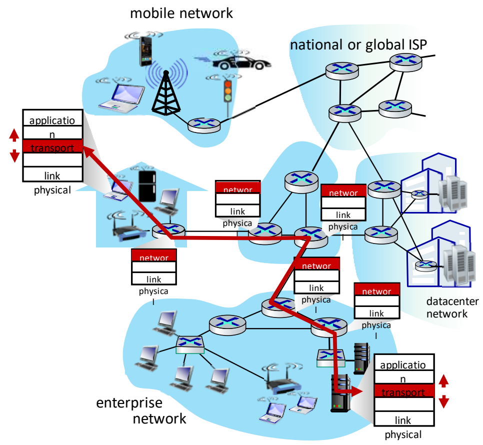
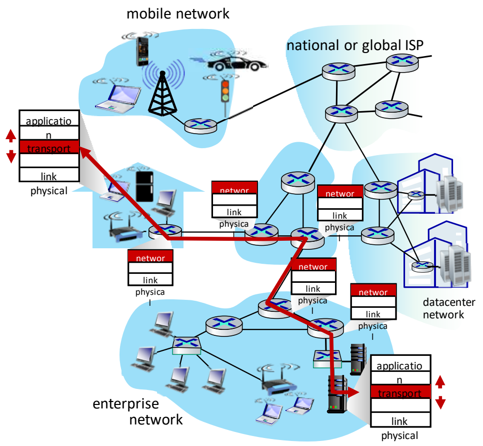
Network layer



Network-layer services and protocols

- transport segment from sending to receiving host
  - sender: encapsulates segments into datagrams, passes to link layer
  - receiver: delivers segments to transport layer protocol
- network layer protocols in *every Internet device*: hosts, routers
- routers:
  - examines header fields in all IP datagrams passing through it
  - moves datagrams from input ports to output ports to transfer datagrams along end-end path



Two key network-layer functions

network-layer functions:

- *forwarding:* move packets from a router's input link to appropriate router output link
- *routing:* determine route taken by packets from source to destination
  - *routing algorithms*

analogy: taking a trip

- *forwarding:* process of getting through single interchange
- *routing:* process of planning trip from source to destination

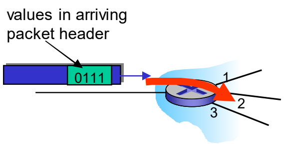
forwarding

routing

Network layer: data plane, control plane

Data plane:

- *local*, per-router function
- determines how datagram arriving on router input port is forwarded to router output port



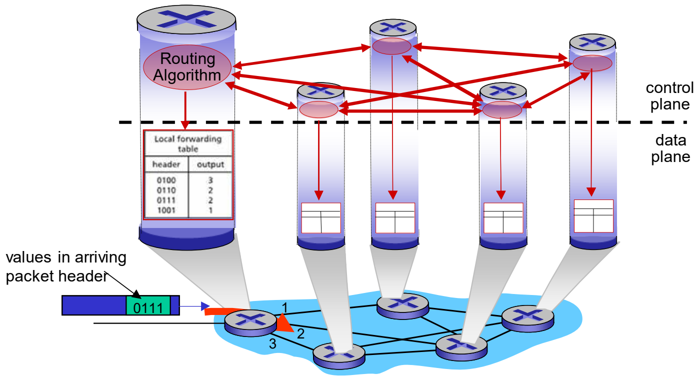
Control plane

- *network-wide* logic
- determines how datagram is routed among routers along end-end path from source host to destination host

- two control-plane approaches:
  - *traditional routing algorithms:* implemented in routers
  - *software-defined networking (SDN)*: implemented in (remote) servers

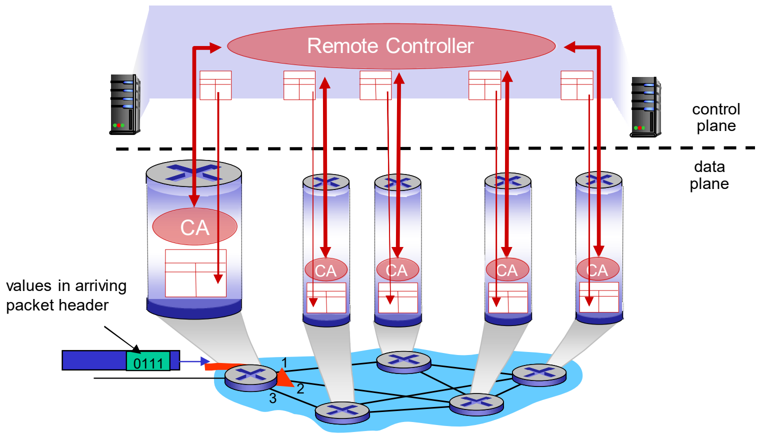
Per-router control plane

Individual routing algorithm components *in each and every router* interact in the control plane



Software-Defined Networking (SDN) control plane

Remote controller computes, installs forwarding tables in routers



Network service model

Q: What *service model* for "channel" transporting datagrams from sender to receiver?

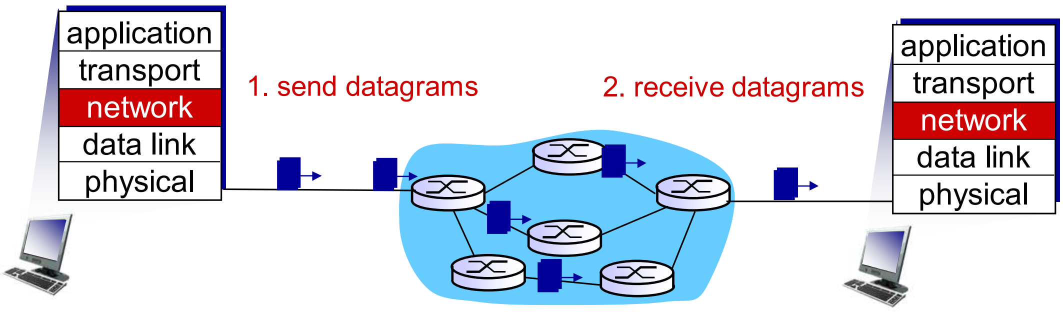
example services for *individual* datagrams:

- guaranteed delivery
- guaranteed delivery with less than 40 msec delay

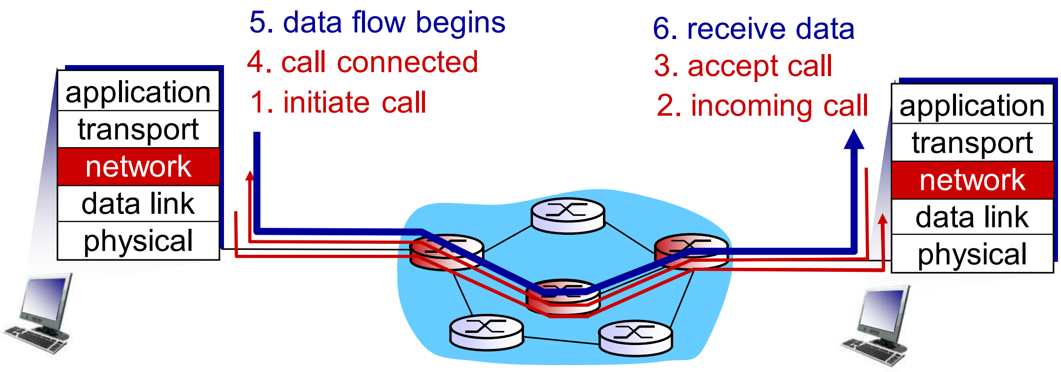
example services for a *flow* of datagrams:

- in-order datagram delivery
- guaranteed minimum bandwidth to flow
- restrictions on changes in inter-packet spacing

Datagram service



Virtual Circuit service



Virtual circuit vs Datagram

	Datagram	Virtual circuit
Circuit creation	No	Yes
State information	Address in every packet	VC address
Routing	No information	Every VC must be identified
Effects of router faults	No	Fault for every VC involved
Congestion control	Complex	Easy

Università di Catania

Network-layer service model

Network Architecture	Service Model	Quality of Service (QoS) Guarantees ?			
		Bandwidth	Loss	Order	Timing
Internet	best effort	none	no	no	no

Internet "best effort" service model

No guarantees on:

1) successful datagram delivery to destination
2) timing or order of delivery
3) bandwidth available to end-end flow

Network-layer service model

Network Architecture	Service Model	Quality of Service (QoS) Guarantees ? Bandwidth	Loss	Order	Timing
Internet	best effort	none	no	no	no
ATM	Constant Bit Rate	Constant rate	yes	yes	yes
ATM	Available Bit Rate	Guaranteed min	no	yes	no
Internet	Intserv Guaranteed (RFC 1633)	yes	yes	yes	yes
Internet	Diffserv (RFC 2475)	possible	possibly	possibly	no

Reflections on best-effort service:

- simplicity of mechanism has allowed Internet to be widely deployed adopted
- sufficient provisioning of bandwidth allows performance of real-time applications (e.g., interactive voice, video) to be “good enough” for “most of the time”
- replicated, application-layer distributed services (datacenters, content distribution networks) connecting close to clients’ networks, allow services to be provided from multiple locations
- congestion control of “elastic” services helps

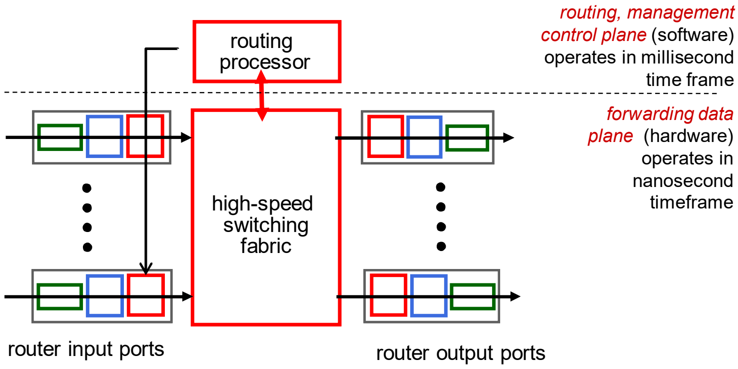
It’s hard to argue with success of best-effort service model

Network layer: "data plane" roadmap

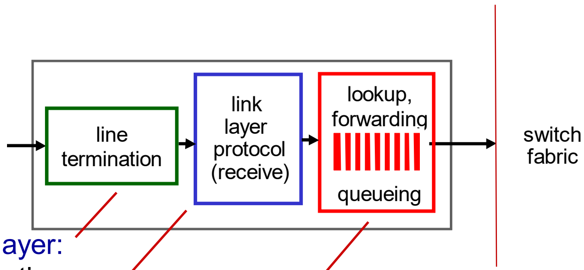
- Network layer: overview
  - data plane
  - control plane
- What's inside a router
  - input ports, switching, output ports
  - buffer management, scheduling
- IP: the Internet Protocol
  - datagram format
  - addressing
  - network address translation
  - IPv6

Router architecture overview

High-level view of generic router architecture:



Input port functions



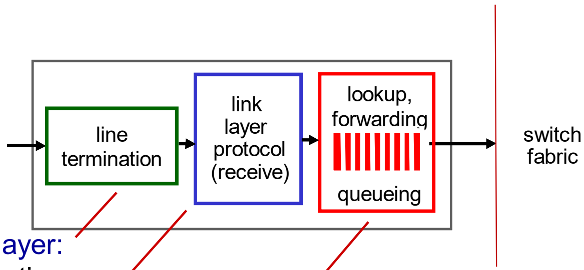
physical layer:
bit-level reception

link layer:
e.g., Ethernet

decentralized switching:

- using header field values, lookup output port using forwarding table in input port memory *("match plus action")*
- goal: complete input port processing at 'line speed'
- input port queuing: if datagrams arrive faster than forwarding rate into switch fabric

Input port functions



physical layer:
bit-level reception

link layer:
e.g., Ethernet

decentralized switching*:*

- using header field values, lookup output port using forwarding table in input port memory *("match plus action")*
- destination-based forwarding: forward based only on destination IP address (traditional)
- generalized forwarding: forward based on any set of header field values

Destination-based forwarding

forwarding table

Destination Address Range	Link Interface
11001000 00010111 00010000 00000000 through 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 through 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 through 11001000 00010111 00011111 11111111	2
otherwise	3

Q: but what happens if ranges don't divide up so nicely?

Destination-based forwarding

forwarding table

Destination Address Range	Link Interface
11001000 00010111 00010000 00000000 through 11001000 00010111 00010111 11111111	0
11001000 00010111 00010000 00000100 through 11001000 00010111 00010000 00000111	3
11001000 00010111 00011000 00000000 through 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 through 11001000 00010111 00011111 11111111	2
otherwise	3

Longest prefix matching

longest prefix match

when looking for forwarding table entry for given destination address, use *longest* address prefix that matches destination address.

Destination Address Range	Link interface
11001000 00010111 00010*** ********	0
11001000 00010111 00011000 ********	1
11001000 00010111 00011*** ********	2
otherwise	3

examples:

11001000 00010111 00010110 10100001 which interface?

11001000 00010111 00011000 10101010 which interface?

Longest prefix matching

longest prefix match

when looking for forwarding table entry for given destination address, use *longest* address prefix that matches destination address.

Destination Address Range	Link interface
11001000 00010111 00010*** ********	0
11001000 00010111 00011000 ********	1
11001000 00010111 00011*** ********	2
otherwise	3

match!

examples:

11001000 00010111 00010110 10100001 which interface?

11001000 00010111 00011000 10101010 which interface?

Longest prefix matching

longest prefix match

when looking for forwarding table entry for given destination address, use *longest* address prefix that matches destination address.

Destination Address Range	Link interface
11001000 00010111 00010*** ********	0
11001000 00010111 00011000 ********	1
11001000 00010111 00011*** ********	2
otherwise	3

match!

examples:

11001000 00010111 00010110 10100001 which interface?

11001000 00010111 00011000 10101010 which interface?

Longest prefix matching

longest prefix match

when looking for forwarding table entry for given destination address, use *longest* address prefix that matches destination address.

Destination Address Range	Link interface
11001000 00010111 00010*** ********	0
11001000 00010111 00011000 ********	1
11001000 00010111 00011*** ********	2
otherwise	3

match!

examples:

11001000 00010111 00010110 10100001 which interface?

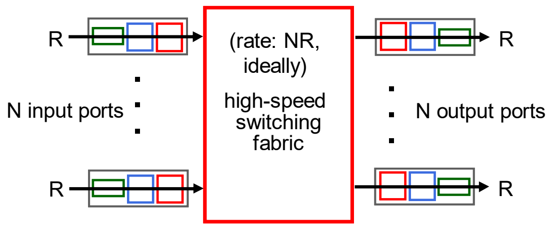
11001000 00010111 00011000 10101010 which interface?

Longest prefix matching

- we'll see *why* longest prefix matching is used shortly, when we study addressing
- longest prefix matching: often performed using ternary content addressable memories (TCAMs)
  - *content addressable:* present address to TCAM: retrieve address in one clock cycle, regardless of table size
  - Cisco Catalyst: ~1M routing table entries in TCAM

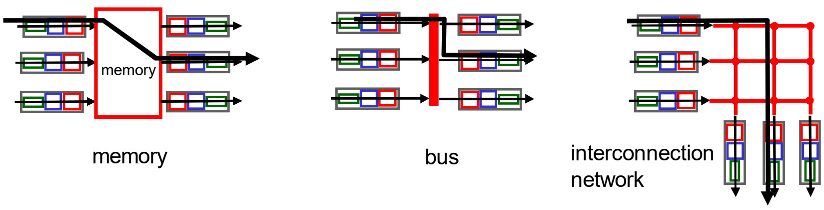
Switching fabrics

- transfer packet from input link to appropriate output link
- switching rate: rate at which packets can be transfer from inputs to outputs
  - often measured as multiple of input/output line rate
  - N inputs: switching rate N times line rate desirable



Switching fabrics

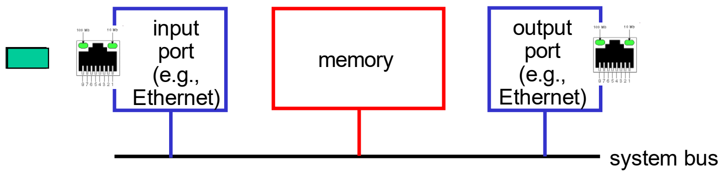
- transfer packet from input link to appropriate output link
- switching rate: rate at which packets can be transfer from inputs to outputs
  - often measured as multiple of input/output line rate
  - N inputs: switching rate N times line rate desirable
- three major types of switching fabrics:



Switching via memory

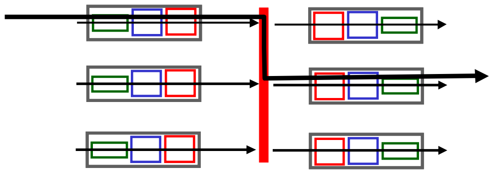
first generation routers:

- traditional computers with switching under direct control of CPU
- packet copied to system's memory
- speed limited by memory bandwidth (2 bus crossings per datagram)



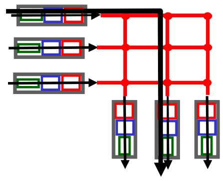
Switching via a bus

- datagram from input port memory to output port memory via a shared bus
- *bus contention:* switching speed limited by bus bandwidth
- 32 Gbps bus, Cisco 5600: sufficient speed for access routers

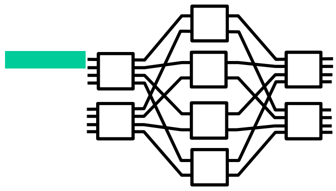


Switching via interconnection network

- Crossbar, Clos networks, other interconnection nets initially developed to connect processors in multiprocessor
- multistage switch: *nxn* switch from multiple stages of smaller switches
- exploiting parallelism:
  - fragment datagram into fixed length cells on entry
  - switch cells through the fabric, reassemble datagram at exit



3x3 crossbar

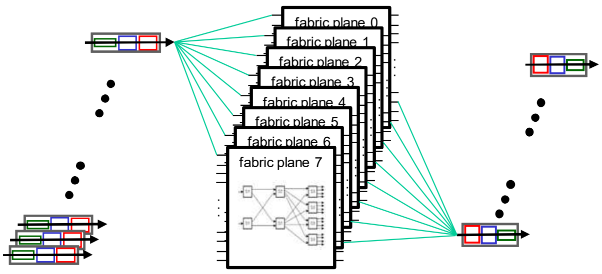


8x8 multistage switch
built from smaller-sized switches

Switching via interconnection network

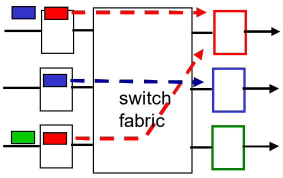
- scaling, using multiple switching “planes” in parallel:
  - speedup, scaleup via parallelism

- Cisco CRS router:
  - basic unit: 8 switching planes
  - each plane: 3-stage interconnection network
  - up to 100’s Tbps switching capacity

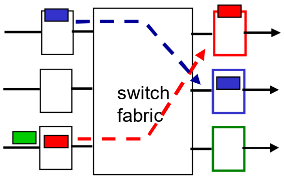


Input port queuing

- If switch fabric slower than input ports combined -> queueing may occur at input queues
  - queueing delay and loss due to input buffer overflow!
- Head-of-the-Line (HOL) blocking: queued datagram at front of queue prevents others in queue from moving forward

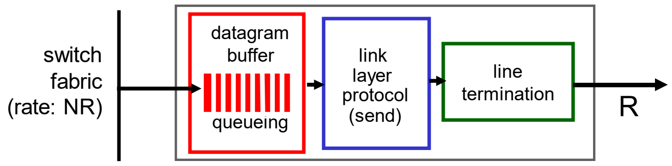


output port contention: only one red datagram can be transferred. lower red packet is *blocked*



one packet time later: green packet experiences HOL blocking

Output port queuing



This is a really important slide

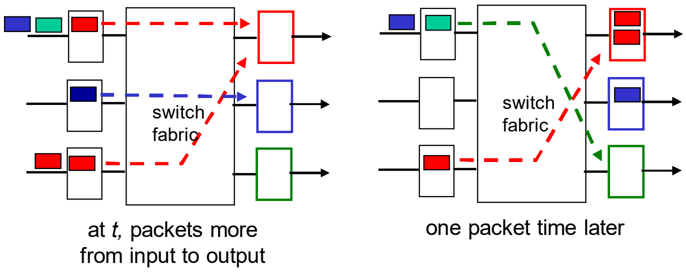
- *Buffering* required when datagrams arrive from fabric faster than link transmission rate. *Drop policy:* which datagrams to drop if no free buffers?

  → Datagrams can be lost due to congestion, lack of buffers

- *Scheduling discipline* chooses among queued datagrams for transmission

  → Priority scheduling – who gets best performance, network neutrality

Output port queuing



- buffering when arrival rate via switch exceeds output line speed
- *queueing (delay) and loss due to output port buffer overflow!*

How much buffering?

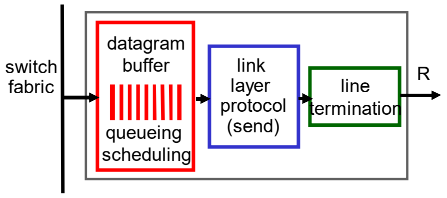
- RFC 3439 rule of thumb: average buffering equal to “typical” RTT (say 250 msec) times link capacity C
  - e.g., C = 10 Gbps link: 2.5 Gbit buffer

- more recent recommendation: with *N* flows, buffering equal to

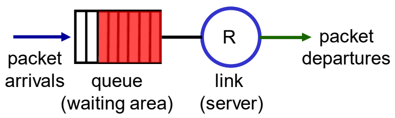
$$\frac{RTT \cdot C}{\sqrt{N}}$$

- but *too* much buffering can increase delays (particularly in home routers)
  - long RTTs: poor performance for realtime apps, sluggish TCP response
  - recall delay-based congestion control: “keep bottleneck link just full enough (busy) but no fuller”

Buffer Management



Abstraction: queue



buffer management:

- drop: which packet to add, drop when buffers are full
  - tail drop: drop arriving packet
  - priority: drop/remove on priority basis
- marking: which packets to mark to signal congestion (ECN, RED)

Packet Scheduling: FCFS

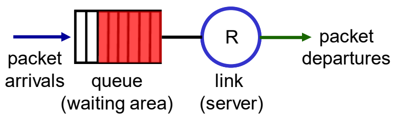
packet scheduling: deciding which packet to send next on link

- first come, first served
- priority
- round robin
- weighted fair queueing

FCFS: packets transmitted in order of arrival to output port

- also known as: First-in-first-out (FIFO)
- real world examples?

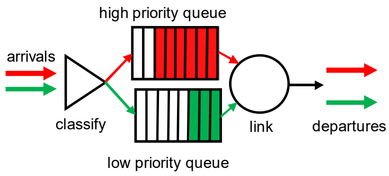
Abstraction: queue

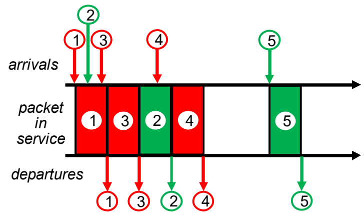


Scheduling policies: priority

Priority scheduling:

- arriving traffic classified, queued by class
  - any header fields can be used for classification
- send packet from highest priority queue that has buffered packets
  - FCFS within priority class

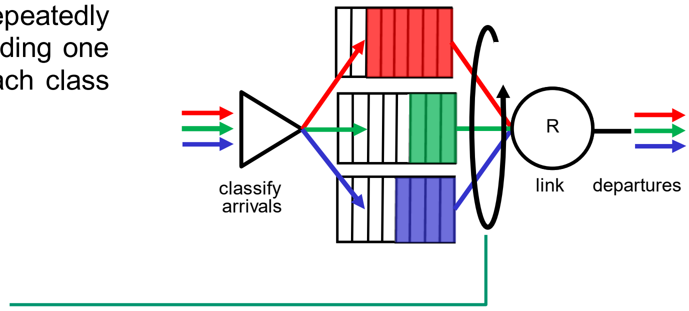




Scheduling policies: round robin

Round Robin (RR) scheduling:

- arriving traffic classified, queued by class
  - any header fields can be used for classification
- server cyclically, repeatedly scans class queues, sending one complete packet from each class (if available) in turn



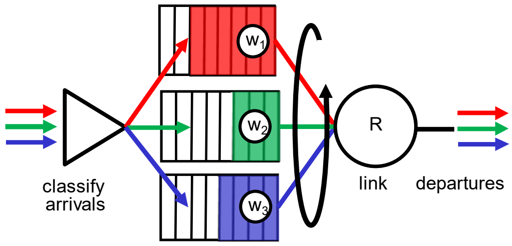
Scheduling policies: weighted fair queueing

Weighted Fair Queuing (WFQ):

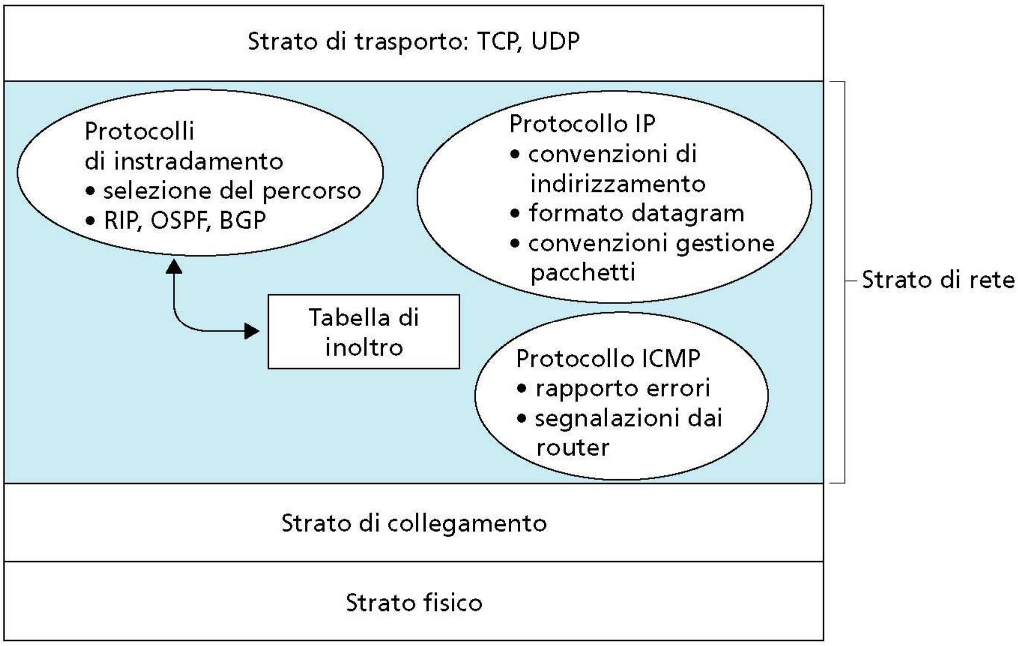
- generalized Round Robin
- each class, *i,* has weight, w_i, and gets weighted amount of service in each cycle:

$$\frac{w_i}{\Sigma_j w_j}$$

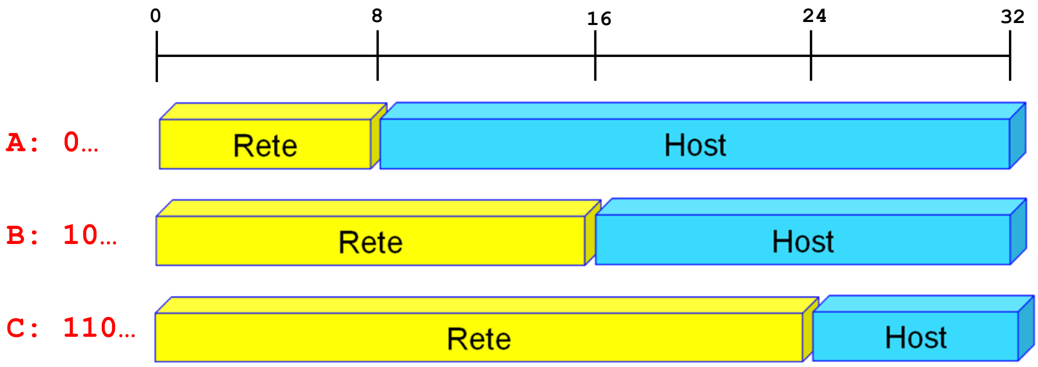
- minimum bandwidth guarantee (per-traffic-class)



IPv4


Strato di trasporto: TCP, UDP
Protocolli di instradamento
• selezione del percorso
• RIP, OSPF, BGP
Protocollo IP
• convenzioni di indirizzamento
• formato datagram
• convenzioni gestione pacchetti
Tabella di inoltro
Protocollo ICMP
• rapporto errori
• segnalazioni dai router
Strato di rete
Strato di collegamento
Strato fisico

Indirizzi IPv4



Classe	Da	A
A	1.0.0.0	127.255.255.255
B	128.0.0.0	191.255.255.255
C	192.0.0.0	223.255.255.255
D (multicast)	224.0.0.0	239.255.255.255
E (riservati)	240.0.0.0	

Indirizzi IPv4

0.0.0.0	This host
00000….xxxxxxxxxx	Host nella subnet
255.255.255.255	Broadcast
127.0.0.0 – 127.255.255.255	Loopback
10.0.0.0 – 10.255.255.255	IP Privati
169.254.0.0 – 169.254.255.255	Zero conf. net. APIPA
172.16.0.0 – 172.31.255.255	IP Privati
192.168.0.0 – 192.168.255.255	IP Privati

Assegnamento Indirizzi IP

Inizialmente era effettuato da IANA (Internet Assigned Numbers Authority) (www.iana.org).

Nel 1992 sono state create le RIR (Regional Internet Registry), per gestire gli indirizzi e i nomi di dominio per ciascuna area assegnata.

Nel 1998 è stata fondata un'organizzazione no-profit denominata ICANN (Internet Corporation for Assigned Network and Numbers, www.icann.org) sotto il controllo del ministero del commercio USA per coordinare tutte le RIR del mondo.

Assegnamento Indirizzi IP

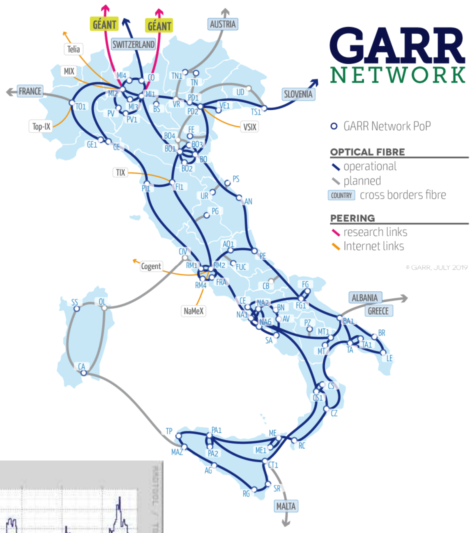
Oggi esistono cinque RIR:

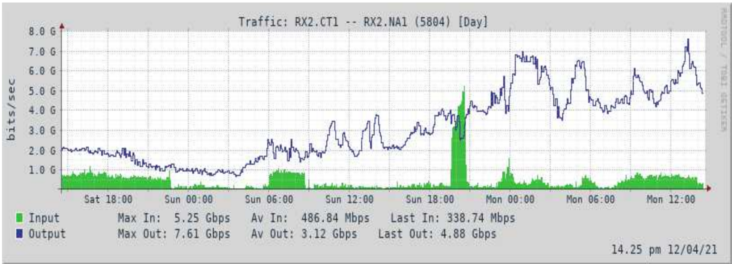
- APNIC (Asia Pacific Network Information Center): www.apnic.net.
- ARIN (American Registry for Internet Numbers): www.arin.net.
- LACNIC (Latin American and Caribbean Internet Addresses Registry): www.lacnic.net.
- RIPE NCC (Réseaux IP Européens Network Coordination Centre): www.ripe.net.
- AFRINIC (African Regional Internet Registry): www.afrinic.org.

Assegnamento Indirizzi IP

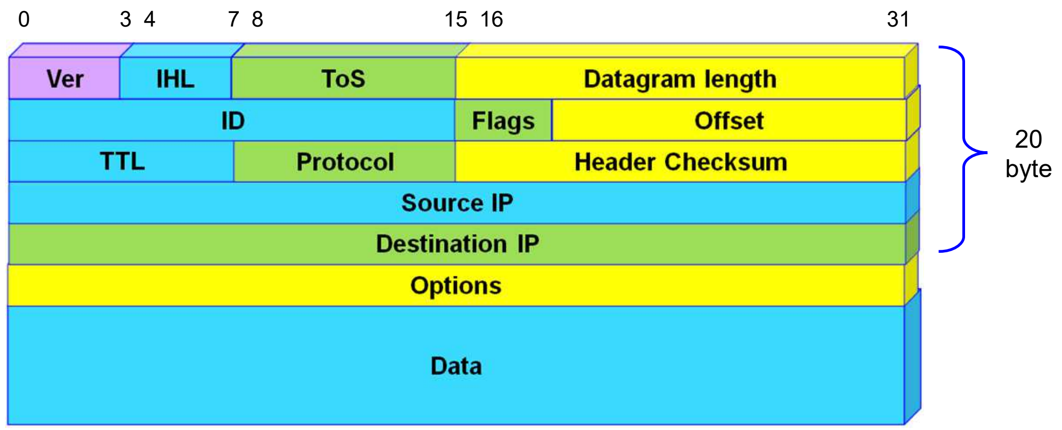
Ai RIR fanno capo i LIR (Local Internet Registries) per gestire gli indirizzi a livello locale.

GARR-LIR

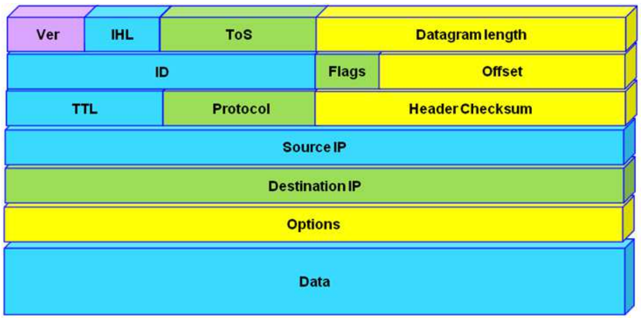




Formato Datagram IPv4



Formato Datagram IPv4



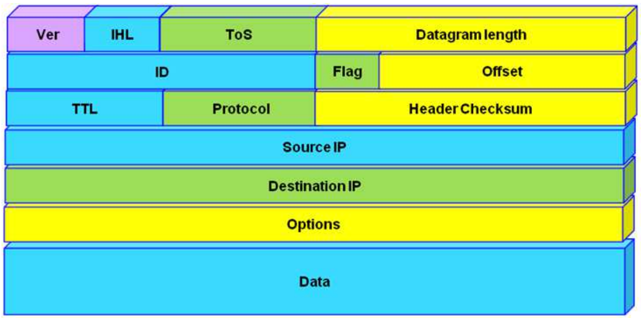
Version: 4 bit.

IHL: 4 bit - dimensione del preambolo, in gruppi da 4 byte

ToS: tipo di servizio: 3 bit di precedenza e 3 flag. Non è usato.

Datagram Length: lunghezza totale del datagram. 16 bit => 64kB

Formato Datagram IPv4



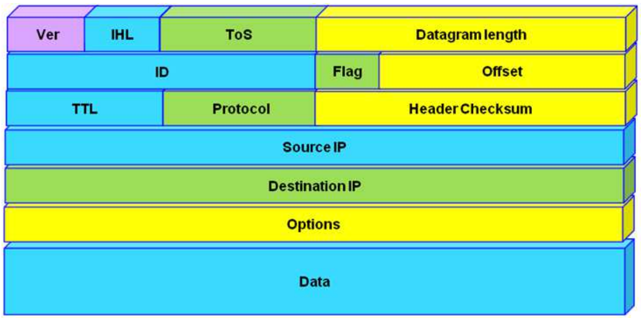
ID: identifica il datagram e i suoi frammenti.

FLAG (3 bit)

- Not used
- DF Don't fragment
- MF: More fragment

Fragment Offset: 13 bit => 8192 frammenti da almeno 8 byte

Formato Datagram IPv4



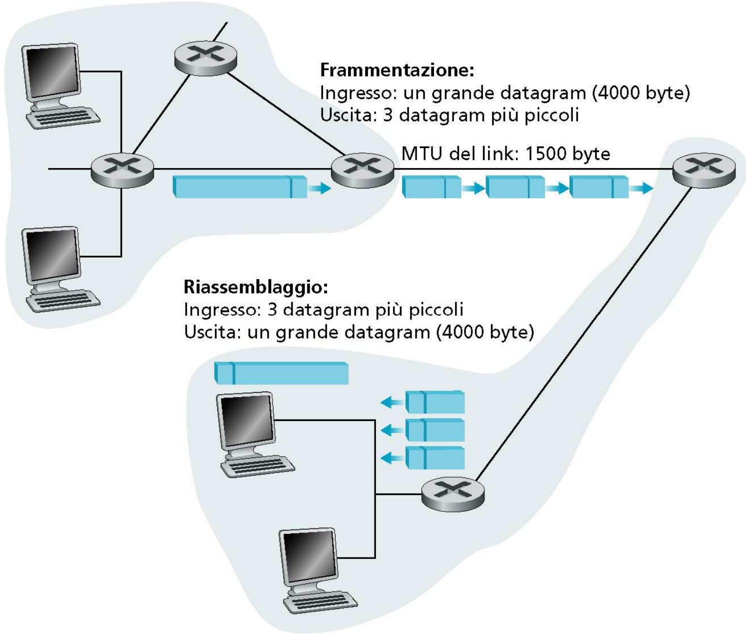
TTL: Time To Live

Protocol: specifica il protocollo di livello di trasporto

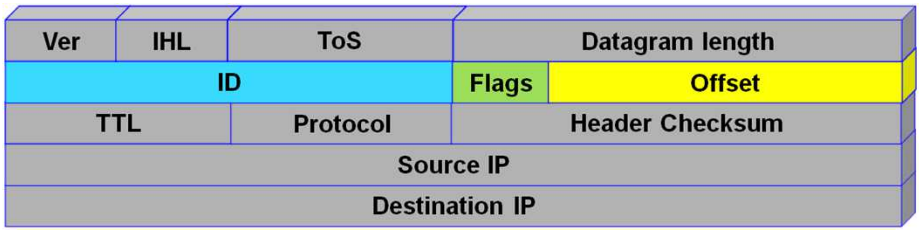
Header checksum: checksum del solo preambolo.

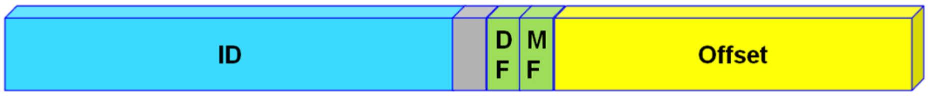
Options

Frammentazione

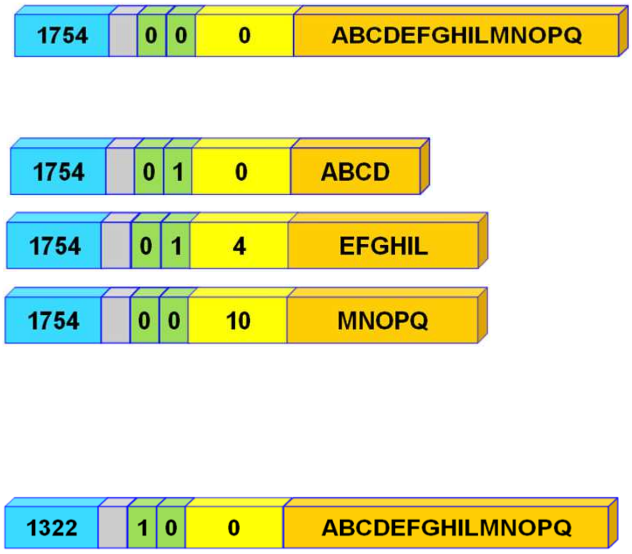


Frammentazione

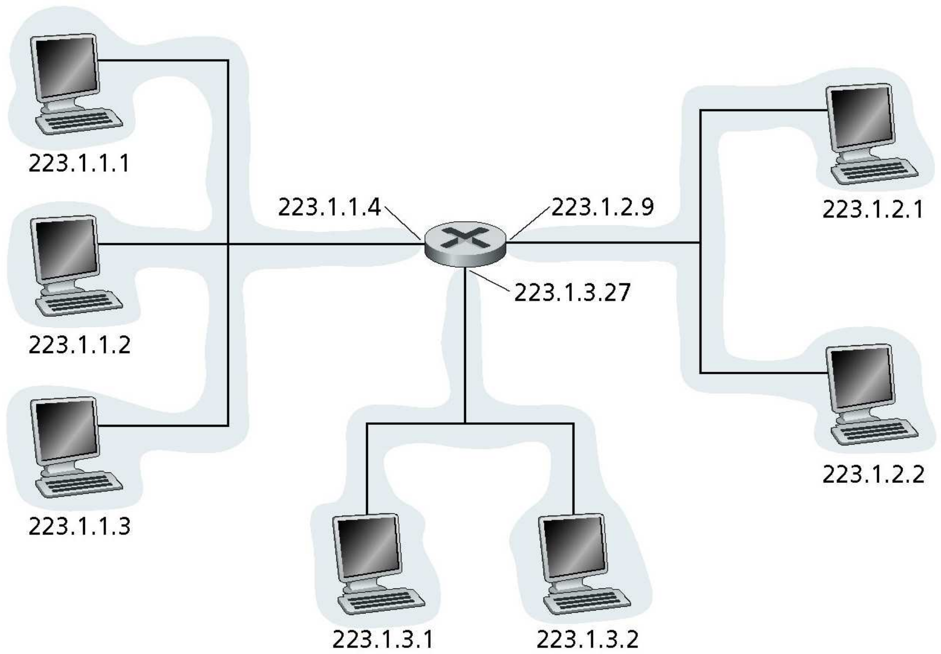




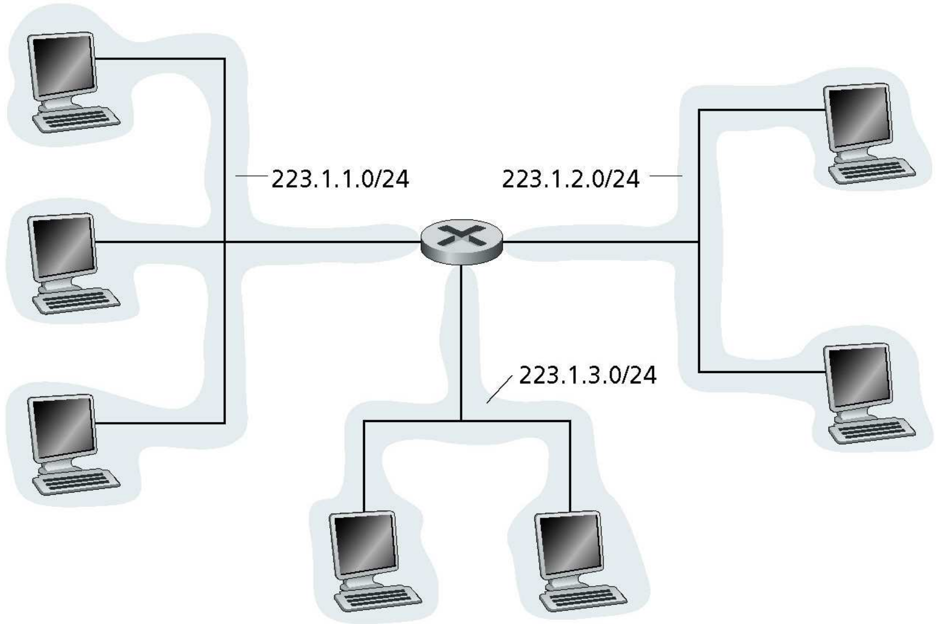
Frammentazione



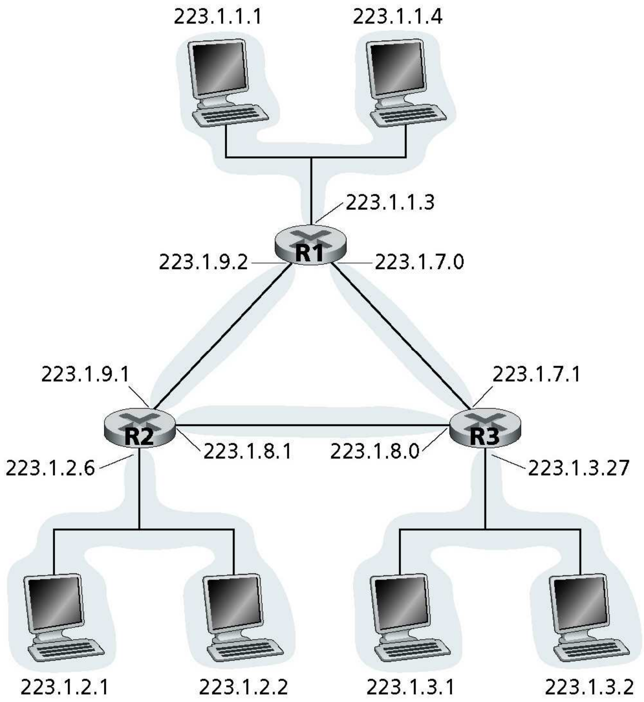
Indirizzamento IPv4



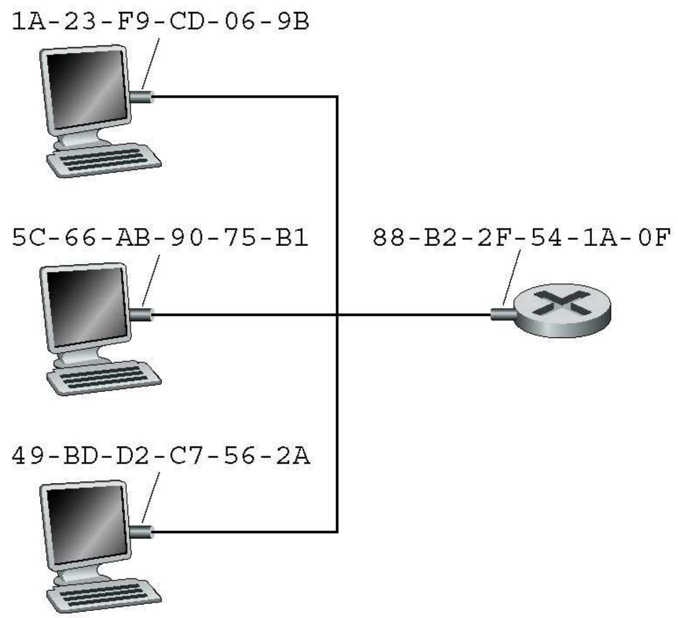
Indirizzamento IPv4



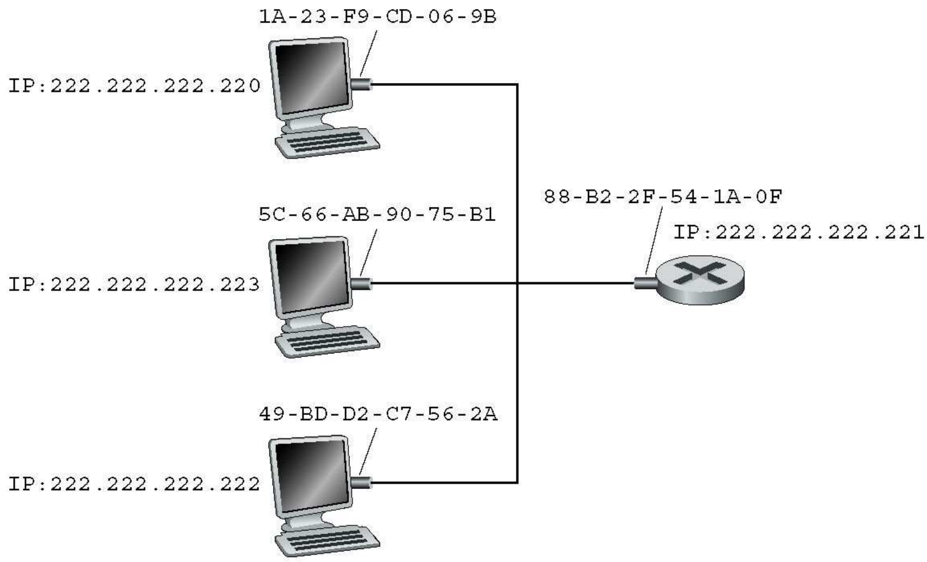
Indirizzamento IPv4



Indirizzamento intraLAN

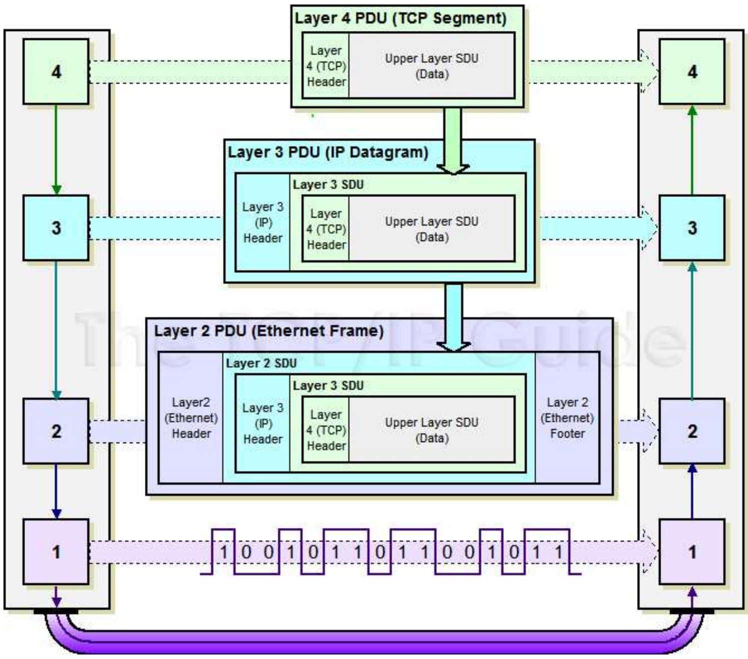


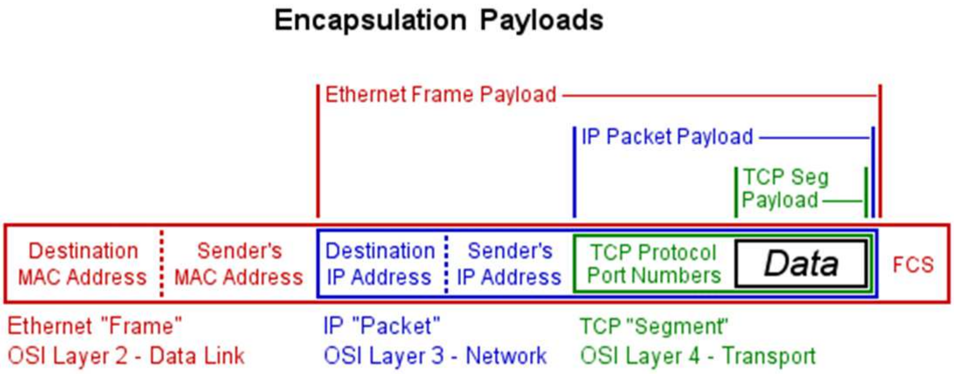
Indirizzamento intraLAN



Università di Catania

Indirizzi LAN e IPv4




Encapsulation Payloads
Ethernet Frame Payload
IP Packet Payload
TCP Seg
Payload
Destination
MAC Address
Sender's
MAC Address
Destination
IP Address
Sender's
IP Address
TCP Protocol
Port Numbers
Data
FCS
Ethernet "Frame"
OSI Layer 2 - Data Link
IP "Packet"
OSI Layer 3 - Network
TCP "Segment"
OSI Layer 4 - Transport

Indirizzi LAN e IPv4

).	Time	Source	Destination	Protocol	Length	Info
30	4.203963853	proxy03.mirror.garr.it	Toshiba-Z830.local	TCP	66	80 → 55492 [SYN, ACK]
31	4.203985642	Toshiba-Z830.local	proxy03.mirror.garr…	TCP	54	55492 → 80 [ACK] Seq=
32	4.209664430	Toshiba-Z830.local	proxy03.mirror.garr…	HTTP	802	GET / HTTP/1.1
33	4.213469694	mil04s25-in-f74.1e100.net	Toshiba-Z830.local	TCP	74	80 → 58828 [SYN, ACK]
34	4.213522703	Toshiba-Z830.local	mil04s25-in-f74.1e1…	TCP	66	58828 → 80 [ACK] Seq=
35	4.222429261	proxy03.mirror.garr.it	Toshiba-Z830.local	TCP	60	80 → 55484 [ACK] Seq=

```
Frame 32: 802 bytes on wire (6416 bits), 802 bytes captured (6416 bits) on interface 0
Ethernet II, Src: Toshiba_dc:c7:5e (e8:e0:b7:dc:c7:5e), Dst: Routerbo_13:e0:1c (4c:5e:0c:13:e0:1c)
Internet Protocol Version 4, Src: Toshiba-Z830.local (192.168.65.167), Dst: proxy03.mirror.garr.it (90.147.160.73)
Transmission Control Protocol, Src Port: 55484, Dst Port: 80, Seq: 1, Ack: 1, Len: 748
Hypertext Transfer Protocol
  ▶ GET / HTTP/1.1\r\n
    Host: mirror.garr.it\r\n
    User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:65.0) Gecko/20100101 Firefox/65.0\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8\r\n
    Accept-Language: it-IT,it;q=0.8,en-US;q=0.5,en;q=0.3\r\n
    Accept-Encoding: gzip, deflate\r\n
    Referer: https://it.search.yahoo.com/\r\n
    Connection: keep-alive\r\n
  ▶ Cookie: __utma=225464966.2039547598.1552470368.1552470368.1552470368.1; __utmb=225464966.1.10.1552470368; __utmc=225
    Upgrade-Insecure-Requests: 1\r\n
    If-Modified-Since: Sat, 10 Nov 2018 08:45:07 GMT\r\n
```

```
000  4c 5e 0c 13 e0 1c e8 e0  b7 dc c7 5e 08 00 45 00   L^······ ···^··E·
010  03 14 93 d1 40 00 40 06  a6 e6 c0 a8 41 a7 5a 93   ····@·@· ····A·Z·
020  a0 49 d8 bc 00 50 60 10  5a be ca 22 4d 02 50 18   ·I···P`· Z··"M·P·
030  00 e5 00 33 00 00 47 45  54 20 2f 20 48 54 54 50   ···3··GE T / HTTP
040  2f 31 2e 31 0d 0a 48 6f  73 74 3a 20 6d 69 72 72   /1.1··Ho st: mirr
050  6f 72 2e 67 61 72 72 2e  69 74 0d 0a 55 73 65 72   or.garr. it··User
060  2d 41 67 65 6e 74 3a 20  4d 6f 7a 69 6c 6c 61 2f   -Agent:  Mozilla/
070  35 2e 30 20 28 58 31 31  3b 20 55 62 75 6e 74 75   5.0 (X11 ; Ubuntu
080  3b 20 4c 69 6e 75 78 20  78 38 36 5f 36 34 3b 20   ; Linux  x86_64;
090  72 76 3a 36 35 2e 30 29  20 47 65 63 6b 6f 2f 32   rv:65.0)  Gecko/2
0a0  30 31 30 30 31 30 31 20  46 69 72 65 66 6f 78 2f   0100101  Firefox/
0b0  36 35 2e 30 0d 0a 41 63  63 65 70 74 3a 20 74 65   65.0··Ac cept: te
0c0  78 74 2f 68 74 6d 6c 2c  61 70 70 6c 69 63 61 74   xt/html, applicat
0d0  69 6f 6e 2f 78 68 74 6d  6c 2b 78 6d 6c 2c 61 70   ion/xhtm l+xml,ap
0e0  70 6c 69 63 61 74 69 6f  6e 2f 78 6d 6c 3b 71 3d   plicatio n/xml;q=
0f0  30 2e 39 2c 69 6d 61 67  65 2f 77 65 62 70 2c 2a   0.9,imag e/webp,*
100  2f 2a 3b 71 3d 30 2e 38  0d 0a 41 63 63 65 70 74   /*;q=0.8 ··Accept
```

Indirizzi LAN e IPv4

o.	Time	Source	Destination	Protocol	Length	Info
30	4.203963853	proxy03.mirror.garr.it	Toshiba-Z830.local	TCP	66	80 → 55492 [SYN, ACK]
31	4.203985642	Toshiba-Z830.local	proxy03.mirror.garr…	TCP	54	55492 → 80 [ACK] Seq=
32	4.209664430	Toshiba-Z830.local	proxy03.mirror.garr…	HTTP	802	GET / HTTP/1.1
33	4.213469694	mil04s25-in-f74.1e100.net	Toshiba-Z830.local	TCP	74	80 → 58828 [SYN, ACK]
34	4.213522703	Toshiba-Z830.local	mil04s25-in-f74.1e1…	TCP	66	58828 → 80 [ACK] Seq=
35	4.222429261	proxy03.mirror.garr.it	Toshiba-Z830.local	TCP	60	80 → 55484 [ACK] Seq=

```
Frame 32: 802 bytes on wire (6416 bits), 802 bytes captured (6416 bits) on interface 0
Ethernet II, Src: Toshiba_dc:c7:5e (e8:e0:b7:dc:c7:5e), Dst: Routerbo_13:e0:1c (4c:5e:0c:13:e0:1c)
Internet Protocol Version 4, Src: Toshiba-Z830.local (192.168.65.167), Dst: proxy03.mirror.garr.it (90.147.160.73)
Transmission Control Protocol, Src Port: 55484, Dst Port: 80, Seq: 1, Ack: 1, Len: 748
Hypertext Transfer Protocol
 ▸ GET / HTTP/1.1\r\n
   Host: mirror.garr.it\r\n
   User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:65.0) Gecko/20100101 Firefox/65.0\r\n
   Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8\r\n
   Accept-Language: it-IT,it;q=0.8,en-US;q=0.5,en;q=0.3\r\n
   Accept-Encoding: gzip, deflate\r\n
   Referer: https://it.search.yahoo.com/\r\n
   Connection: keep-alive\r\n
```

```
000 4c 5e 9c 13 e0 1c e8 e0 b7 de c7 5e 08 00 45 00
010 03 14 93 d1 40 00 40 06 a6 e6 c0 a8 41 a7 5a 93
020 a0 49 d8 bc 00 50 60 10 5a be ca 22 4d 02 50 18
030 00 e5 00 33 00 00 47 45 54 20 2f 20 48 54 54 50
040 2f 31 2e 31 0d 0a 48 6f 73 74 3a 20 6d 69 72 72
050 6f 72 2e 67 61 72 72 2e 69 74 9d 0a 55 73 65 72
060 2d 41 67 65 6e 74 3a 20 4d 6f 7a 69 6c 6c 61 2f
070 35 2e 30 20 28 58 31 31 3b 20 55 62 75 6e 74 75
080 3b 20 4c 69 6e 75 78 20 78 38 36 5f 36 34 3b 20
090 72 76 3a 36 35 2e 39 29 20 47 65 63 6b 6f 2f 32
0a0 30 31 30 30 31 30 31 20 46 69 72 65 66 6f 78 2f
0b0 36 35 2e 30 0d 0a 41 63 63 65 70 74 3a 20 74 65
0c0 78 74 2f 68 74 6d 6c 2c 61 70 70 6c 69 63 61 74
0d0 69 6f 6e 2f 78 68 74 6d 6c 2b 78 6d 6c 2c 61 70
0e0 70 6c 69 63 61 74 69 6f 6e 2f 78 6d 6c 3b 71 3d
0f0 39 2e 39 2c 69 6d 61 67 65 2f 77 65 62 70 2c 2a
100 2f 2a 3b 71 3d 30 2e 38 0d 0a 41 63 63 65 70 74
```

Eth mac1

Eth mac2

Eth type

IP

TCP

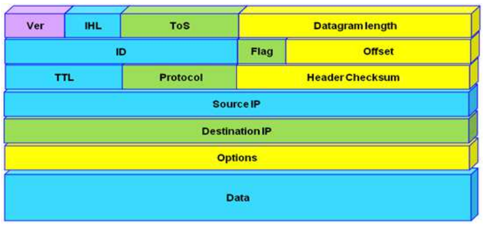
HTTP

Indirizzi LAN e IPv4

No.	Time	Source	Destination	Protocol	Length	Info
30	4.203963853	proxy03.mirror.garr.it	Toshiba-Z830.local	TCP	66	80 → 55492 [SYN, ACK]
31	4.203985642	Toshiba-Z830.local	proxy03.mirror.garr…	TCP	54	55492 → 80 [ACK] Seq=
32	4.209664430	Toshiba-Z830.local	proxy03.mirror.garr…	HTTP	802	GET / HTTP/1.1
33	4.213469694	mil04s25-in-f74.1e100.net	Toshiba-Z830.local	TCP	74	80 → 58828 [SYN, ACK]
34	4.213522703	Toshiba-Z830.local	mil04s25-in-f74.1e1…	TCP	66	58828 → 80 [ACK] Seq=
35	4.222429261	proxy03.mirror.garr.it	Toshiba-Z830.local	TCP	60	80 → 55484 [ACK] Seq=

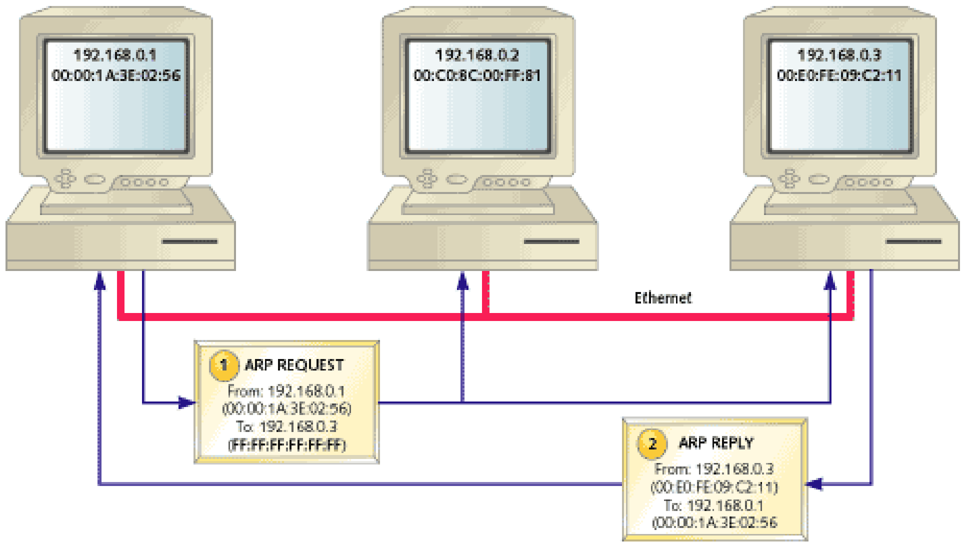
```
Frame 32: 802 bytes on wire (6416 bits), 802 bytes captured (6416 bits) on interface 0
Ethernet II, Src: Toshiba_dc:c7:5e (e8:e0:b7:dc:c7:5e), Dst: Routerbo_13:e0:1c (4c:5e:0c:13:e0:1c)
Internet Protocol Version 4, Src: Toshiba-Z830.local (192.168.65.167), Dst: proxy03.mirror.garr.it (90.147.160.73)
Transmission Control Protocol, Src Port: 55484, Dst Port: 80, Seq: 1, Ack: 1, Len: 748
```

```
45 00 03 14
93 d1 40 00
40 06 a6 e6
c0 a8 41 a7
5a 93 a0 49
d8 bc 00 50
60 10 5a be
ca 22 4d 02
50 18 00 e5
00 33 00 00
```




```
Ver 04    IHL 05    ToS 03    DL 0314 (788 = 802-6-6-2)
ID 93D1   Flag + Offset 010 0000000000000
TTL 40 (64)      Protocol 06 (TCP)      HC A6E6
Source IP    C0 (192) A8 (168) 41 (065) A7 (167)
Dest   IP    5A (090) 93 (147) A0 (160) 49 (073)
```

ARP request



192.168.0.1
00:00:1A:3E:02:56
192.168.0.2
00:C0:8C:00:FF:81
192.168.0.3
00:E0:FE:09:C2:11
Ethernet
1 ARP REQUEST
From: 192.168.0.1
(00:00:1A:3E:02:56)
To: 192.168.0.3
(FF:FF:FF:FF:FF:FF)
2 ARP REPLY
From: 192.168.0.3
(00:E0:FE:09:C2:11)
To: 192.168.0.1
(00:00:1A:3E:02:56)

ARP table

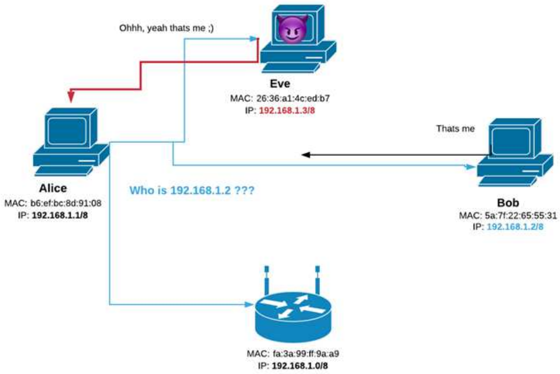
Indirizzo IP	Indirizzo LAN	TTL
222.222.222.221	88-B2-2F-54-1A-0F	13:45:00
222.222.222.223	5C-66-AB-90-75-B1	13:52:00

Vulnerabilità

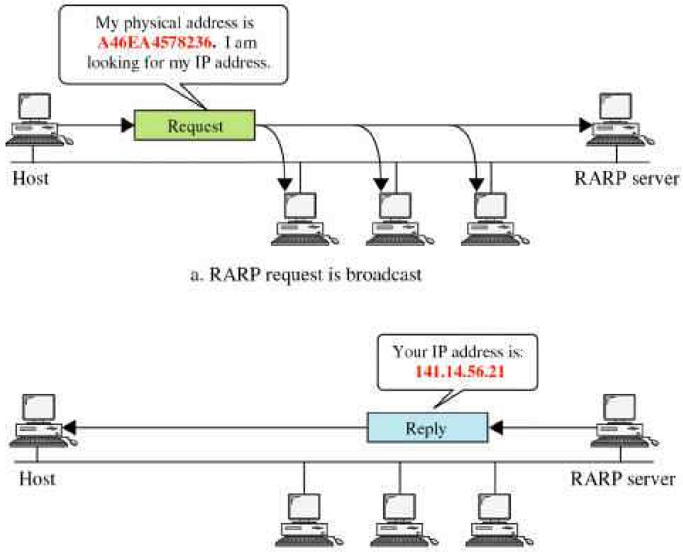
- ARP non prevede autenticazione.
- ARP è un protocollo senza stato: una ARP Reply può essere mandata senza una precedente ARP Request.
- Un host che riceve una pacchetto ARP deve aggiornare la sua ARP cache.

Conseguenze

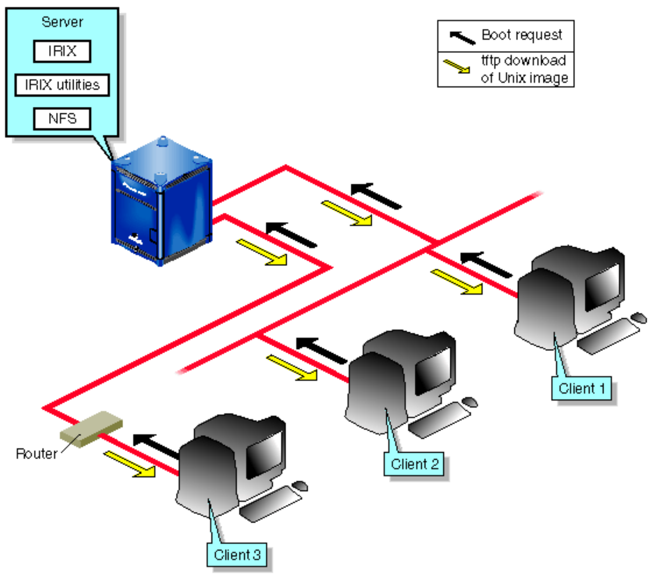
È possibile dirottare il traffico IP.



RARP

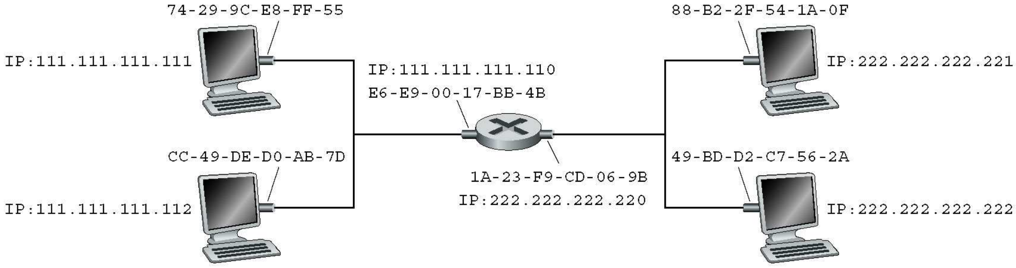

My physical address is A46EA4578236. I am looking for my IP address.
Request
Host
RARP server
a. RARP request is broadcast
Your IP address is: 141.14.56.21
Reply
Host
RARP server

BOOTP

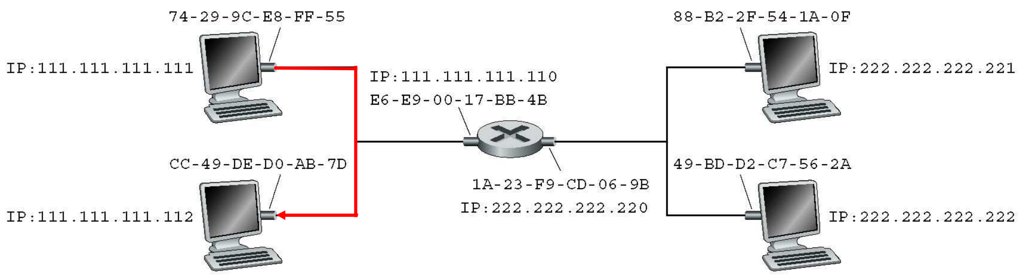


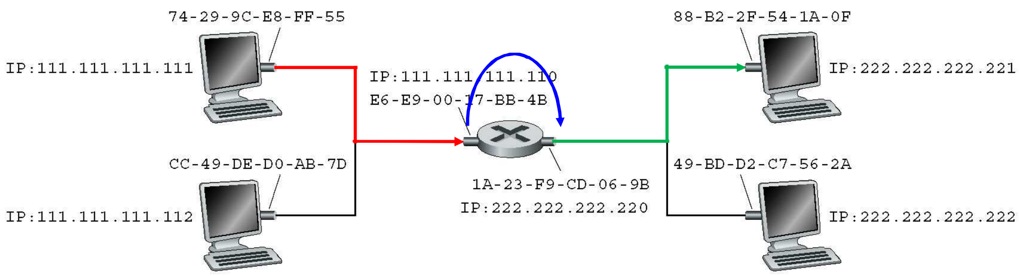
Server
IRIX
IRIX utilities
NFS
Boot request
tftp download of Unix image
Router
Client 1
Client 2
Client 3

LAN interconnesse



LAN interconnesse





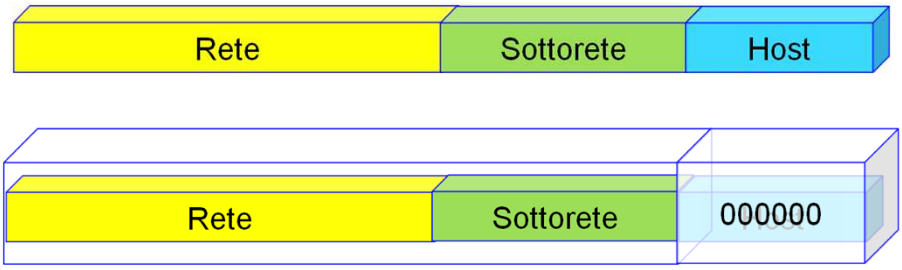
Le maschere di sottorete

CIDR (Classless Inter-Domain Routing)

VLSM (Variable Length Subnet Masking)

a.b.c.d/x

Le maschere di sottorete

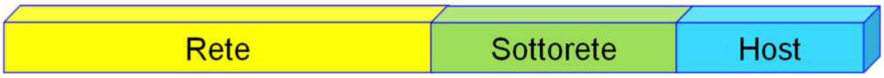



```
10010111.01100001.11111100.01000010    151.097.252.066
11111111.11111111.11111110.00000000    255.255.254.000
```

AND

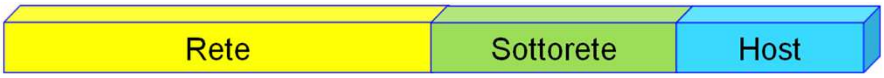
```
10010111.01100001.11111100.00000000    151.097.252.000
```

Le maschere di sottorete



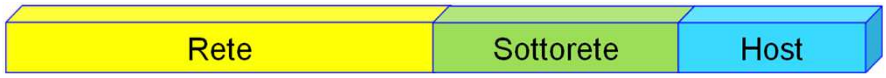

```
151.097.252.000/23
10010111.01100001.11111100.00000000

10010111.01100001.11111100.00000001 151.097.252.001
10010111.01100001.11111100.00000010 151.097.252.002
…
10010111.01100001.11111101.11111110 151.097.253.254


151.97.252.0/23    Tutta la sottorete

151.97.253.255     Broadcast nella sottorete
```

Le maschere di sottorete




```
191.15.12.0/30
10111111.00001111.00001100.00000000    network
10111111.00001111.00001100.00000001
10111111.00001111.00001100.00000010
10111111.00001111.00001100.00000011    broadcast

191.15.12.24/30
10111111.00001111.00001100.00011000    network

191.15.12.25
191.15.12.26
191.15.12.27    10111111.00001111.00001100.00011011
```

Le maschere di sottorete



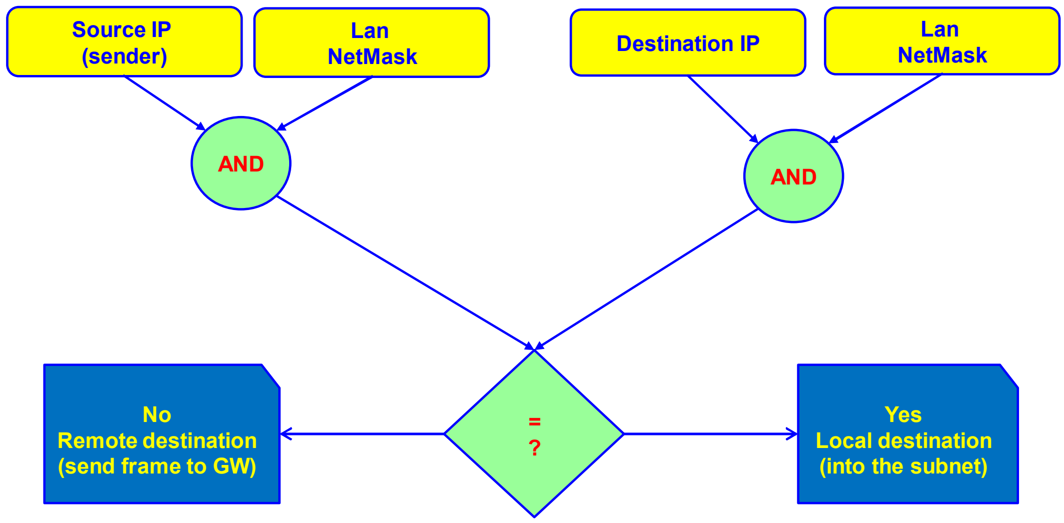
/31

A 31-bit subnet mask is often used for an interface that is the endpoint of a point-to-point network. The use of 31-bit subnet masks for IPv4 point-to-point links is described in RFC 3021.

/32

A 32-bit subnet mask defines a network with only one IP address.

Netmask



Source IP
(sender)
Lan
NetMask
Destination IP
Lan
NetMask
AND
AND
=
?
No
Remote destination
(send frame to GW)
Yes
Local destination
(into the subnet)

Università di Catania

Tabella host

Tabella di rilancio in A

Rete di destinazione	Router successivo	Numero salti
223.1.1.0/24		1
223.1.2.0/24	223.1.1.4	2
223.1.3.0/24	223.1.1.4	2

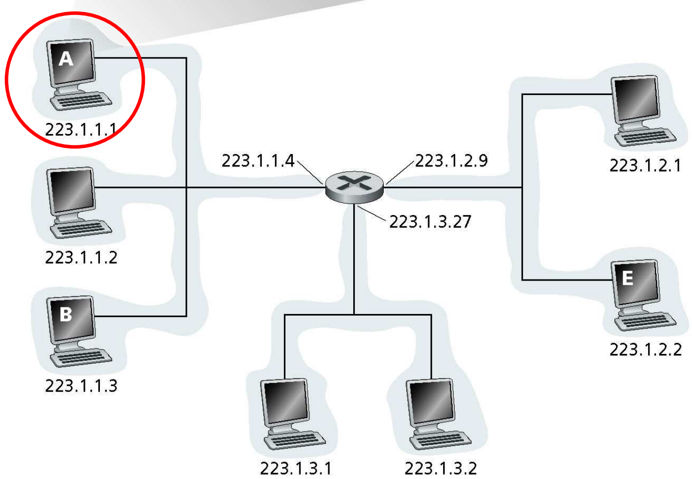
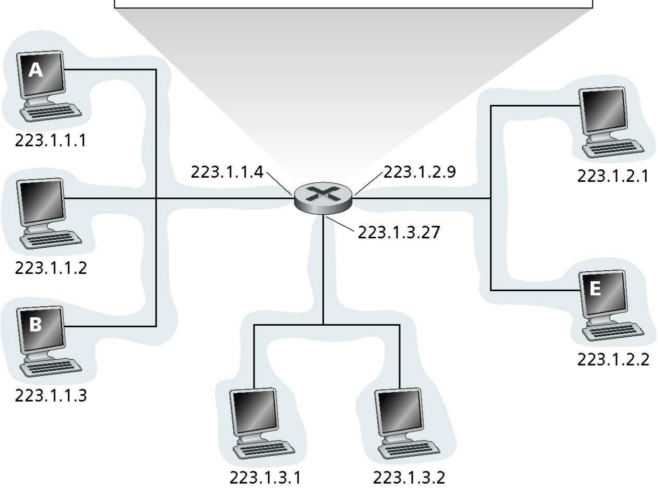


Tabella di routing

Rete di destinaz.	Router success.	N. hop	Interfaccia
223.1.1.0/24	—	1	223.1.1.4
223.1.2.0/24	—	1	223.1.2.9
223.1.3.0/24	—	1	223.1.3.27

Tabella di rilancio nel router



Example: routing table

```
Scheda Ethernet Ethernet:

   Suffisso DNS specifico per connessione: dmi.unict.it
   Indirizzo IPv6 locale rispetto al collegamento : fe80::2430:ee60:719:1e6c%6
   Indirizzo IPv4. . . . . . . . . . . . : 192.168.65.156
   Subnet mask . . . . . . . . . . . . . : 255.255.255.0
   Gateway predefinito . . . . . . . . . : 192.168.65.1

Scheda Ethernet Ethernet 3:

   Suffisso DNS specifico per connessione:
   Indirizzo IPv6 locale rispetto al collegamento : fe80::c5f:4e6:d1c1:c3d2%39
   Indirizzo IPv4. . . . . . . . . . . . : 192.168.56.1
   Subnet mask . . . . . . . . . . . . . : 255.255.255.0
   Gateway predefinito . . . . . . . . . :


Scheda LAN wireless Wi-Fi:

   Suffisso DNS specifico per connessione: wluct.unict.it
   Indirizzo IPv6 locale rispetto al collegamento : fe80::b176:bbcd:561:97ae%54
   Indirizzo IPv4. . . . . . . . . . . . : 151.97.62.134
   Subnet mask . . . . . . . . . . . . . : 255.255.248.0
   Gateway predefinito . . . . . . . . . : 151.97.56.1

   Stato supporto. . . . . . . . . . . . : Supporto disconnesso
   Suffisso DNS specifico per connessione: wluct.unict.it
```

Example: routing table

```
C:\> route print
===========================================================================
Elenco interfacce
 55...........................Wintun Userspace Tunnel
  6...00 1c c2 4b 8a b3 ......Realtek USB GbE Family Controller
 39...0a 00 27 00 00 27 ......VirtualBox Host-Only Ethernet Adapter
 18...00 ff 51 02 c2 ab ......TAP-Windows Adapter V9
 35...38 fc 98 b0 b3 56 ......Microsoft Wi-Fi Direct Virtual Adapter
  4...3a fc 98 b0 b3 55 ......Microsoft Wi-Fi Direct Virtual Adapter #2
 54...38 fc 98 b0 b3 55 ......Intel(R) Wi-Fi 6 AX200 160MHz
 17...38 fc 98 b0 b3 59 ......Bluetooth Device (Personal Area Network)
  1...........................Software Loopback Interface 1
 28...94 8f 85 51 5b 53 ......Generic Mobile Broadband Adapter
===========================================================================
```

Example: routing table

```
IPv4 Tabella route
===========================================================================
Route attive:
     Indirizzo rete                 Mask          Gateway     Interfaccia      
Metrica
          0.0.0.0          0.0.0.0      151.97.56.1    151.97.62.134     45
        127.0.0.0        255.0.0.0         On-link         127.0.0.1    331
        127.0.0.1  255.255.255.255         On-link         127.0.0.1    331
  127.255.255.255  255.255.255.255         On-link         127.0.0.1    331
      151.97.56.0    255.255.248.0         On-link     151.97.62.134    301
    151.97.62.134  255.255.255.255         On-link     151.97.62.134    301
    151.97.63.255  255.255.255.255         On-link     151.97.62.134    301
     192.168.56.0    255.255.255.0         On-link      192.168.56.1    281
     192.168.56.1  255.255.255.255         On-link      192.168.56.1    281
   192.168.56.255  255.255.255.255         On-link      192.168.56.1    281
        224.0.0.0        240.0.0.0         On-link         127.0.0.1    331
        224.0.0.0        240.0.0.0         On-link      192.168.56.1    281
        224.0.0.0        240.0.0.0         On-link     151.97.62.134    301
  255.255.255.255  255.255.255.255         On-link         127.0.0.1    331
  255.255.255.255  255.255.255.255         On-link      192.168.56.1    281
  255.255.255.255  255.255.255.255         On-link     151.97.62.134    301
===========================================================================
Route permanenti:
  Nessuna
```

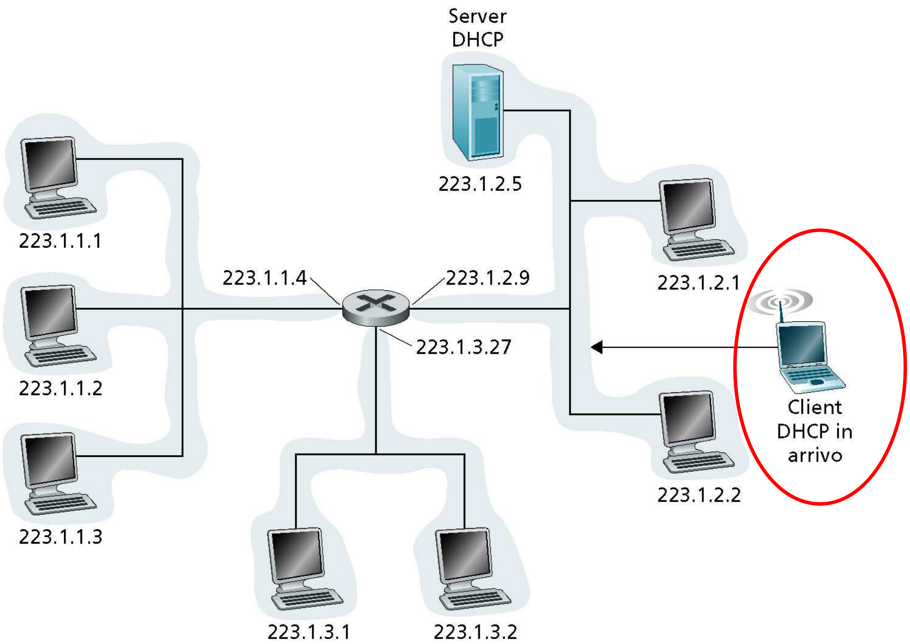
Example: routing table

```
IPv4 Tabella route
===========================================================================
Route attive:
     Indirizzo rete             Mask          Gateway     Interfaccia
Metrica
          0.0.0.0          0.0.0.0      192.168.65.1   192.168.65.156     25
        127.0.0.0        255.0.0.0          On-link         127.0.0.1    331
        127.0.0.1  255.255.255.255          On-link         127.0.0.1    331
  127.255.255.255  255.255.255.255          On-link         127.0.0.1    331
     192.168.56.0    255.255.255.0          On-link      192.168.56.1    281
     192.168.56.1  255.255.255.255          On-link      192.168.56.1    281
   192.168.56.255  255.255.255.255          On-link      192.168.56.1    281
     192.168.65.0    255.255.255.0          On-link    192.168.65.156    281
   192.168.65.156  255.255.255.255          On-link    192.168.65.156    281
   192.168.65.255  255.255.255.255          On-link    192.168.65.156    281
        224.0.0.0        240.0.0.0          On-link         127.0.0.1    331
        224.0.0.0        240.0.0.0          On-link      192.168.56.1    281
        224.0.0.0        240.0.0.0          On-link    192.168.65.156    281
  255.255.255.255  255.255.255.255          On-link         127.0.0.1    331
  255.255.255.255  255.255.255.255          On-link      192.168.56.1    281
  255.255.255.255  255.255.255.255          On-link    192.168.65.156    281
===========================================================================
Route permanenti:
  Nessuna
```

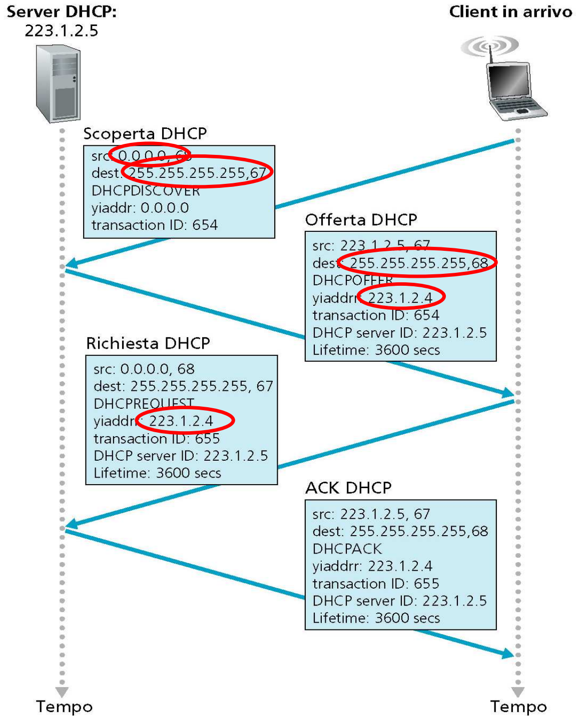
Example: routing table

```
IPv6 Tabella route
===========================================================================
Route attive:
 Interf Metrica Rete Destinazione      Gateway
  1    331 ::1/128                  On-link
 39    281 fe80::/64                On-link
  6    281 fe80::/64                On-link
 39    281 fe80::c5f:4e6:d1c1:c3d2/128
                                    On-link
  6    281 fe80::2430:ee60:719:1e6c/128
                                    On-link
  1    331 ff00::/8                 On-link
 39    281 ff00::/8                 On-link
  6    281 ff00::/8                 On-link
===========================================================================
Route permanenti:
  Nessuna
```

Assegnamento degli indirizzi: DHCPv4



DHCPv4


Server DHCP:
223.1.2.5
Client in arrivo
Scoperta DHCP
src: 0.0.0.0, 68
dest: 255.255.255.255,67
DHCPDISCOVER
yiaddr: 0.0.0.0
transaction ID: 654
Offerta DHCP
src: 223.1.2.5, 67
dest: 255.255.255.255,68
DHCPOFFER
yiaddrr: 223.1.2.4
transaction ID: 654
DHCP server ID: 223.1.2.5
Lifetime: 3600 secs
Richiesta DHCP
src: 0.0.0.0, 68
dest: 255.255.255.255, 67
DHCPREQUEST
yiaddrr: 223.1.2.4
transaction ID: 655
DHCP server ID: 223.1.2.5
Lifetime: 3600 secs
ACK DHCP
src: 223.1.2.5, 67
dest: 255.255.255.255,68
DHCPACK
yiaddrr: 223.1.2.4
transaction ID: 655
DHCP server ID: 223.1.2.5
Lifetime: 3600 secs
Tempo
Tempo

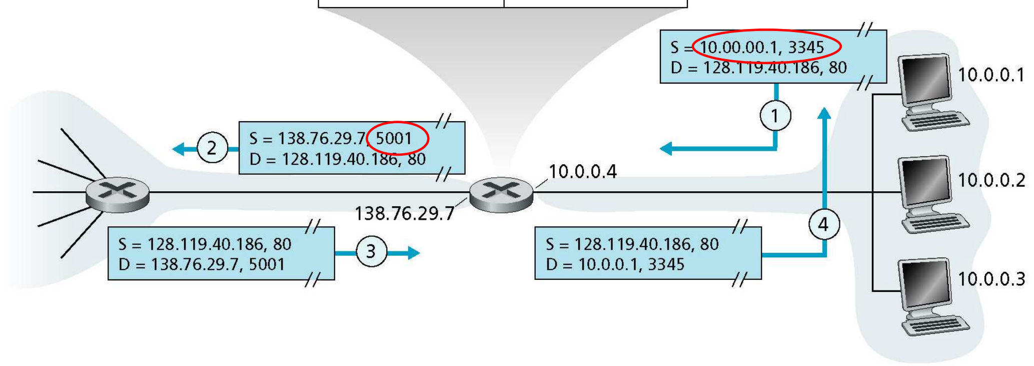
DHCPv4

Offerta DHCP

src: 223.1.2.5, 67
dest: 255.255.255.255,68
DHCPOFFER
yiaddrr: 223.1.2.4
transaction ID: 654
DHCP server ID: 223.1.2.5
Lifetime: 3600 secs

NAT

Tabella di traduzione NAT	
Lato WAN	Lato LAN
138.76.29.7, 5001	10.00. 00.1, 3345
. . .	. . .



IPv6

"If Y2K was technology's ticking time bomb, then IPv6 is a slow, deadly gas leak."

IPv6

Limiti di IPv4

- Esaurimento dello spazio degli indirizzi
- Scalabilità del routing
- Nuovi servizi

Famous Last Words

"I think there is a world market for maybe five computers"
– Thomas Watson (IBM), 1943

"640K ought to be enough for anybody"
– Bill Gates (MS), 1981

"32 bits should be enough address space for the Internet"
– Vint Cerf (ARPA), 1977

IPv4 Address Allocation

IPv4 adresses Available in October 17, 2010 according to the RIPE

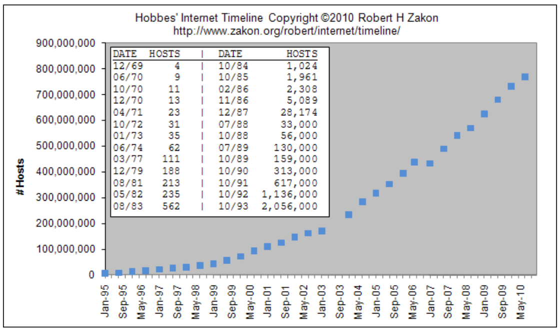
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

xx	Used
xx	Available
xx	Not useable

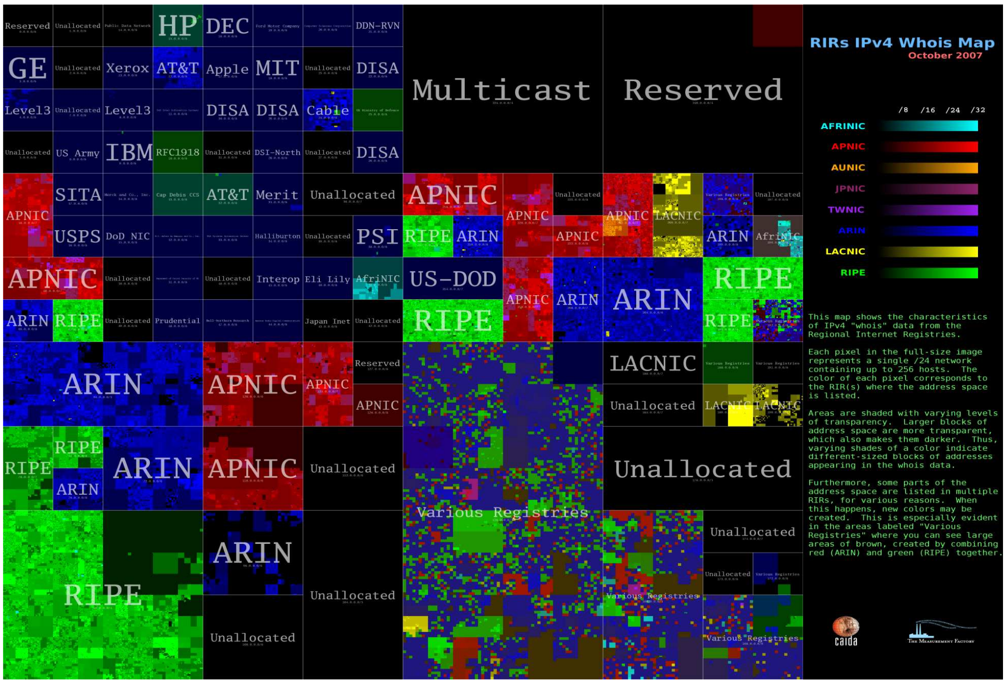
xx	Allocated in Jan. 2010	xx	Allocated in May 2010
xx	Allocated in Feb. 2010	xx	Allocated in June 2010
xx	Allocated in Apr. 2010	xx	Allocated in Aug. 2010

xx	Allocated in Oct. 2010

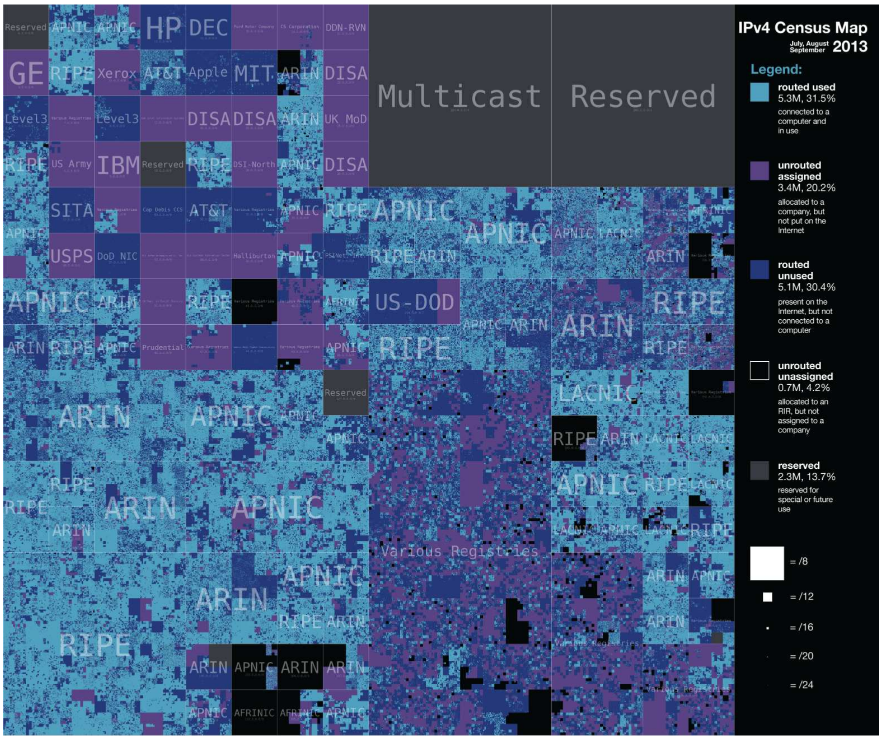
Total	256
Free	12
Percent	4,69%



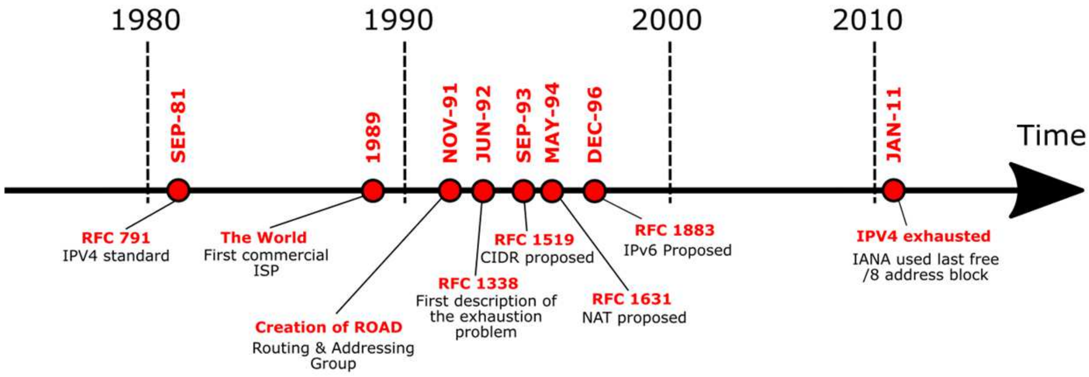
IPv4 Address Allocation



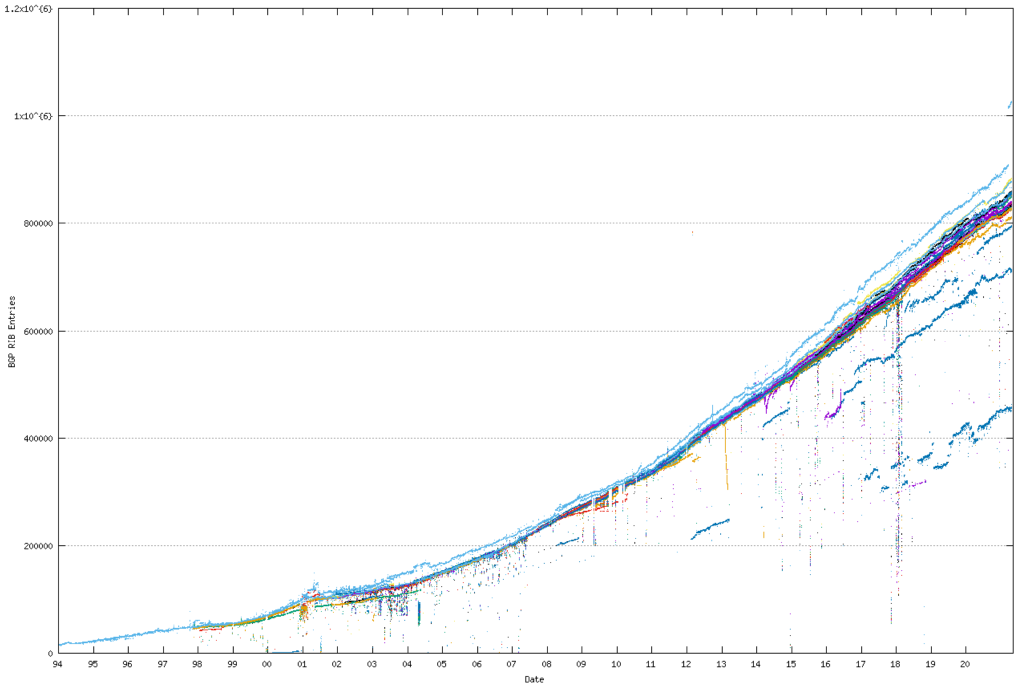
IPv4 Address Allocation



IPv4 Address Allocation



IPv4 routing

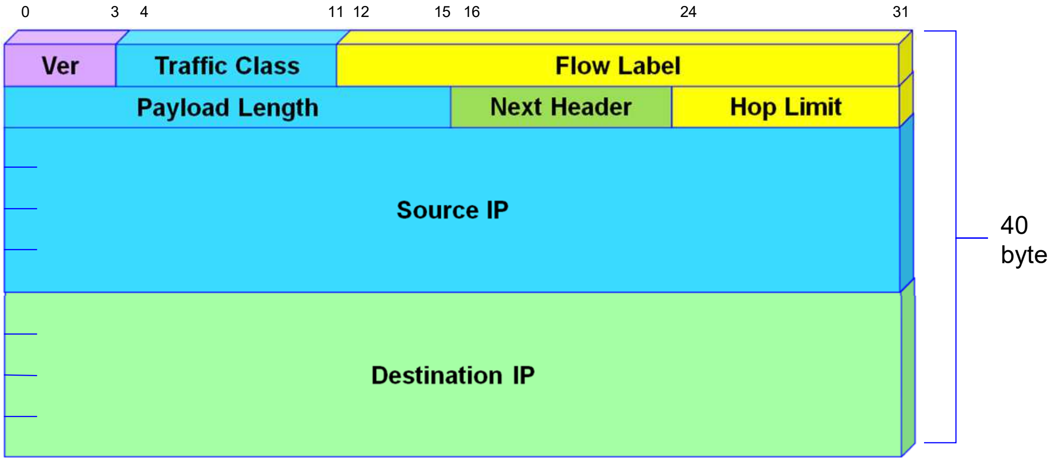


IPv6

Nuovi Servizi richiesti :

- Sicurezza
- Autoconfigurazione (Plug & Play)
- Gestione della Qualità del Servizio (QoS)
- Indirizzamento Multicast
- Indirizzamento host mobili

IPv6

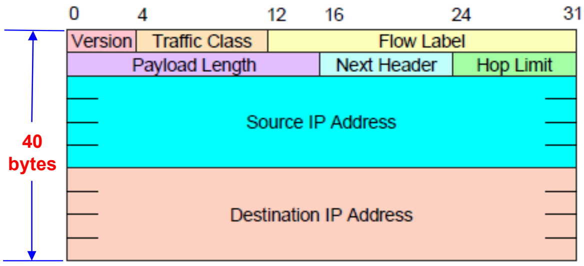


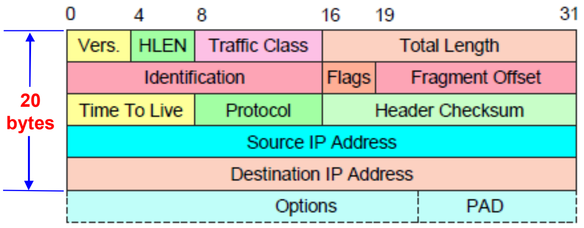
2^{128} = 340282366920938463463374607431768211456 ≈ 3×10^{38}
2^{32} = 4294967296 ≈ 4×10^{9}

Sup. terrestre ≈ 5.1×10^{14} m^2 ⇒ 6.6×10^{23} indirizzi IPv6 per m^2
Stelle nell'universo (osservabile) ≈ 10^{24} ⇒ 10^{14} indirizzi per stella

Confronto delle intestazioni

IPv6





IPv4

Rimossi
- ID, Flags, Offset
- ToS, HLEN
- Header Checksum

Cambiati
- Total length ⇨ payload
- Protocol ⇨ next header
- TTL ⇨ hop limit

Aggiunti
- Traffic class
- Flow label

Espansi
- address: 32 ⇨ 128 bits

Indirizzi IPv6

0000 0000	Riservati (includono IPv4)
0000 001	Indirizzi OSI NSAP (deprecated)
0000 010	Indirizzi Novell IPX
001	Indirizzi unicast globali
010	Indirizzi per service providers
100	Indirizzi geografici
1111 1110 10	Indirizzi unicast link-local
1111 1110 11	Indirizzi unicast site-link (deprecated)
1111 1111	Multicast

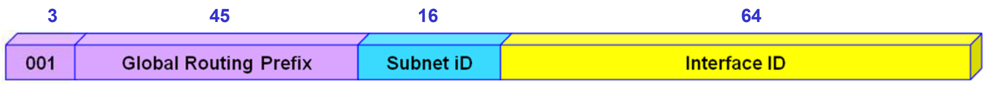
Indirizzi IPv6

```
8000:0000:0000:0000:0123:4567:89AB:CDEF

8000::123:4567:89AB:CDEF


2001:760:4804:0:0:0:0:105
2001:760:4804::/48
```

`::151.97.1.1`	indirizzo IPv4 mappato su IPv6
`::1`	loopback
`::`	non specificato

Indirizzi IPv6

Global Unicast addresses



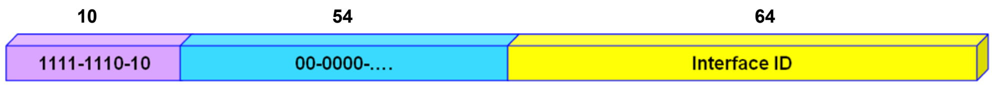
2 001:760:4804: 0: 0:0:0:105

Global Routing Prefix: identifica il sito.

Subnet ID: viene gestito all'interno del sito

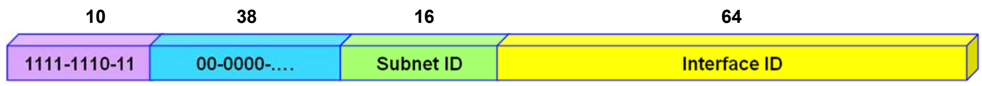
Indirizzi IPv6

Indirizzi locali: **`FE80::/64`** **`(1111-1110-1000-0000)`**



Si possono usare solo tra nodi dello stesso link

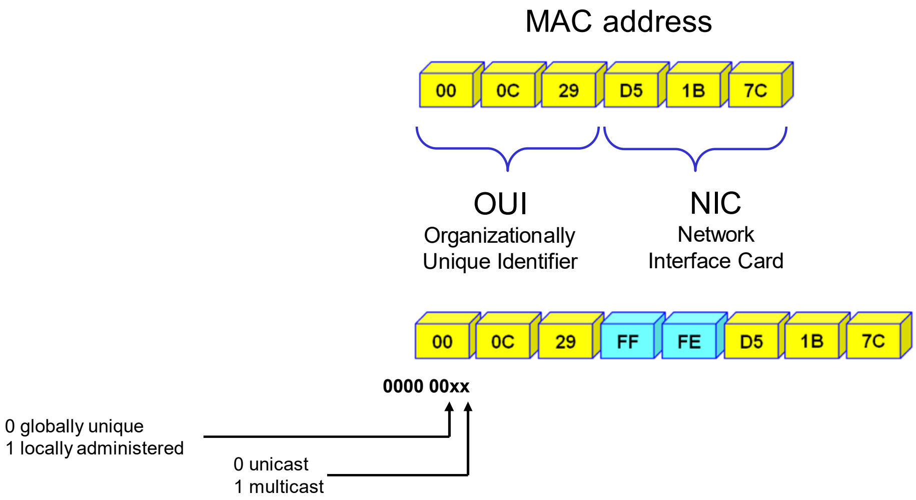
Indirizzi di sito: **`FEC0::/48`** **`(1111-1110-1100-0000)`**



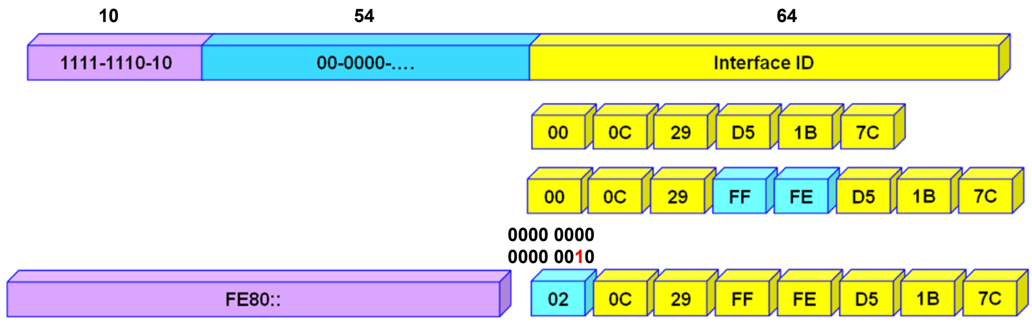
Si possono usare solo tra nodi di uno stesso sito. Sono sconsigliati da RFC 3879

Entrambi non possono essere usati su Internet.

Indirizzi IPv6 EUI-64 (RFC 2373)



Indirizzi IPv6 EUI-64 (RFC 2373)




```
[root@LabReti ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0C:29:D5:1B:7C
          inet addr:192.168.19.138  Bcast:192.168.19.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fed5:1b7c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:11 errors:0 dropped:0 overruns:0 frame:0
          TX packets:13 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1421 (1.3 KiB)  TX bytes:1382 (1.3 KiB)
          Interrupt:19 Base address:0x2000
```

IPv6 local link address non EUI-64

Windows OS non usa EUI-64 L'indirizzo IPv6 è calcolato partendo da un numero casuale. Manca il gruppo **FFFE** nella parte centrale.

Windows PowerShell

```
PS C:\Users\sr> Get-NetIPConfiguration | Select-Object IPv6LinkLocalAddress

IPv6LinkLocalAddress
--------------------
{fe80::d5d4:6f5f:f278:f690%15}
{fe80::1c47:c345:2175:cba8%29}
{fe80::b81e:9f55:9b3c:a407%30}
{fe80::19b4:e575:a36f:3593%27}
{fe80::8dc7:92:7a6f:e9f3%13}
{fe80::35f9:7e55:522c:edf1%16}
{fe80::2cdf:9c5f:ffb6:2b68%5}

PS C:\Users\sr>
```

Windows PowerShell

```
PS C:\Users\sr> Get-NetAdapter

Name                      InterfaceDescription                    ifIndex Status       MacAddress             LinkSpeed
----                      --------------------                    ------- ------       ----------             ---------
VMware Network Adapte...8 VMware Virtual Ethernet Adapter for ...      30 Up           00-50-56-C0-00-08       100 Mbps
Ethernet 2                Realtek USB GbE Family Controller #2         29 Up           00-E0-4C-36-01-78         1 Gbps
Connessione di rete B...2 Bluetooth Device (Personal Area Ne...#2      27 Disconnected E4-F8-9C-F8-3B-CA         3 Mbps
Connessione alla rete ... TAP-Windows Adapter V9                       16 Disconnected 00-FF-5B-71-7C-E2         1 Gbps
VMware Network Adapte...1 VMware Virtual Ethernet Adapter for ...      15 Up           00-50-56-C0-00-01       100 Mbps
Wi-Fi                     Intel(R) Dual Band Wireless-AC 3160          13 Disconnected E4-F8-9C-F8-3B-C6         0 bps
Ethernet                  Realtek PCIe FE Family Controller             5 Disconnected 20-47-47-C4-EF-67         0 bps

PS C:\Users\sr>
```

Università di Catania

IPv6 addresses

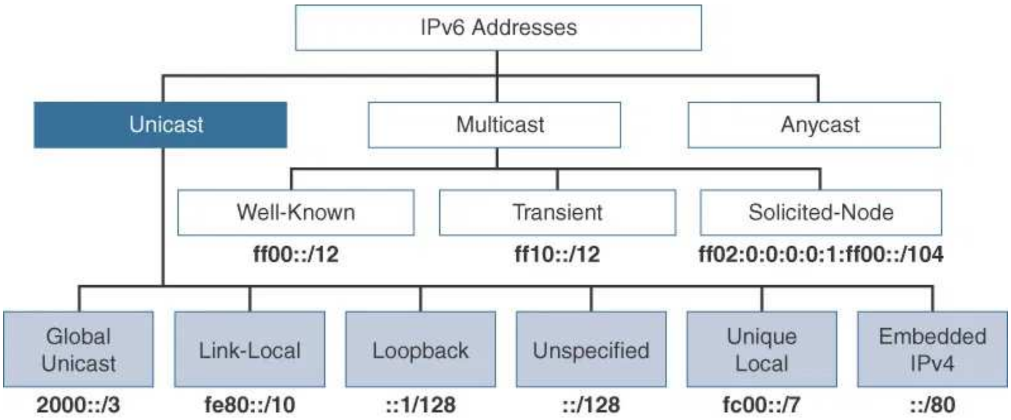
Indirizzi Unicast (global) **2xxx: .... 3xxx : ....**

Indirizzi Multicast **FFxx: ......**

Multicast Address	Scope	Group Within the Scope
FF01:0:0:0:0:0:0:1	Node-local	All-nodes address
FF01:0:0:0:0:0:0:2	Node-local	All-routers address
FF02:0:0:0:0:0:0:1	Link-local	All-nodes address
FF02:0:0:0:0:0:0:2	Link-local	All-routers address
FF02:0:0:0:0:0:0:5	Link-local	OSPF IGP
FF02:0:0:0:0:0:0:6	Link-local	OSPF IGP designated routers
FF02:0:0:0:0:0:0:D	Link-local	All PIM routers
FF02:0:0:0:0:0:0:16	Link-local	All MLDv2-capable routers
FF02:0:0:0:0:0:1:2	Link-local	All DHCP agents
FF02:0:0:0:0:1:FFXX:XX XX	Link-local	Solicited-node address
FF05:0:0:0:0:0:0:2	Site-local	All-routers address
FF05:0:0:0:0:0:1:3	Site-local	All DHCP servers

Indirizzi Anycast

IPv6 addresses



IPv6 over Ethernet

IPv6 Ethernet encapsulation (RFC 2464)

I pacchetti IPv6 sono incapsulati in frame Ethernet esattamente come per i pacchetti IPv4: cambia solo il campo Ethertype (86DD invece che 0800).

Neighbour discovery (NDP) (RFC 4861)

NDP fornisce i seguenti servizi:

- Router discovery
- Prefix discovery
- Parameter discovery
- Address Autoconfiguration
- Address resolution
- Next-hop determination
- Neighbor unreachability detection
- Duplicate address detection
- Redirect messages

Neighbour Discovery Protocol

- Address Resolution Protocol (ARP),
- Internet Control Message Protocol (ICMP) router discovery,
- ICMP Redirect message

ARP non fa parte di ICMPv4. Aver incorporato il meccanismo di risoluzione degli indirizzi mac in ICMPv6, rende il tutto più indipendente dal tipo di mezzo fisico e consente l'utilizzo di meccanismi di sicurezza

Neighbour Discovery Protocol

NDP prevede 5 messaggi:

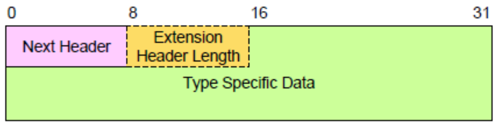
- Router Solicitation (ICMPv6 type 133)
- Router Advertisement (ICMPv6 type 134)
- Neighbor Solicitation (ICMPv6 type 135)
- Neighbor Advertisement (ICMPv6 type 136)
- Redirect (ICMPv6 type 137)

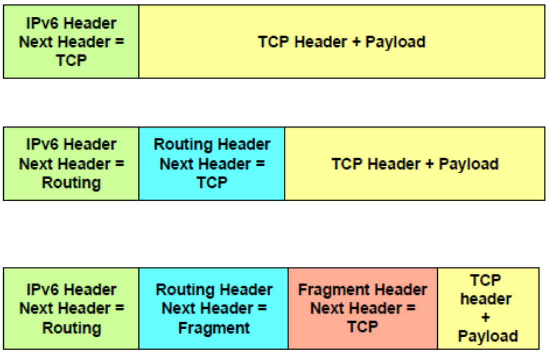
Classi di Traffico IPv6

000 -111 = time insensitive (could be discarded)
1000 -1111 = priority (should not be discarded)

0 = uncharacterized
1 = filler (NetNews)
2 = unattended transfer (mail)
4 = bulk (ftp)
6 = interactive (telnet)
7 = Internet control
8 = video
15 = low quality audio

Header opzionali





Header opzionali

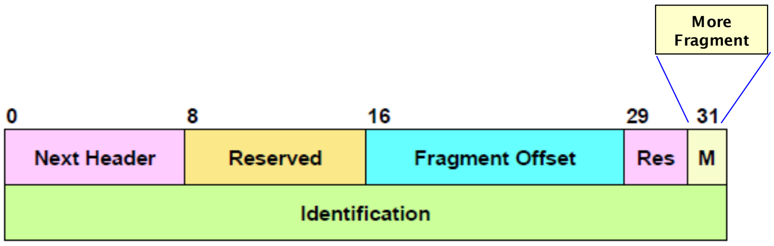
0	HBH	Hop-by-Hop option (IPv6) (Jumbogram)
1	ICMP	Internet Control Message (IPv4)
2	IGMP	Internet Group Management (IPv4)
3	GGP	Gateway-to-Gateway Protocol
4	IP	IP in IP (IPv4 encapsulation)
5	ST	Stream
6	TCP	Transmission Control
8	EGP	Exterior Gateway Protocol
9	IGP	Any private interior gateway
16	CHAOS	Chaos
17	UDP	User Datagram
29	ISO-TP4	ISO Transport Protocol Class 4
36	XTP	XTP
43	RH	Routing header (IPv6)
44	FH	Fragmentation header (IPv6)
45	IDRP	Inter-Domain Routing Protocol
46	RSVP	Reservation Protocol
50	ESP	Encapsulating Security Payload
51	AH	Authentication header (IPv6)
54	NHRP	NBMA Next Hop Resolution Protocol
58	ICMP	Internet Control Message (IPv6)
59	Null	No next header (IPv6)
60	DOH	Destination Options header (IPv6)
80	ISO-IP	ISO Internet Protocol (CLNP)
83	VINES	VINES
88	IGRP	IGRP
89	OSPF	OSPF (Open Shortest Path First)
93	AX.25	AX.25 Frames

Frammentazione

In IPv4 qualunque nodo può frammentare.

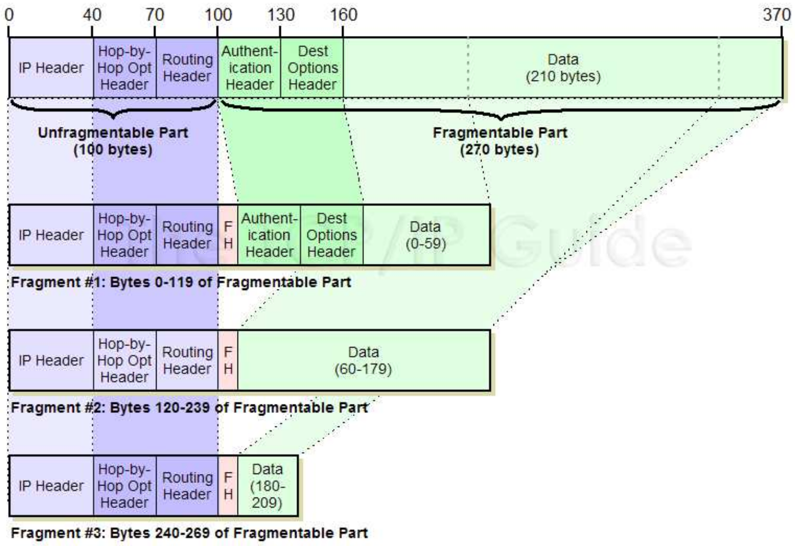
In IPv6 la frammentazione è scoraggiata. (RFC 1981 – MTU Path discovery for IPv6). Il minimo MTU è 1280 byte, con 1500 raccomandati.

Solo il mittente può frammentare, tramite un Fragment Header.



Frammentazione

Il pacchetto IPv6 può essere frammentato solo dopo il FH. Tutto ciò che è prima non deve essere frammentato.



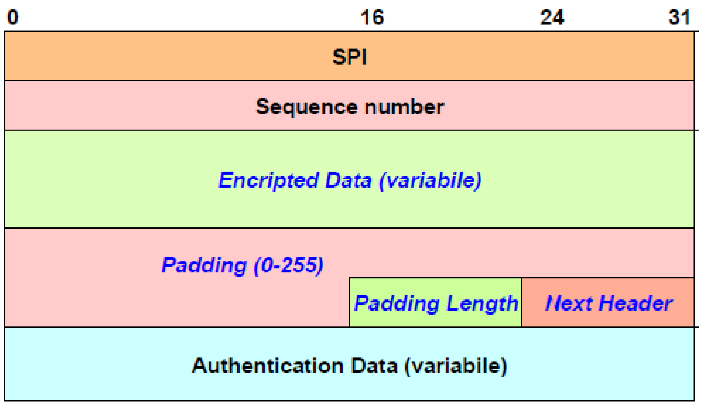
Sicurezza

Authentication Header Garantisce l'autenticità e la correttezza del pacchetto.

0	8	16	31
Next Header	Payload Length	Reserved	
Security Parameters Index (SPI)			
Sequence number			
Authentication Data (variabile)			

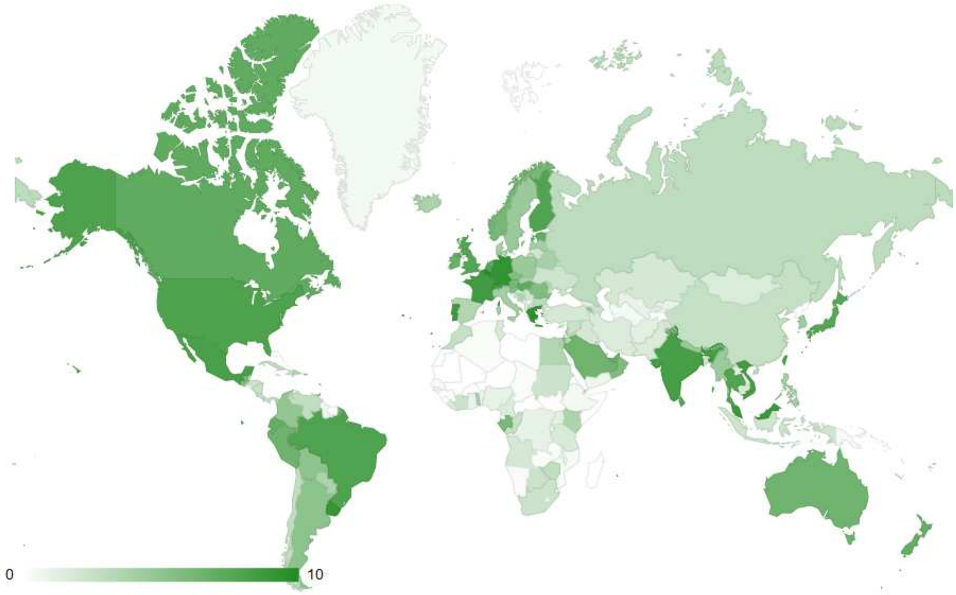
Sicurezza

Encrypted security payload Header. Fornisce la cifratura del payload.



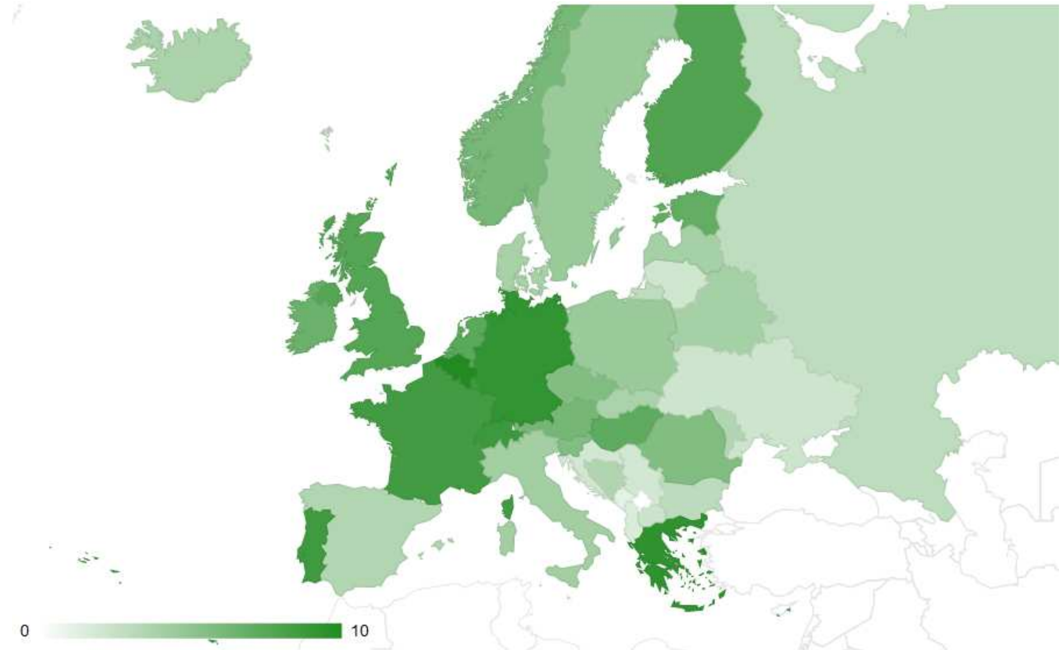
Università di Catania

Diffusione di IPv6



https://6lab.cisco.com/stats/index.php

Diffusione di IPv6 in Europa



Diffusione di IPv6 in Italia

Projection of IPv6 %-age of IPv6-Enabled Web Browsers (courtesy Google) in Italy

%-age of IPv6-Enabled Web Browsers (courtesy Google)

Metric to display: allocated prefixes - announced prefixes - alive prefixes - global routing table - IPv6 web browsers (Google) - IPv6 web browsers (APnic) - IPv6 web browsers (Akamai) - IPv6 web servers

Country to plot: Italy

Predict for 365 day(s) in the future,
based on data from the 2997 last days,
using a regression: Quadratic (2nd order)
Recompute

In the last 531 days, the IPv6 metric has doubled.

Mi piace Piace a 84 persone. Iscriviti per vedere cosa piace ai tuoi amici.

Thanks to Google for their Chart tools, to Erik Kline for his suggestion and to Tom-Alexander for his polynomial regression package.

Diffusione di IPv6 in Italia

Projection of IPv6 %-age of IPv6-Enabled Web Browsers (courtesy Google) in Italy

%-age of IPv6-Enabled Web Browsers (courtesy Google)

Metric to display: allocated prefixes - announced prefixes - alive prefixes - global routing table - IPv6 web browsers (Google) - IPv6 web browsers (APnic) - IPv6 web browsers (Akamai) - IPv6 web servers

Country to plot: Italy

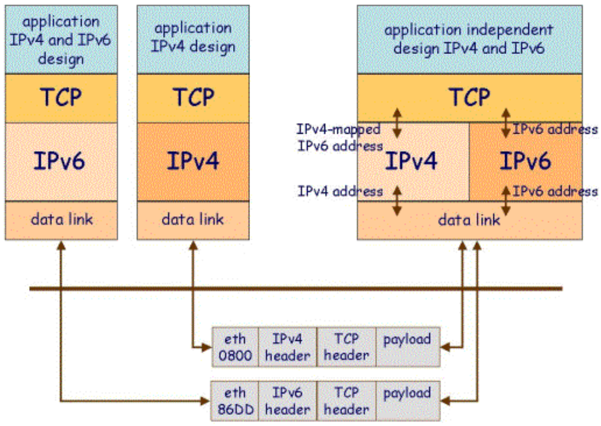
Predict for 365 day(s) in the future,
based on data from the 3338 last days,
using a regression: Quadratic (2nd order)
Recompute

In the last 891 days, the IPv6 metric has doubled.

Thanks to Google for their Chart tools, to Erik Kline for his suggestion and to Tom-Alexander for his polynomial regression package.

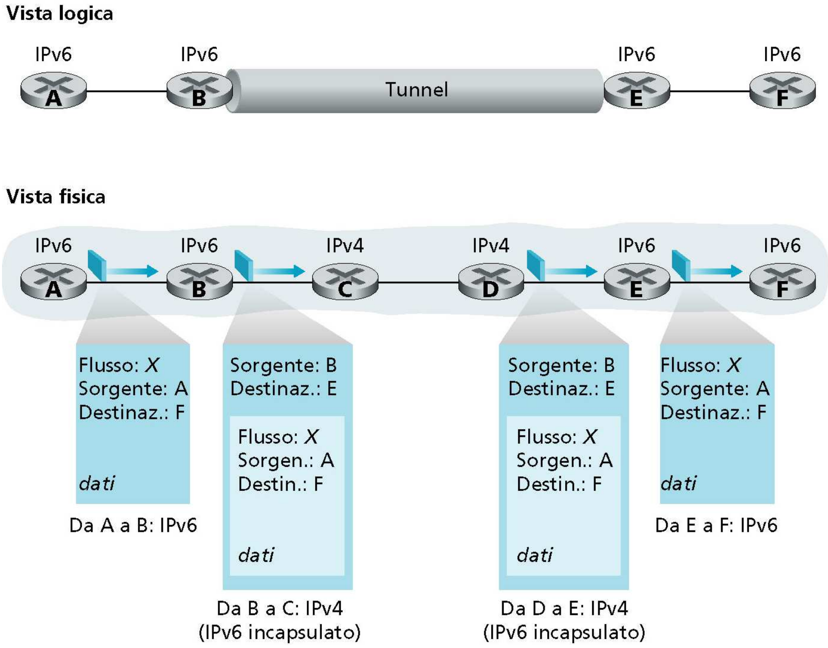
Migrazione IPv4 IPv6

L'approccio dual-stack



Migrazione IPv4 IPv6

Tunneling



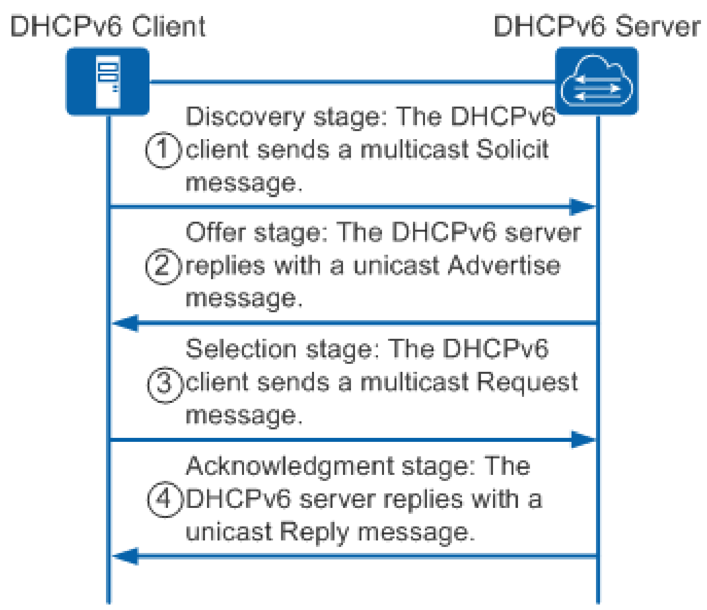
ICMPv4

ICMP Tipo	Codice	Descrizione
0	0	risposta al messaggio di eco (a ping) - *echo replay*
3	0	rete di destinazione irraggiungibile - *destination network unreachable*
3	1	host di destinazione irraggiungibile - *destination host unreachable*
3	2	protocollo di destinazione irraggiungibile - *destination protocol unreachable*
3	3	porta di destinazione irraggiungibile - *destination port unreachable*
3	6	rete di destinazione sconosciuta - *destination network unknown*
3	7	host di destinazione sconosciuto - *destination host unreachable*
4	0	strozzamento della sorgente (controllo della congestione) - *source quench*
8	0	richiesta di eco - *echo request*
9	0	annuncio dal router - *router advertisement*
10	0	scoperta del router - *router discovery*
11	0	TTL scaduto - *TTL expired*
12	0	cattiva intestazione IP - *IP header bad*

ICMPv6

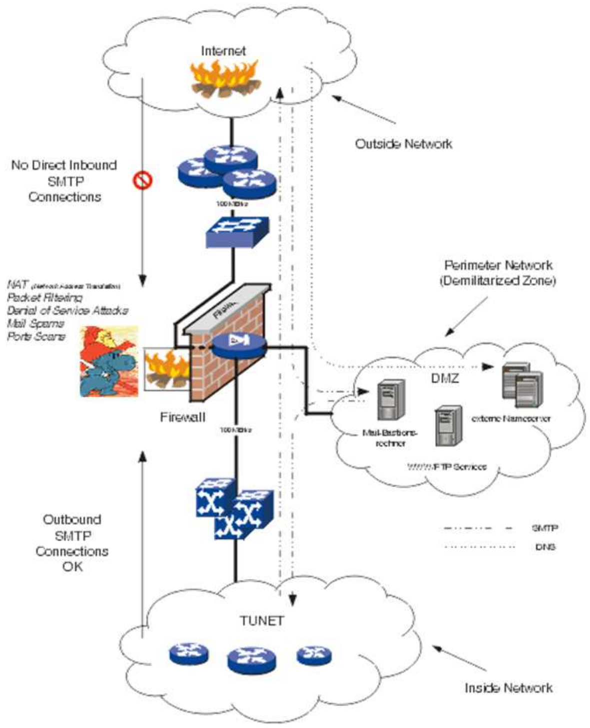
Type	Meaning
1	Destination Unreachable
2	Packet Too Big
3	Time Exceeded
4	Parameter Problem
128	Echo Request
129	Echo Reply
130	Group Membership Query
131	Group Membership Report
132	Group Membership Reduction
133	Router Solicitation
134	Router Advertisement
135	Neighbor Solicitation
136	Neighbor Advertisement
137	Redirect
138	Router Renumbering

DHCPv6



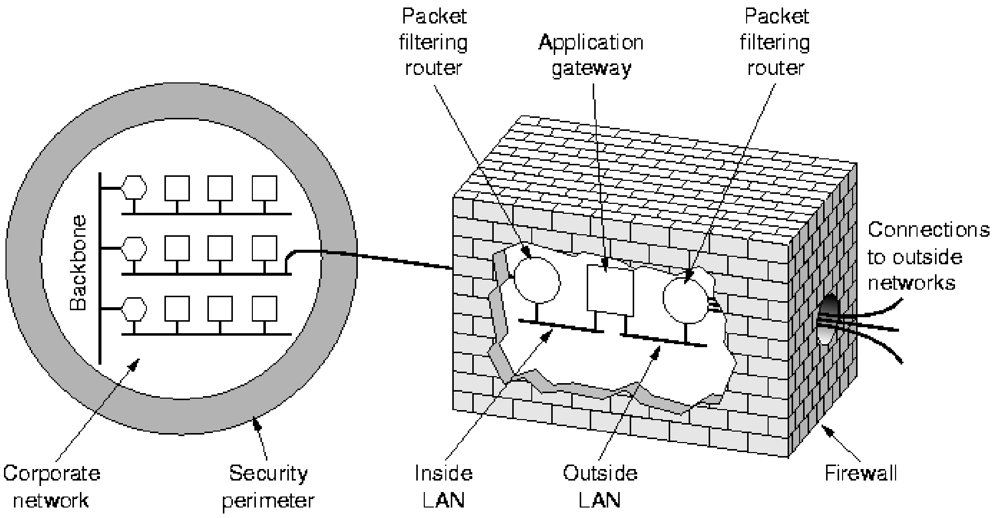
DHCPv6 Client
DHCPv6 Server
1 Discovery stage: The DHCPv6 client sends a multicast Solicit message.
2 Offer stage: The DHCPv6 server replies with a unicast Advertise message.
3 Selection stage: The DHCPv6 client sends a multicast Request message.
4 Acknowledgment stage: The DHCPv6 server replies with a unicast Reply message.

Firewall



Un firewall è un sistema di sicurezza della rete informatica che limita il traffico Internet in entrata, in uscita o all'interno di una rete privata. Questo software o unità hardware-software dedicata funziona bloccando o consentendo in maniera selettiva i pacchetti di dati.

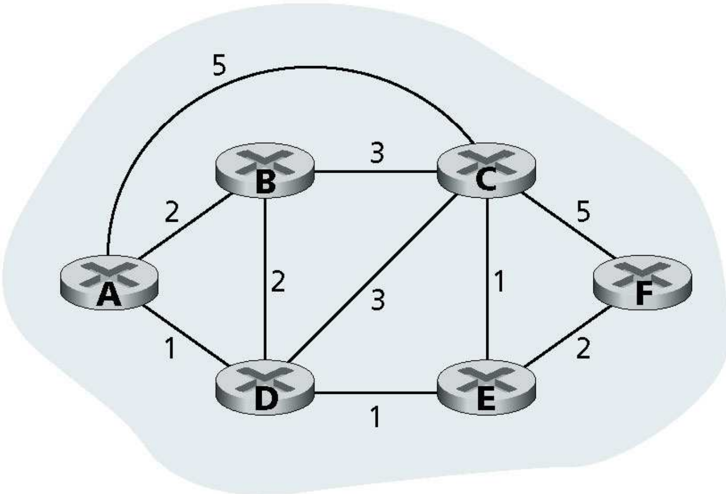
Firewall


Packet
filtering
router
Application
gateway
Packet
filtering
router
Backbone
Connections
to outside
networks
Corporate
network
Security
perimeter
Inside
LAN
Outside
LAN
Firewall

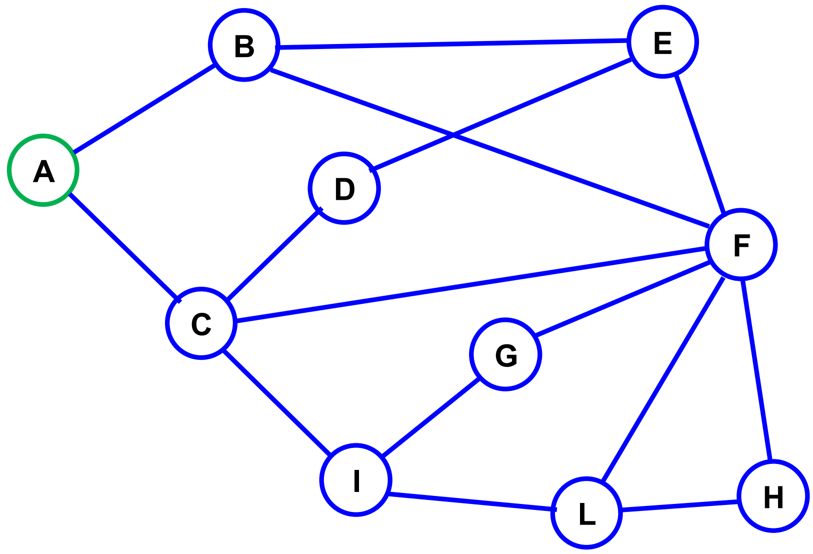
Algoritmi di routing

- Statici
- Dinamici

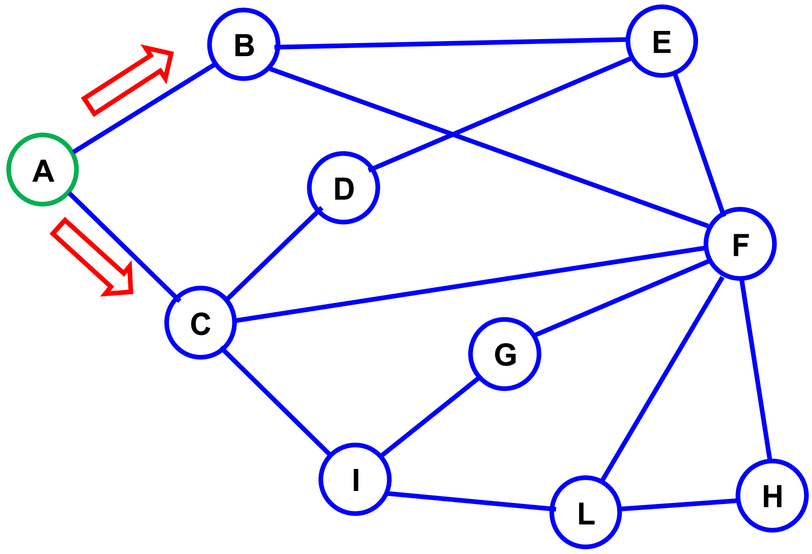
- Locali
- Globali



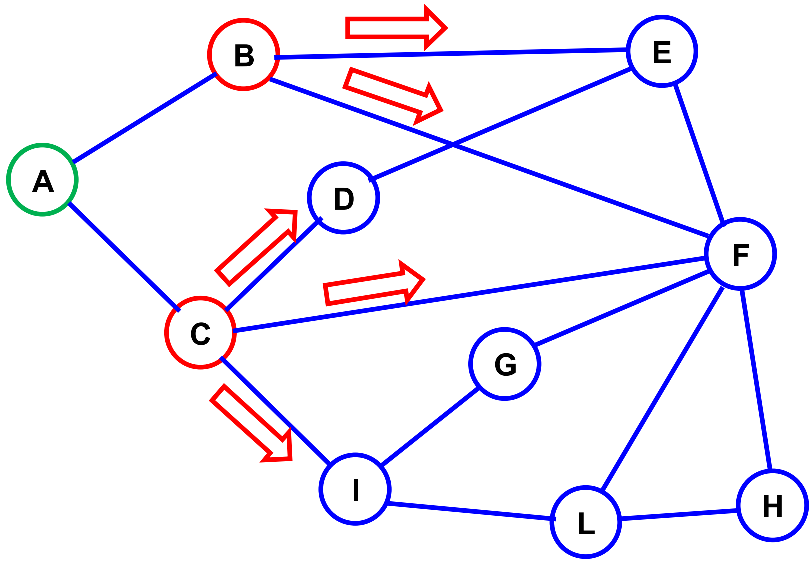
Flooding



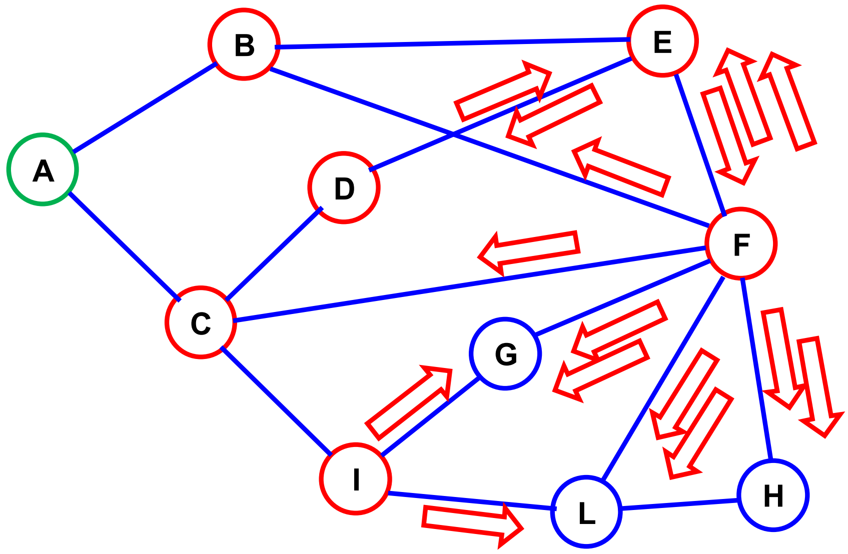
Flooding



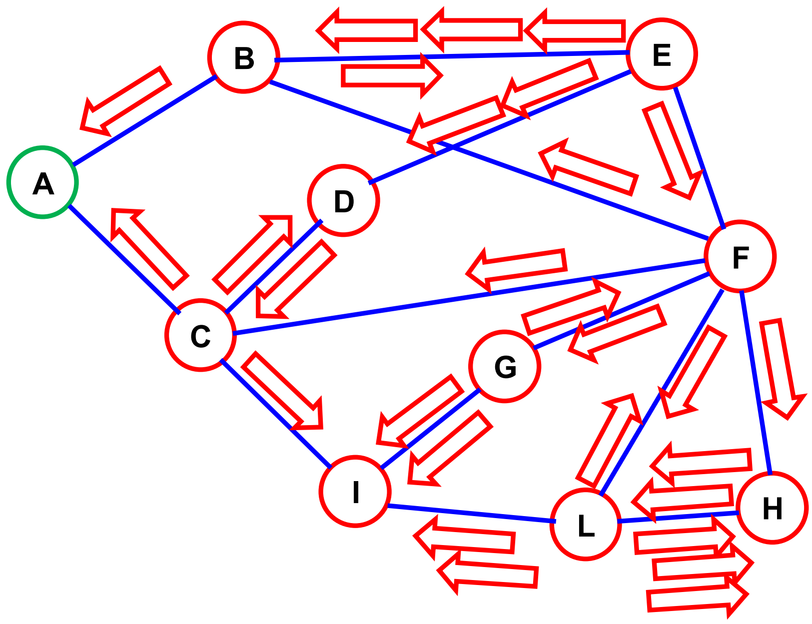
Flooding



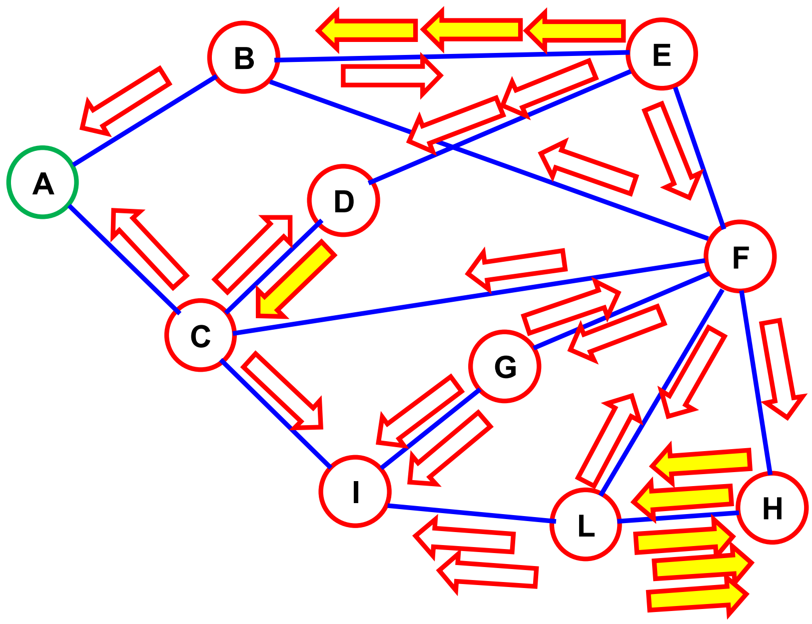
Flooding



Flooding



Flooding



Flooding

Vantaggi:
- Trova un percorso verso la destinazione, se esiste.
- Trova il percorso migliore.

Svantaggi:
- Crea un numero elevato di pacchetti duplicati, saturando la rete.

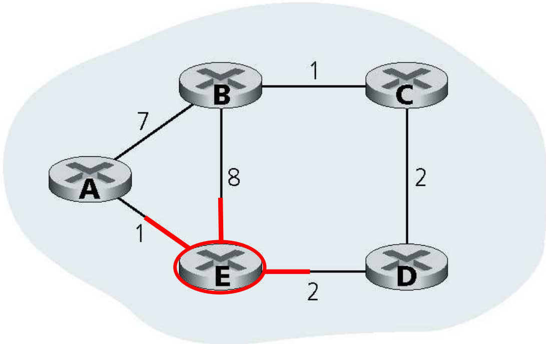
Il flooding deve essere controllato in qualche modo.

Flooding

Controllo del flooding

1) Usare un ID per i pacchetti
2) Usare un HOP limit
3) Eliminare i cicli dal grafo
4) Inoltrare solo lungo uno Spanning Tree

Distance Vectors

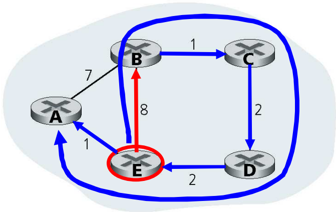


costo verso la destinazione via

$D^E()$	A	B	D
A	1	14	5
B	7	8	5
C	6	9	4
D	4	11	2

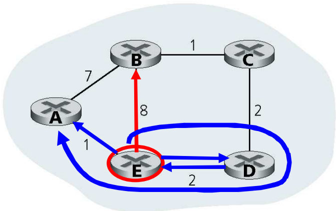
destinazione

Distance Vectors



$D^E()$	A	B	D
A	1	14	5
B	7	8	5
C	6	9	4
D	4	11	2

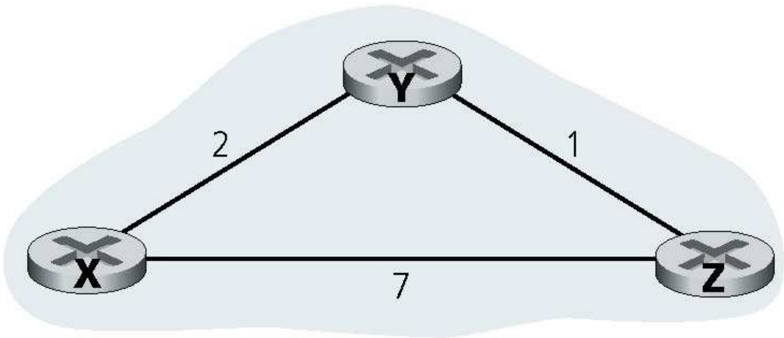
destinazione



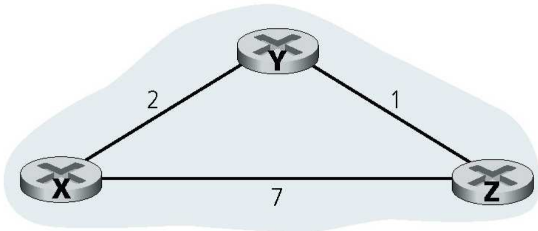
$D^E()$	A	B	D
A	1	14	5
B	7	8	5
C	6	9	4
D	4	11	2

destinazione

Distance Vectors



Distance Vectors



Uscita (columns) / Destinazione (rows)

X	Y	Z
Y	∞	∞
Z	∞	∞

X	Y	Z
Y	∞	∞
Z	∞	∞

X	Y	Z
Y	2	∞
Z	∞	7

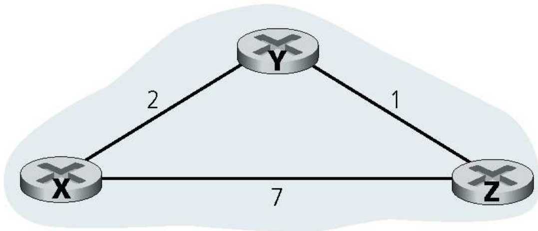
Y	X	Z
X	∞	∞
Z	∞	∞

Y	X	Z
X	2	∞
Z	∞	1

Z	X	Y
X	∞	∞
Y	∞	∞

Z	X	Y
X	7	∞
Y	∞	1

Distance Vectors



Uscita

Destinazione

X	Y	Z
Y	∞	∞
Z	∞	∞

X	Y	Z
Y	∞	∞
Z	∞	∞

X	Y	Z
Y	2	∞
Z	∞	7

X	
Y	2
Z	7

Y	X	Z
X	∞	∞
Z	∞	∞

Y	X	Z
X	2	∞
Z	∞	1

Y	
X	2
Z	1

Z	X	Y
X	∞	∞
Y	∞	∞

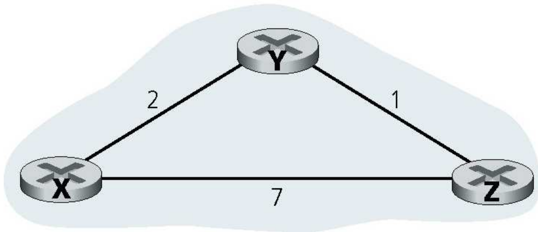
Z	X	Y
X	7	∞
Y	∞	1

Z	
X	7
Y	1

Distance Vectors

Uscita / Destinazione

X	Y	Z
Y	∞	∞
Z	∞	∞



X	Y	Z
Y	∞	∞
Z	∞	∞

X	Y	Z
Y	2	∞
Z	∞	7

X	
Y	2
Z	7

X	Y	Z
Y	2	8
Z	3	7

Y	X	Z
X	∞	∞
Z	∞	∞

Y	X	Z
X	2	∞
Z	∞	1

Y	
X	2
Z	1

Y	X	Z
X	2	8
Z	9	1

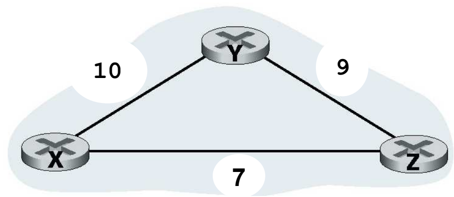
Z	X	Y
X	∞	∞
Y	∞	∞

Z	X	Y
X	7	∞
Y	∞	1

Z	
X	7
Y	1

Z	X	Y
X	7	3
Y	9	1

Distance Vectors: miglioramenti



Uscita

Destinazione

X	Y	Z
Y	∞	∞
Z	∞	∞

X	Y	Z
Y	10	∞
Z	∞	7

X	Y	Z
Y	10	16
Z	19	7

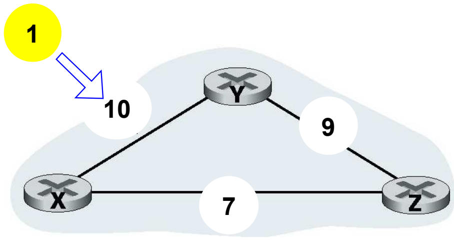
Y	X	Z
X	10	∞
Z	∞	9

Y	X	Z
X	10	16
Z	17	9

Z	X	Y
X	7	∞
Y	∞	9

Z	X	Y
X	7	19
Y	17	9

Distance Vectors: miglioramenti



Uscita

Destinazione

X	Y	Z
Y	∞	∞
Z	∞	∞

X	Y	Z
Y	10	16
Z	19	7

X	Y	Z
Y	1	16
Z	19	7

Per Z:

uscita X verso Y

Y	X	Z
X	10	16
Z	17	9

Y	X	Z
X	1	16
Z	17	9

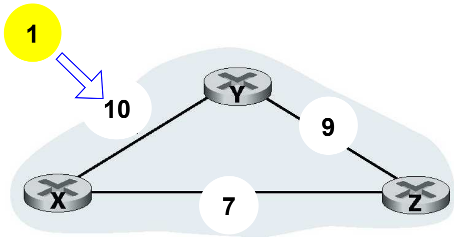
Z	X	Y
X	7	19
Y	17	9

Z	X	Y
X	7	19
Y	17	9

Z	X	Y
X	7	10
Y	8	9

Università di Catania

Distance Vectors: miglioramenti



Uscita (columns) / Destinazione (rows)

X	Y	Z
Y	∞	∞
Z	∞	∞

X	Y	Z
Y	10	16
Z	19	7

X	Y	Z
Y	1	16
Z	19	7

X	
Y	1
Z	7

X	Y	Z
Y	1	16
Z	10	7

Y	X	Z
X	10	16
Z	17	9

Y	X	Z
X	1	16
Z	17	9

Y	
X	1
Z	9

Y	X	Z
X	1	16
Z	8	9

Z	X	Y
X	7	19
Y	17	9

Z	X	Y
X	7	19
Y	17	9

Z	
X	7
Y	9

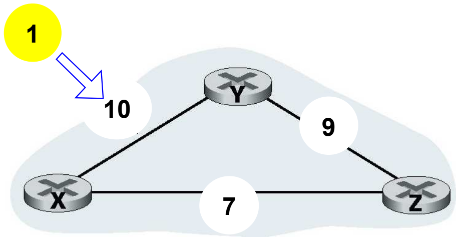
Z	X	Y
X	7	10
Y	8	9

Distance Vectors: miglioramenti

Uscita

Destinazione

X	Y	Z
Y	∞	∞
Z	∞	∞



X	Y	Z
Y	10	16
Z	19	7

X	Y	Z
Y	1	16
Z	19	7

X	
Y	1
Z	7

X	Y	Z
Y	1	16
Z	10	7

Y	X	Z
X	10	16
Z	17	9

Y	X	Z
X	1	16
Z	17	9

Y	
X	1
Z	9

Y	X	Z
X	1	16
Z	8	9

Z	X	Y
X	7	19
Y	17	9

Z	X	Y
X	7	19
Y	17	9

Z	
X	7
Y	9

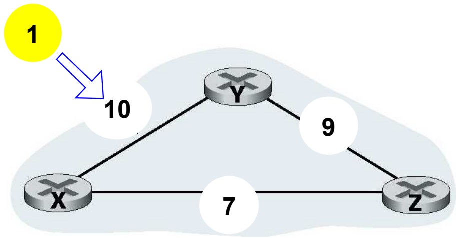
Z	X	Y
X	7	10
Y	8	9

Distance Vectors: miglioramenti

Uscita

Destinazione

X	Y	Z
Y	∞	∞
Z	∞	∞



X	Y	Z
Y	10	16
Z	19	7

X	Y	Z
Y	1	16
Z	19	7

X	
Y	1
Z	7

X	Y	Z
Y	1	16
Z	10	7

X	
Y	1
Z	7

Y	X	Z
X	10	16
Z	17	9

Y	X	Z
X	1	16
Z	17	9

Y	
X	1
Z	9

Y	X	Z
X	1	16
Z	8	9

Y	
X	1
Z	8

Z	X	Y
X	7	19
Y	17	9

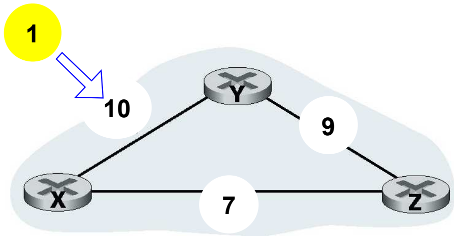
Z	X	Y
X	7	19
Y	17	9

Z	
X	7
Y	9

Z	X	Y
X	7	10
Y	8	9

Z	
X	7
Y	8

Distance Vectors: miglioramenti



Uscita / Destinazione

X	Y	Z
Y	∞	∞
Z	∞	∞

X	Y	Z
Y	1	16
Z	10	7

X	
Y	1
Z	7

X	Y	Z
Y	1	15
Z	9	7

Y	X	Z
X	1	16
Z	8	9

Y	
X	1
Z	8

Y	X	Z
X	1	16
Z	8	9

Z	X	Y
X	7	10
Y	8	9

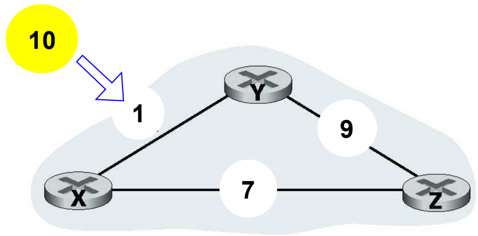
Z	
X	7
Y	8

Z	X	Y
X	7	10
Y	8	9

Distance Vectors: peggioramenti

Uscita / Destinazione

X	Y	Z
Y	∞	∞
Z	∞	∞



X	Y	Z
Y	1	15
Z	9	7

Y	X	Z
X	1	16
Z	8	9

Z	X	Y
X	7	10
Y	8	9

X	Y	Z
Y	10	16
Z	9	7

Y	X	Z
X	10	16
Z	8	9

Z	X	Y
X	7	10
Y	8	9

X	
Y	10
Z	7

Y	
X	10
Z	8

Z	
X	7
Y	8

X	Y	Z
Y	10	15
Z	18	7

Y	X	Z
X	10	16
Z	17	9

Z	X	Y
X	7	19
Y	17	9

X	
Y	10
Z	7

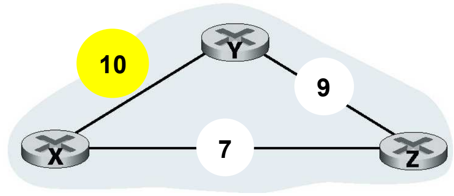
Y	
X	10
Z	9

Z	
X	7
Y	9

Distance Vectors: peggioramenti

Uscita / Destinazione

X	Y	Z
Y	∞	∞
Z	∞	∞



X	
Y	10
Z	7

Y	
X	10
Z	9

Z	
X	7
Y	9

X	Y	Z
Y	10	15
Z	18	7

Y	X	Z
X	10	16
Z	17	9

Z	X	Y
X	7	19
Y	17	9

X	Y	Z
Y	10	16
Z	19	7

Y	X	Z
X	10	16
Z	17	9

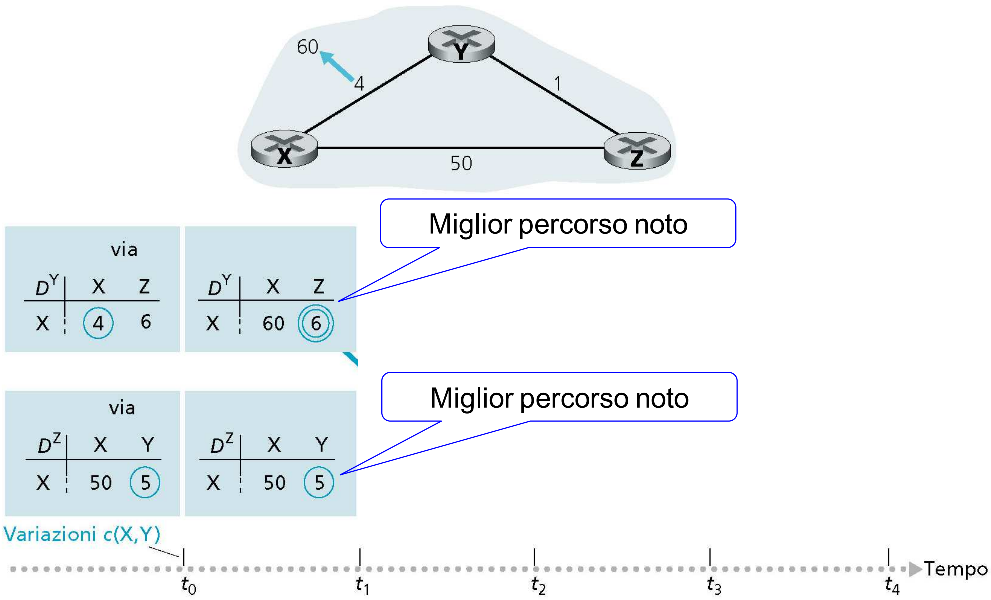
Z	X	Y
X	7	19
Y	17	9

X	
Y	10
Z	7

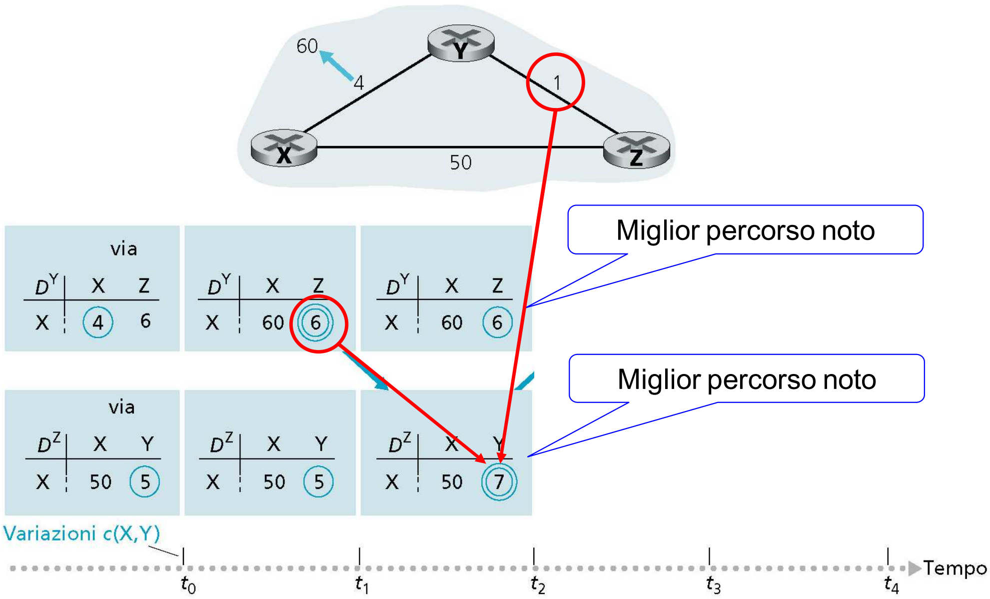
Y	
X	10
Z	9

Z	
X	7
Y	9

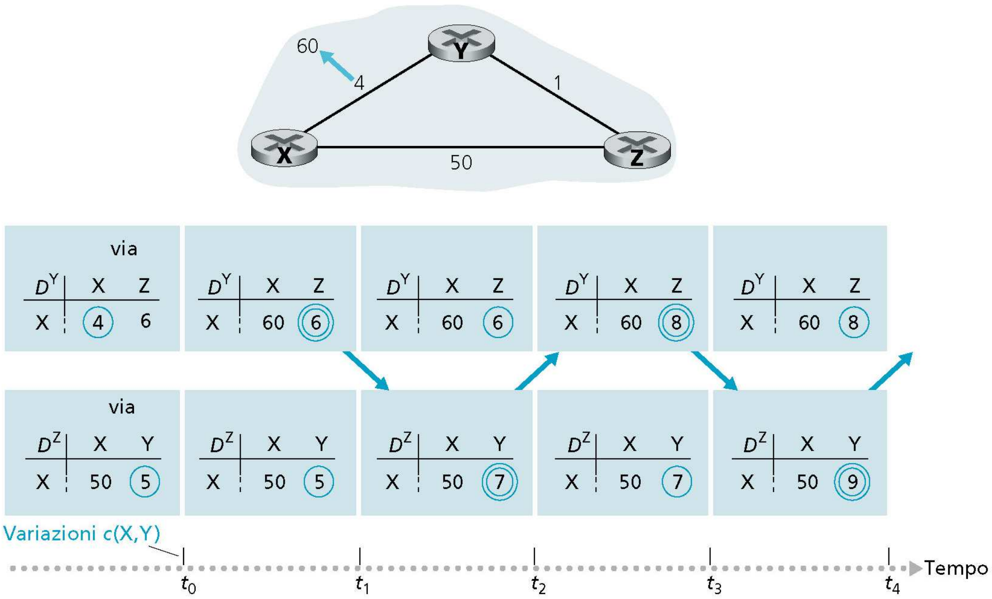
Distance Vectors



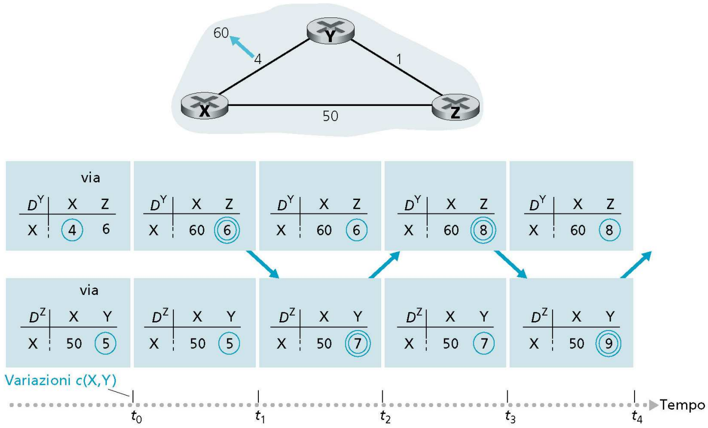
Distance Vectors



Conteggio all'infinito



Poisoned reverse



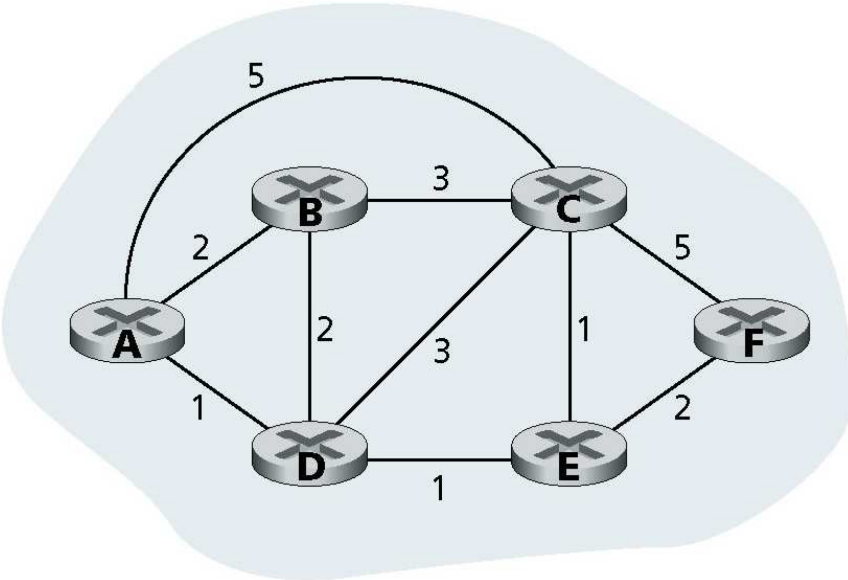
Possibile soluzione: inviare ∞ al nodo che si ha come uscita migliore per raggiungere un altro nodo

(es: z dice a y che la sua distanza verso x è ∞ se il suo percorso migliore verso x parte da y).

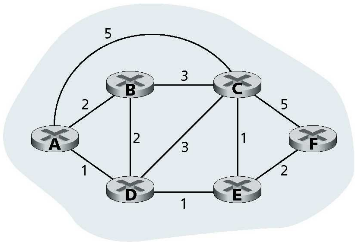
Poison Reverse è un metodo di evitamento del loop utilizzato nel Routing Information Protocol (RIP) nel Distance Vector Routing Protocol (DVRP) che consente a un'interfaccia abilitata RIP di impostare il costo del percorso (cioè impostare il conteggio degli hop che generalmente varia da 0 a 15) che ottiene dai router vicini a 16 (qui 16 sarà considerato infinito) che mostra che un determinato percorso non è raggiungibile, quindi rimanda indietro il percorso. Quando il percorso adiacente riceve questo percorso con costo 16, elimina i percorsi inutili dalla sua tabella di routing che impedisce ulteriormente il looping.

Poison Reverse è usato per affrontare i problemi di conteggio all'infinito e si può immaginare come un inverso del metodo Split Horizon. Con l'aiuto del veleno inverso, possiamo pubblicizzare la pubblicità del percorso che sarebbe soppressa da un orizzonte diviso con una distanza di infinito.

Link State Routing



Link State Routing



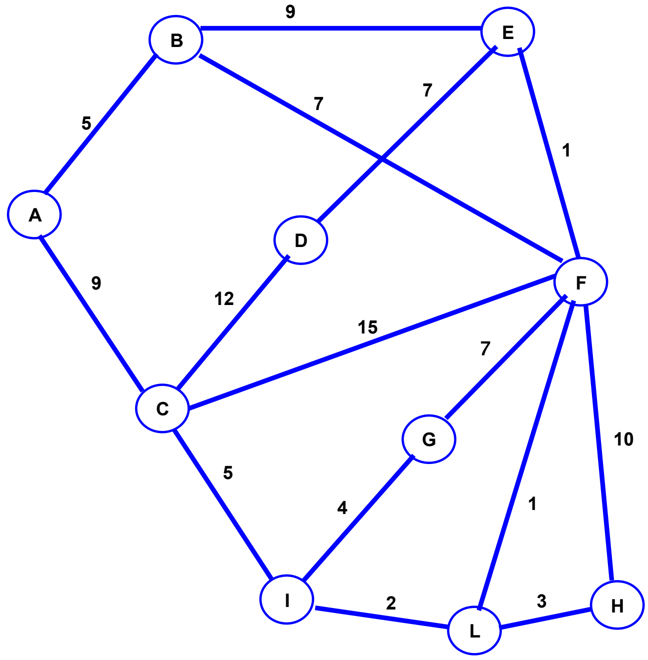
La prima versione di Arpanet utilizzava il Distance Vector Routing.

L'algoritmo venne sostituito dopo il 1979 con il Link State Routing in quanto il primo non teneva conto delle capacità dei canali ma solo dei tempi di ritardo di trasmissione.

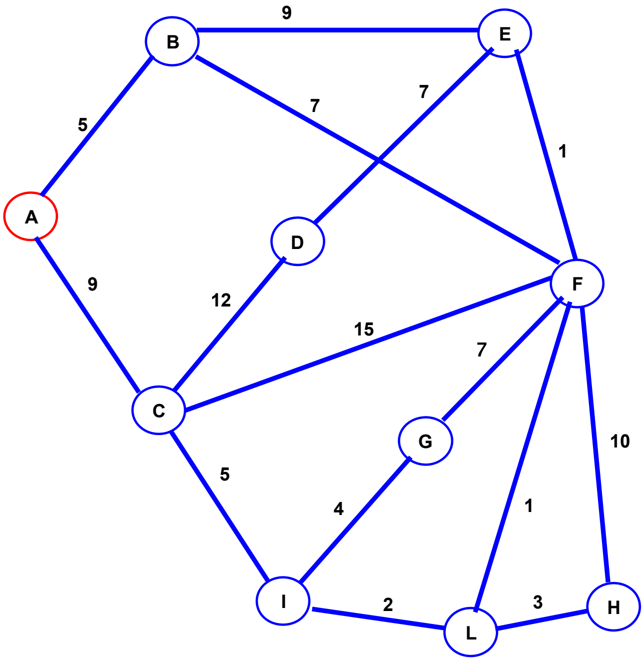
Con questo algoritmo, ogni router indaga sui propri vicini e determina i costi di comunicazione. Spedisce quindi le informazioni in broadcast, in modo che ogni router possa calcolare i percorsi ottimali.

Il calcolo è basato sull'algoritmo di Dijkstra.

Dijkstra



Dijkstra

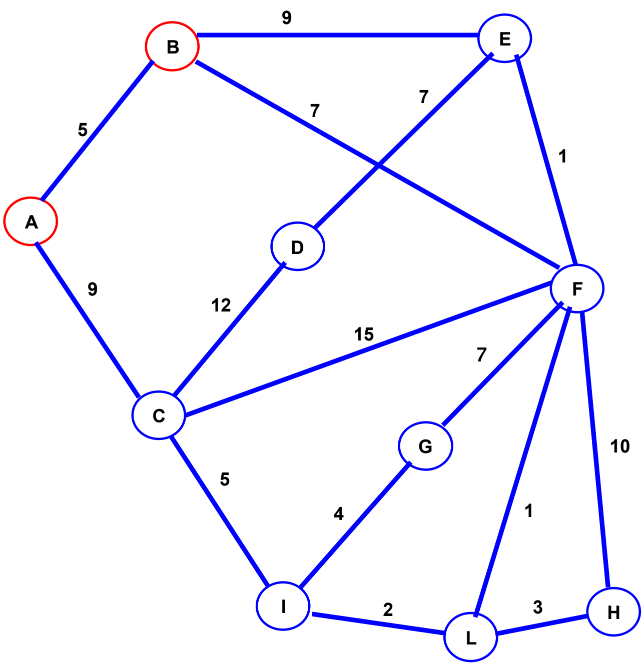


A

B = 5

C = 9

Dijkstra

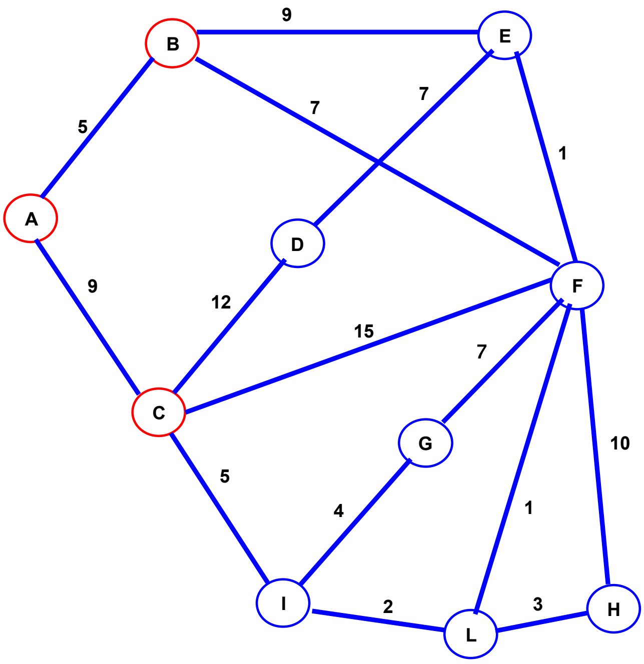


A
B = 5

C = 9
E = 5+9 = 14
F = 5+7= 12

Università di Catania

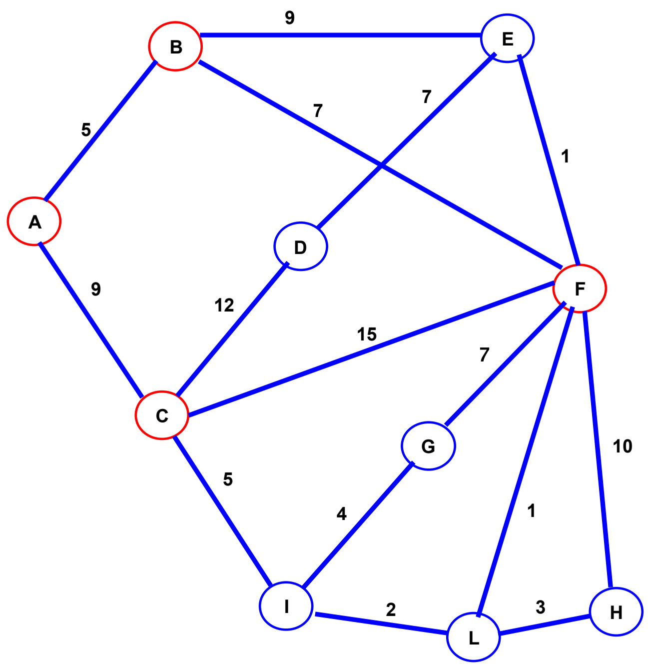
Dijkstra



A
B = 5
C = 9

E = 5+9 = 14
F = 5+7= 12
D = 9+12= 21
F = 9+15= 24
I = 9+5 = 14

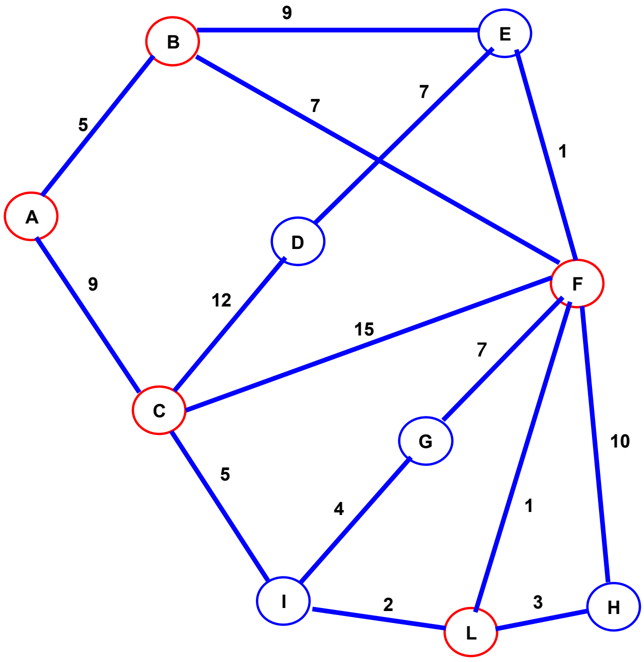
Dijkstra



A
B = 5
C = 9
F = 12

E = 5+9 = 14
D = 9+12= 21
I = 9+5 = 14
E = 12+1 = 13
G = 12+7 = 19
H = 12+10 = 22
L = 12+1= 13

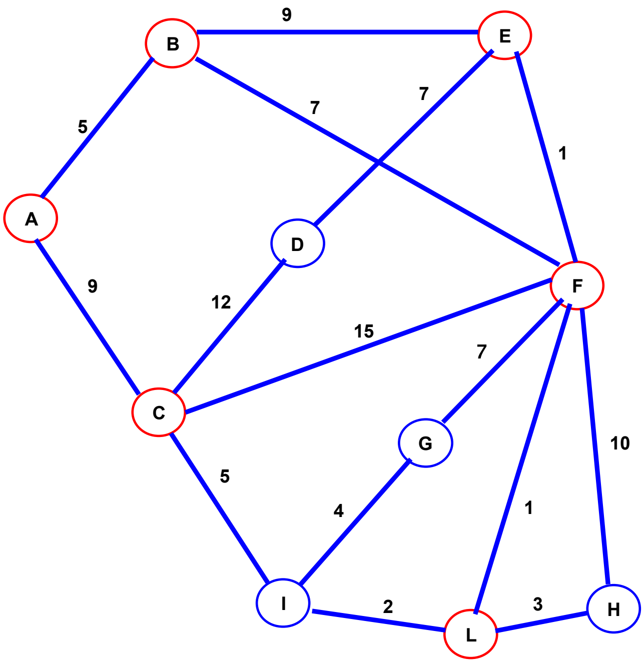
Dijkstra



A
B = 5
C = 9
F = 12
L = 13

E = 5+9 = 14
D = 9+12= 21
I = 9+5 = 14
E = 12+1 = 13
G = 12+7 = 19
H = 12+10 = 22
I = 13+2 = 15
H = 13+3 = 16

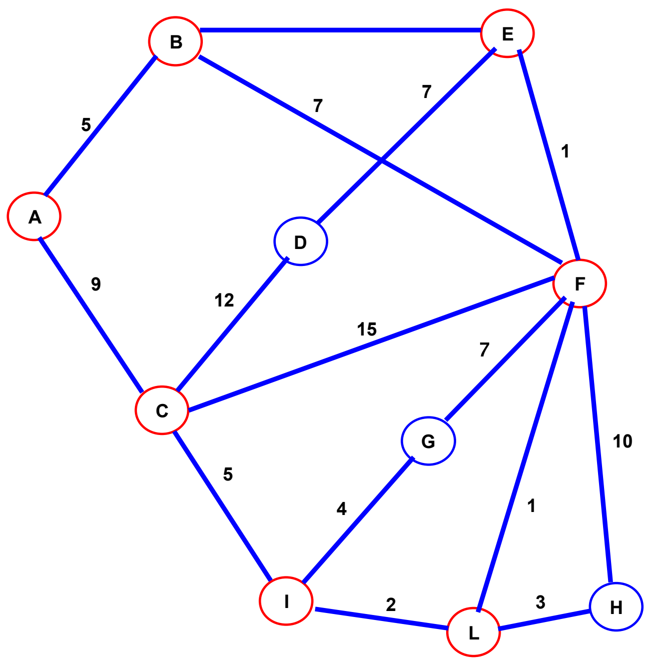
Dijkstra



A
B = 5
C = 9
F = 12
L = 13
E = 13

D = 9+12= 21
I = 9+5 = 14
G = 12+7 = 19
H = 12+10 = 22
I = 13+2 = 15
H = 13+3 = 16
D = 13+7 = 20

Dijkstra

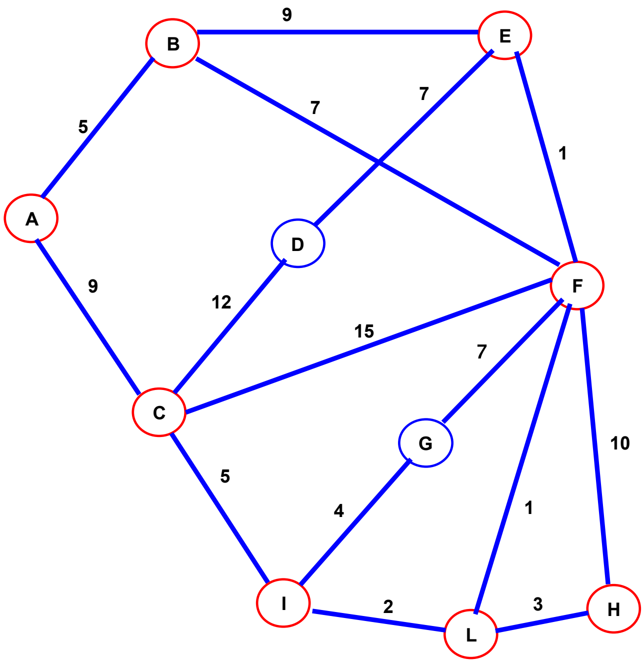


A
B = 5
C = 9
F = 12
L = 13
E = 13
I = 14

D = 9+12= 21
G = 12+7 = 19
H = 12+10 = 22
H = 13+3 = 16
D = 13+7 = 20
G = 14+4 = 18

Università di Catania

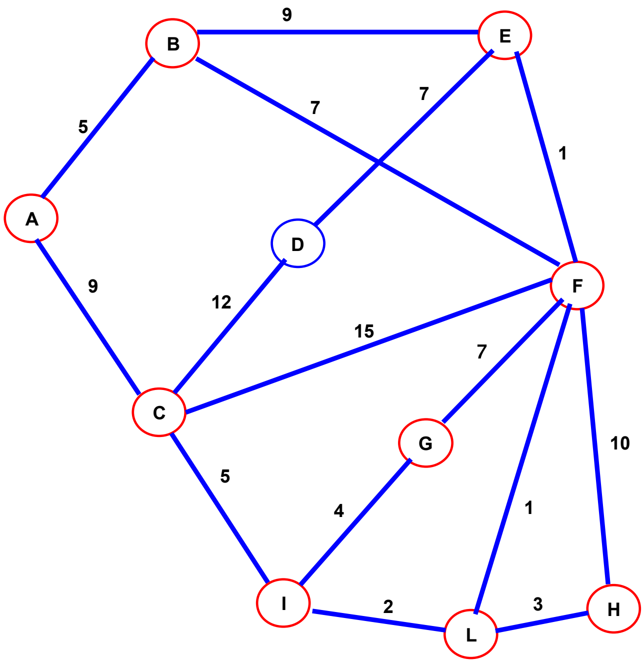
Dijkstra



A
B = 5
C = 9
F = 12
L = 13
E = 13
I = 14
H = 16

D = 9+12= 21
G = 12+7 = 19
D = 13+7 = 20
G = 14+4 = 18

Dijkstra

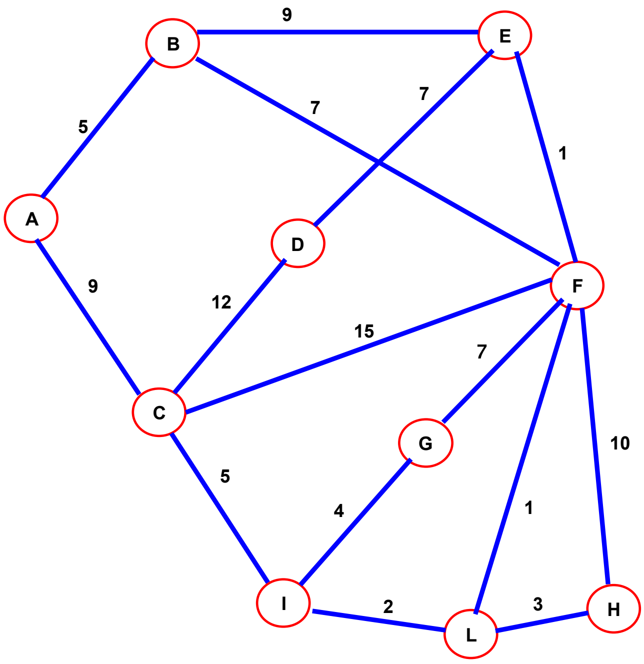


A
B = 5
C = 9
F = 12
L = 13
E = 13
I = 14
H = 16
G = 18

D = 9+12= 21
D = 13+7 = 20

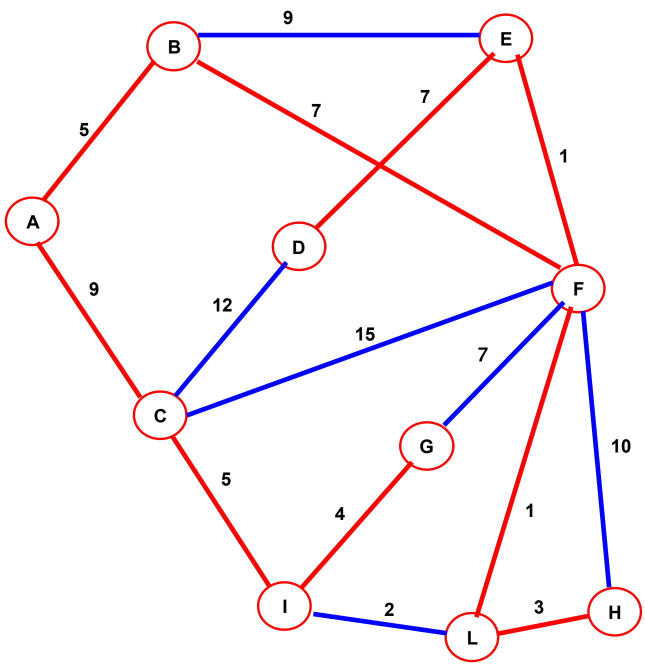
Università di Catania

Dijkstra



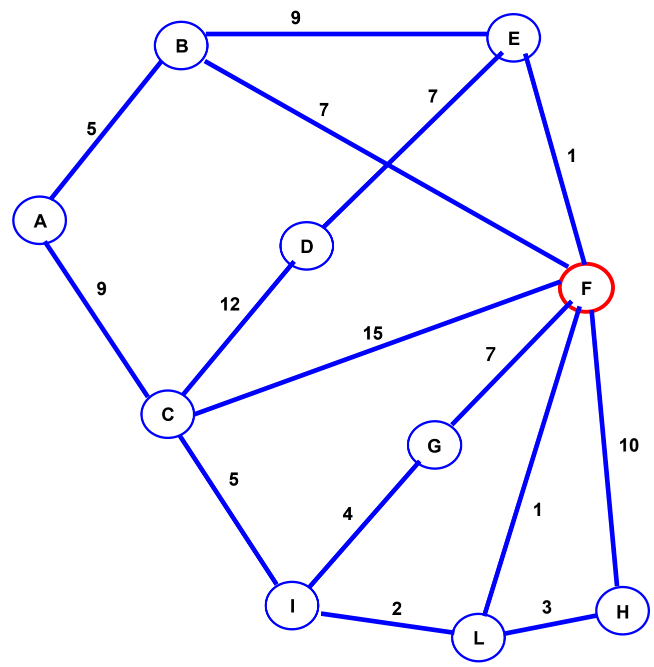
A
B = 5
C = 9
F = 12
L = 13
E = 13
I = 14
H = 16
G = 18
D = 20

Dijkstra



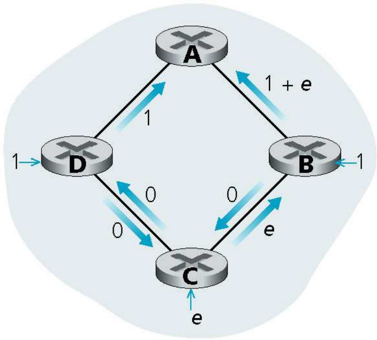
A
B = 5 A B
C = 9 A C
F = 12 A B F
L = 13 A B F L
E = 13 A B F E
I = 14 A C I
H = 16 A B F L H
G = 18 A C I G
D = 20 A B F E D

Dijkstra



Link State Routing

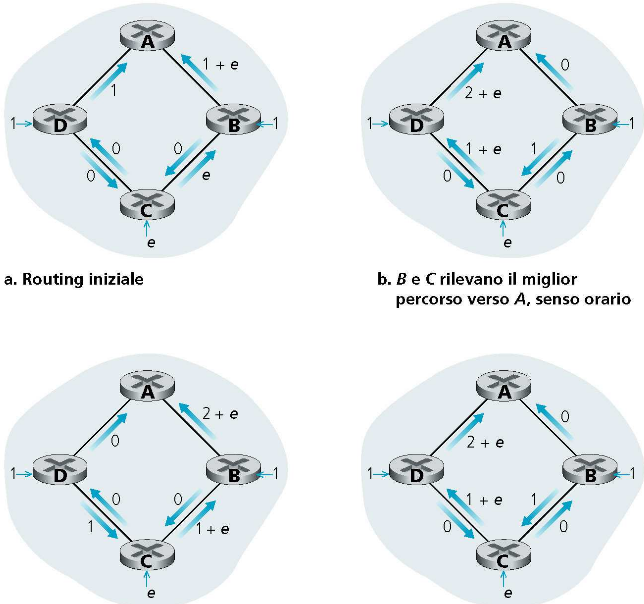
Nel LSR ogni router informa periodicamente i propri vicini sulla sua situazione dei link.



a. Routing iniziale

Nella figura i nodi B, C e D ricevono traffico dall’esterno diretto verso il nodo A.

Link State Routing



a. Routing iniziale

b. *B* e *C* rilevano il miglior percorso verso *A*, senso orario

c. *B*, *C* e *D* rilevano il miglior percorso verso *A*, senso antiorario

d. *B*, *C* e *D*, rilevano il miglior percorso verso *A*, senso orario

Confronto tra DV e LSR

DV	LSR
Decentralizzato	Globale
Messaggi solo per i vicini	Messaggi in Broadcast
Informazioni riguardanti tutte le destinazioni	Informazioni riguardanti solo i propri link
Conteggio all'infinito	Problemi di sincronia (oscillazioni)
Messaggi solo per variazioni sui link	Maggiore traffico di messaggi (invii periodici)
Poco robusto nei confronti di nodi maliziosi	Robusto agli attacchi

Internet Routing

Principi base:

- routing dinamico
- Intra-AS e Inter-AS routing

Internet è organizzata in “autonomous systems” (AS).

AS definition: unit of routing policy, either a single network or a group of networks that is controlled by a common network administrator on behalf of a single administrative entity (such as a university, a business enterprise, or a business division)

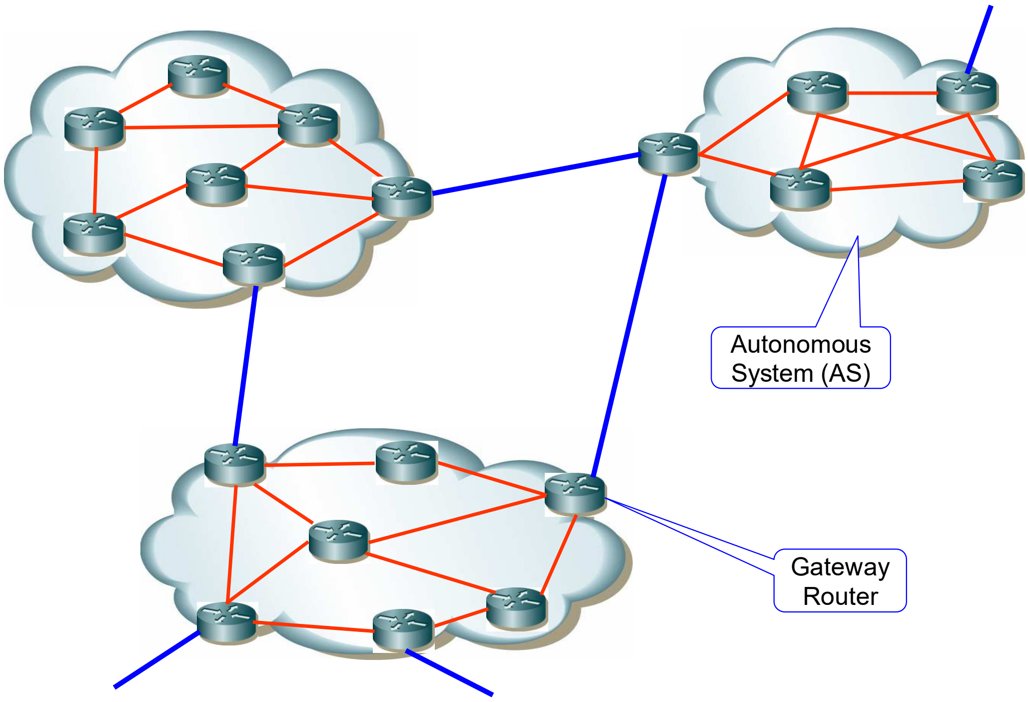
Ogni AS è internamente fortemente connessa.

Internet Routing

Protocolli:

- Interior Gateway Protocols (IGPs) all’interno di AS (RIP, OSPF, HELLO, IS-IS)
- Exterior Gateway Protocols (EGPs) tra AS (EGP, BGP-4)

AS



RIP

Routing Information Protocol:

- RIP v1 (RFC 1058)
- RIP v2 (RFC 1723, 2453).

- Basato su Distance Vector.
- Metrica utilizzata: HOP (numero di salti effettuati)

RIP v1

Il numero massimo di HOP ammesso è 15**:** con reti troppo estese la convergenza risulterebbe lenta.

Oltre tale valore il router è considerato irraggiungibile.

Le Routing Table (RT) vengono scambiate ogni 30 secondi.

Se un percorso per un router non viene aggiornato entro 180 secondi, la sua distanza è posta a infinito.

Trascorsi altri 120 secondi (garbage-collection timer) il router viene eliminato dalla RT.

Messaggi RIP v1

RIP v1 utilizza due tipi di messaggi :

- REQUEST: per chiedere informazioni ai nodi adiacenti
- RESPONSE: per inviare informazioni di routing

Una tabella di routing contiene:

- Indirizzo di destinazione
- Distanza dalla destinazione (in hop)
- Next hop: router adiacente a cui inviare i pacchetti
- Timeout
- Garbage-collection timer

Datagrammi RIP v1

Lunghezza variabile fino a 512 byte (max 25 reti di destinazione).

Command	Version	Must Be Zero
Address Family Identifier		Must Be Zero
IP Address		
Must Be Zero		
Must Be Zero		
Metric		

Command : 1 = richiesta, 2= aggiornamento.

Version : versione del protocollo.

Address Family Identifier : sempre 2 per il protocollo IP

IP Address : indirizzo di destinazione (rete o sottorete)

Metric : hop count (valore compreso tra 1 e 15)

RIP v2

Caratteristiche di RIP v2

- Indirizzamento CIDR e VLSM
- Autenticazione dei messaggi
- Specifica del next hop
- Split horizon, poison reverse

Datagrammi RIP v2

0	7 15	31
Command	Version	0000
0xFFFF		Autentication Type
Autentication		
Address Family Ident.		Route Tag
IP Address		
Subnet Mask		
Next Hop		
Metric		

Command : 1 = richiesta, 2= aggiornamento.

Version : versione del protocollo.

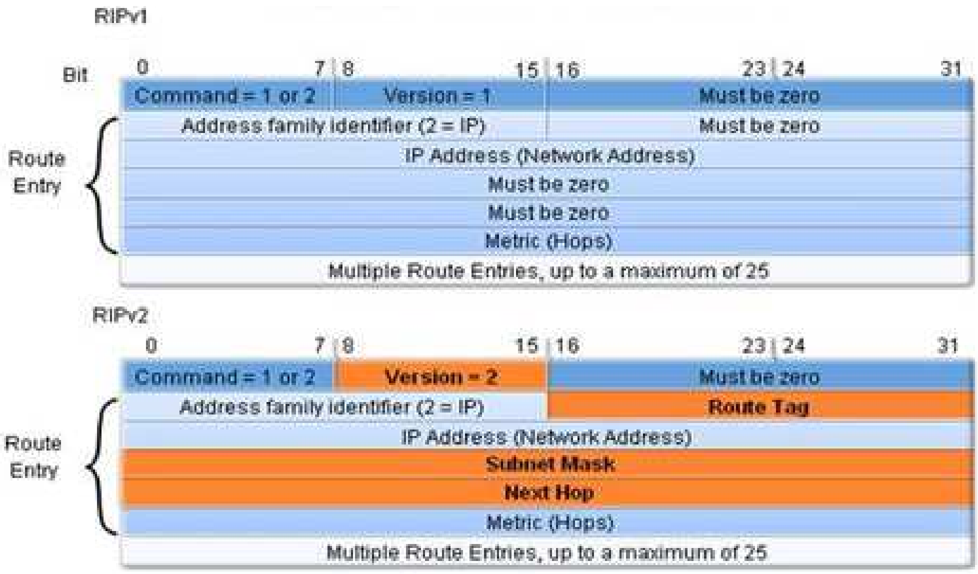
Address Family Identifier : sempre 2 per il protocollo IP

Route Tag: identificazione di route esterni

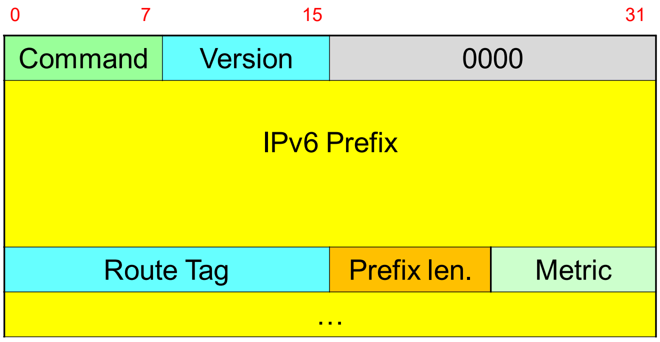
IP Address : indirizzo di destinazione (rete o sottorete)

Metric : hop count (valore compreso tra 1 e 15)

Datagrammi RIP v2



RIPng



RIPng è basato su RIPv2, ma non ne rappresenta un’estensione.
È pensato esclusivamente per IPv6 (non supporta IPv4).

RIPng ha le stesse caratteristiche di RIPv2 tranne l’autenticazione

OSPF

Open Shortest Path First

- OSPFv1 (RFC 1131);
- OSPFv2 (RFC 2178, 2328).

- Pensato per sostituire RIP
- Si basa sul Link State Routing
- Va bene per reti di dimensioni grandi
- I messaggi sono autenticati
- Consente l'utilizzo di metriche differenti

OSPF

Il protocollo OSPF si compone di tre parti:

- HELLO: scoperta e verifica dei vicini
- EXCHANGE: sincronizzazione iniziale del DB
- FLOODING: aggiornamento del DB

Database di Link State

Ogni router ha un database composto da "link state record"

I vari DB vengono aggiornati e sincronizzati tramite i "Link State Advertisement" (LSA).

Gli LSA sono emessi:

- quando un router riscontra un nuovo router adiacente
- quando un router – link si guasta
- quando il costo di un link cambia
- periodicamente, tipicamente ogni 30 minuti

Link State Advertisement

LS Age	Options	Ls Type
Link State ID		
Advertisement Router		
Link State Sequence Number		
Link State Checksum	Lenght	
LS Data		

Header comune OSPF

Version	Type	Message Lenght
Router ID		
Area ID		
Checksum		Autentication Type
Autentication Data		
Rest of the OSPF Message		

Type:

1. Hello
2. DB Description
3. LS Request
4. LS Update
5. LS ACK

Router ID: router che ha generato il messaggio

AreaID: Ip dell'area a cui si riferisce il messaggio

Tipi di datagrammi OSPF

Hello: Trasmissione periodica per la scoperta dei vicini

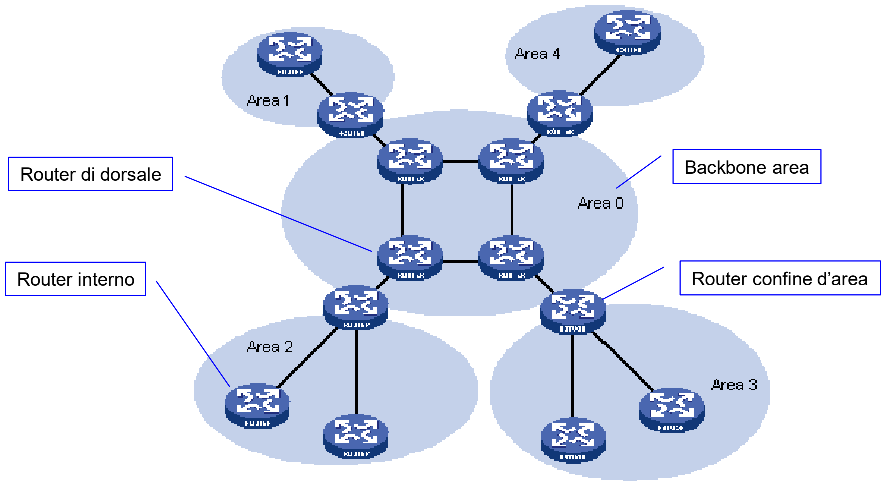
Database Description: usato per lo scambio dell'informazione dei Link-State di ogni router

LS Request: richiesta di parti specifiche del database di Link-State di un vicino

Link-State Update: trasferisce annunci di link-state ai vicini

Link-State Acknowledgments: invia una conferma di ricezione di un update di link-state

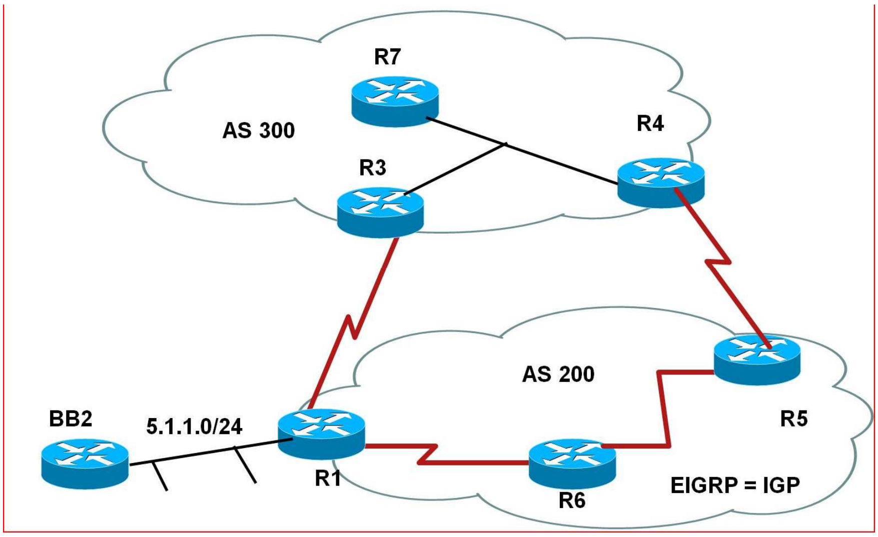
OSPF



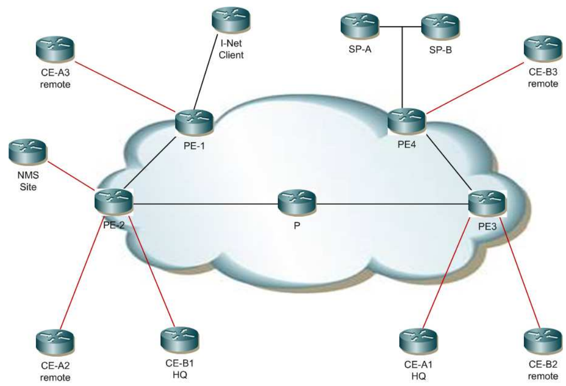
Area 1
Area 4
Router di dorsale
Backbone area
Area 0
Router interno
Router confine d'area
Area 2
Area 3

Università di Catania

BGP v4



BGP



- AS Border Router (ASBR): router connesso ad altri sistemi autonomi
- BGP speaker: router che supporta il protocollo BGP (un BGP speaker non necessariamente coincide con un AS border router)
- BGP Neighbors: coppia di BGP speaker che si scambiano informazioni di instradamento inter-AS
  - interni: se appartengono allo stesso AS
  - esterni: se appartengono ad AS diversi