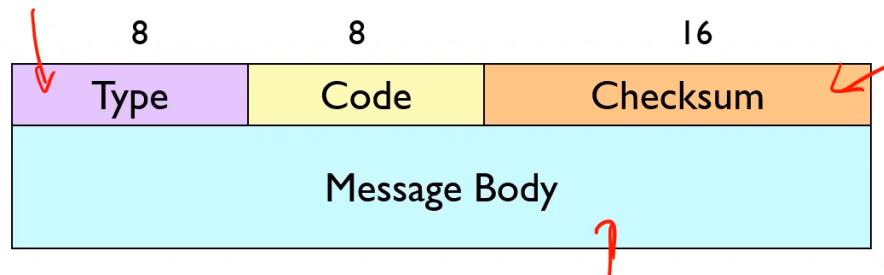


Figure 8: Icmp Format

Messaggi di errore:

- 1: destination unreachable;
- 2: packet too big;
- 3: time exceeded;
- 4: parameter problem;

Echo:

- 128: echo request;
- 129: echo reply;

### 3.4.2 Error Message

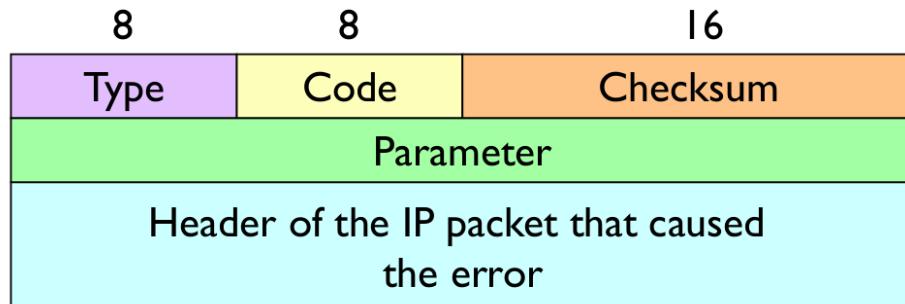


Figure 9: Icmpv6 Error Message

**Echo** I messaggi di echo sono: echo request ed echo reply.

### 3.4.3 Neighbor Solicitation

Mandato da un host nella sotto rete.

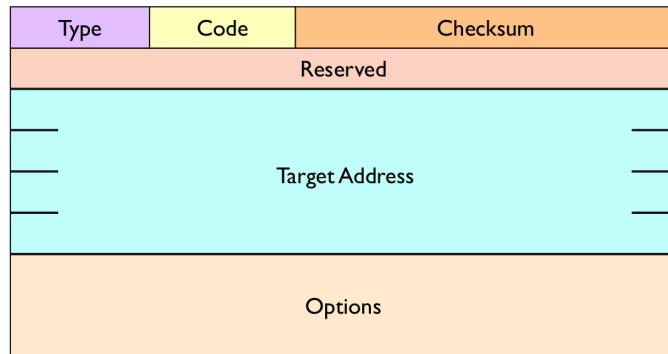


Figure 10: Neighbor Solicitation

### 3.4.4 Neighbor Advertisement

Vengono aggiunti 3 flag nel campo reserved:

- router: indica se il pacchetto arrivo da un router;
- solicited: specifica se il nodo è stato sollicitato o meno;
- override: specifica se l'host cache deve essere sovrascritta;

L'indirizzo MAC viene messo nel campo options, mentre l'ip di sorgente viene messo nel campo target address.

### 3.4.5 Group Management

Per gestire le comunicazioni multicast in un link local (sottorete) non si utilizza più il protocollo IGMP ma si usa ICMPv6, per gestire il join ai diversi gruppi si utilizzano tre tipi di messaggi:

- Multicast Listener Query;
- Multicast Listener Report;
- Multicast Listener Done;

Un router inizia mandando un query a tutti gli host collegati utilizzando l'indirizzo ff02::1 (all host multicast address, solo host e non router), in questo modo vengono esposti i nodi che vogliono partecipare ad uno specifico gruppo. In teoria ha senso che tutti quanti gli host non appena ricevono la Query rispondano con la relativa Report, perché in questo modo il router può avere la lista completa di tutti gli interessati.

Ma nel tentativo di ottimizzare le trasmissioni, si è pensato che il router non dovesse possedere l'intera lista delle query con tutti i gruppi multicast, questo perché il router dovrà semplicemente forwardare nella rete il pacchetto multicast quando è presente almeno 1 host nel gruppo multicast. È stato dunque inserito un timer randomico sugli host, che viene fatto partire quando viene ricevuta una query. Il primo host che manda un report viene registrato sia dal router che da tutti gli interessati al gruppo broadcast bloccando i loro timer.

Quando uno degli host si disconnette invia un Done. I router tengono un timeout per ogni entry nella loro tabella in caso un host si disconnetta senza inviare un done.

## 3.5 Configurazione

Le informazioni necessarie sono:

- address prefix;
- interface id;
- default gateway;
- dns server;
- host name;
- domain name;
- MTU maximux transmission unit;

Per creare queste informazioni:

- stateful config: informazioni date da un DHCP;
- stateless config: generate automaticamente;
- ibrida (stateless DHCP);

### 3.5.1 Interface ID

Si può configurare manualmente, ottenere dal DHCP oppure generati automaticamente. Nel modo di configurare i 64 bit bassi non si hanno garanzie che siano univoci, esiste un protocollo che permette il controllo e l'univocità.

### EUI-48 to EUI-64 Mapping

- EUI = Extended Unique Id
- OUI = Organization Unique Id

Per creare l'interface ID in modo automatico si può sfruttare la tecnica del mapping, si prende il MAC del interfaccia e viene mappato nel seguente modo:

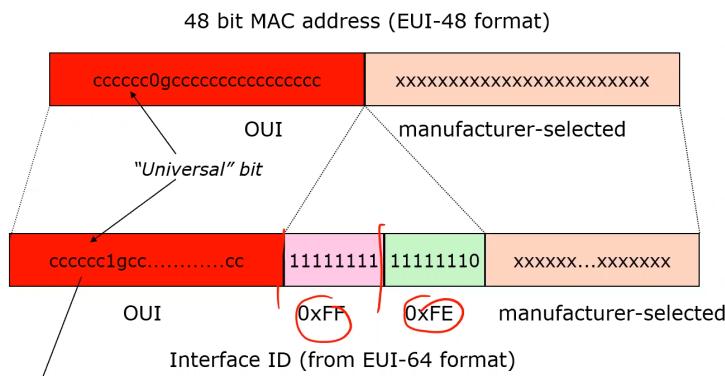


Figure 11: Mac To Id Mapping

Il settimo bit viene settato a 0 il MAC è universale (la scheda di rete è configurata da un'organizzazione), mentre viene settato a 1 se il MAC è stato assegnato manualmente (quindi l'indirizzo sarà necessariamente univoco nella sottorete), questa è una convenzione ma non la regola.

**Privacy Extension Algorithm** Per evitare che l'interface Id possa essere calcolato da qualcun'altro ch conosca il MAC del mio dispositivo si usa un approccio per garantire maggior privacy ed aumentare la sicurezza.

Per generare l'interface Id si prendono 64 bit random e 64 bit generati del MAC mapping e poi viene fatto un hash. In fine il settimo bit viene settato a 0 perché non si ha comunque la certezza che l'interface Id sia univoco.

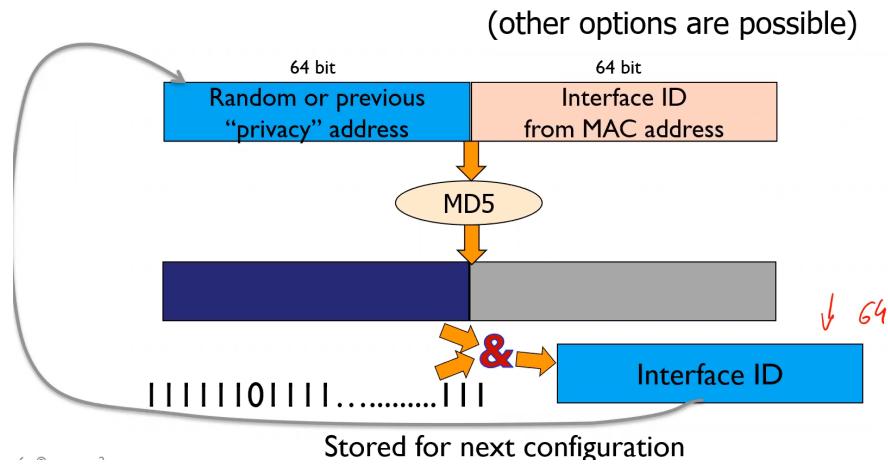


Figure 12: Privacy Extension Alogorithm

### 3.5.2 Address Prefix

Anche in questo caso è possibile configurare la parte alta dell'indirizzo ip in modo manuale oppure in modo automatico. Per ottenere queste informazioni in modo automatico esistono due messaggi:

- **Router Solicitation:** mandato a tutti i router con indirizzo multicast ff01::2.



Figure 13: Router Solicitation

- **Router Advertisement:** può essere una risposta ad una solicitaion, M (Managed Address Configuration): se settato ad 1 indica che l'indirizzo è disponibile via DHCP; O (Other Configuration): parametri come il DNS server; Reachable Time: tempo in cui il router è disponibile; Retrans Timer: tempo in cui l'indirizzo è disponibile; solitamente l'indirizzo viene messo nel campo options;

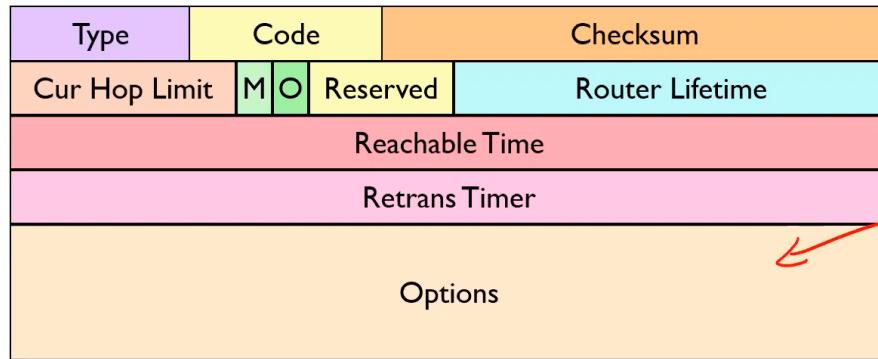


Figure 14: Router Advertisement

Quando un host si collega ad una rete potrà fare una solicitation e ricevere una advertisement oppure i router periodicamente mandano una advertisement.

**Options** Le opzioni hanno un loro formato particolare:



Figure 15: Options Format

Le informazioni sul prefisso sono:

- L: ad 1 se il prefisso può essere on-link;
- A: ad 1 se il prefisso può essere usato con una configurazione autonoma;

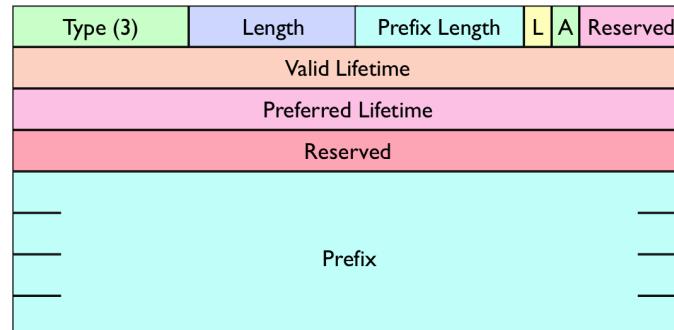


Figure 16: Prefix Information Options

Un'altra opzione è il **Link Layer Address Option**, si mette il MAC del default gateway.

Un messaggio molto importante è l'**ICMP Redirect**, in una rete ci sono più router, l'host manda i pacchetti al suo default gateway, se per raggiungere un altro host un altro è router è più vicinino allora i pacchetti vengono rediretti in quel router.

### 3.5.3 Duplicate Address Detection (DAD)

L'host manda un messaggio di DAD ogni volta che viene effettuata una configurazione. Per verificare che non esistano indirizzi duplicati viene mandato una neighbor solicitation con l'indirizzo appena configurato, se un host risponde a questa richiesta l'indirizzo viene cambiato.

## 3.6 Stateless Config

- Il link local address viene generato automaticamente;
- successivamente viene fatta una DAD;
- l'host si iscrive al Solicited Node Multicast Address (configurando il MAC multicast e inviando una ICMP multicast listener report);
- abilita le comunicazioni on-link;

## 3.7 Stateful Config

- viene inviata un router solicitation;
- si ascolta la router advertisement;
- si crea un indirizzo col prefisso annunciato;
- si prova la sua unicità con la DAD;
- ci si iscrive al solicited node multicast address (configurando il MAC multicast e inviando una ICMP multicast listener report);

Un'altra grossa vantaggio è il **Renumbering**, tramite l'advertisement si riassegnano gli indirizzi in modo automatico.

### 3.8 Scope

Quando un dispositivo ha più interfacce, gli indirizzi generati automaticamente saranno gli stessi, per distinguere le due interfacce l'host tiene conto a quale interfaccia mandare un messaggio. Per distinguere le due interfacce si mette %x, dove x è l'id dell'interfaccia.

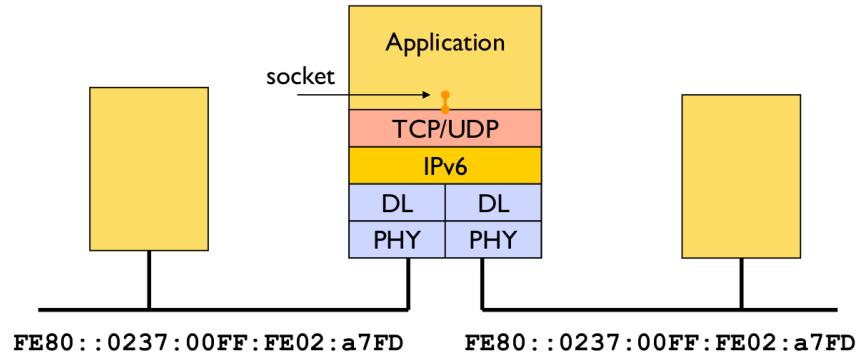


Figure 17: Ipv6 Scope

## 4 IPv6 Transitioning

IPv6 è ancora in via di utilizzo parziale, perciò si devono capire i problemi che sorgono nell'utilizzo di questo nuovo protocollo, si possono identificare 4 fasi di transizione, in cui IPv6 diventa incrementalmente più utilizzato fino a diventare lo standard:

- **Network ipv6 isolati:** host di interfaccia dual stack, con ipv6 in ipv4 tunneling;

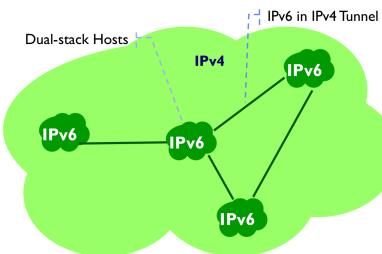


Figure 18: Isolated Ipv6

- **ipv6 island grow:** le isole diventano più grandi con soli host ipv6 all'interno, gli host di interfaccia sono dual stack translating devices;

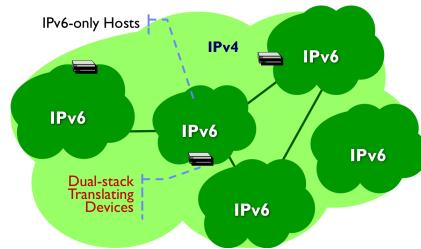


Figure 19: Ipv6 Island Grow

- **native ipv6 connectivity;**

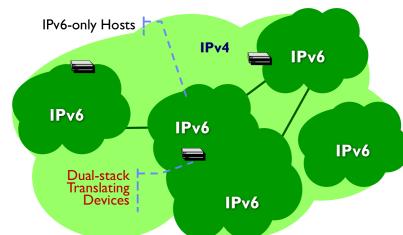


Figure 20: Native Ipv6 Connectivity

- **ipv6 takes over:** ipv4 isolati, ipv4 in ipv6 tunneling;

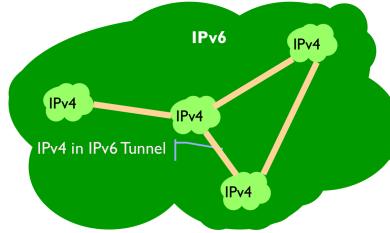


Figure 21: Ipv6 Takes Over

## 4.1 Isolated IPv6 networks

Si fa del **Tunneling**, partendo da un pacchetto ipv6 che si interfaccia ad una rete ipv4, il router crea un header ipv4 e incapsula il pacchetto ipv6, il ricevitore con un indirizzo ipv4 dovrà poi riprendere il pacchetto ipv6 originale, per far in modo che questo avvenga i router devono essere necessariamente dual stack. Esistono dei vari protocolli che permettono di avere un mapping delgi indirizzi ipv6 algi indirizzi ipv4 automatico o manuale.

Una di queste soluzioni è avere entrambi gli host in dual stack (**host-centered solutions**):

- **IPv4-compatible addresses:** assegno solo degli indirizzi ipv6 compatibili con ipv4, quando devo inviare un messaggio a destinazione gli ultimi 32 bit dell'indirizzo ipv6 (::/96) sono i bit dell'indirizzo ipv4, dunque non ho conflitti;

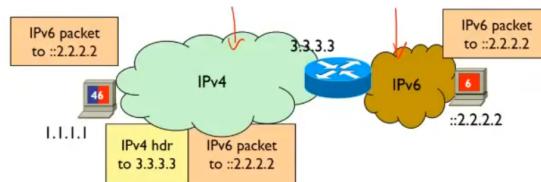


Figure 22: Ipv4 Compatible Addresses

- **6over4:** sfrutta il multicast dell'ipv4 e dell'ipv6, però anche se in ipv4 è presente il multicast i provider non lo abilitano all'interno delle loro reti;
- **ISATAP:** l'ipv6 avrà su un prefisso comune (fe80::5efe) e gli ultimi 32 bit saranno dell'indirizzo ipv4;
- **neighbor discovery:** viene distribuita un lista dal DNS dove esistono i mapping tra indirizzi ipv4 e indirizzi ipv6, una delle limitazione è che ad ogni indirizzo ipv6 deve essere associato un hostname;

Posso avere anche delle soluzioni **network-centered**, dove esistono host nativi in ipv6 e degli host dual stack:

- **6to4**: una delle principali limitazioni del mapping era il basso numero di bit utilizzati, allora una possibile soluzione è quella di andare a mettere l'indirizzo ipv4 nella parte alta dell'ipv6,

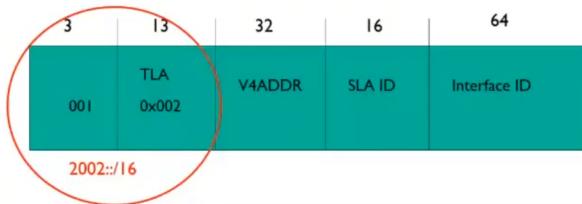


Figure 23: 6To4

- **tunnel broker**: esiste un entità nel mezzo che fa da broker per gli indirizzi da andare ad utilizzare per il tunnel;

Soluzioni **scalabili, carrier-grade solutions**, quando le isole ipv6 continuano a crescere si avranno anche scambi di pacchetti ipv4 attraverso reti native ipv6. Questo tipo di setup dovrà supportare ancora ipv4 clients e connessioni di host ipv6 a server in ipv4.

Il mapping di un indirizzo IP era proprio del NAT (map da ipv4 a ipv4), un dispositivo in grado di fare sia routing che natting viene detto Customer Premises Equipment (CPE). Alcune varianti del NAT è quello nelle reti cellulari, dove le antenne a cui si connettono il nat assegna un indirizzo pubblico all'host, per questo viene detta Large Scale NAT (LSN). Esiste anche un soluzione con entrambi gli approcci dove un LSN mappa delle reti private, in cui è presente il NAT per la traduzione. Alcune soluzioni di mapping tra indirizzi dipendono da dove viene inserito il NAT.

- **AFTR (address family transmission router)**: grantisce agli host ipv4 di parlare con host ipv4 attraverso infrastrutture nel mezzo ipv6, l'AFTR tipicamente ha due funzionalità: accelleratori hardware del nat e del tunneling;
- **Dual Stack Lite (DS lite)**: con questa soluzione si gestiscono tutti i casi in cui l'ISP abbia un backbone in ipv6, ds lite prevede che l'indirizzo ipv4 arrivi al CPE e che poi venga fatto un tunneling verso il Large Scale NAT (AFTR) e poi viene fatto il natting, lo svantaggio è che il piano di indirizzamento deve essere pubblico altrimenti si potrebbero creare delle collisioni, le principali limitazioni sono: il NAT non è sotto il controllo del consumatore perché un unico NAT gestisce tutti i clienti, il port forwarding non può essere abilitato nel CPR;

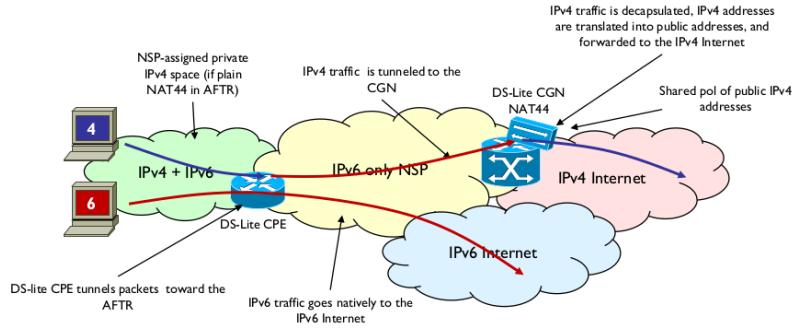


Figure 24: Ds Lite

- **A+P (address plus port):** in questa soluzione rispetto a ds lite il NAT viene spostato dall'AFTR al CPR diventando sotto il controllo del consumatore, aumentando la scalabilità, quando si manda un pacchetto in ipv4 passo da un indirizzo privato ad uno natato e poi si fa un tunneling verso il AFTR, questo comporta l'assegnazione di un indirizzo ipv4 pubblico al CPR, il motivo è che quando si farà il natting è necessario avere un ip pubblico che si interfacci con la rete e che poi viene tunnelato. Se però l'ISP ha scarsità di indirizzi ipv4 si più utenti avranno gli stessi indirizzi ip con dei range di porte diversi assegnati;

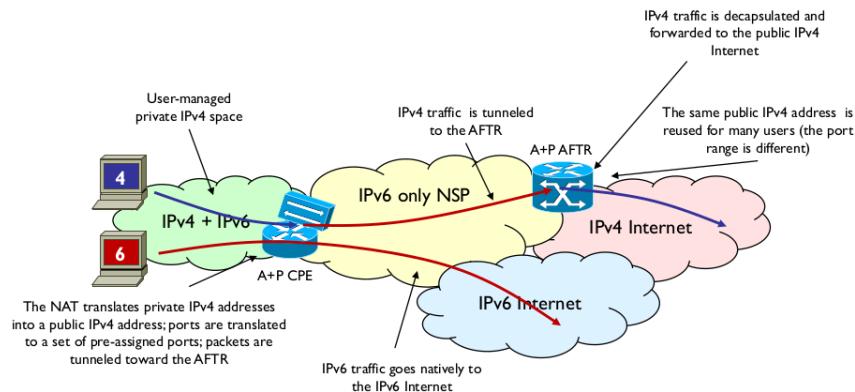


Figure 25: A+P

- **MAP (mapping address and port):** è un mix tra ds lite e a+p, in cui si tenta di avere una configurazione stateless, dove non si allocano dei range di porte al CPE, ma un set di porte, il vantaggio è che un border relay (AFTR) è in grado di ricostruire le informazioni a partire dall'indirizzo. Le caratteristiche sono:

- un client ipv4 viene mappato su un indirizzo univoco ipv6 caratterizzato dal prefisso del router CPE;
- l’indirizzo pubblico dell’host di destinazione viene mappato nel border relay;
- map-e: map con incapsulamento, il pacchetto ipv6 viene messo nel payload di un pacchetto ipv4;
- map-t: map con traduzione, in un pacchetto ipv6 l’header viene sostituito con un uno ipv4;

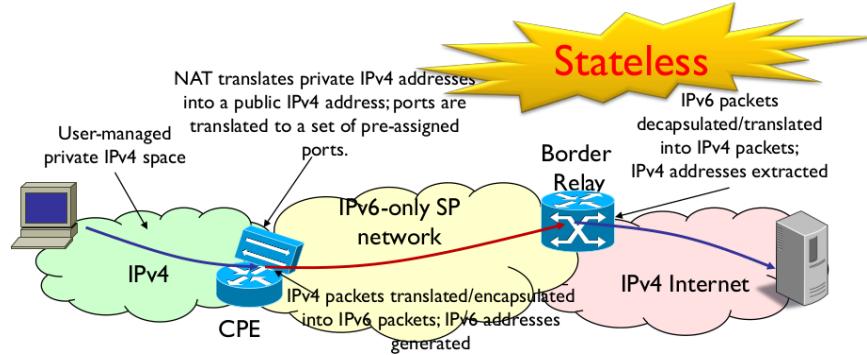


Figure 26: Map

- **Port set:** per generare i set di porte si utilizzano i 16 bit delle porte, ad ogni CPE viene assegnato un PSID (Port Set ID) e un indirizzo ipv4 pubblico, ogni set di porte sono rappresentate da: a bit maggiori di 0 (altrimenti si andrebbero a toccare le well-known ports), k bit di PSID, m bit (fino a 16 bit totali);
- **CPE ipv6 address:** si basa sul fatto che l’indirizzo ipv6 del CPE contiene delle informazioni sul CPE stesso e legate in qualche modo all’host ipv4 che ci vuole raggiungere, in modo che il border relay possa ricostruirsi l’indirizzo ipv6 a partire dall’indirizzo ipv4 da cui arrivano le risposte, perciò l’indirizzo si compone di:
  - \* ruole ipv6 prefix: prefisso che identifica l’istanza di map;
  - \* EA bits: parte dell’indirizzo ipv4 pubblico e una porzione del port set;
  - \* subnet id: compatibilità con gli indirizzi ipv6 (sempre messi a 0);
  - \* interface id: informazioni dell’indirizzo ipv4;

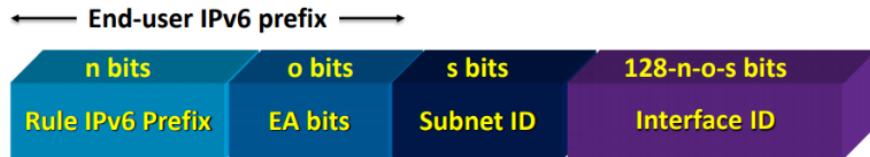


Figure 27: Cpe Ipv6 Address

- map-e:

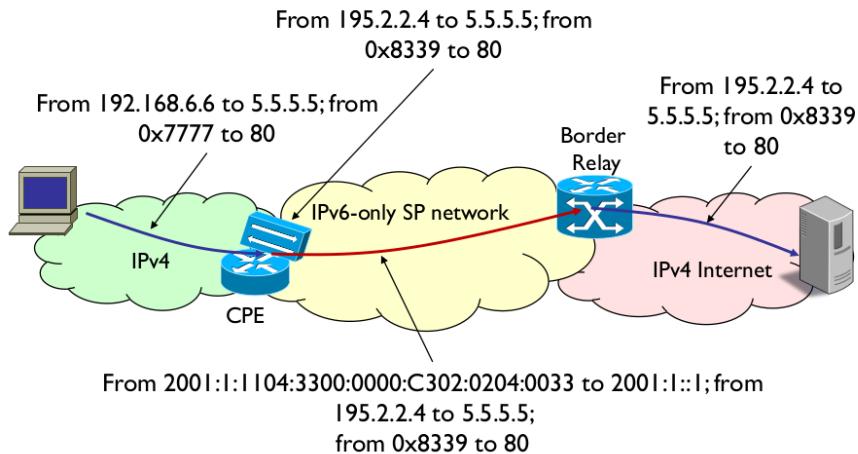


Figure 28: Map E

- map-t:

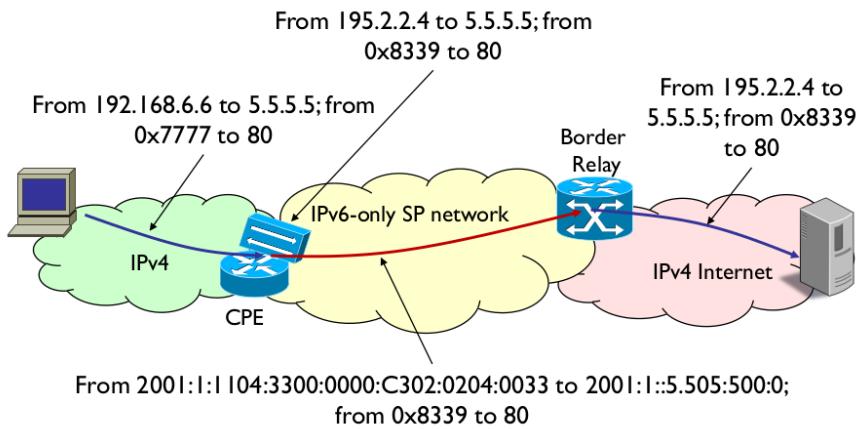


Figure 29: Map T

- **NAT64 + DNS64:** lo scenario è quando un client ipv6 vuole parlare con un indirizzo ipv4, questa operazione andrà fatta con un NAT64, l'indirizzo però

viene mappato dal DNS64. Per effettuare una risoluzione del nome, il DNS64 chiede a cui arriva una richiesta AAAA (ipv6) la inoltra al DNS del server ipv4 che risponde con un errore, allora il DNS64 manda una richiesta A (ipv4), il DNS6 mappa l'indirizzo ipv4 ricevuto ad un indirizzo ipv6 che inoltra al client, l'indirizzo ha un prefisso conosciuto (usato per le risoluzioni tradotte da ipv4) e la parte finale con l'indirizzo ipv4 del server, la richiesta che poi viene fatta passando per il NAT64 effettuerà una traduzione, creando l'header ipv4 a partire dall'indirizzo ipv4 dagli ultimi bit nell'indirizzo ipv6;

## 5 Wireless and Cellular Networks

### 5.1 Wireless LAN

Le caratteristiche del link wireless sono:

- un link Wireless ha un degrado maggiore del segnale, rispetta ad una fibra ottica;
- è soggetto a interferenze;
- problema del **fading**, causato dai rimbalzi del segnale su ostacoli;

Esistono vari standard dello IEEE 802.11 wireless LAN, i più moderni arrivano fino a 5GHz, il problema con le alte frequenze è

Tutte le implementazioni utilizzano il protocollo di accesso **CSMA/CA**.

Un **access point** o una **base station** serve una **Basic Service Set (BSS)**, dentro la BSS si trovano sia gli access point che gli host.

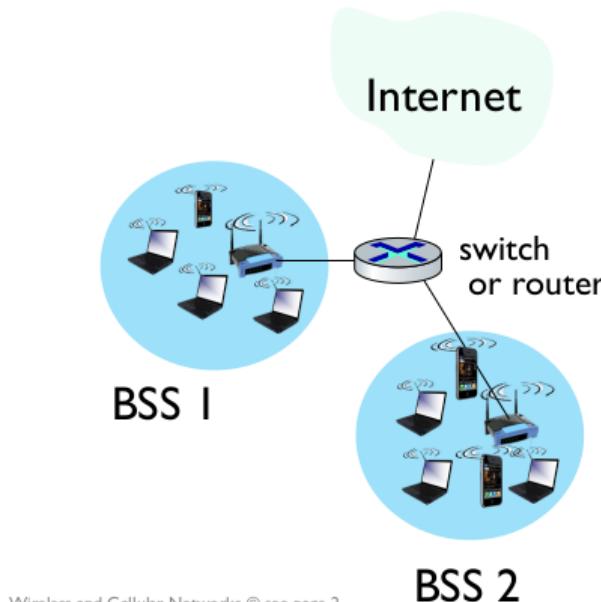


Figure 30: BSS

Un dispositivo fa il **sensing** per cercare un canale operativo, che provengono di vari access point, ascoltando per il **beacon frame** che serve ad agganciarsi ad un access point, ci sono diversi beacon fram ed il dispositivo si collega all'access point con il segnale più forte, il beacon frame contiene:

- il nome dell'AP (SSID);

- l'indirizzo MAC;

Per poter accedere ad una rete wifi, oggigiorno, hanno tutte bisogno di autenticazione, e tipicamente sarà presente un DHCP per ottenere una configurazione IP.

## 5.2 CSMA/CA

Il protocollo fa il sensing del canale (CSMA = carrier sense multiple access), ed la collision avoidance (CA), il motivo è che in una rete wireless il mezzo di trasmissione è l'aria che è un mezzo condiviso. Il sender:

- si fa il sense del canale, si aspetta un tempo di DIFS e si trasmettono i dati;
- se si fa il sense ed il canale è occupato si aspetta, per applicare la CA parte un random exponential backoff timer quando sento il canale occupato (in ethernet il timer partiva solo quando avveniva una collisione!);

Per evitare le collisione gli host mandano un piccolo pacchetto , per sprecare la minor banda possibile, detti RTS (ready to send) usando CSMA, l'AP rispondono in broadcast agli host con un CTS (clear to send) per un degli host che ha mandato l'RTS, dopichè l'host a cui l'AP ha mandato il CTS che inizia a trasmettere una trama e l'AP manda sempre in broadcast un ACK all'host che ha trasmesso la trama.

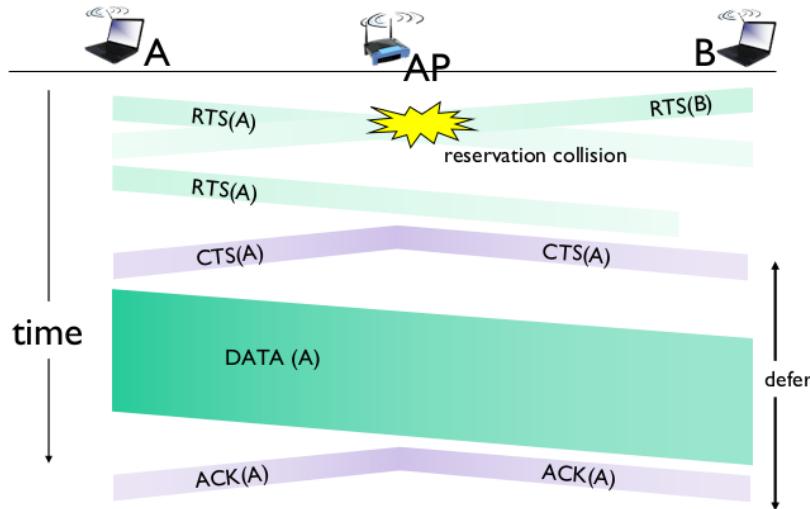


Figure 31: Collision Avoidance

Una trama 802.11 è fatta da:

- frame control: è composto da:

- protocol version;

- type: RTS, CTS, ACK, data;
- power mgt;
- ...;
- duration: la durata che ci mette la trama per essere trasfertita;
- address 1: indirizzo dell’interfaccia MAC dell’AP, il motivo per cui è presente questo indirizzo è che nelle reti wireless bisogna prima passare dell’AP (violando in qualche modo il principio su cui si basa il livello link nelle reti ethernet, dove se uno switch è presente il pacchetto viene direttamente inoltrato al mac di destinazione, senza passare dal router);
- address 2: indirizzo sorgente;
- address 3: indirizzo dell’interfaccia del router a cui l’AP è collegato;

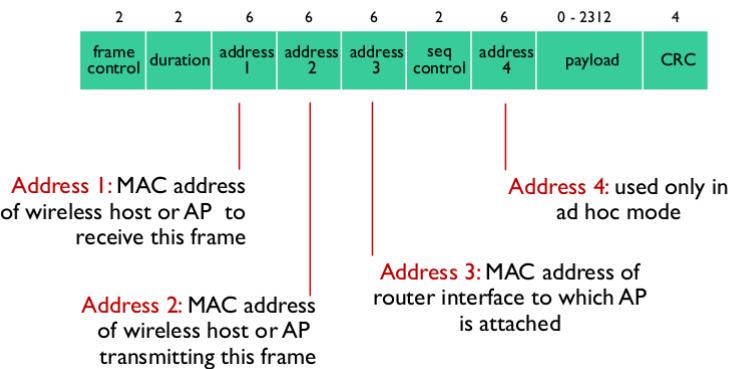


Figure 32: 802.11 Frame

In una rete wireless va gestita anche la mobilità, questo è improbabile in una rete wifi, solitamente il movimento è fatto in una subnet, ovvero in una rete in cui sono presenti più AP collegati allo stesso router di una sottorete.

Un’altra capacità dei dispositivi è il poter entrare in **sleep mode**, un nodo manda all’AP questa richiesta, se l’AP riceve delle trame da mandare al nodo li bufferizza finché il nodo non si sveglia, un nodo si riattiva quando un beacon frame viene mandato, nel beacon frame vengono anche segnalati gli host per cui ci sono dei messaggi in coda, allora il nodo capisce che deve svegliarsi e colleziona le trame, oppure continua a dormire.

### 5.3 Cellular Networks

Una rete cellulare è una rete che cerca di coprire un’area geografica molto vasta attraverso le **celle**, dove il terminale utente si muove anche su lunghe distanze, gestendo il cambiamento da una cella all’altra, detto **handover**.

La forma e le dimensioni di una cella sono determinate da:

- potenza emessa;
- altezza;
- il guadagno dell'antenna: indica quanto un'antenna è buona nella trasmissione;
- morfologia del territorio;
- condizioni di propagazione: se nevica il segnale sarà più attenuato;

Le celle utilizzate in pratica sono:

- Una **macrocella** viene realizzata con un'antenna posizionata molto in alto;



Figure 33: Macrocell

- **Microcella** realizzata con antenne non molto posta in alto,

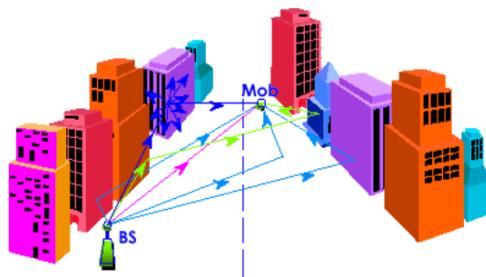


Figure 34: Microcell

Nelle reti cellulari non si usa CSMA/CD, le tecniche di condivisione sono: fdma, tdma, cdma, sdma. Per quasi tutte le reti cellulari si usa FDMA con riutilizzo dell

frequenze, sfruttando la distanza fisica che separa le celle. Vengono creati dei **cluster** di celle in cui vengono usate tutte le frequenze disponibili, al di fuori di questo cluster si possono riutilizzare le frequenze, il numero di celle presenti in un cluster si indica con  $G=x$ . Nei cluster le celle che hanno le stesse frequenze si chiamano **co-channel**.

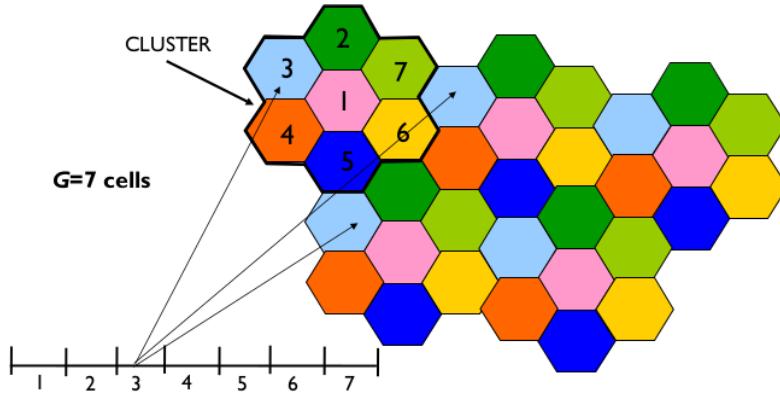


Figure 35: 7 Cell Cluster ( $G=7$ )

Per soddisfare più utenti si possono diminuire le dimensioni delle celle, soddisfando meno utenti per area ma incrementando l'ottimizzazione di utilizzo delle frequenze, se però si inizia a diminuire troppo la dimensione sorgono dei problemi:

- aumento dei costi per creare le maggiori celle;
- diminuendo il numero di celle in un cluster aumenta l'interferenza, la distanza tra i co-channel diventa più piccola, creando interferenze maggiori nella stessa frequenza;

Tecniche di frequency reuse:

- **Splitting:** coesistenza di micro celle e macro celle, esempio: in campagna sarebbe meglio gestita con macro celle più copertura e meno persone da gestire, in città ha più senso usare delle micro celle, meno copertura per più utenti da gestire e più ostacoli da evitare;
- **Cell Shaping:** per evitare degli handover mette le micro celle in posti dove le persone so che rimarranno ferme, e per le persone che si muovono avrà delle macro celle che coprono un area più ampia;
- **Power Control:** è una tecnica per evitare di sprecare più batteria del necessario, per decidere la potenza da utilizzare; le strategie sono: open loop, closed loop, ...; nell'open loop
- **Sectoring:** si vanno a considerare delle antenne con capacità di trasmissioni non omnidirezionali, diminuendo le interferenze in una certa direzione;

- **Tilting:** non usare un angolo di 90 gradi per le trasmissioni, limitando le interferenze;
- **Creating femtocell:** si creano delle celle al volo quando ne ho bisogno dove ne ho bisogno, esempio: uno stadio non avrebbe senso di essere gestito ogni giorno della settimana, se non quando lo stadio si riempie di gente per un evento;

jArchittura ...

## 5.4 Basi Procedures

- **Registrazione:** fornire un associazione ad un rete cellulare, la registrazione viene fatta ogni qual volta un utente voglia accedere ad un servizio;
- **Mobilità:** le procedure legate alla mobilità sono:
  - **Roaming:** il roaming è la capacità di un terminale di essere tracciabile in una rete, tenendo i log su ogni cella in cui è stato attaccato. Per effettuare il roaming la rete ricorda in quale location area il terminale si trova, più celle adiacenti formano una location area, non è detto che una location area sia fatta da celle di un solo operatore. Ognuna di queste location area ha un ID detto Location Area Id (LAI);
  - **Location updating:** è l'operazione che un utente deve fare ogni qual volta si muove e cambia location area, un terminale si accorge di aver cambiato LA quando visualizza un LAI diverso;
  - **Paging:** come si fa ed essere tracciabili? La rete conosce la LA in cui il terminale si trova, ma non la specifica cella, allora quando arriva un messaggio per host x, il sistema manda un **paging message** in broadcast in tutta la LA (simile ad un ARP request), il motivo per cui si manda un messaggio in broadcast in una location area è limitare il numero delle location updating;
  - **Handover:** procedura molto complessa per continuare a mantenere la connessione passando da una cella all'altra, si classificano in: intra-cell vs inter-cell, (soft vs hard) sono collegati ad entrambe le base station, sono collegati prima ad una e poi ad un'altra, (MT vs BS) inizializzata da MT a dalla BS (tipicamente), (Backward vs Forward) procedure gestite dalla cella di arrivo o dalla cella di partenza;

## 5.5 Evoluzione delle reti cellulari

La prima generazione (1G) era una tecnologia completamente in analogico in FDMA, sostituita dal **GSM (Europa) 2G**, in Europa adottato con FDMA/TDMA, il passo

in avanti è stato il passaggio dall'analogico al digitale e l'avvento degli sms, con il **2.5G GPRS/EDGE** è la prima tecnologia di fornire servizi a pacchetti.

**3G** dove il focus è improntato al miglioramento dello scambio di contenuti multi-mediali, che adotta il CDMA in USA, UMTS in Europa. L'estensione del **3.5G**, va a migliorare l'UMTS che diventa HSPA aumentando di fatto la grandezza dei dati scambiati.

**4G** conosciuto come LTE, ha un focus sul creare un'architettura ben integrata con i servizi TPC/IP forniti da internet, vengono create delle antenne MIMO in cui si riesce a trasmettere su più canali in contemporanea, aumentando di molto il bitrate. Per la volta la rete è interamente in IP, anche per chiamate vocali.

**5G** cerca di unificare diverse tecnologie di accesso wireless, di fatto annullando le diversità tra rete cellulare e wifi. Il 5G utilizza delle **mmWave** (micro onde) permettono di avere un throughput maggiore. Si cerca di offrire dei servizi aggiuntivi come l'**edge computing**, spostando le computazioni da un server ad dei nodi vicini alla BS, cercando di fornire dei servizi personalizzati attraverso la virtualizzazione dei servizi di rete (come ad esempio un firewall), detto **NFV (Network Function Visualization)**. Queste funzioni di rete vengono combinate in una catena di rete, utilizzando anche il source routing di ipv6. Esiste anche l'**SDN (Software Define Networking)**, serve per eseguire tutte queste funzioni ho bisogno di un disallineamento tra piano di controllo e piano dati, il router avrà bisogno di un sistema general purpose per poter gestire questi due servizi in modo differente, gestendo in maniera opportuna la catena di servizi.

## 5.6 GSM

Offre servizi come:

- voce a (13 kbit/s o 6.5 kbit/s);
- sms;
- servizi supplementari;

L'architettura GSM è fatta da:

- **Mobile Station:** si tratta di dispositivi in grado di connettersi;
- **Subscriber Identity Module (SIM):** per connettersi alla rete, il dispositivo ha bisogno di una sim, che contiene dei parametri per autenticazione e di altri dati, uno di questi è l'IMSI che è molto simile ad un MAC, il dispositivo + sim formano un **mobile terminale (MT)**;
- **Base Station Subsystem (BSS):** la BS è formata da una Basa Transceiver Station BTS che è in grado di modulare su più frequenze i diversi canali di accesso, e da una Base Station Controller BSC, che gestisce tutta la logica delle

rete, come ad esempio la gestione dall'handover, manda inoltre il segnale di paging per localizzare un utente, e converte i 13kbit per la voce a 64kbit/s (il motivo è perché la rete cellulare usa PCM64);

- **Network and Switching Subsystem (NSS):** la BSS è collegata a:

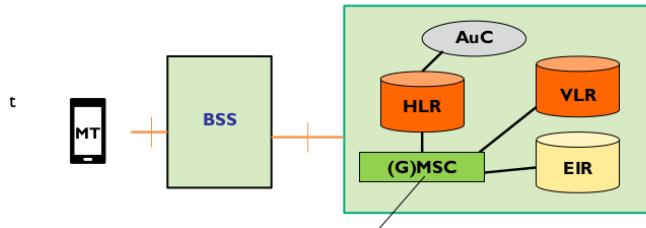


Figure 36: Nss

- MSC (Mobile Switching Center): gestisce l'autenticazione, il routing tra il GSM ed altri network;
- HLR (Home Location Register): è il database della rete a cui appartiene il dispositivo home, nella quale vengono registrati i parametri dell'utente, come le chiavi crittografiche, dati di mobilità, è unico per tutta la rete;
- VLS (Visitor Location Register): contiene le informazioni legati ad un MT presente in quel momento nella rete, mantiene i valori quando l'utente si è spostato;
- AuC (Authentication Center): tramite una tecnologia challenge and response, l'utente si può autenticare;
- EIR (Equipment Identity Register): mantiene un registro di tutti i dispositivi rubati;
- **GMS Frequenze:** le frequenze allocate al GMS sono: 850, 900, 1800, 1900 MHz. Le frequenze in uplink sono diverse da quelle in downlink;
- **Canali:** si usa un approccio FDMA/TDMA, la divisione è in 32 frequenze portanti, inoltre ogni canale viene diviso in 8 slot temporali. All'interno degli slot le informazioni vengono organizzate in burst:

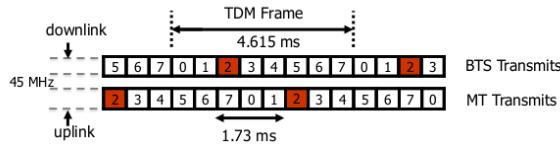


Figure 37: Up Down Link

- **Timing Advance:** ci sono dei ritardi di propagazione, comportando dei problemi con la trasmissione negli slot assegnati, per questo viene usato il timing advance, misurando i ritardi di propagazione si inizia a trasmettere con un anticipo temporale pari a al ritardo di propagazione, infatti all'inizio e alla fine del burst sono presenti dei bit inutili, detti di guardia, per evitare sovrapposizioni con gli altri canali, anche per problemi di disallineamento;

- **Struttura burst:**

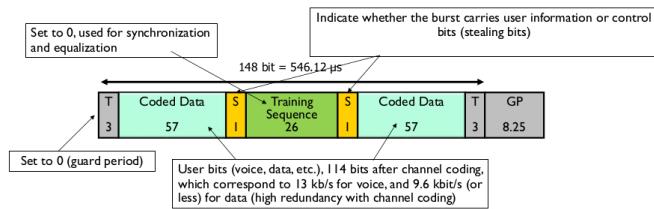


Figure 38: Struttura Burst

- **Canali Logici:** i canali logici sono mappati sui canali fisici che identificano cosa viene trasmesso all'interno dei canali, divisi in traffic channel e control channel.

## 5.7 LTE

Una delle caratteristiche principali è il rimpiazzamento del CDMA con l'FDMA (OFDMA: FDM dove le frequenze portanti sono più vicine tra di loro ed ortogonalni tra di loro).

L'LTE è composto da:

- LTE utilizza 3 diverse bande di frequenza:
  - 2600 MHz per città;
  - 1800 MHz medio gittata;
  - 800 MHz lunghe gittate, pochi utenti, pochi ostacoli;

- **Terminologia:**

- **downlink (DL):**
- **uplink (UL):**
- **User plane:**
- **control plane:**

- **Architettura:** è formata da RAN Radio Access Network, ... , le BSS vengono chiamate eNodeB,

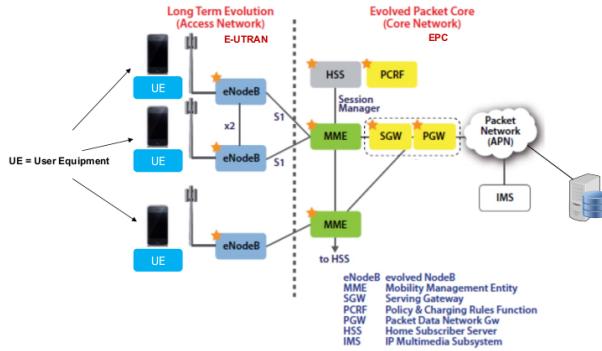


Figure 39: Architettura Lte

- **ECP:** l'EPC è stato ridefinito da zero (clean-slate), gestisce anche le risorse radio;
- **Bearers:** i bearer sono dei tunnel che trasportano il traffico dall'UE ad altri elementi dell'architettura, questi tunnel vengono creati per inviare il traffico attraverso alcuni nodi ritenuti i migliori per gestire il servizio dell'utente, si possono creare dei bearer per dei servizi specifici. I bearer hanno dei nomi specifici:
  - radio bearer: connessione tra UE e eNodeB;
  - s1 bearer: connessione tra eNodeB e l'S-GW (Service Gateway);
  - s5 bearer: connette il S-GW al P-GW.
- **E-UTRAN:** consistono per la maggior parte di eNodeB, la loro funzione principale è la gestione delle risorse, compressione degli header per migliorare le prestazioni, sicurezza, connessione all'EPC;
- **Control/Data Plane:** controllo: protocolli per la mobilità, sicurezza, autenticazione, dati: nuovi protocolli per per livello link e livello fisico,

In LTE c'è una separazione, esistono **Control/Data Plane**:

- **Link Layer Protocol:** formato da:

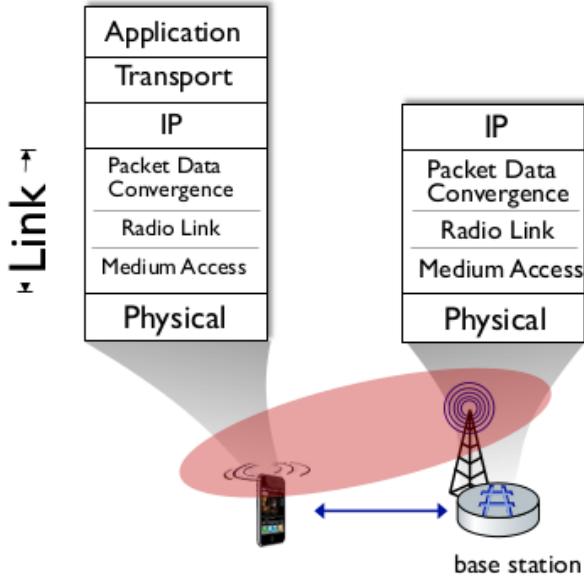


Figure 40: Link Layer

- packet data convergence: compressione e encryption;
- radio link control (RLC): potrebbe essere necessario frammentare i pacchetti e la ricomposizione, inoltre permette un reliable data transfer che permette una verifica dei dati, come nel wifi;
- medium access: richiesta di accesso, uso di slot radio;
- **First Hop:** per accedere ai canali si utilizza un approccio OFDM con TDM, gli slot sono molto piccoli, grazie a questi slot piccoli gli algoritmi di assegnamento potranno assegnare più slot ad un singolo utente, tutti allocati in maniera dinamica;
- **Canali Fisici:** la divisione fatta da OFDM/TDM vengono detti canali, in alcuni canali vengono riservati degli slot specifici per segnali di controllo, esiste dunque un divisione tra canali di controllo e canali di dato;
- **Associazione di un nodo:** la BS fa un broadcast di un **primary sync signal** ogni 5ms su ogni frequenza, il terminale che riceve questo signal ne sceglie uno basandosi sulla potenza del segnale e aspetta il **second sync signal** che contiene delle informazioni sulla BS, il primary da solo il segnale di aggancio, scelta la BS si inizia l'autenticazione e si inizializza il data plane;
- **Sleep Mode:** dopo 100ms di inattività il terminale entra il light sleep e viene periodicamente svegliato ogni 100ms per controllare se ci sono dei messaggi

pending, se i secondi di inattività diventano 5-10s, il terminale va in deep sleep, in cui la connessione va riassociata ad una BS, se il terminale deve ricevere dei messaggi la BS può svegliare il terminale dal deep sleep;

## **6 5G**

Lo scopo è creare una rete wireless unificata (nuovo ordine mondiale), insieme ad altri servizi.

Per offrire tutti questi servizi bisogna disclocare i calcoli vicino all'utente (edge della rete), si ha bisogno di **network slice**, ovvero porzioni di rete dedicate ad un utente, ho bisogno di un controllore centralizzato (SDN) per assegnare le risorse.

Alcuni use cases:

- eMBB (enhanced mobile brodband): definisce come in una rete si possono creare delle ditribuzioni di servizi streaming per utenti mobili;
- mMTC (massive machine type communication): comunicazioni tra macchine industriali;
- URLLC (ultra reliable low latency comunication): comunicazioni con latenze di 1ms;

Le tecnologie usate:

- forme d'onda avanzate;
- MIMO avanzate, con un throught maggiore di quelle di LTE;
- uso delle micro onde;

Core network:

- SDN: controller centralizzato;
- NFV: funzioni virtuali eseguite;

## **7 VPN (Virtual Private Network)**

Si utilizzano le VPN quando ci si vuole connettere in una rete privata passando dalla rete, per questo motivo stabilire un tipo di comunicazione va protetta. Gli elementi chiave di queste connessioni sono:

- **tunnel**: encapsulamento del traffico su un canale condiviso, che in alcune situazione non potrebbe essere disponibile;

- **VPN gateway:** dispositivo ad-hoc per aprire e terminare dei tunnel, che dovrà supportare un protocollo specifico (non sempre il VPN gateway non supporta tutti i protocolli);

Il motivo per cui esistono le VPN è che l'alternativa sarebbe quella di posare un cavo direttamente collegato alla rete, dunque è comodo avere un modo di connettersi ad una rete privata su una rete condivisa. Un altro motivo è l'utilizzo di un indirizzo IP locato in un altro paese.

In base al livello dello stack ISO/OSI è possibile implementare una VPN.

Gli scenari di attuazione di VPN può essere fatta in una azienda solo internamente (**Intranet VPN**) oppure può essere condivisa con più aziende o vari clienti, quindi dovranno essere presi in considerazione aspetti di sicurezza e privacy, entrando in gioco anche diversi livelli di firewall e varie autorizzazioni che vengono assegnate, si deve tenere in considerazione anche il problema dell'overlapping, ovvero avere più host con lo stesso indirizzo privato.

Quando si parla di connessione ad internet attraverso la VPN esistono due tipi di accesso:

- **Centralized Internet Access:** si ha accesso alla sottorete ed a internet;
- **Distributed Internet Access:** solo il traffico indirizzato alla sottorete passa per la VPN, mentre il traffico verso internet passa normalmente dal default gateway;

## 7.1 Deployment Model

Con deployment model ci si focalizza sull'infrastruttura che permette di utilizzare una VPN. Le VPN devono garantire autenticazione dei peer, integrità dei dati e confidenzialità. Quando si fa il deploy possono essere possibili diversi tipi di tunneling.

- **s2s (site 2 site) VPN tunneling:** una sottorete 1 comunica con una sottorete 2 con un tunnel;
- **e2e (end 2 end) VPN tunneling:** due computer in due sottoreti separate vengono connessi con un tunnel;
- **remote VPN tunneling:** un singolo host si collega con un border gateway delle sottorete;

Adoperare un s2s ha dei costi maggiori, infatti i due border gateway (che sono ai bordi delle sottoreti) devono gestire costi di overhead, tunnel multipli e quelli della crittografia, mentre col e2e non si ha questo problema di gestione. Solitamente una buona soluzione implementativa è l'utilizzo in contemporanea di e2e e s2s.

Quando si parla di tunnel esistono diversi metodi di creazione, a differenza della soluzione si avrà una differenza tra performance e sicurezza, queste ultime sono inversamente proporzionali:

- **Overlay Model:** nell'overlay model i tunnel vengono creati tra i due endpoint, non preccopandosi del routing, l'overlay model è una soluzione customer provisioner;
- **Peer Model:** in una soluzione peer model viene creato un tunnel spezzato tra i vari router in cui si passa, quindi esisteranno tanti tunnel quanti sono i router da cui si deve passare per raggiungere la destinazione, il vantaggio di questa soluzione è che si può indirizzare il traffico per un percorso specifico, lo svantaggio è che ad ogni hop il pacchetto viene decapsulato per poi essere reincapsulato in un nuovo pacchetto quando si fa il forward, il peer model è una soluzione di provider provisioner (l'ISP deve provvedere a rendere sicura il traffico tra i peer, che solitamente appartengono a lui);

Per effettuare un implementazione di una VPN esistono due tipi di dispositivi, il CE (Customer Equipment) che appartiene all'utente e gestisce il tunneling a partire dall'uscita della sottorete, ed il PE (Provider Equipment) che appartiene all'ISP, solitamente si tratta di una rete collegata che implementa il VPN, questo servizio non è più disponibile all'uscita di questa della rete di PE.

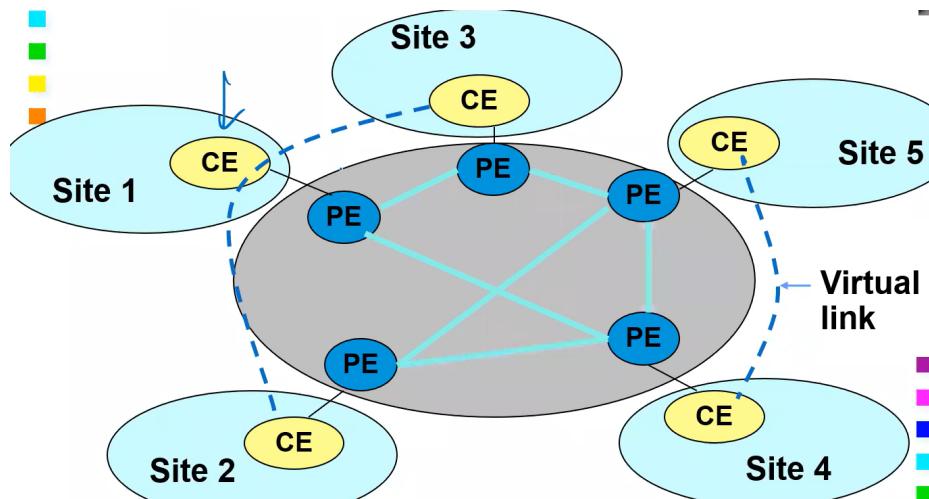


Figure 41: Ce Pe

Il tunneling in una VPN non è altro che l'incapsulamento di un pacchetto IP (header e payload) in un altro pacchetto IP (nel caso di VPN a livello IP), questo nuovo payload viene criptato con i protocolli GRE o IPsec.

## 7.2 Livelli delle VPN

Esistono anche le VPNs di livello 2, vengono dette **Virtual Private LAN Service**, viene usata per emulare le funzionalità delle LAN, serve anche per connettere parti

della sottorete. Il tunnel viene fatto attraverso l'incapsulamento di una trama di livello 2 in pacchetto IP, per questo motivo è necessario avere una rete locale con una tecnologia di livello 3.

Le VPN di livello 3, hanno un routing basato su Peer o Overlay, mentre i CE possono essere connessi con e2e, s2s, remote. L'impacchettamento può essere fatto in GRE o IPsec.

Le VPN di livello 4, hanno una encapsulazione basata su TCP che ottiene un alto livello di sicurezza attraverso SSL.

### 7.3 Protocolli per VPN

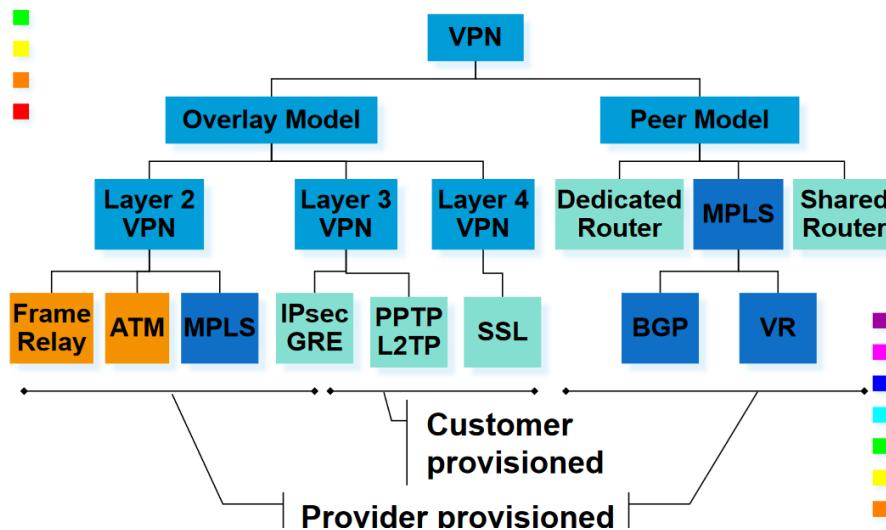


Figure 42: Protocolli Vpn

#### 7.3.1 GRE

Il protocollo **GRE** (Generic Routing Encapsulation) è uno standard per l'encapsulazione di un pacchetto all'interno di un altro, il suo header è fatto da:

- Flags: presenza di campi addizionali;
- s: se la destinazione non è raggiunta quando si è percorsa la lista dei router il pacchetto viene distrutto;
- recur: massimo numero di encapsulazioni successive;
- sequence number: questo campo non riesce ad evitare i replay attack;
- route: è possibile specificare un indirizzo per fare del routing, viene usato insieme al flag s;

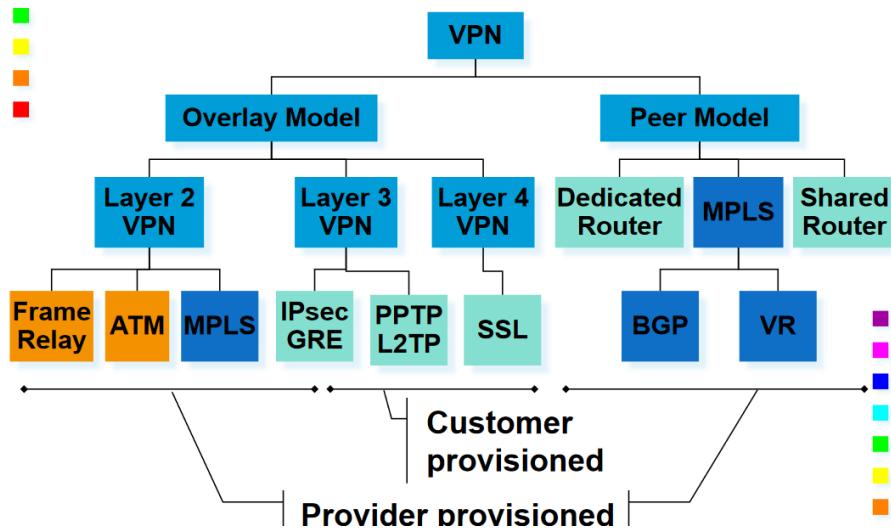


Figure 43: Header Protocollo Gre

Del protocollo GRE esiste anche una versione 1 che migliora gli acknowledge.

## 7.4 L2TP

Soluzione di tipo provider provisioner.

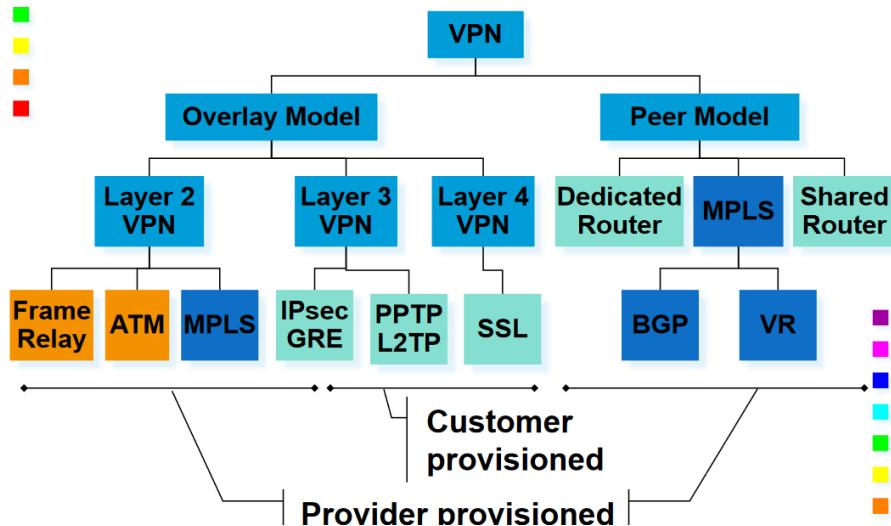


Figure 44: L2Tp Schema

Il protocollo **L2TP** (Layer 2 Tunneling Protocol) è un protocollo di livello 2, che la caratteristica di essere indipendente dal livello 2, non viene utilizzato in pratica perchè richiede un ulteriore encapsulamento in IPsec, rendendo tutto il protocollo abbastanza inutile. L2TP utilizza due dispositivi per creare un tunnel, il LAC (L2TP

Access Concentrator) a cui gli utenti si collegano, poi questo crea un tunnel verso il LNS (L2TP Network Server), che poi si collega alla rete locale dell'azienda.

PPTP, il point to point protocol nasce come una soluzione customer provider, ma anche esso ha una cifratura debole oltre ad avere un sistema di chiavi proprietario.

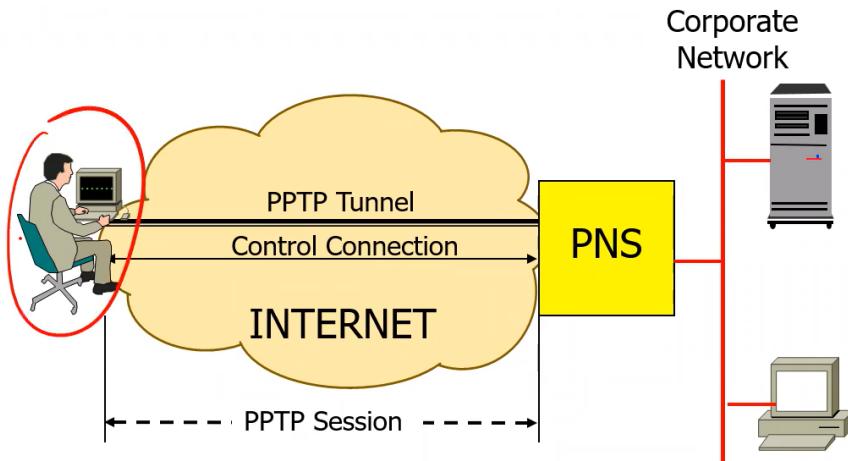


Figure 45: Ptpp Schema

#### 7.4.1 IPsec

Il protocollo IPsec, si basa su altri due protocolli:

- AH: fornisce l'intergrità;
- ESP: fornisce confidenzialità e/o intergrità;

IPsec ha due modalità di funzionamento in base al tipo di VPN che si ha (e2e vs s2s).

Per scambiare le chiavi IPsec si basa sul protocollo IKE.

#### 7.4.2 SSL

Le soluzioni che implementa ssl sono:

- s2s;
- e2e (source service access);
- remote access;
- tunneling su tpc o upd;

Il motivo per cui si preferisce una soluzione SSL invece che su IPsec è che SSL non si interfaccia con il kernel, è poco complesso, è il più sicuro, non ci sono problemi di attraversamento del NAT. SSL funziona molto bene con soluzioni che utilizzano uno scenario di network, ad esempio scenari client-server.

Per rendere un protocollo sicuro basta incapsularlo in un tunnel ssl, ad esempio http diventa https che utilizza ssl per cifrare tutto il traffico.

Per il livello 4 quando si ha un s2s, l'header A to B solitamente è incapsulato (può essere un header di livello 3 o 4) l'header IP è tra i due VPN gateway, se si ha un e2e solo il payload è incapsulato, mentre l'header IP è tra A e B.

## 8 Routing

Il routing viene definito come la creazione delle tabelle di routing, spesso confuso con il forwarding che è l'azione di spedire il pacchetto su una delle diverse interfaccie basandosi sulle tabelle di routing.

Il **proactive routing** (routing) consiste nel scegliere in modo dinamico o statico il percorso migliore per raggiungere un nodo, il percorso migliore è basato su diverse metriche come: il percorso più corto, il percorso con il maggiore throughput, ...

Mentre l'**on-the-fly routing** (forwarding) si basa sullo spedire un pacchetto su una delle interfaccie basandosi su: label swapping, indirizzi, source routing (indirizzamento del traffico), a differenza degli algoritmi di forwarding si possono avere dei risultati diversi.

Gli algoritmi di routing si distinguono in:

- **statici**: le routing table possono essere configurate manualmente in modo statico, detto fixed directory routing, anche il flooding è un algoritmo statico perchè si mandano pacchetti a tutti i nodi indistintamente;
- **dinamici**: gli algoritmi dinamici si dividono in: centralizzato, distribuito, isolato;

### 8.1 Centralized

Esiste un **Routing Control Center** (RCC), che è a conoscenza di tutti i nodi della rete, decide per ognuno la propria routing table riuscendo anche a fare del load balancing tra il traffico, la gestione della rete viene fatto in modo ottimo ma la svantaggio è che l'RCC diventa un collo di bottiglia ed un single point of failure, si può anche ridondare l'RCC limitando questi due lati negativi.

### 8.2 Isolated

La scelta del percorso che deve essere presa da pacchetto deve essere fatto in modo isolato per ogni nodo, infatti non vengono scambiate informazioni.

### 8.3 Distributed

Gli algoritmi distribuiti vengono basati sul tipo di informazioni che vengono scambiate tra i diversi nodi.

Uno degli algoritmi è il **Distance Vector**, si tratta dell'implementazione dell'algoritmo di Bellman-Ford, dove vengono scambiate informazioni solo tra i vicini, ovvero i nodi direttamente connessi, in cui ogni router avrà la sua routing table. Le informazioni scambiate sono solo le distanze tra i vicini, per far questo c'è una fase detta di **transitorio** in cui tutti i nodi riescono a scambiare le informazioni tra di

loro. Il problema del transitorio è molto grave, infatti il transitorio non esiste solo quando c'è un cold start ma anche quando ci sono dei guasti nella rete, infatti nascono delle problematiche che sono:

- black hole;
- count to infinity: scenario in cui dei router continuano a scambiarsi informazioni su un nodo non raggiungibile, incrementando la sua distanza all'infinito;

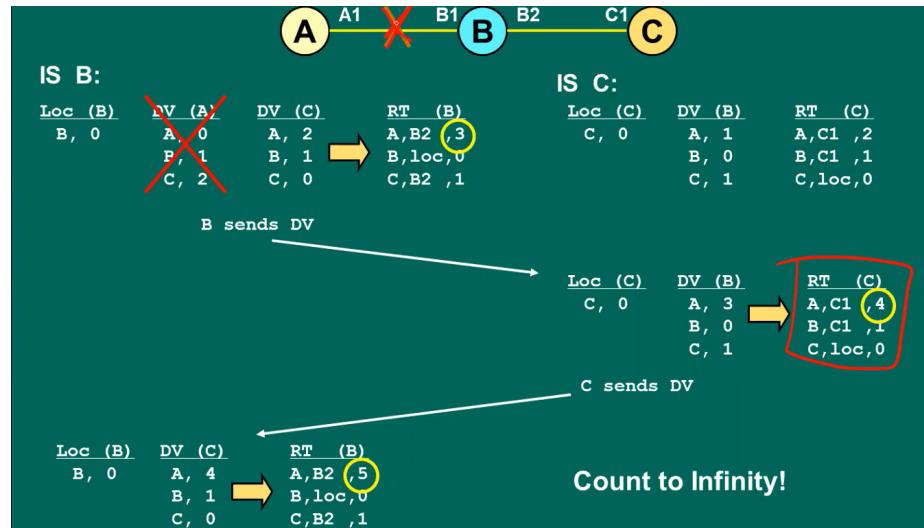


Figure 46: Count To Infinity

- bouncing effect: il messaggio viene rimbalzato tra due nodi per sempre;

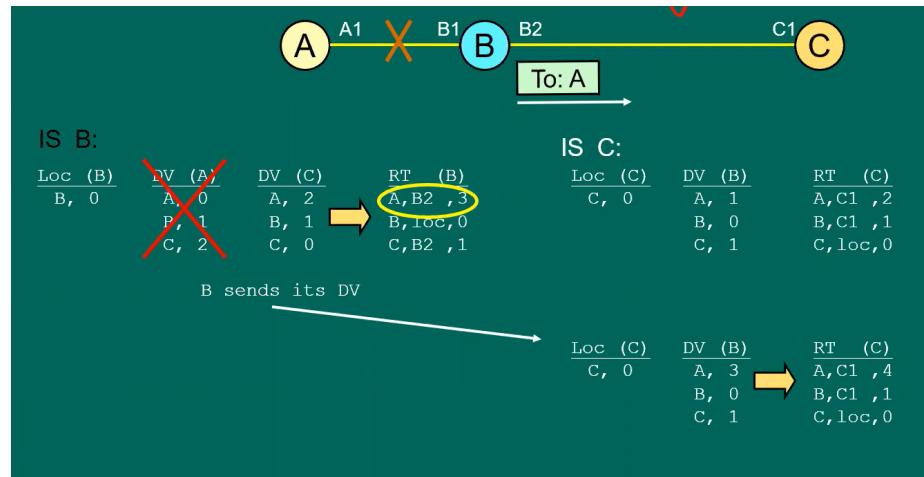


Figure 47: Bouncing Effect

Le soluzioni del count to infinity e bouncing effect sono:

- **split horizon:** la variante si basa sul principio: *"se C raggiunge A attraverso B, è inutile per B provare a raggiungere A attraverso C"*, questo serve per evitare la creazione di loop. Così facendo B non propaga sul distance vector l'informazione di A. In particolare il DV di C verso B non contiene destinazioni reggiunte attraverso B.

Su una rete completamente connesse esiste ancora il problema del transitorio perché i nodi non sono a conoscenza di tutti i path. Il problema si risolve quando vengono inviati nuovamente i DV, però i DV verranno inviati anche al nodo connesso al guasto "avvelenando" la sua RT, questo problema viene risolto dal path hold down;

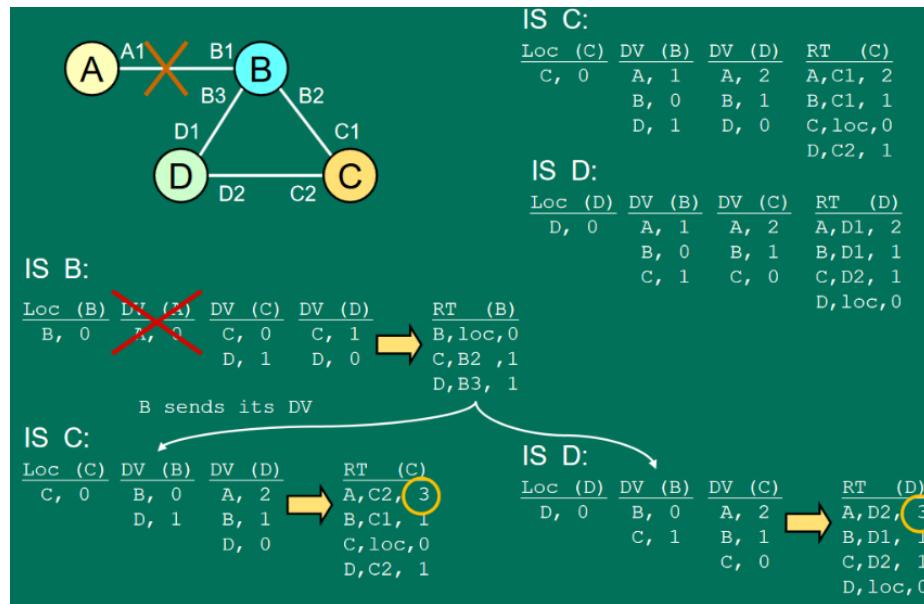


Figure 48: Split Horizon Transitorio

- **Path Hold Down:** *"Se un link L fallisce, tutte le destinazioni raggiungibili attraverso il link L sono considerate irraggiungibili per un certo periodo di tempo"*, in questo modo nessun percorso di A viene calcolato.

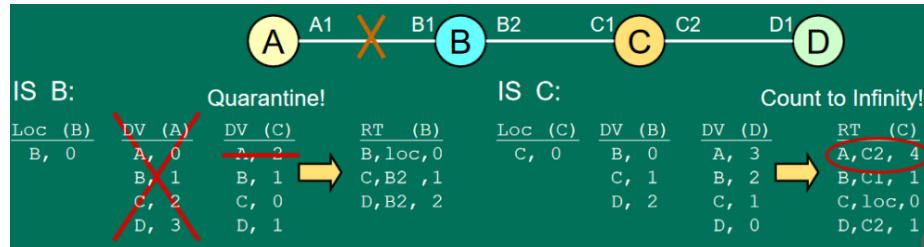


Figure 49: Path Hold Down

Il path hold down non risolve il problema del count to infinity, questo può essere superato dall'utilizzo combinato con lo split horizon;

- **Route poisoning:** quando un link si guasta la distanza viene impostata direttamente a infinito, evitando il count to infinity, questo algoritmo però è molto lento, infatti viene sconsigliato per reti molto grandi;
- **Path Vector:** questo algoritmo elimina il problema dei loop, al posto del DV viene inviato un path vector dove viene inserito il costo di un link ed il percorso da cui il PV arriva;
- **Link State:** in questo algoritmo ogni nodo attraverso le informazioni dei suoi vicini riesce a calcolare la topologia della rete, questa informazione viene detta link state, il link state viene poi condiviso tra tutti i nodi, in questo modo ogni nodo possiede la mappa di tutta la topologia della rete. Per calcolare poi le RT viene usato l'algoritmo di Djikstra.

## 8.4 Internet Routing Architecture

...

# 9 Multi-Protocol Label Switching (MPLS)

MPLS nasce con il supporto di molti protocolli. *"MPLS è l'abilitazione per la Nuova Rete (IP) Pubblica a Bandlarga"*, per rete pubblica si intende una rete dove viaggia del traffico proveniente da diversi , il motivo per cui non si possono utilizzare i router IP per fare questo tipo di infrastruttura è che per decidere la strada che un pacchetto deve prendere si basa sull'indirizzo di destinazione creano un collo di bottiglia , MPLS è una tecnologia che deve superare questo problema. Per superare questo problema negli anni passati si doveva creare una "cipolla" di protocolli per dirigere il traffico in modo da creare congestioni, grazie ad MPLS si utilizza un solo tipo di commutatore collegato con gli switch ottici, rimuovendo tutta la ridondanza, MPLS è stato progettato per funzionare con IP e per abbattere i costi degli operatori.

L'idea di MPLS è: non inoltrare i pacchetti IP in base all'indirizzo di destinazione, ma in base ed una etichetta piazzata davanti al pacchetto, è per questo motivo che MPLS è multiprotocollo.

Il motivo per cui si è adottata questa tecnologia è il problema del **longest prefix matching**, che richiede del tempo per la ricerca all'interno della tabella di routing e lo spazio utilizzato per immagazzinare le strutture dati di supporto. Il vantaggio di avere un etichetta è che il valore di quell'etichetta è il numero della riga della tabella dove si trova il next hop, molto più facile e veloce. Un altro motivo per cui l'utilizzo di etichette è importante è la possibilità di realizzare del **Traffic Engineering**, ovvero determinare come viene distribuito il traffico in una rete.

MPLS introduce un paradigma **connection-oriented** nelle reti IP.

MPLS utilizza una architettura che non arriva all'endsystem, per evitare ciò che avvenne con IPv6 dove andavano cambiati tutti i software. MPLS può essere adottato in modo incrementale, esiste una **rete MPLS** dove sono presenti:

- **Label edge router**: questi router si trovano ai bordi della rete che inseriscono e rimuovono le etichette;
- **Label router**: router che fanno il routing delle etichette;

All'ingresso della rete devono essere creati **Label Switched Path** (LSP), ovvero le "connessioni" create all'interno della rete. Le etichette vengono cambiate ad ogni hop, se si volesse utilizzare una etichetta per ogni connessione, servirebbe un'etichetta per ogni possibile percorso, il che è molto oneroso, cambiare l'etichetta fa sì che la sua lunghezza non sia lunga.

I punti chiave sono:

- **intestazione**: dove viene messa l'etichetta;
- protocolli per distribuire l'etichetta;
- protocolli di routing potenziati, in cui si vuole in percorso più breve ed un link carico più del 50%;

## **9.1 Shim Header 9 MULTI-PROTOCOL LABEL SWITCHING (MPLS)**

---

Storia di MPLS: ..., venne usato per utilizzare utilizzare la scambio di pacchetti IP su delle reti ATM.

### **9.1 Shim Header**

L'header MPLS è fatto da uno o più moduli da aggiungere tra il livello 2 e livello 3. In ogni modulo viene inserita l'**etichetta** (20 bit), **bit sperimentali** (3 bit ridondanti), **bottom of stack** (1 bit) il border router quando toglie le etichette deve sapere qual'è l'ultimo modulo alla fine dello stack, **TTL** (8 bit) con un funzionamento identico al TTL dell'header IP.

Se il protocollo di livello 2 è un protocollo connection-oriented si utilizza la label di quel protocollo lasciando intatto lo standard, il motivo per cui si possono utilizzare è che l'hardware rimane invariato, ma il software implementa MPLS.

### **9.2 Forwarding Equivalence Class (FEC)**

La FEC viene associato ad ogni pacchetto che fa parte di un LSP,

gli LSR devono scegliere un'etichetta (**label binding**), una volta fatto il binding si deve creare la riga nella tabella di routing (**label mapping**), un'etichetta deve essere comunicata agli LSR vicini (**label distribution**)

Label Binding deve essere fatto dal router che sta a valle di un link, ovvero dal nodo che riceverà pacchetti con quell'etichetta (**downstream binding**), inoltre i pacchetti con la stessa FEC devono essere ricevuti con la stessa label, ed il nodo in upstream (a monte) deve essere notificato, inoltre il label binding deve può essere unsolicited o ...;

Label Mapping quando arriva una label viene fatto il mapping alla nuova label e viene fatto il forwarding al next hop. Il nodo a valle vedrà arrivare pacchetti con la stessa label scelti da lui. Il routing viene deciso con i protocolli di routing.

Label Distribution quando un router ha fatto il binding farà una label distribution per notificare ai vicini la scelta della label per una FEC.

Il label binding può anche essere fatto in modo dinamico, le modalità con cui viene scatenato sono: scatenato dai dati oppure da dei messaggi di controllo. Quando il binding è control driven esistono due modalità di operamento:

- **topology based:** la creazione di LSP è data dalla scoperta di percorsi di routing;
- **creazione esplicita:** quando un router decide che serve un LSP per un FEC, allora viene creato, questa segnalazione parte da un label edge router;

Per distribuire i label esistono 3 soluzioni diverse:

- protocolli di routing (BGP): solo in topology based;

## **9.3 VPN via MPLS**

---

- **label distribution protocol (LDP)**: evoluzione del protocollo inventato da cisco;
- **resource reservation protocol (RSVP)**: implementata da quasi tutti i costruttori, dove i nodi prenotano delle risorse per un tipo di pacchetto e poi la conferma viene propogata al contrario;

Modi di Routing:

- hop-by-hop routing;
- ...

### **9.3 VPN via MPLS**

(Layer 2) Si può creare una **pseudo wire emulation end-to-end**: due label edge router creano una LSP tra di loro, quindi in traffico viene trasportato solo tra questi due CE, vengono usati due label per maggiore scalabilità, il problema è che le connessioni devo essere fatte manualmente.

(Layer 3) La rete mpls che viene usata per collegare due sottoreti aziendali, i ce possono scambiarsi dei dati sulle posizioni.

I dispositivi CE scambiano informazioni col la sottorete dell'azienda per conoscere gli host presenti all'interno, parlando direttamente con il PE. Il PE viene configurato manualmente per sapere l'interfaccia a quale rete è collegata, infatti se tra le sottoreti collegati esistono dei duplicati la configurazione serve a distinguere se quale interfaccia mandare i pacchetti.

I router P non hanno informazioni dei nodi all'interno della rete MPLS, quindi si

...

I router PE scambiano informazioni di routing attraverso il protocollo IBGP (interior border gateway protocol), per creare un LSP per collegare la rete MPLS al raggiungimento dell'altra rete aziendale, i label edge router per scambiare informazioni tra di loro utilizzeranno EBGP (exterior BGP).

Per immagazzinare le informazioni vengono utilizzate le **VRF** (vpn routing and forwarding) table, questo serve a distinguere le destinazioni.

... da finire ...

### **9.4 6BE**

Lo scenario è che ci sono diverse reti IPv6 che devono scambiare informazioni attraverso una rete MPLS con IPv4. Il problema è che sul piano di controllo scambiano informazioni di routing con gli indirizzi ipv4, mentre i capi sono con indirizzi ipv6. I PE a questo punto diventeranno dual stack e sarà il loro compito di creare l'LSP con l'altre label edge e creare un comunicazione tra i due punti, questa comunicazione è detta 6PE.

## 10 Reti ottiche

Nelle reti ottiche la commutazione è ottica, questa tecnologia è nata insieme al WDM (wavelength division multiplexing), dove vengono presi più segnali e vengono messi sulla stessa fibra, esistono due tipi di WDM:

- Dense WDM: si possono trasmettere centinaia di messaggi;
- Coarse WDM: si può mettere qualche onda;

Nell'ottica si usano delle zone di frequenze predefinite dette finestre, per cui l'attenuazione del segnale è minimo, il coarse usa un segnale per ogni finestra, mentre il dense riesce a mettere molte su ogni frequenze e non solo nelle finestre.

Quando ho una fibra su cui passano più canali ottici per si utilizza un wavelength switching ( $\lambda$  switching), con questi switch ottici si costruirà una rete che compone la parte centrale della rete, dove ai bordi sono collegati ai router, attraverso dei canali ottici due router potranno comunicare tra di loro.

Gli switch ottici possono essere:

- con un core ottico vs elettronico;
- cross connect (configurazione statica) vs switch (configurazione dinamica);
- wavelength conversion;

Gli switch ideali hanno un **optical core** (completamente ottico), vengono usati solo degli specchi, per far questo si utilizza la tecnologia MEMS (micro electro mechanical systems), dove si costruiscono delle superfici riflettenti dell'ordine dei micrometri che sono orientabili con una differenza di potenzia, essendo che viene usata pochissima corrente non si deve dissipare la potenza prodotta.

Mentre venivano progettati gli switch ottici con i meccanismi si possono utilizzare gli switch con core elettronico, dove in entrata si prendono i bit di un canale e verranno commutati su un nuovo canale, dove viene fatto sparare un nuova laser.

### 10.1 Piano di Controllo

Per poter fare del provisioning o della protection in caso di guasti si ha bisogno di un piano di controllo.

Il motivo per cui si encapsula un pacchetto IP con un protocollo di livello 2 è perchè IP non ha nessuna informazione su dove inizia o finisce il pacchetto.

### 11 Qualità del servizio

Ad esempio nello streaming video si utilizza un ritardo costante nella ricezione, per far sì che il buffer al ricevitore sia sempre con pacchetti da far vedere all'utente e per dare tempo ai pacchetti che arrivino dalla rete. Con le applicazioni multimediali o **real time**, si deve far sì che i dati arrivino con un ritardo controllato.

Se si vogliono fornire dei ritardi bassi e costanti, esistono delle contromisure:

- **classificazione del traffico;**
- **algoritmi di scheduling**, per i pacchetti che competono per delle risorse;
- controllo sul traffico che entra nella rete;
- QoS routing;

La classificazione va fatta in base alla quintupla: dest IP, src IP, transport protocol, dest port, src port. Basandosi su queste informazioni si può classificare le applicazioni in real time.

Tra gli algoritmi di scheduling uno dei più efficienti è il **weighted fair queuing** (WFQ), che riesce a garantire che ogni applicazione della rete riesce ad avere il QoS che ha richiesto.

Soluzioni standard al QoS in IP. Uno è **IntServ** (internet's ambitious solution), si garantisce un'allocazione di risorse ad ogni applicazione della rete che ne richiede usando RSVP, e si garantisce il QoS.