POLITECNICO
MILANO 1863

Fondamenti di Internet e Reti

Antonio Capone, Matteo Cesana,
Ilario Filippini, Guido Maier

4 - Livello di Rete (parte 3)

Antonio Capone, Matteo Cesana,
Ilario Filippini, Guido Maier

Agenda

- Il protocollo IPv4
- Protocolli di gestione di IP
  - ICMP
  - ARP e RARP
  - DHCP
- Network Address Translation (NAT)

Agenda

- Il protocollo IPv4
- Protocolli di gestione di IP
  - ICMP
  - ARP e RARP
  - DHCP
- Network Address Translation (NAT)

Il servizio di comunicazione offerto da IP

- *Connectionless*
  - progettato secondo un paradigma *packet-oriented (o datagram)*
  - Due pacchetti (o datagrammi) destinati alla stesso host possono "essere trattati" in maniera diversa

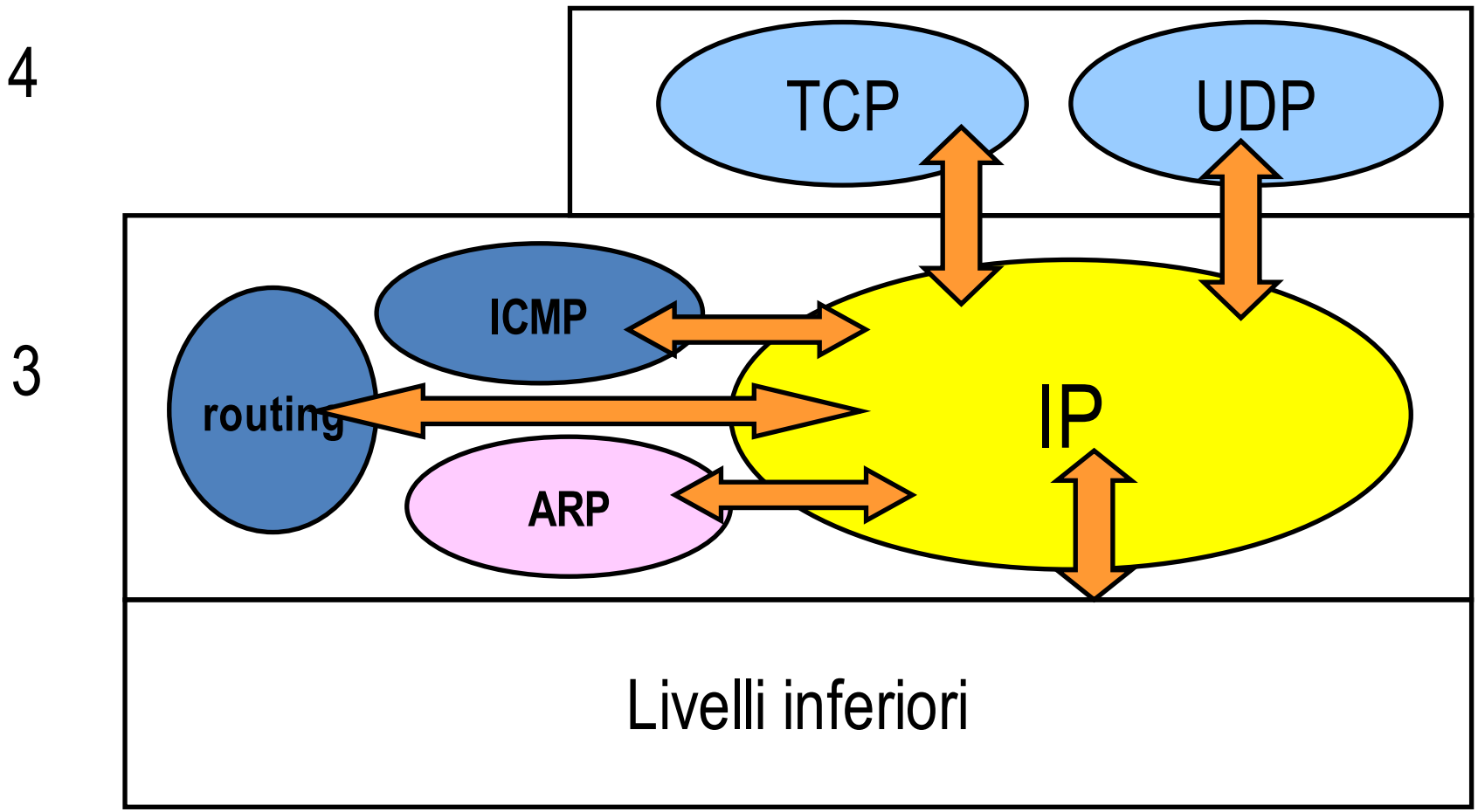
- Non affidabile
  - Consegna *best-effort* dei datagrammi senza garanzia di successo
  - Analogia con il servizio postale ordinario

Altri Servizi Offerti da IP

- Indirizzamento: assegna un indirizzo universalmente riconosciuto

- Frammentazione/Deframmentazione: frammenta/deframmenta i pacchetti se il livello locale (livello 2) lo richiede (IP è pensato per funzionare su molteplici tecnologie di livello inferiore)

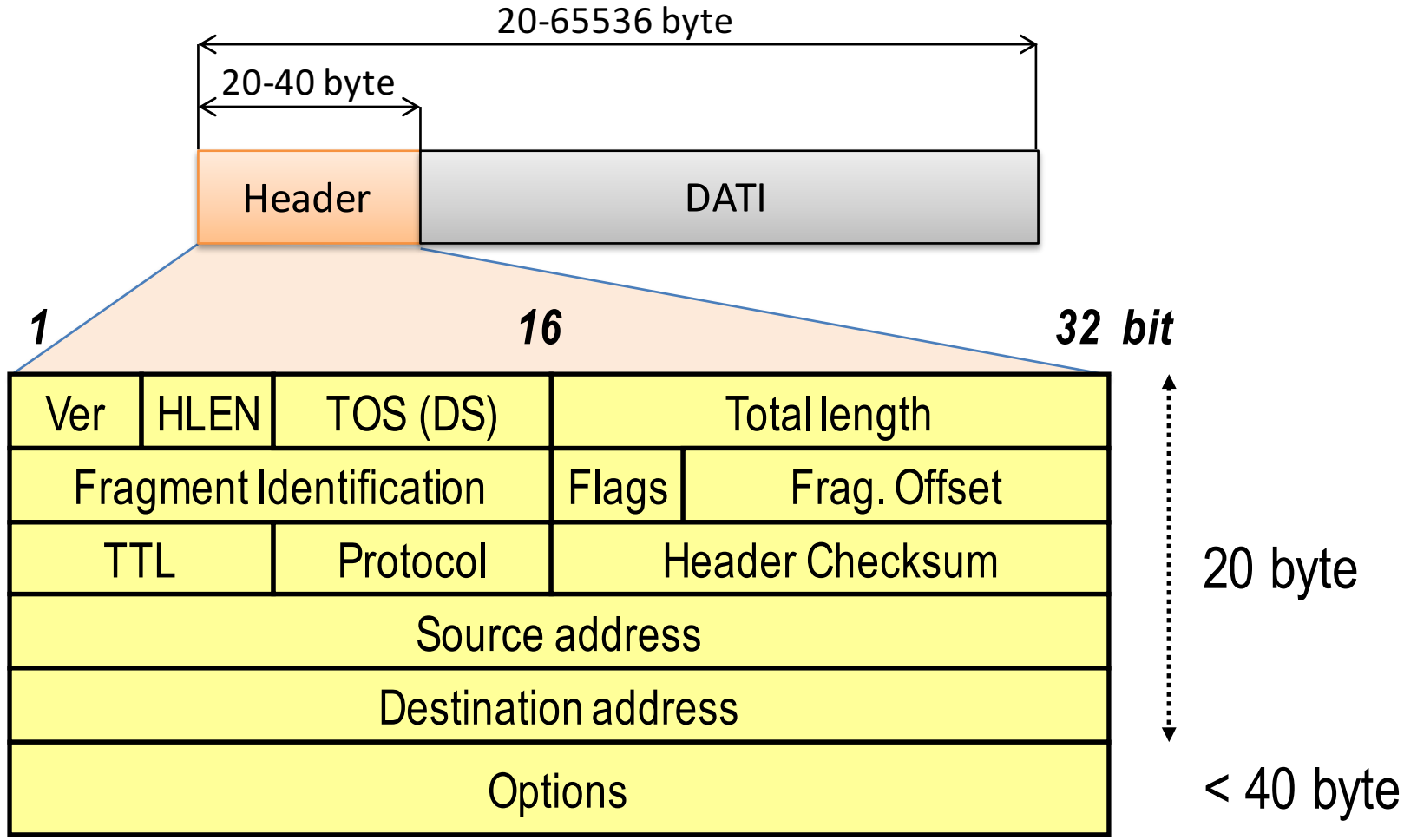
Lo stack IP base



Internet Protocol (IP): caratteristiche

- Le unità informative IP (IP-PDU) sono chiamate datagrammi IP o pacchetti IP
  - IP header + payload di dati che ospita la PDU del livello di trasporto
- Esegue la rivelazione di errori e la segnalazione di errori
- Garanzia sul max lifetime: il datagram IP viene eliminato se non consegnato entro un preassegnato time-to-live

Formato del datagramma IP



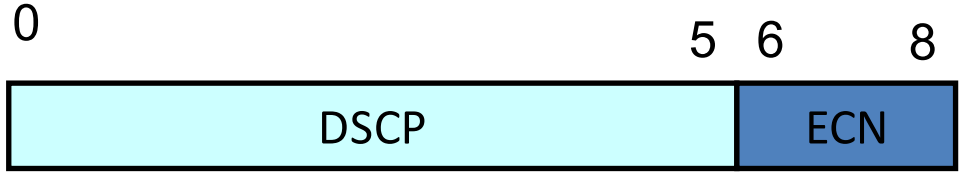
- A causa dell'importanza dei campi indirizzo (a 32 bit), si suole raffigurare il datagram a righe di 32 bit (pura rappresentazione grafica)

I campi dell'header IP

- Ver (4 bit):
  - *Version*: indica la versione del protocollo; IPv4, IPv6. Se il campo VER non corrisponde alla versione del protocollo implementata sul router ricevente, il pacchetto viene scartato.
- HLEN (4 bit)
  - *Header length*: indica la lunghezza dell'header del pacchetto espressa in parole da 32 bit (max 64 byte)
- Total length (16 bit):
  - Indica la lunghezza totale del pacchetto in byte: valore massimo 2^{16}=65536; una volta sottratta la dimensione dell'header dà la lunghezza del payload. Serve solo se il livello sottostante effettua padding riempitivo.

I campi dell'header IP

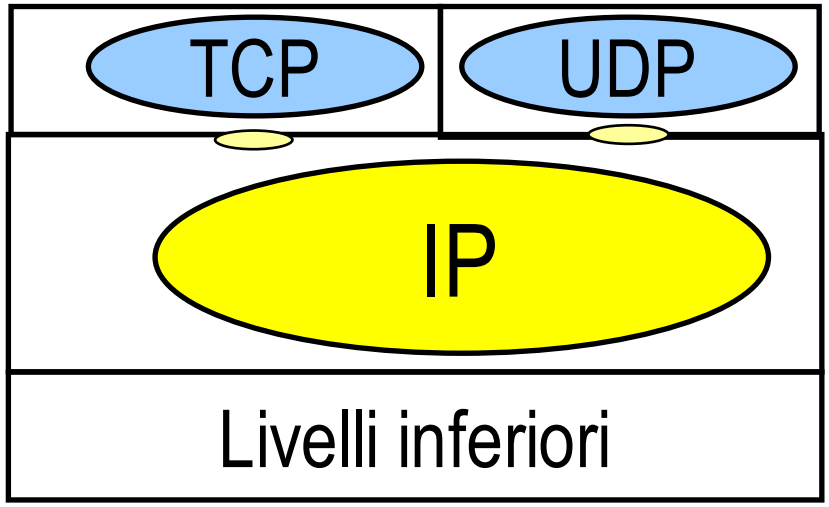
- TOS *type of service* (8 bit)
  - Ha subito diversi cambi di significato nel tempo, sempre legati alla gestione del pacchetto nelle code dei router
  - Il più recente è:



- Differentiated Services Code Point
  - Default (0)
  - Expedited Forwarding /Voice Admit (Basso Ritardo)
  - Assured Forwarding (Differenti mix di classe di priorità di servizio e precedenza di dropping in caso di congestione)
  - Class selector (8 differenti classi di servizio)
- Explicit Congestion Notification
  - Usato nei router per segnalare un'imminente congestione (e drop dei pacchetti) alle destinazioni

Il campo *Protocol*

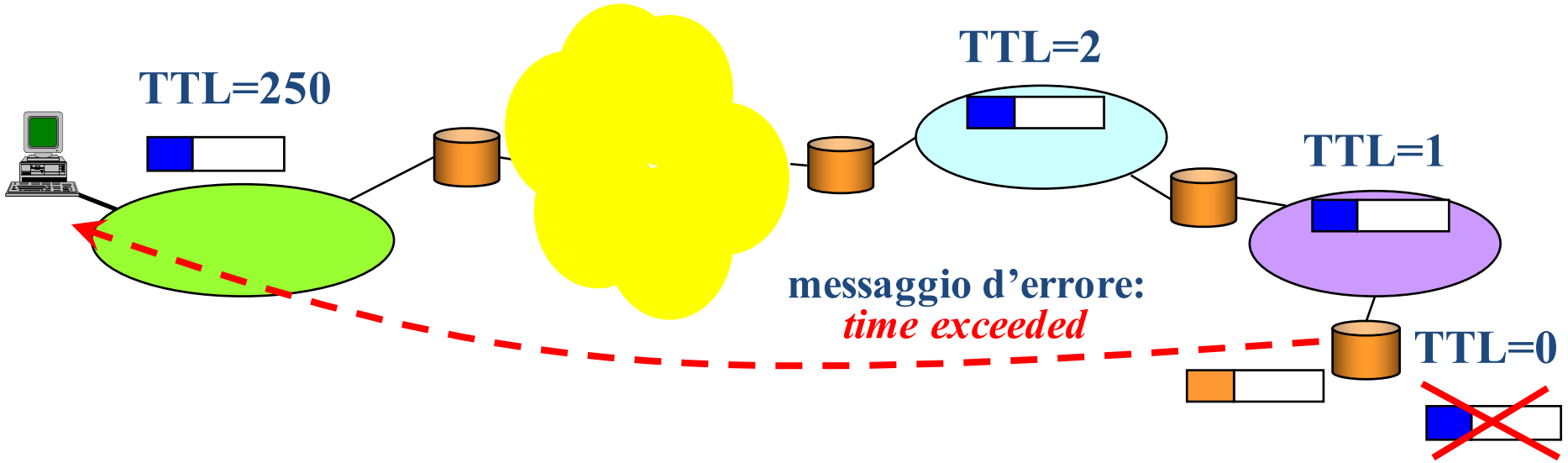
- E' un codice che indica il protocollo di livello superiore
- Più protocolli di livello superiore possono usare IP (multiplazione)
- Il codice identifica il SAP (*Service Access Point*) tra IP e il protocollo di livello superiore



Valore	Protocollo
1	ICMP
2	IGMP
6	TCP
17	UDP
89	OSPF

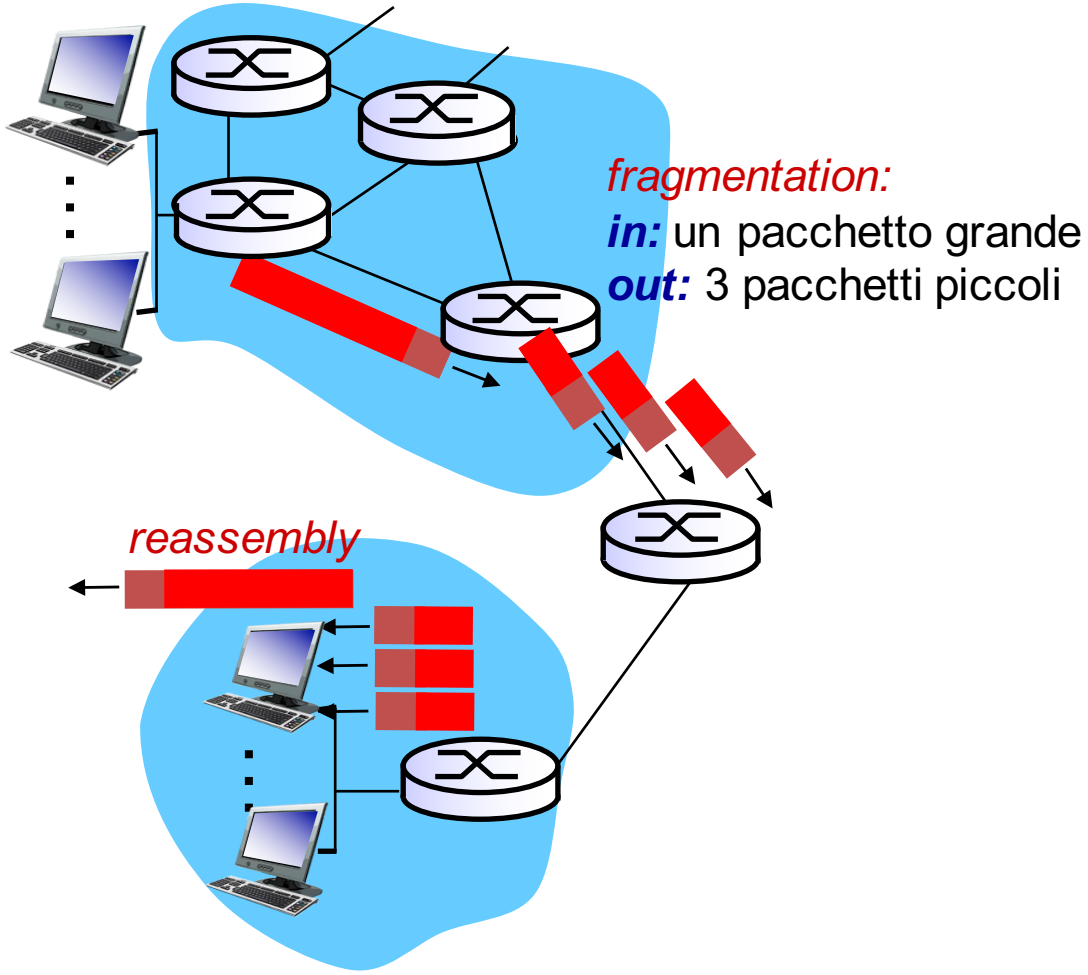
Il campo *Time To Live (TTL)*

- Il campo *TTL* viene settato ad un valore elevato da chi genera il pacchetto e viene decrementato da ogni router attraversato
- Se un router decrementa il valore e questo va a zero, il pacchetto viene scartato e viene generato un messaggio di errore verso la sorgente
  - NOTA: TTL decrementato in uscita dal router
- *Time-out* sulla validità di un pacchetto

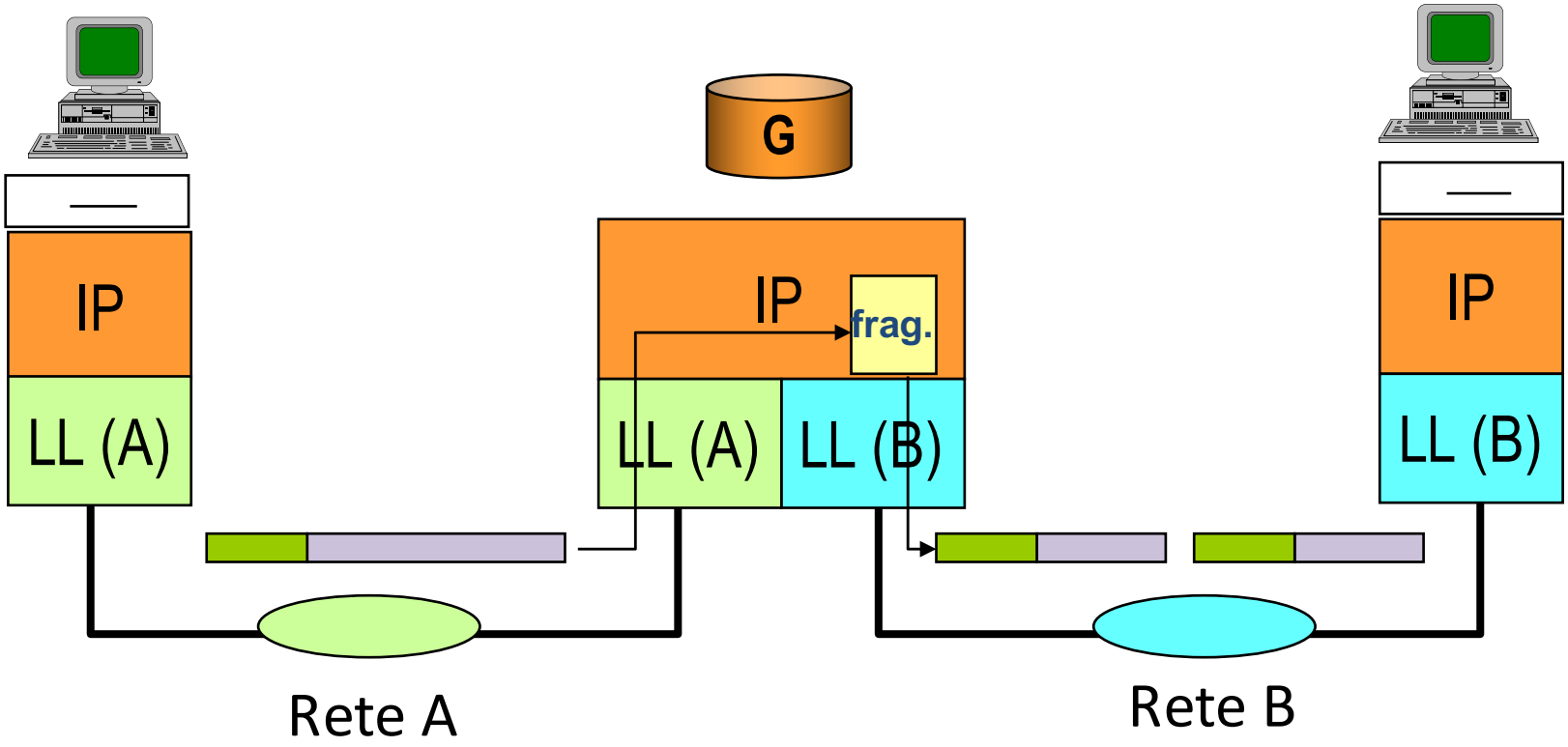


Frammentazione e ricostruzione

- I link della rete impongono un limite alla dimensione delle trame di livello 2 detto MTU (Max. Transfer Unit)
  - Diversi link hanno diversi MTU
- Questo costringe a dividere un datagram IP troppo lungo per stare in una sola trama in più "frammenti", che vengono riassemblati dalla destinazione



La Frammentazione

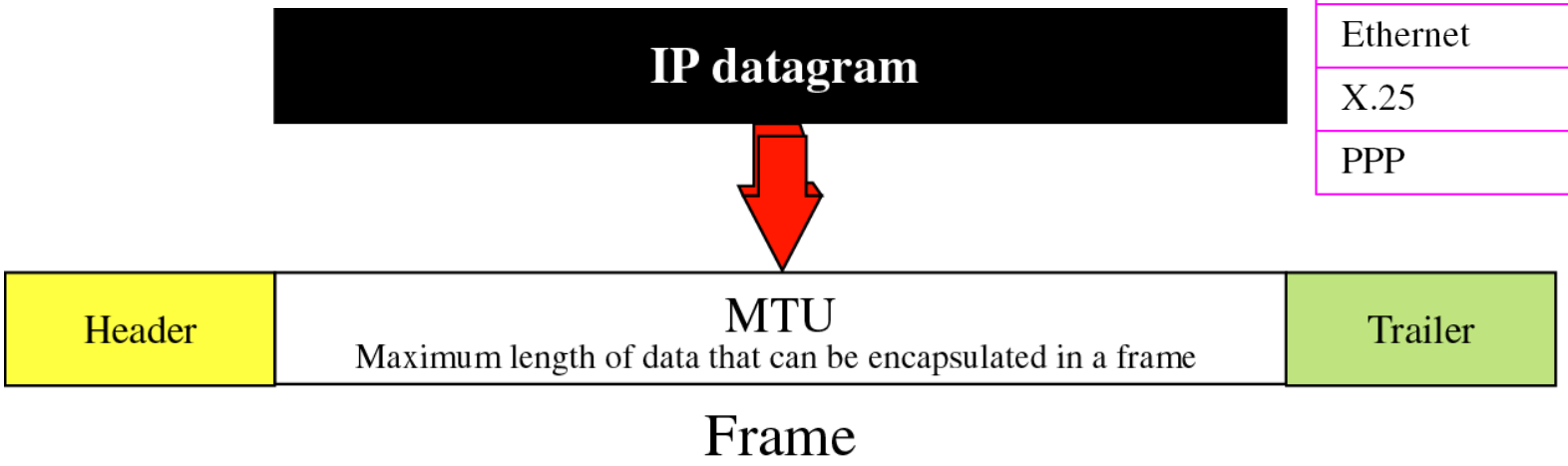


Frammentazione e ricostruzione

- Il pacchetto IP è trasportato in una trama
  - La limitazione sul payload massimo della trama (MTU) può provocare la frammentazione
  - In tal caso, è richiesta la ricostruzione alla destinazione

Lunghezza Ethernet frame: [64, 1518] byte
(lungh. min 64 byte, header = 18 byte)

Protocol	*MTU*
Hyperchannel	65,535
Token Ring (16 Mbps)	17,914
Token Ring (4 Mbps)	4,464
FDDI	4,352
Ethernet	1,500
X.25	576
PPP	296



La Frammentazione (3)

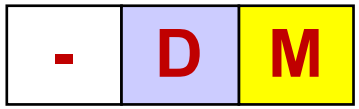
- Prima di passare il pacchetto al livello inferiore IP divide il pacchetto in frammenti ciascuno con il proprio header
- Un frammento di un pacchetto può essere frammentato ulteriormente lungo il cammino
- I frammenti verranno ricomposti dall'entità IP del destinatario indicato nell'header IP (frammenti di uno stesso pacchetto possono seguire diversi percorsi)
- I campi *Identification, Flags e Frag. Offset* sono usati per questo scopo

I campi usati per la frammentazione (1)

- Identification (16 bit)
  - E' un campo che identifica tutti i frammenti di uno stesso pacchetto in modo univoco. E' scelto dall'IP che effettua la frammentazione
- Frag. Offset (13 bit)
  - I byte del pacchetto originale sono numerati da 0 al valore della lunghezza totale. Il campo *Frag. Offset* di ogni frammento riporta il numero di sequenza del primo byte del frammento (in parole da 8 byte).
  - *esempio:* se un pacchetto di 2000 byte viene diviso in due da 1000 il primo frammento avrà un Frag *Offset* pari a 0 e il secondo pari a 1000/8

I campi usati per la frammentazione (2)

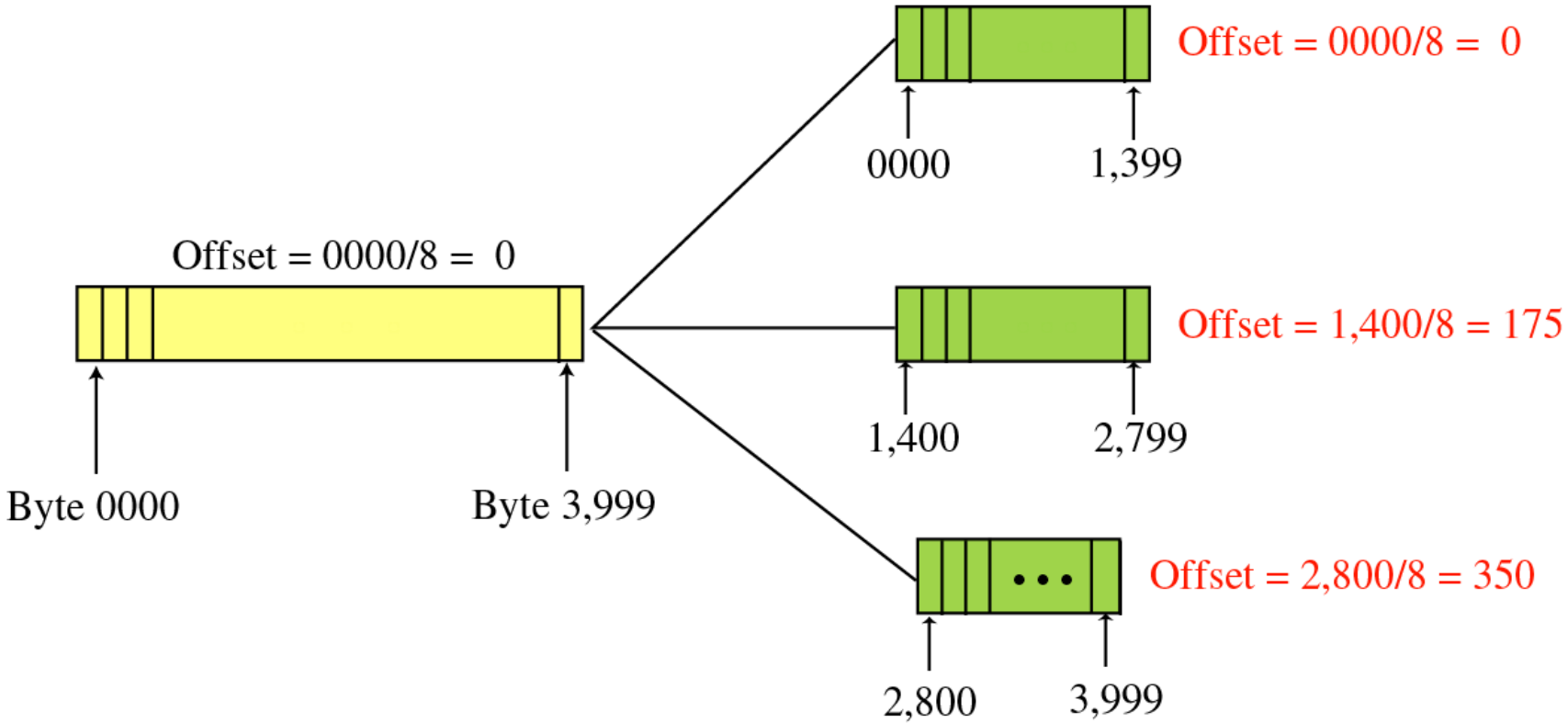
- Flags



  - Il bit M (*More*) è pari a 0 solo nell'ultimo frammento
  - Il bit D (*Do not fragment*) viene posto a 1 quando non si vuole che lungo il percorso venga applicata la frammentazione
    - In questo caso, se la frammentazione fosse necessaria, il pacchetto sarebbe scartato e verrebbe generato un messaggio di errore

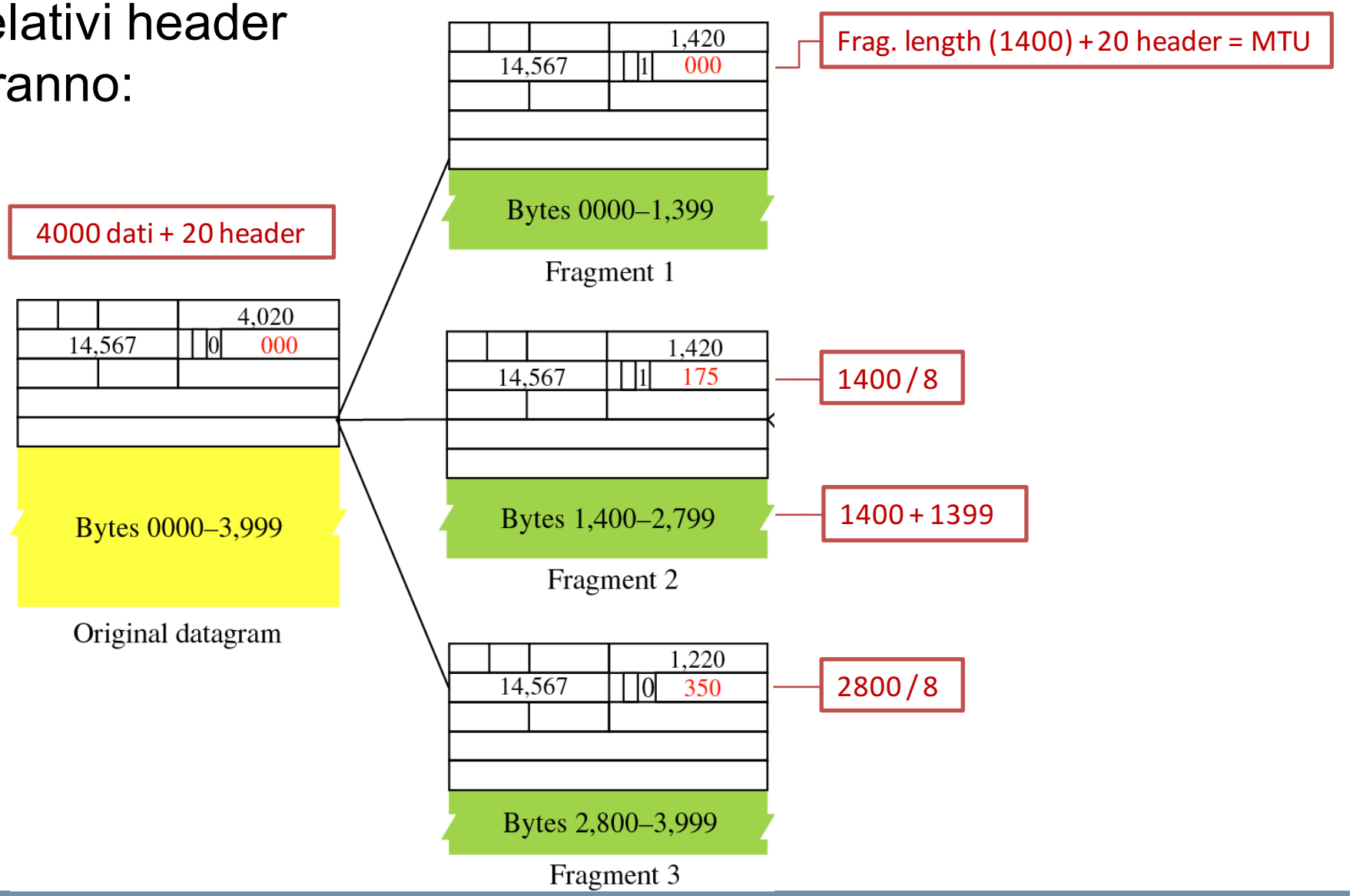
Esempio frammentazione (1)

- Se MTU = 1420 byte
- Il massimo payload, se non ci sono options, è:
  - MTU – max(header) = 1400 byte



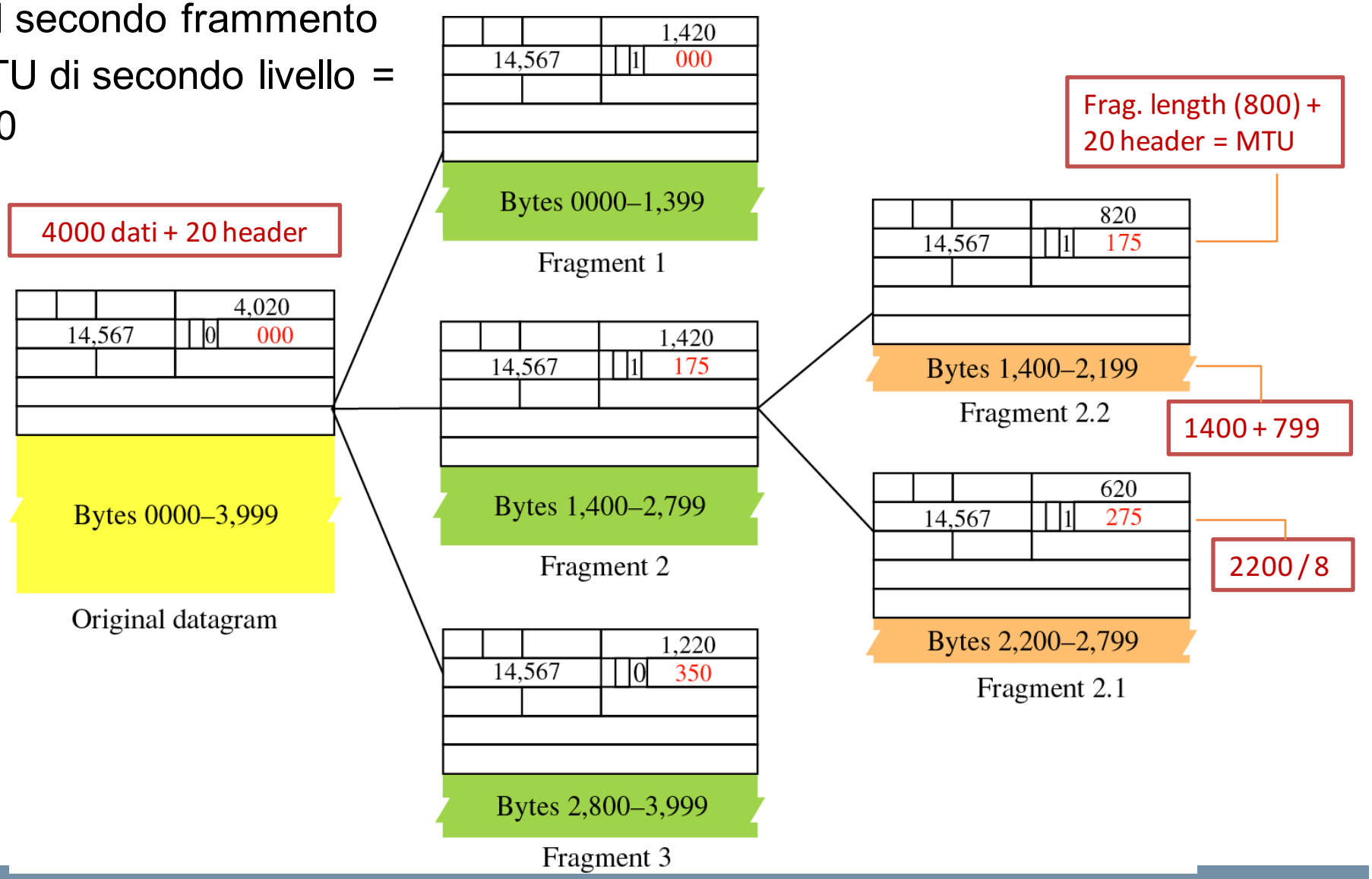
Esempio frammentazione (2)

- I relativi header saranno:



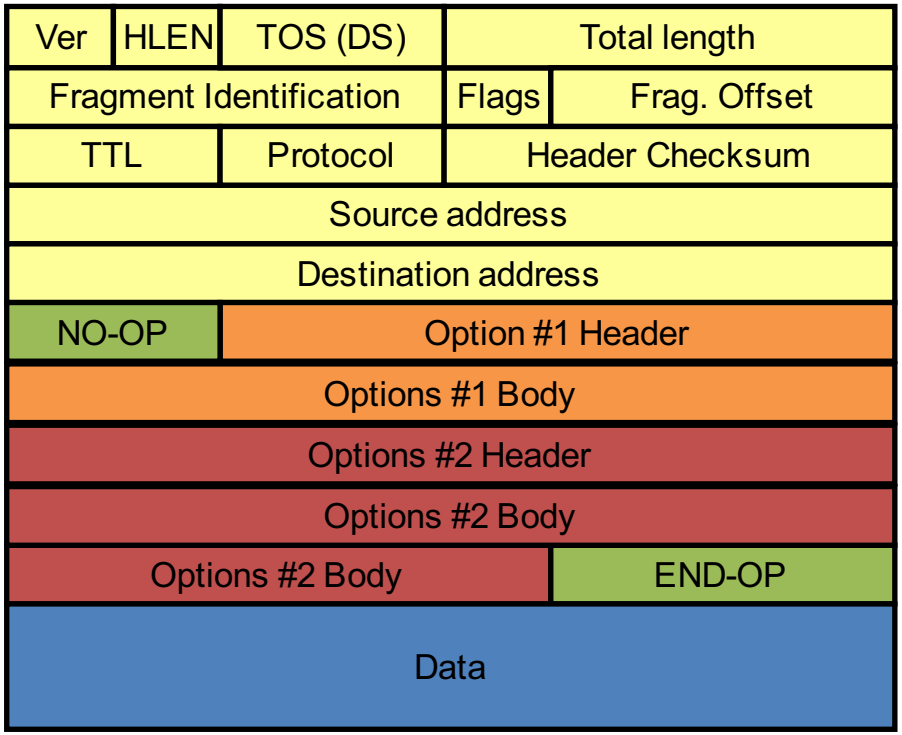
Esempio frammentazione (3)

- Ulteriore frammentazione del secondo frammento
- MTU di secondo livello = 820

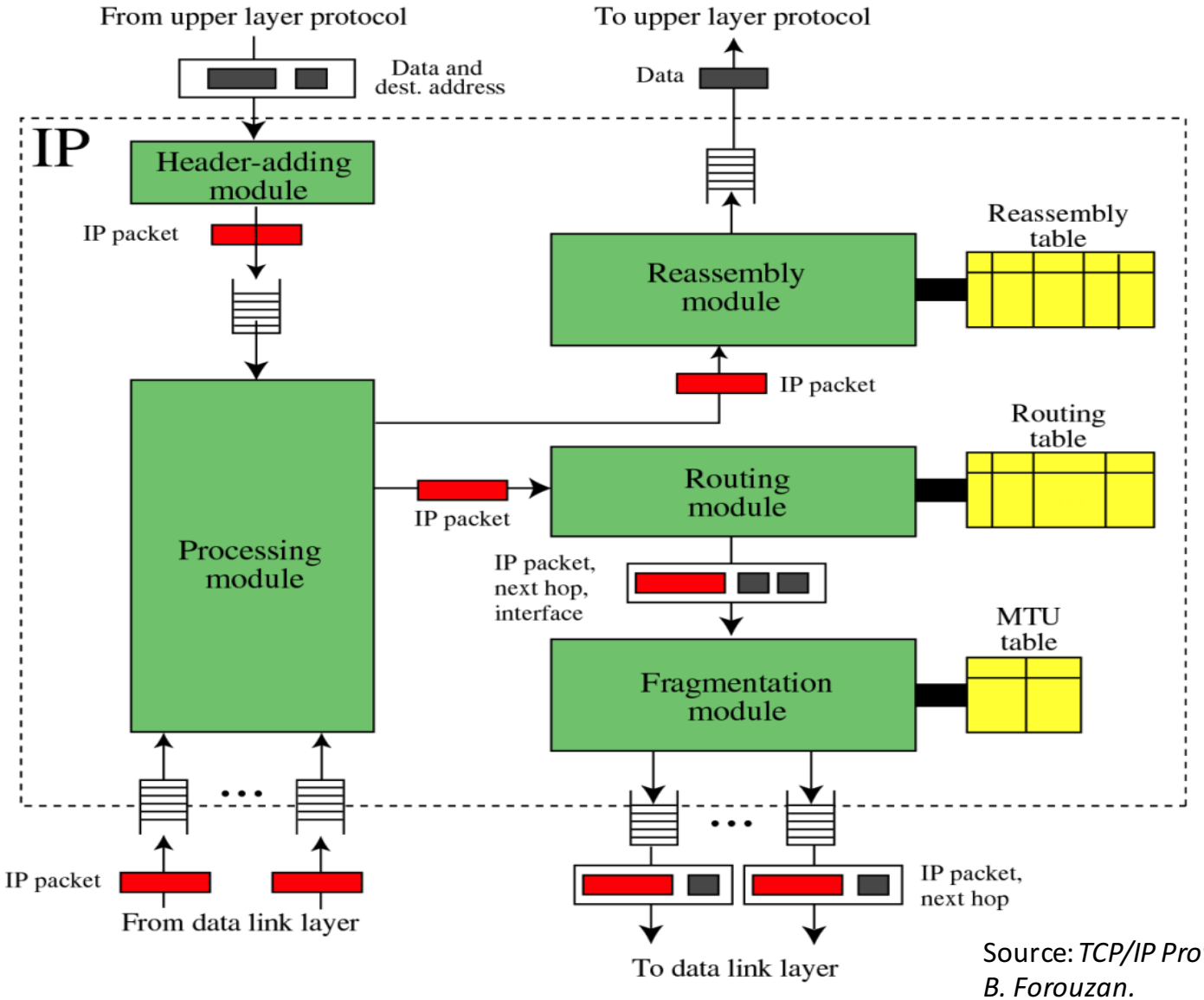


Le Opzioni

- La parte iniziale dell'header IP è di 20 byte ed è sempre presente
- Campi opzionali possono allungare l'header fino ad un massimo di 60 byte
- Originariamente pensate per *Testing e Debugging*
  - Record Route, Timestamp, Source Routing, etc.
- Ora raramente usate, anzi tipicamente filtrate dai router perché considerate a rischio sicurezza



Struttura Implementativa protocollo IP



Source: *TCP/IP Protocol Suite, B. Forouzan.*

Agenda

- Il protocollo IPv4
- Protocolli di gestione di IP
  - ICMP
  - ARP e RARP
  - DHCP
- Network Address Translation (NAT)

Protocolli accessori

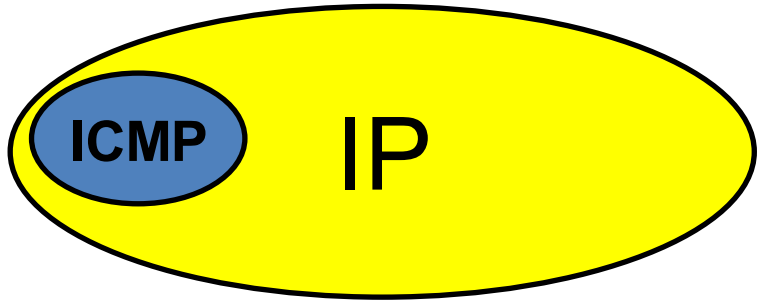
- Il protocollo IP richiede alcune funzioni accessorie di gestione e controllo che vengono svolte da apposite protocolli
  - Internet Control Message Protocol (ICMP): RFC 792
    - Trasferisce messaggi di segnalazione tra router e host (o host e host)
  - Address Resolution Protocol (ARP): RFC 826
    - Usato per scoprire gli indirizzi di livello 2 associate agli indirizzi IP nell'ambito di una rete IP
  - Reverse Address Resolution Protocol (RARP): RFC 903
    - Usato per scoprire l'indirizzo IP da parte di un host che si connette ad una rete IP
  - Dynamic Host Configuration Protocol (DHCP): RFC 2131
    - Usato per assegnare dinamicamente gli indirizzi agli host di una rete

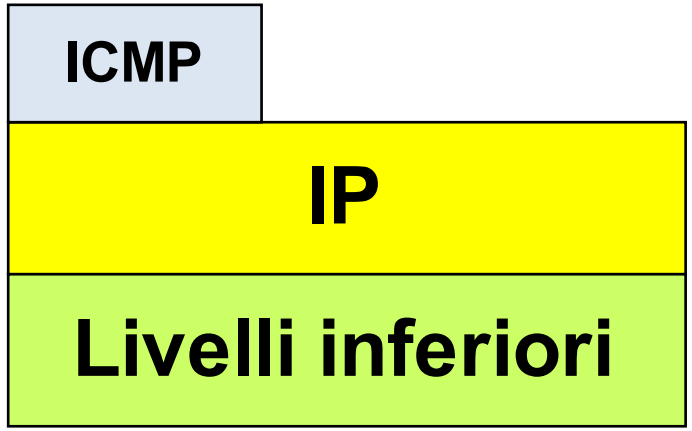
Agenda

- Il protocollo IPv4
- Protocolli di gestione di IP
  - ICMP
  - ARP e RARP
  - DHCP
- Network Address Translation (NAT)

Internet Control Message Protocol (ICMP)

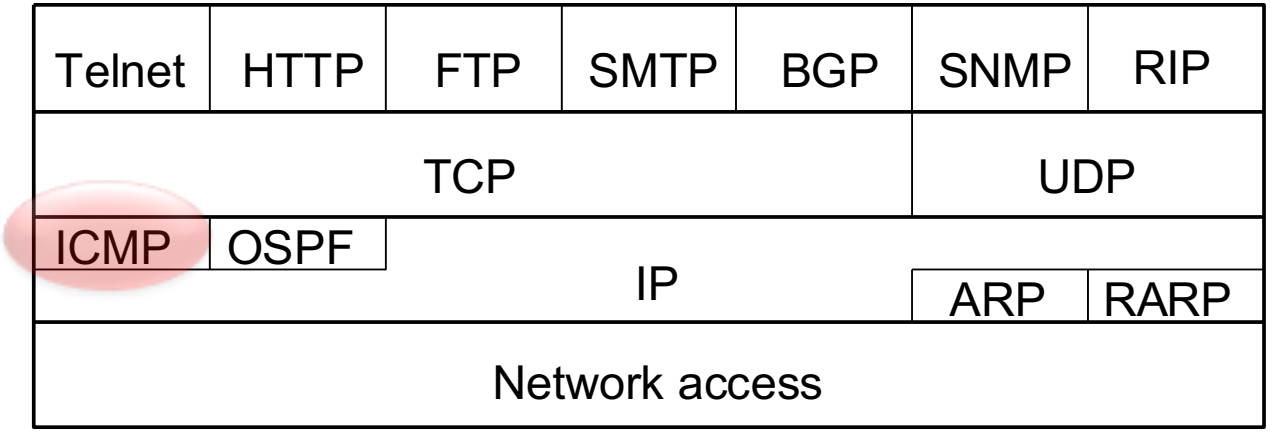
- E' un protocollo per messaggi di servizio fra host e router per informazioni su errori e fasi di attraversamento della rete
  - Da questo punto di vista può essere considerato come parte di IP
- I messaggi ICMP sono incapsulati e trasportati da IP
  - Da questo punto di vista può essere considerato un utente di IP



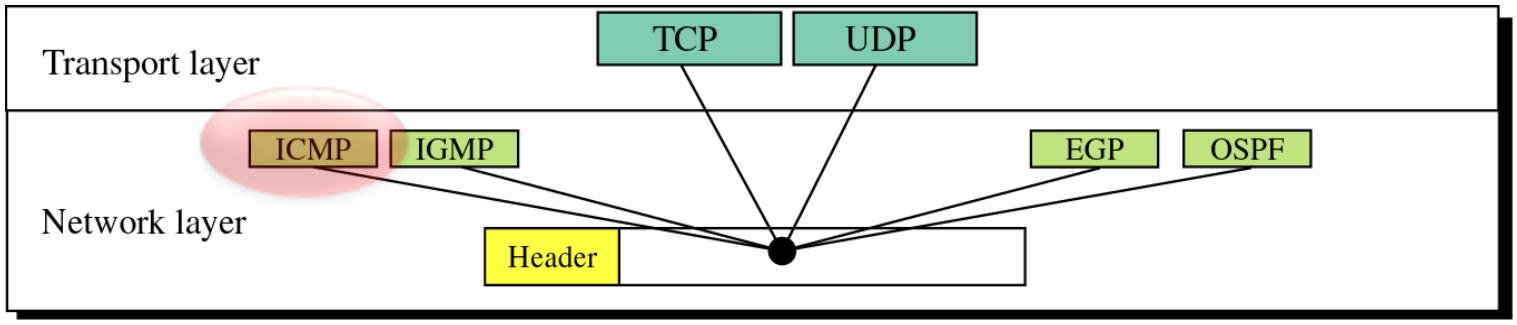


Internet Control Message Protocol (ICMP)

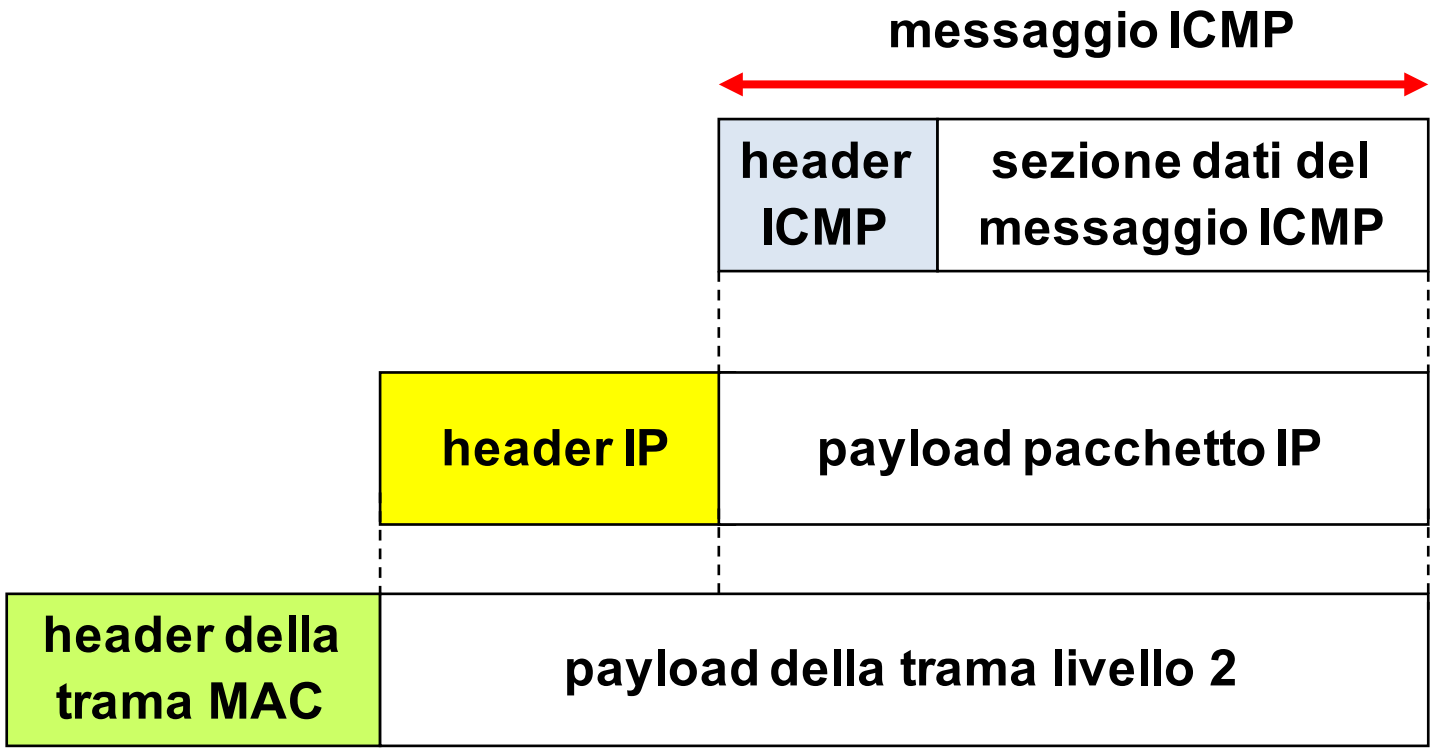
- TCP/IP protocol stack
-



- Violazione dei principi dell'architettura a strati

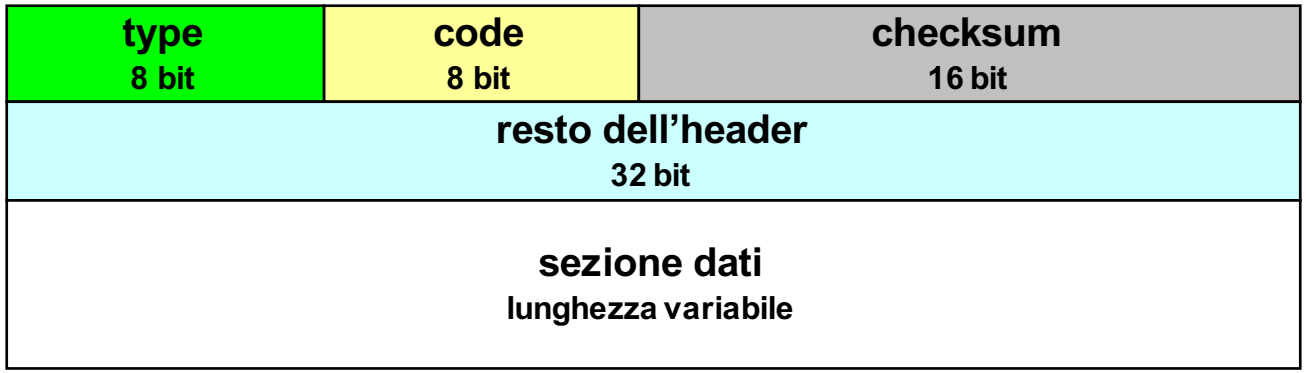


Internet Control Message Protocol (ICMP)



- Nel pacchetto IP il campo Protocol indica il codice dell'ICMP
- I messaggio ICMP viaggia all'interno del pacchetto IP

Formato e tipi di messaggio



- **Error Reporting**
  - Destination Unreachable (type 3)
  - Source Quench (type 4)
  - Time Exceeded (type 11)
  - Parameter Problem (type 12)
  - Redirection (type 5)
- **Query**
  - Echo Request/Reply (type 8,0)
  - Timestamp Request/Reply (type 13/14)
  - Address Mask Request/Reply (type 17/18)
  - Router Solicitation/Advertisment (type 10/9)

Funzionalità di Error Reporting

- ICMP non corregge errori, ma si limita a segnalarli.
- L'evento errore è notificato alla sorgente del pacchetto IP che lo ha causato
- Eventi gestiti
  - *Destination Unreachable* (type 3)
  - *Source Quench* (type 4)
  - *Time Exceeded* (type 11)
  - *Parameter Problem* (type 12)
  - *Redirection* (type 5)

- I messaggi di errore contengono l'header del pacchetto IP che li ha generati e i suoi primi 8 byte di dati

Destination Unreachable

type (3)	code (0-12)	checksum
non usato (0)		
header + primi 64 bit del pacchetto IP che ha causato il problema		

- Quando un router scarta un pacchetto per qualche motivo normalmente genera un messaggio di errore che invia alla sorgente del pacchetto
- Nel campo code è codificato il motivo che ha causato l'errore
- Ovviamente la generazione del messaggio avviene solo nei casi in cui il router può accorgersi del problema

Destination unreachable

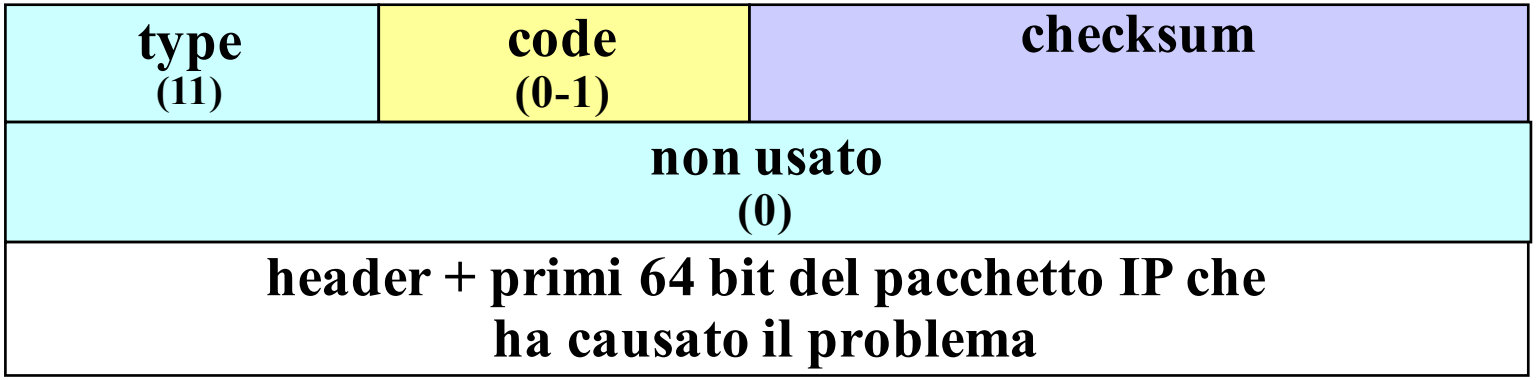
type (3)	code (0-12)	checksum
non usato (0)		
header + primi 64 bit del pacchetto IP che ha causato il problema		

Alcuni Code:

- 0 network unreachable
- 1 host unreachable
- 2 protocol unreachable
- 3 port unreachable
- 4 fragmentation needed and DF set
- 5 source route failed
- ...

Generato solo da un router in cui non è presente una riga nella tabella di routing per la rete di destinazione e non è definito un default router

Time exceeded



- Code 0 (inviato dai router)
  - Il messaggio di time exceeded viene usato quando il router decrementando il TTL lo pone a 0
  - Il messaggio di time exceeded viene inviato alla sorgente del pacchetto
- Code 1 (inviato dalla destinazione)
  - viene usato dalla destinazione quando non tutti i frammenti di un pacchetto arrivano entro un tempo massimo

Parameter problem

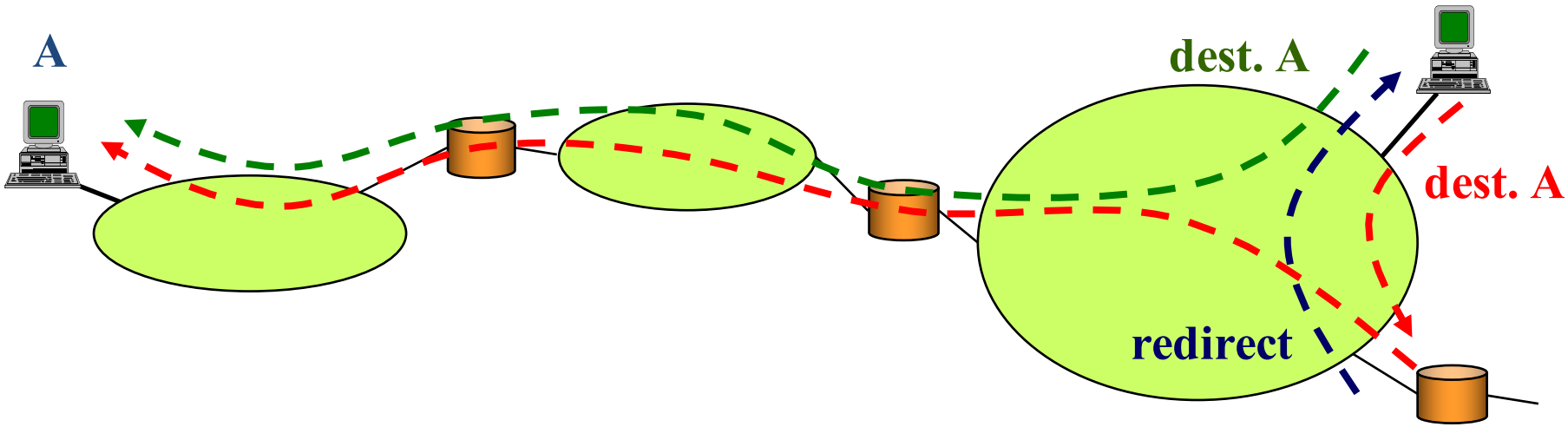
type (12)	code (0-1)	checksum
pointer	non usato (0)	
header + primi 64 bit del pacchetto IP che ha causato il problema		

- Code 0
  - se l'header di un pacchetto IP ha una incongruenza in qualcuno dei suoi campi viene inviato il messaggio di parameter problem; il campo pointer punta al byte del pacchetto che ha causato il problema
- Code 1
  - viene usato quando un'opzione non è implementata o qualche parte del campo Options manca

Redirect

type (5)	code (0-3)	checksum
indirizzo IP del router		
header + primi 64 bit del pacchetto IP		

- Questo messaggio viene usato quando si vuole che la sorgente usi per quella destinazione un diverso router

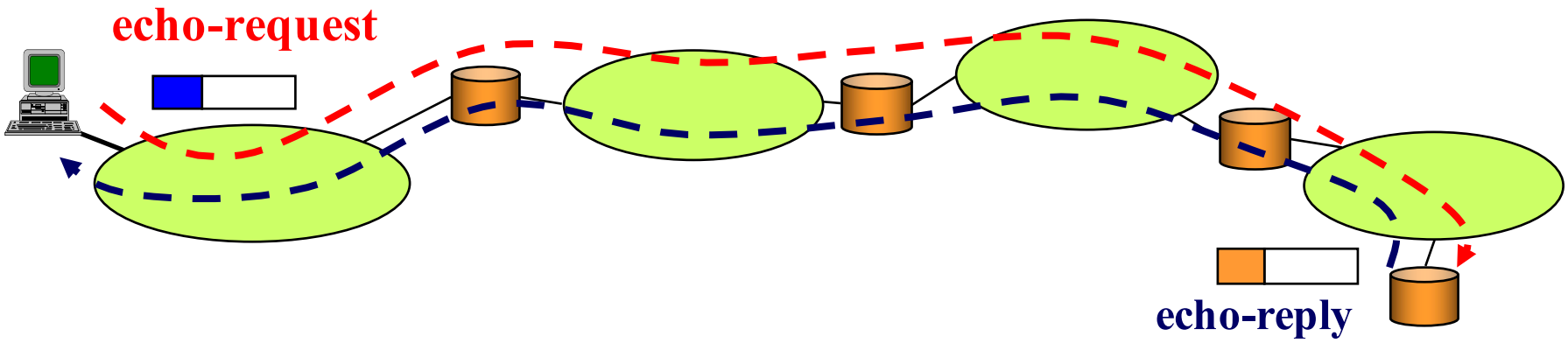


Funzionalità di diagnostica

- Coppie di messaggi secondo il paradigma domanda/risposta
- Tipi di messaggi:
  - *Echo Request/Reply* (type 8/0)
  - *Timestamp Request/Reply* (type 13/14)
  - *Address Mask Request/Reply* (type 17/18)
  - *Router Solicitation/Advertisment* (type 10/9)

Funzionalità di Echo

- I messaggi di *Echo-request* e *Echo-reply* sono usati per verificare la raggiungibilità e lo stato di un host o un router
- Quando un nodo IP riceve un messaggio di *Echo-request* risponde immediatamente con un messaggio di *Echo reply*

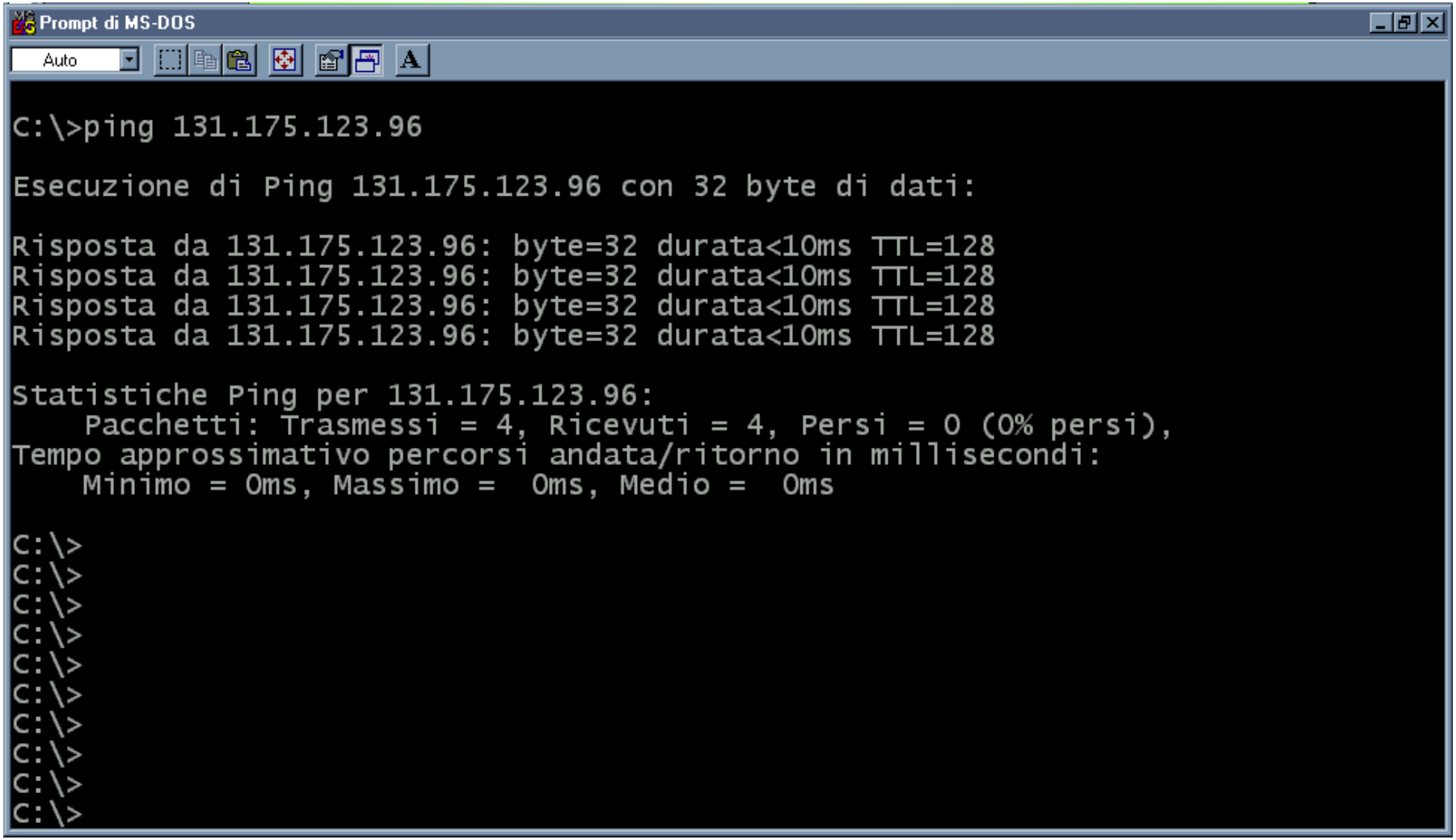


Messaggi Echo

type (8 request, 0 reply)	code (0)	checksum
identifier		sequence number
optional data		

- Il campo *identifier* viene scelto dal mittente della richiesta
- Nella risposta viene ripetuto lo stesso *identifier* della richiesta
- Più richieste consecutive possono avere lo stesso *identifier* e differire per il *sequence number*
- Una sequenza arbitraria può essere aggiunta dal mittente nel campo optional data e deve essere riportata uguale nella risposta
  - Confrontando sequenza inviata e ricevuta il mittente può contare eventuali errori

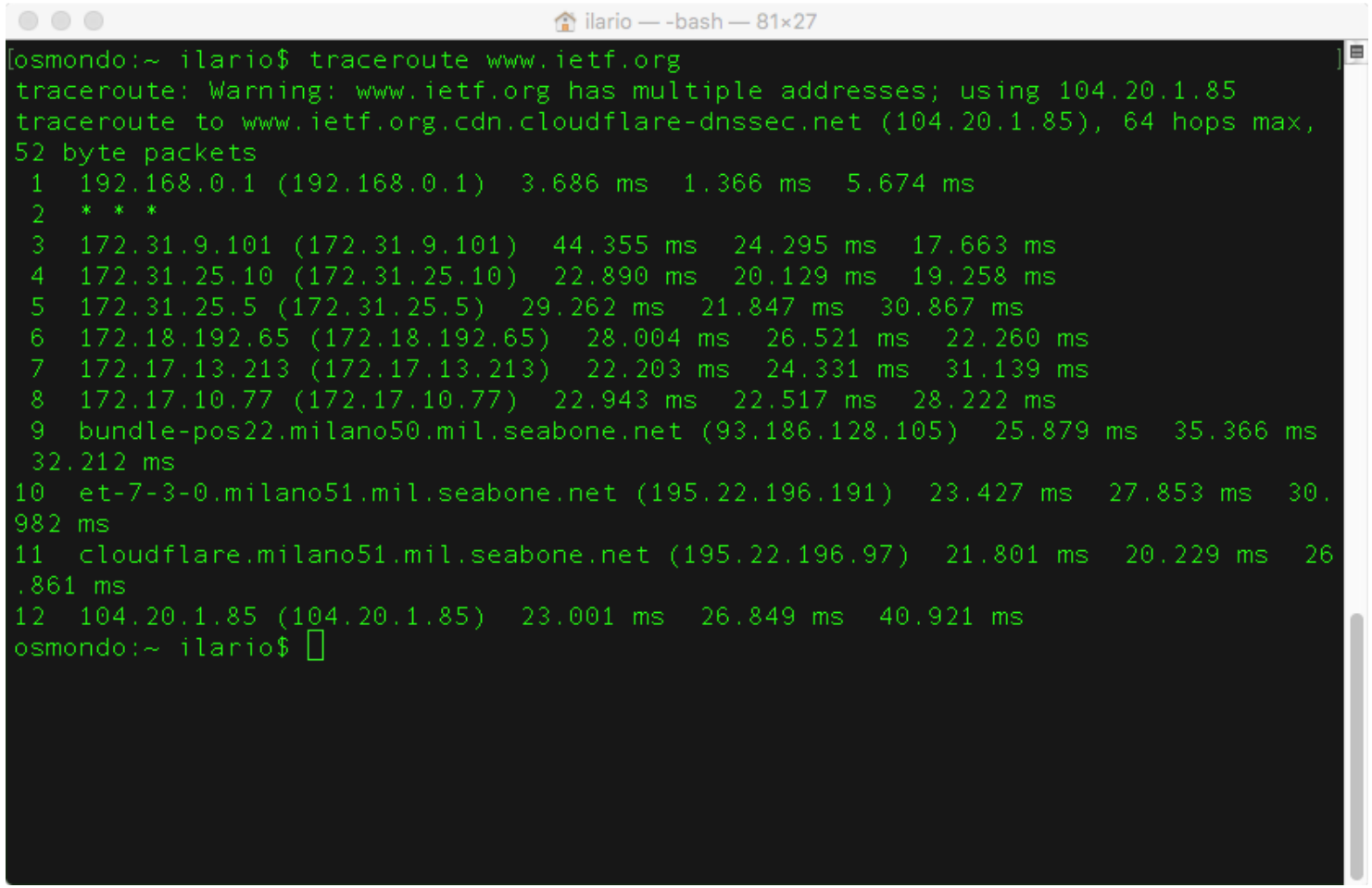
Uso Messaggi di Echo: PING



Wireshark: Ping

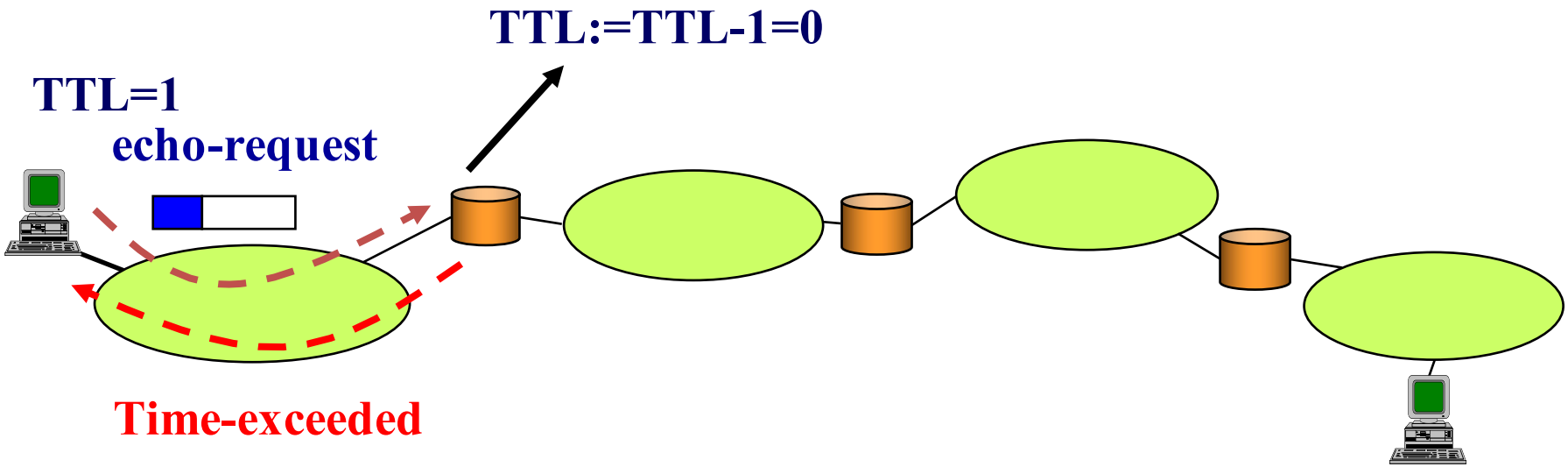
- File cattura : `icmp-ethereal-trace-1`
- Attività:
  - Controlliamo indirizzi IP e numeri di porta
    - Perché non ci sono numeri di porta?
  - Quali valori del campo Identifier e Sequence Number hanno i pacchetti della sequenza?
  - Consideriamo un coppia di ICMP request/reply
    - Come variano i campi Type e Code
    - Come variano i campi Identifier e Sequence Number

Uso dei messaggi ICMP: applicativo di traceroute



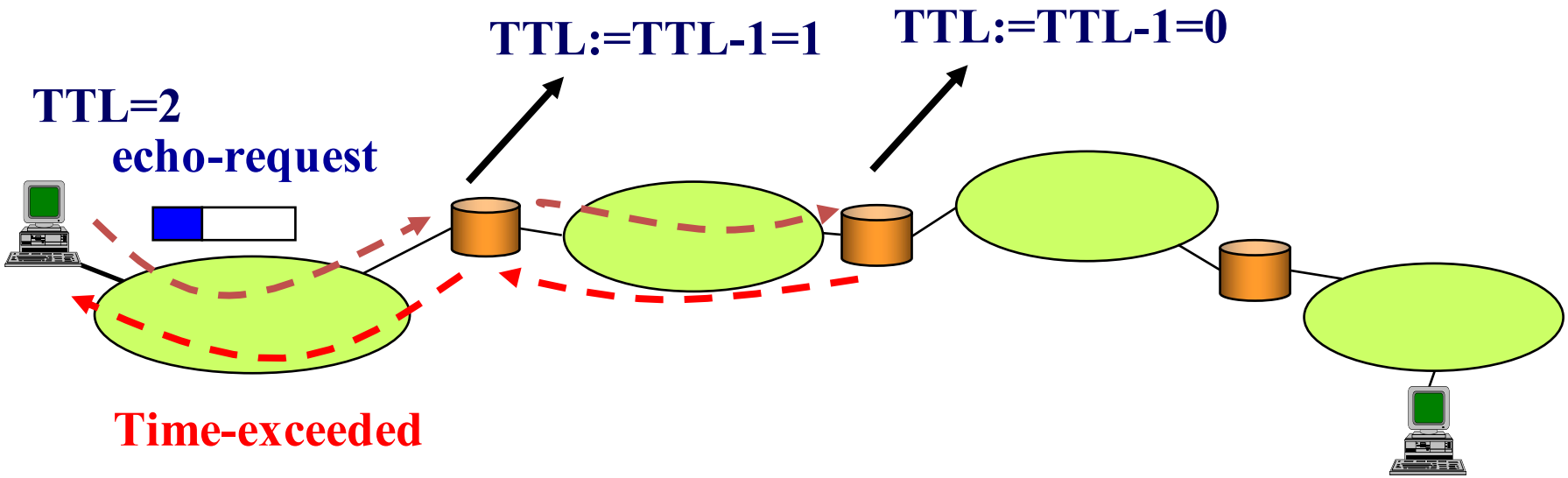
Traceroute: come funziona?

- Il *traceroute* usa (normalmente) messaggi di *Echo-request* verso la destinazione (notare: non un router, ma un host!)
- I primo messaggio ha il TTL=1



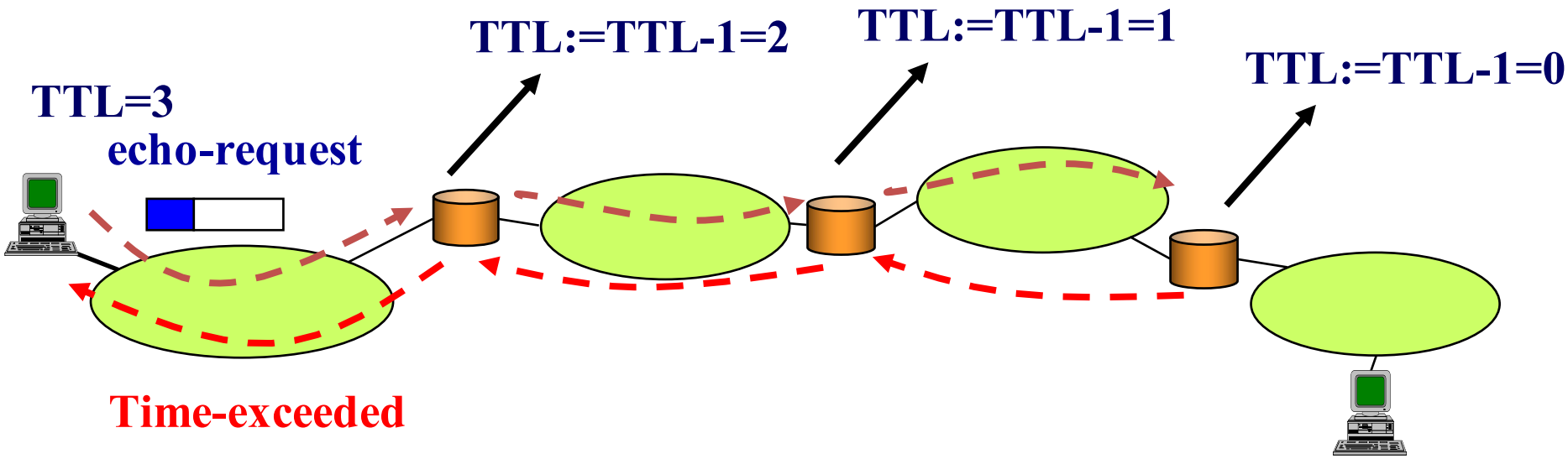
Traceroute: come funziona?

- I secondo messaggio ha il TTL=2



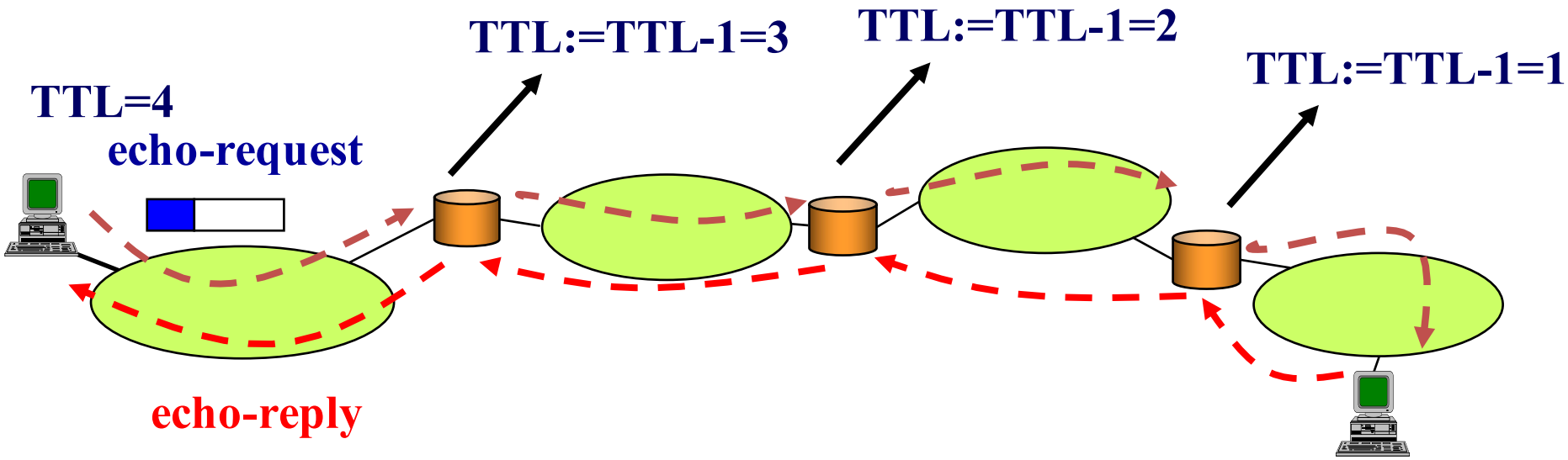
Traceroute: come funziona?

- I terzo messaggio ha il TTL=3, e così via …



Traceroute: come funziona?

- Alla fine il destinatario risponderà con un'Echo reply e così il mittente sa di aver esplorato tutta la via



Traceroute: come funziona?

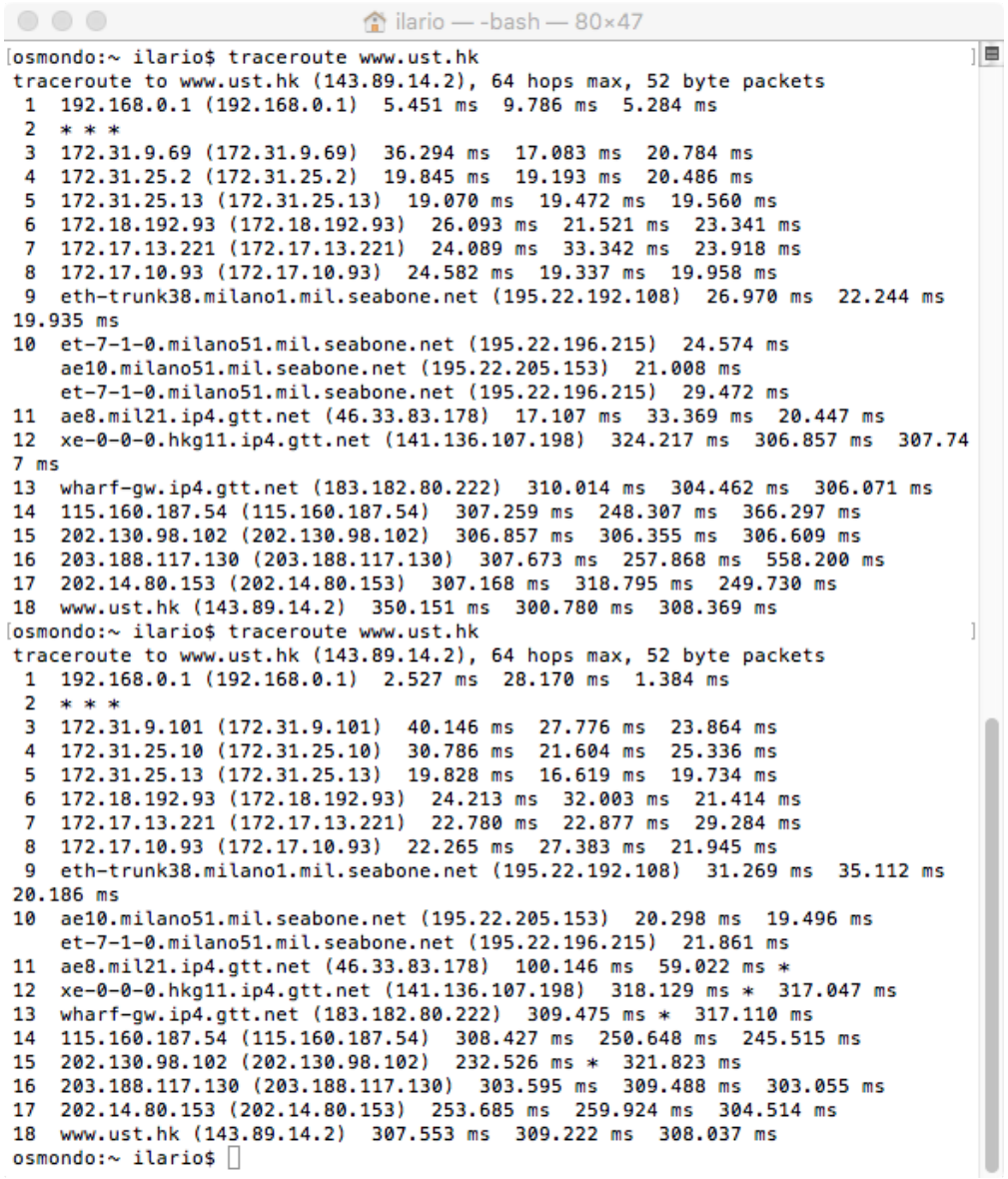
- *Alternativa:* E' possibile usare pacchetti UDP invece degli ECHO requests.
  - I pacchetti vengono inviati con TTL decrescente al destinatario, su una porta particolare raramente usata.
    - Vengono scartati dai router ed inviata un messaggio Time-exceeded
  - Quando il destinatario riceve il pacchetto, lo scarta perché la porta non è usata e si genera un messaggio ICMP Port Unreachable che torna al mittente.
    - Il messaggio svolge lo stesso ruolo dell'ECHO reply

- *Nota:* TTL interviene solo nel momento in cui si deve fare un forward. La regola è:
  - al momento di forwardare il pacchetto decrementa TTL;
  - mai forwardare un pacchetto con TTL = 0.
- Quindi un host può accettare un pacchetto con TTL = 0, se è la sua destinazione finale (accade se la destinazione è sulla stessa rete della sorgente).

Wireshark: Traceroute

- File cattura : `ip-ethereal-trace-2`
- Attività:
  - Quanti messaggi ICMP vengono inviati per ogni valore di TTL ?
  - Esaminiamo le ultime 3 coppie di pacchetti.
    - Perché sono diversi dagli altri?
  - Confrontiamo con la cattura: `traceroute-unix`
    - Quali numeri di porta vengono usati?
    - Come cambia il campo Protocol del header IP?
    - A cosa servono le richieste DNS? (Si confronti con lo screenshot nella slide successiva
    - Come termina la sequenza di messaggi?

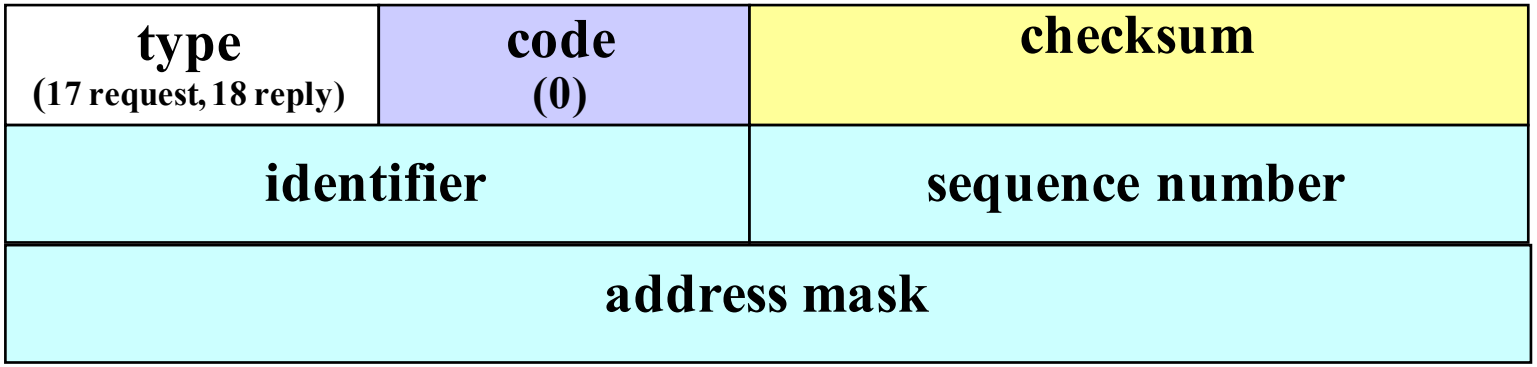
Wireshark: Screenshot traceroute-unix



Wireshark: Traceroute

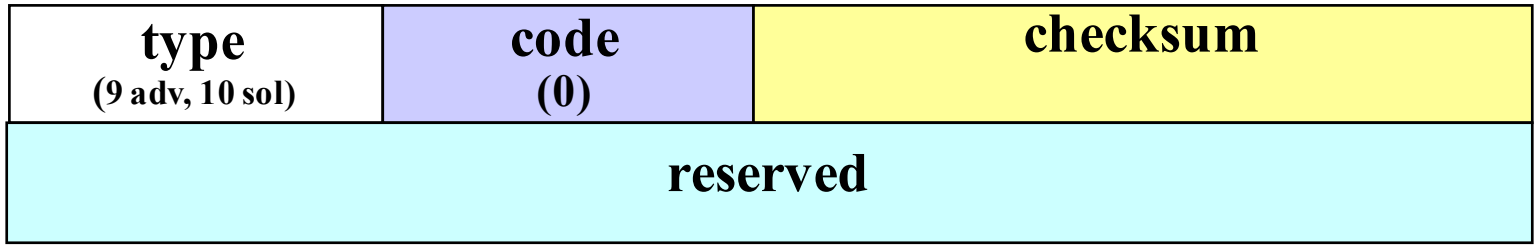
- File cattura : `ip-ethereal-trace-1`
- Attività:
  - Quanto sono lunghi i messaggi di ICMP a partire dal pacchetto #8?
  - Quanto sono lunghi i messaggi di ICMP a partire dal pacchetto #93?
    - Cosa succede al messaggio ICMP Echo Request?
    - Si analizzino i campi dell'header IP dei frammenti.
  - Quanto sono lunghi i messaggi di ICMP a partire dal pacchetto #218?
    - Cosa succede al messaggio ICMP Echo Request?
    - Quanto è lungo il messaggio ICMP Echo Reply di questa sequenza?

Address mask request e reply



- Un host invia un messaggio *Address Mask Request* al/ai router della rete locale per conoscere la propria netmask.
- Un router risponde con un messaggio *Address Mask Reply* che contiene la subnet mask per la rete locale.
- Il campo address mask viene riempito dal destinatario
- Attualmente poco usato perché sostituito da DHCP

Router solicitation e advertisement



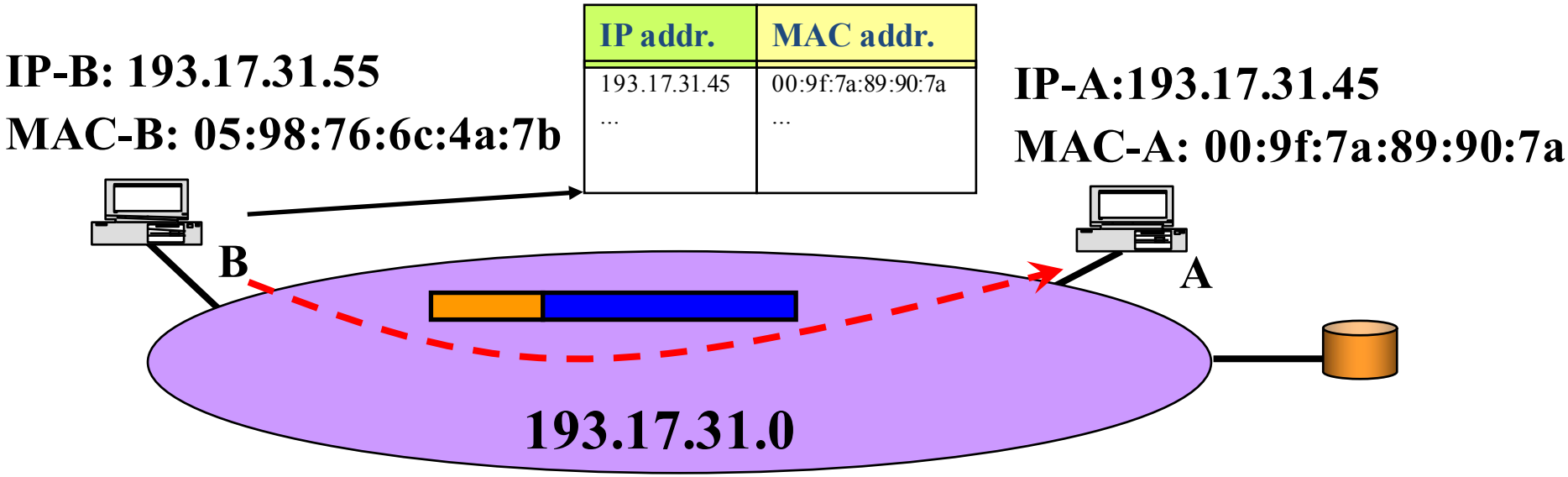
- I router inviano messaggi di *Router Advertisement* per annunciare la loro presenza in rete e fornire informazioni utili, come l'indirizzo IP delle interfacce
- L'invio è periodico e configurabile dall'amministratore del router (7-10 minuti)
- Gli host rimangono in attesa dei messaggi *Router Advertisement*
  - Possono essere sollecitati con messaggi di *Router Solicitation* inviati sulla rete locale
- Attualmente poco usato perché sostituito da DHCP

Agenda

- Il protocollo IPv4
- Protocolli di gestione di IP
  - ICMP
  - ARP e RARP
  - DHCP
- Network Address Translation (NAT)

Indirizzi IP e indirizzi fisici

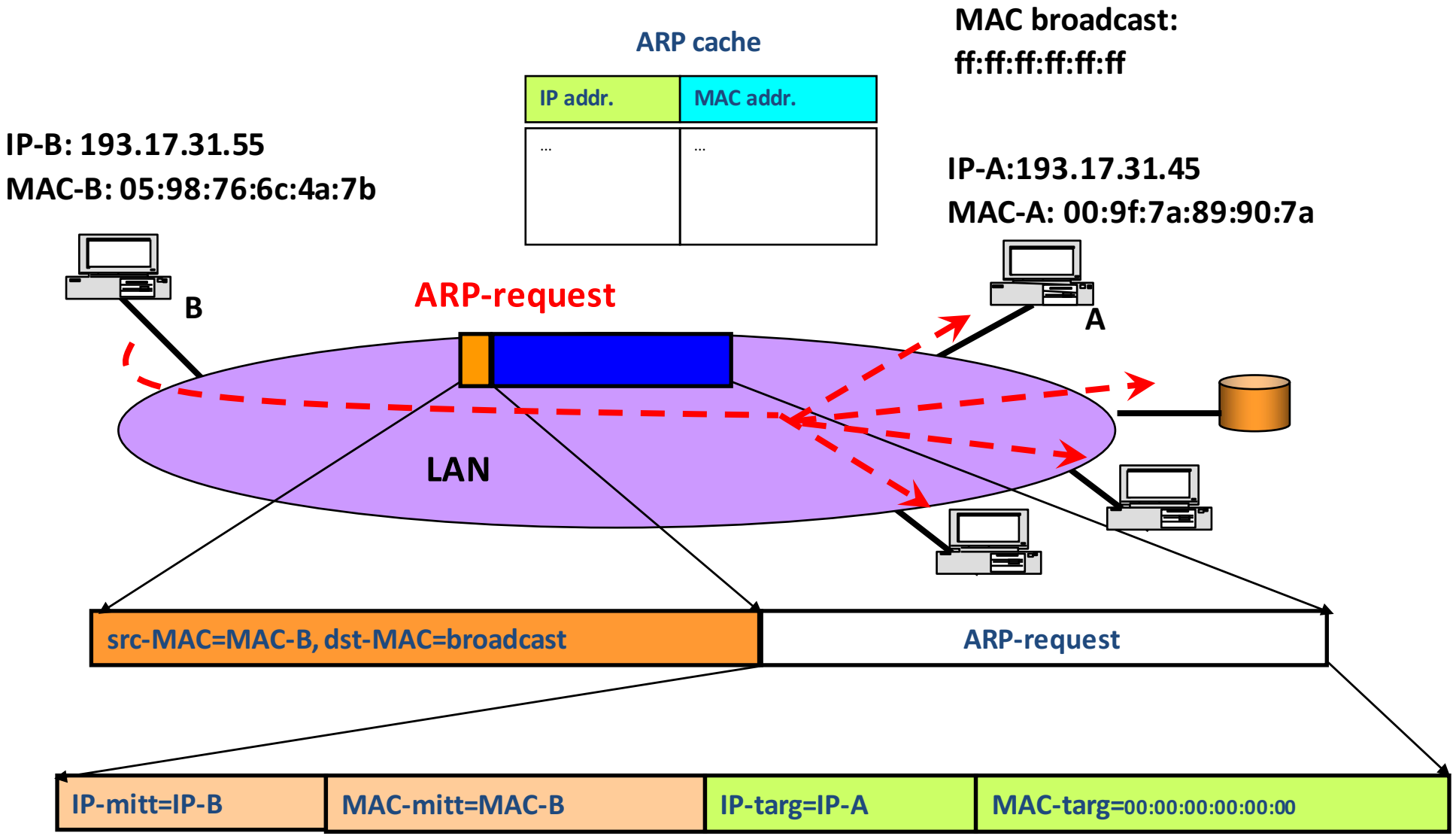
- Illustrando le tecniche di inoltro abbiamo ipotizzato la presenza di una tabella di corrispondenza tra indirizzi IP e indirizzi di livello inferiore (indirizzi fisici)
- Queste tabelle vengono create dinamicamente da ciascun host mediante il protocollo ARP



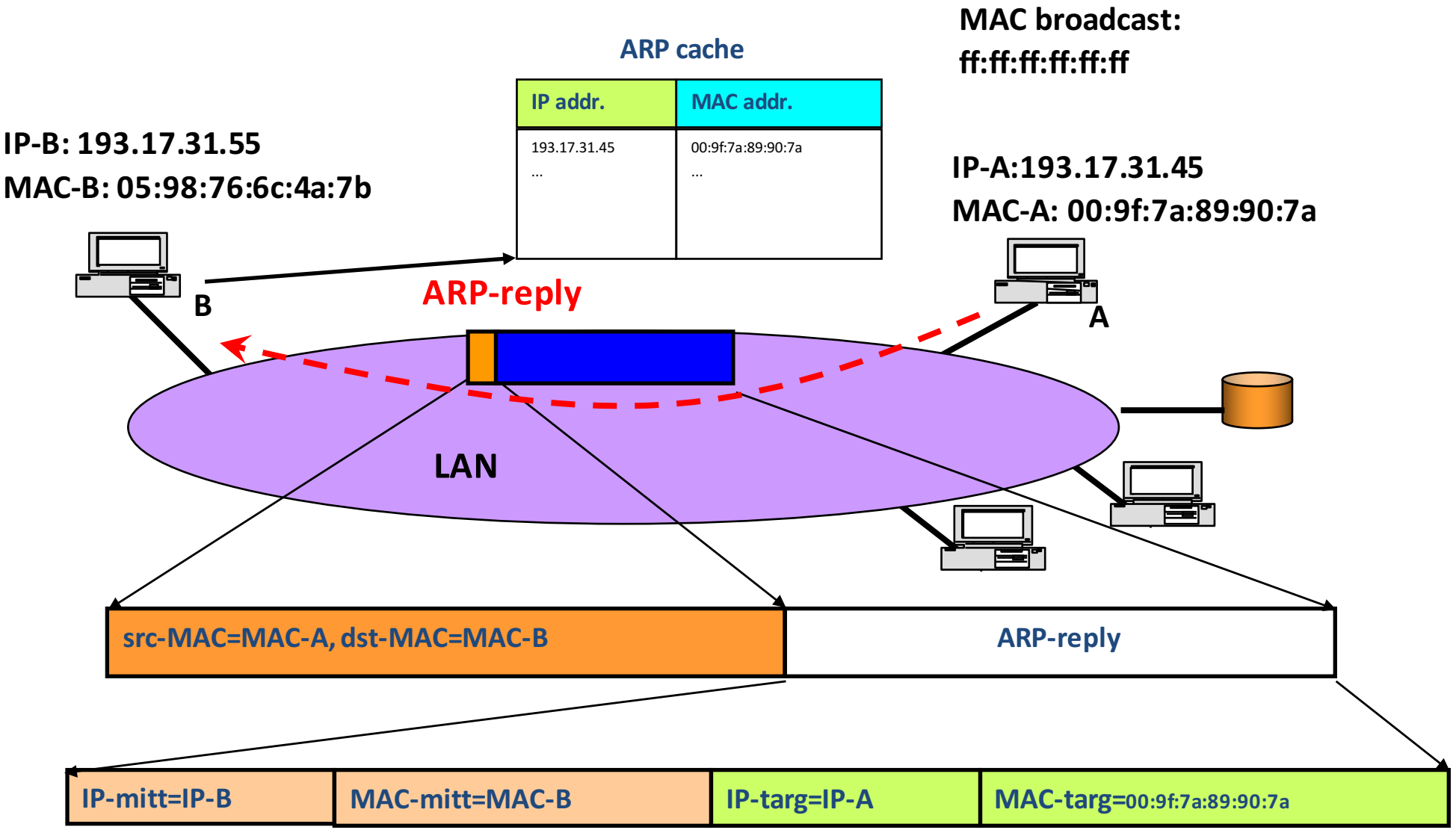
Address Resolution Protocol (ARP, RFC 826)

- Si basa sulla capacità di indirizzamento broadcast della rete locale
- Quando nella tabella memorizzata nell'host (denominata *ARP-cache*) non è presente l'indirizzo MAC cercato viene generato un messaggio di *ARP-request*
- La *ARP-request* viene inviata in broadcast e contiene l'indirizzo IP di cui si chiede il corrispondente indirizzo MAC
- L'host che riconosce l'indirizzo IP come proprio invia una *ARP-reply* direttamente a chi aveva inviato la richiesta con l'indicazione del proprio indirizzo MAC

ARP (*Address Resolution Protocol*)

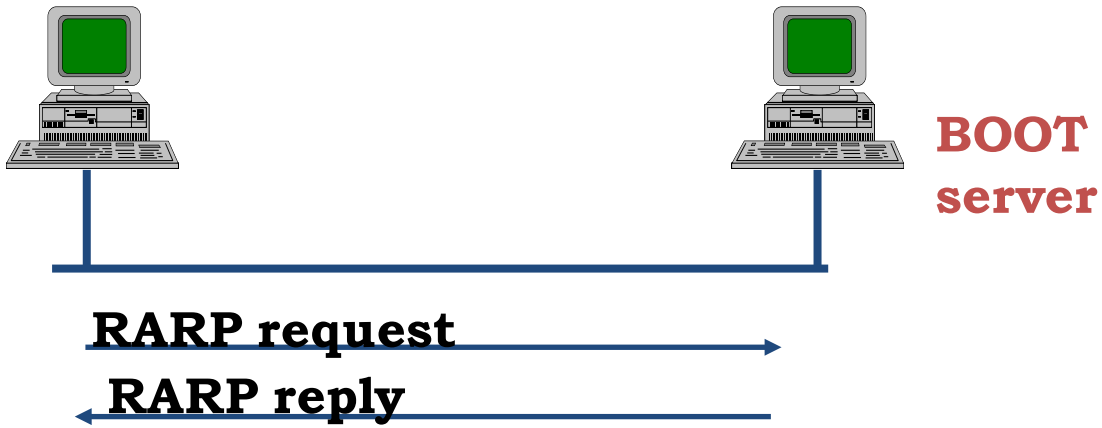


ARP (Address Resolution Protocol)



Assegnamento indirizzi IP - RARP (Reverse ARP)

- Il protocollo ARP consente di associare ad un indirizzo IP noto un indirizzo fisico non noto usando la capacità di broadcast della rete sottostante
- Il protocollo RARP (Reverse ARP) è in grado di effettuare l'operazione inversa:
  - Un host che conosce il proprio indirizzo fisico chiede di sapere il proprio indirizzo IP
  - Utile per macchine diskless che effettuano il bootstrap in rete
  - *Ma non è più usato perché sostituito da DHCP !!!*



Agenda

- Il protocollo IPv4
- Protocolli di gestione di IP
  - ICMP
  - ARP e RARP
  - DHCP
- Network Address Translation (NAT)

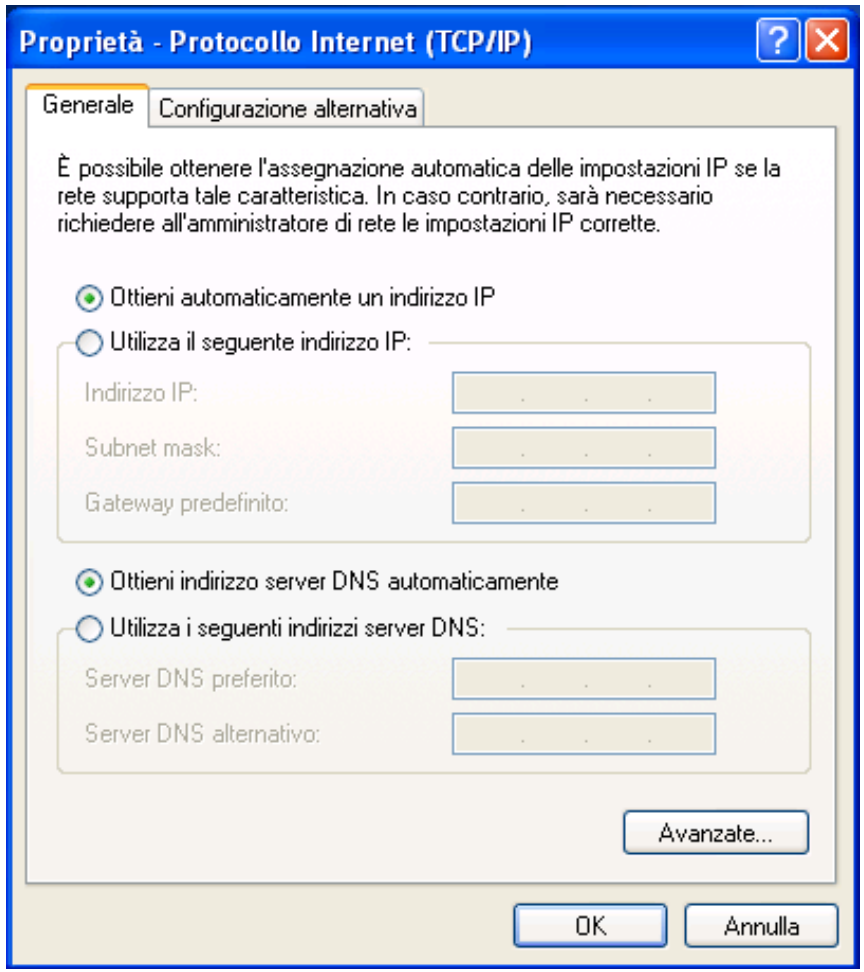
Gestione dinamica degli indirizzi

- Le procedure statiche di assegnamento degli indirizzi sono poco flessibili
- Può essere comodo non configurare i singoli host con l'indirizzo IP, ma usare un *server* per memorizzare tutte le configurazioni
- In molti casi non è necessario avere un'associazione stabile tra i due indirizzi ma si può usare un'associazione dinamica (più host degli indirizzi disponibili):
  - host spesso inattivi (es. collegamenti remoti con rete d'accesso telefonica)
  - host che usano IP solo per rari scambi di informazioni

Indirizzi dinamici

- Supponiamo di avere un *server* in grado di fornire l'indirizzo IP ad un *host* su richiesta
- Sono possibili diversi casi:
  - associazioni statica: il server ha una tabella di corrispondenza tra indirizzi fisici e indirizzi IP e all'arrivo di una richiesta consulta la tabella e invia la risposta
  - associazione automatica: la procedura di corrispondenza nella tabella è automatizzata dal server
  - associazione dinamica: l'insieme di indirizzi IP è più piccolo degli host che possono usarlo

Indirizzi dinamici

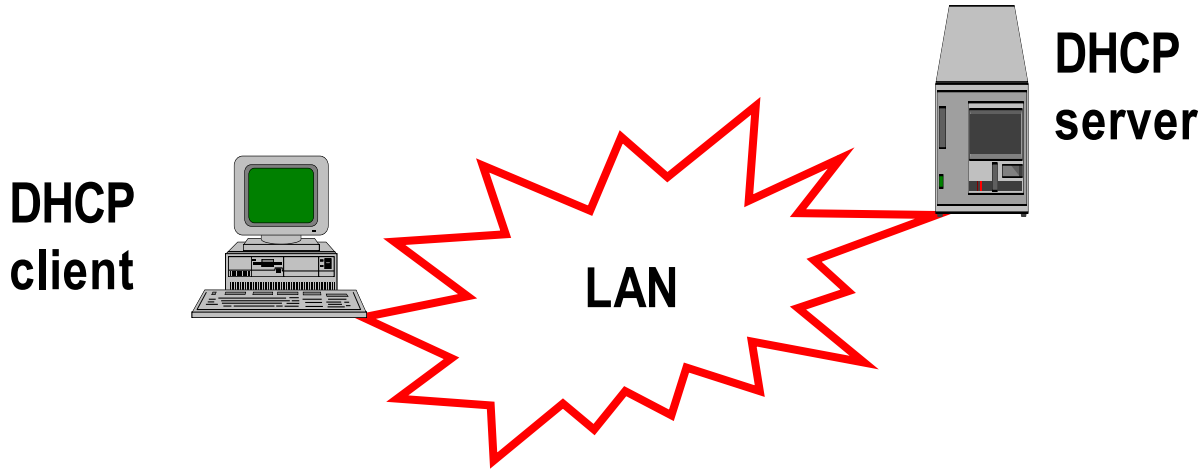


Associazione Dinamica

- Il caso dell'allocazione dinamica è utile in situazioni nelle quali gli host non necessitano di avere sempre un indirizzo IP
- L'associazione deve essere temporanea (uso di *timeout* o procedure di rilascio esplicito)
- E' possibile che all'arrivo di una richiesta non vi siano indirizzi disponili (rifiuto della richiesta)
- Il dimensionamento del numero di indirizzi IP segue gli stessi principi del dimensionamento di un fascio di circuiti in telefonia

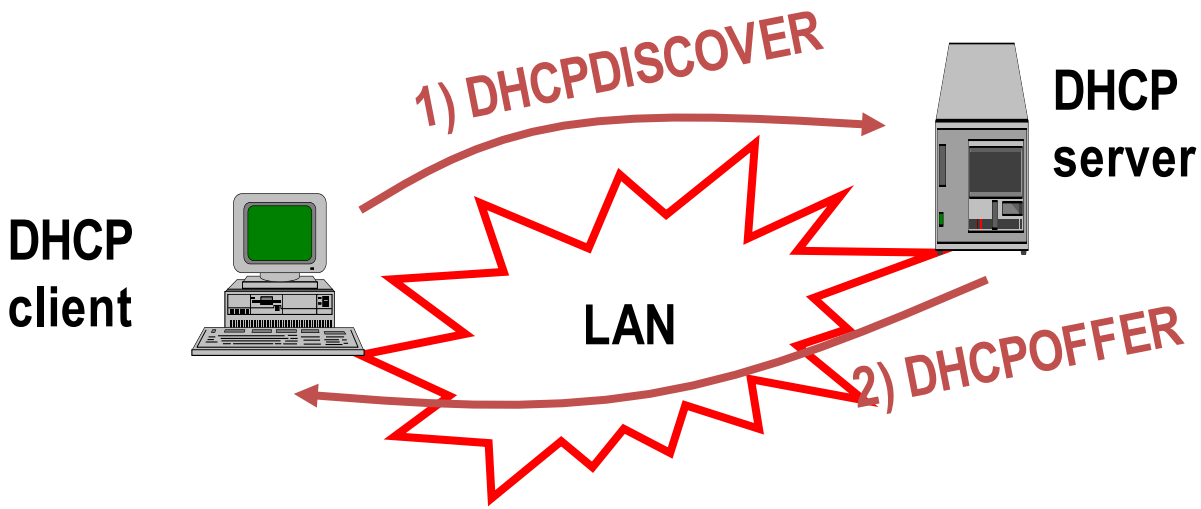
Dynamic Host Configuration Protocol (DHCP)

- Per la configurazione di indirizzi IP non si usa il RARP, ma un protocollo più evoluto derivato dal BOOTP
- E' un protocollo di tipo client-server detto DHCP (RFC 2131)



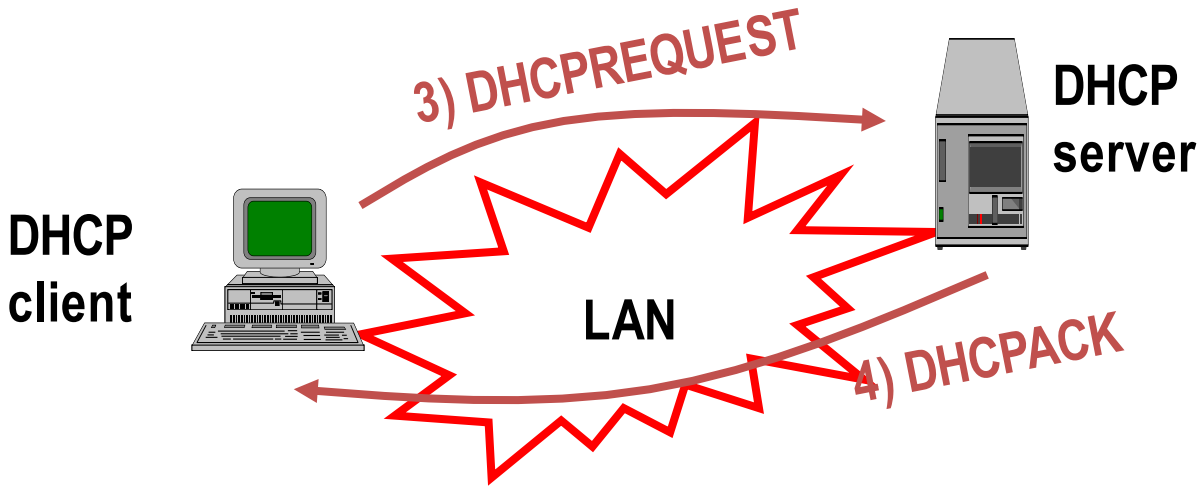
DHCP (1)

- Un client che deve configurare il proprio stack IP invia in broadcast un messaggio di DHCPDISCOVER contenete il proprio indirizzo fisico
- Il server risponde con un messaggio di DHCPOFFER contente un proprio identificativo e un indirizzo IP proposto



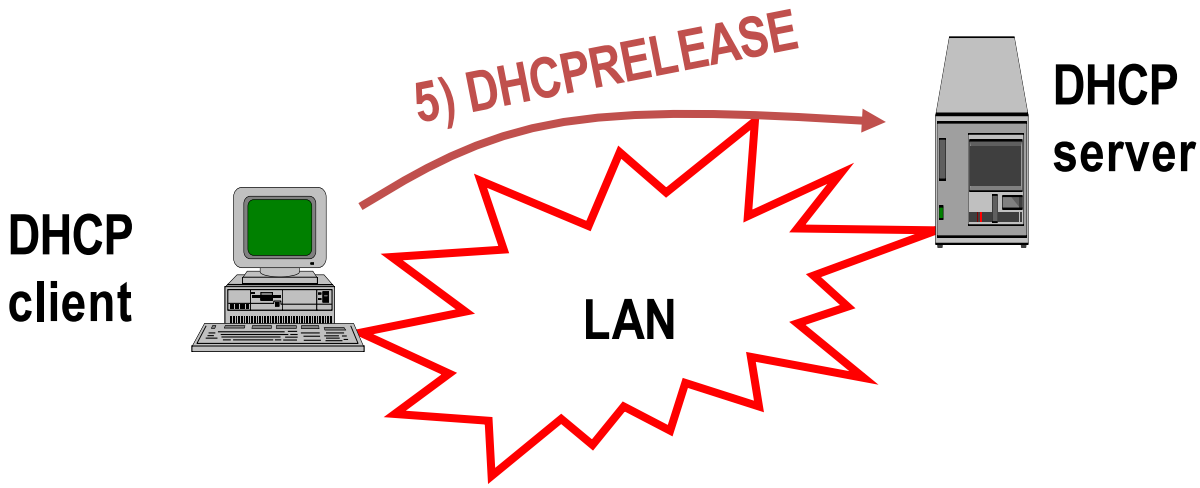
DHCP (2)

- Il client può accettare l'offerta inviando una DHCPREQUEST contenente l'identificativo del server (anche questo messaggio viene inviato in broadcast)
- Il server crea l'associazione con l'indirizzo IP e manda un messaggio di DHCPACK contenente tutte le informazioni di configurazione necessarie

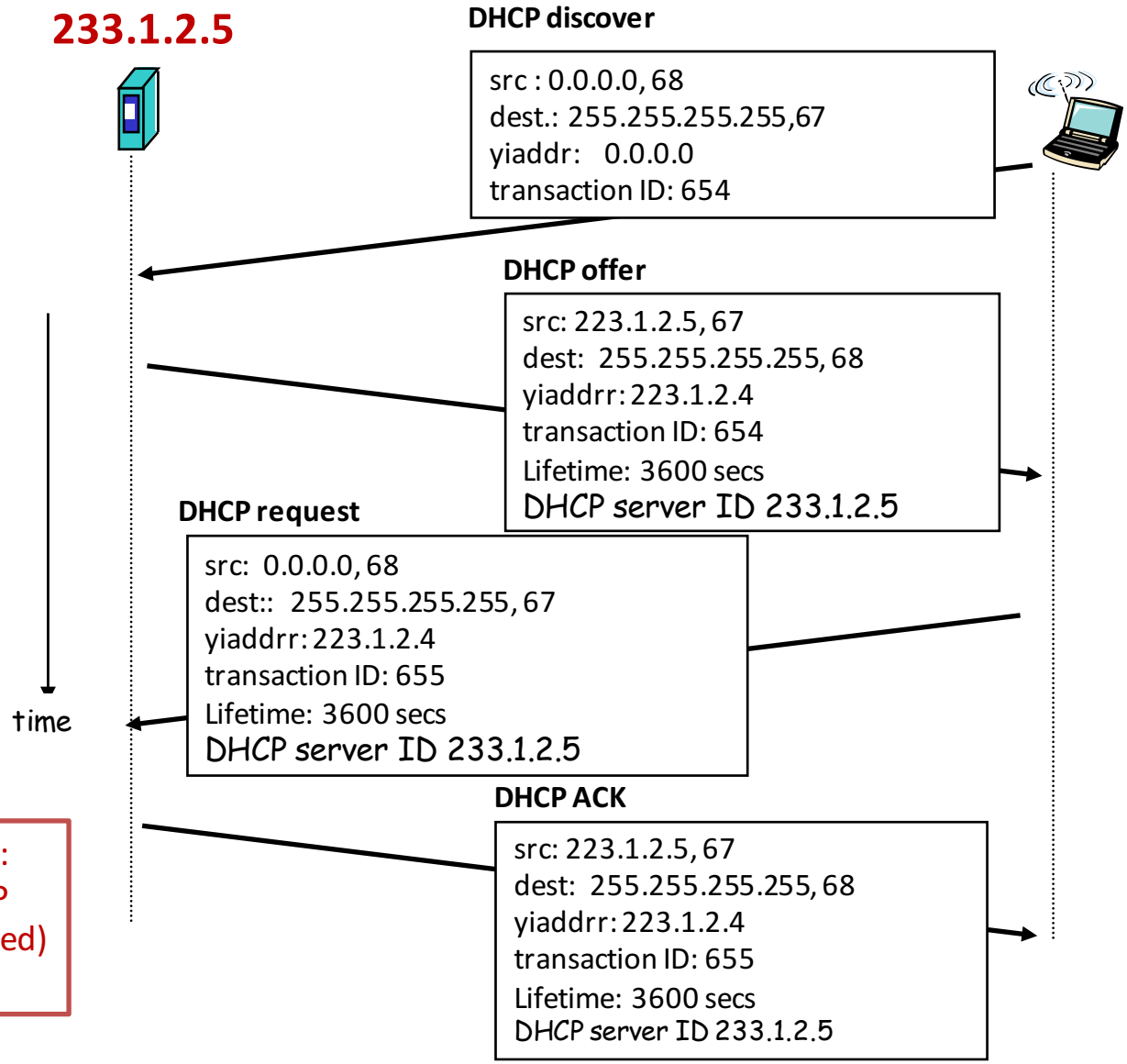


DHCP (3)

- Parametri di configurazione
  - IP address
  - Netmask
  - Gateway
  - DNS server
- Il rilascio dell'indirizzo avviene con l'invio di un messaggio di DHCPRELEASE da parte del client



Scambio di messaggi DHCP

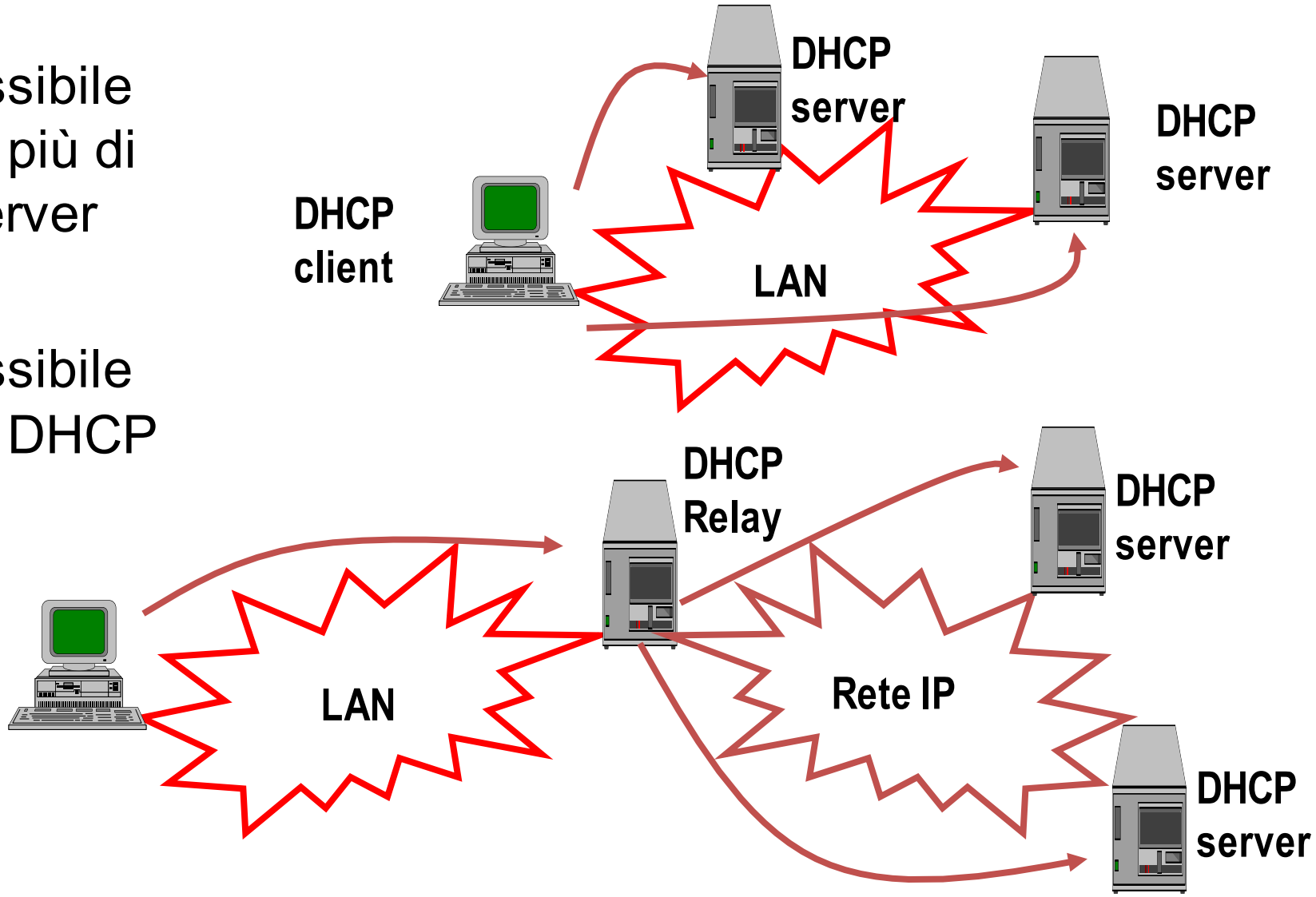


Prestare attenzione a:

- le porte per DHCP
- il broadcast (limited) a livello IP

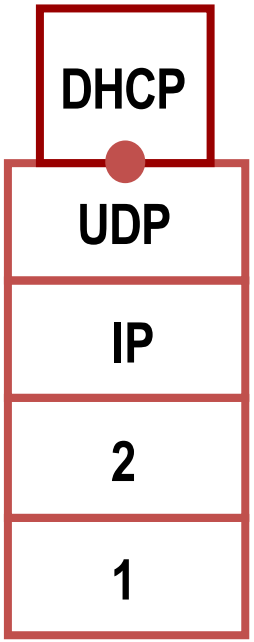
DHCP (4)

- E' possibile avere più di un server
- E' possibile usare DHCP Relay



Trasporto dei messaggi

- DHCP si appoggia su UDP per il trasporto dei messaggi
- I messaggi dei client fino all'assegnamento dell'indirizzo IP hanno:
  - ind. di sorgente: 0.0.0.0
  - ind. di destinazione: 255.255.255.255
  - porta sorgente: 68
  - porta destinazione: 67

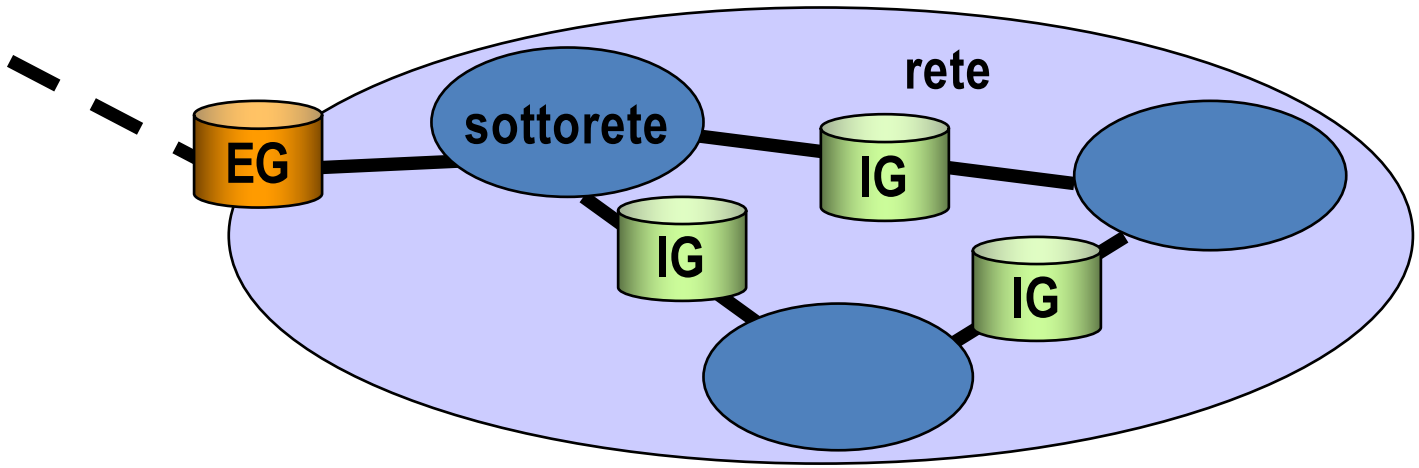


Agenda

- Il protocollo IPv4
- Protocolli di gestione di IP
  - ICMP
  - ARP e RARP
  - DHCP
- Network Address Translation (NAT)

Reti Private e *Intranet*

- Le reti private si sono evolute grazie alla tecnologia IP e sono passate da grandi reti collegate a livello 2 *(bridge)* a reti collegate con *router* IP
- Una *intranet* non è altro che una rete privata che utilizza tecnologia di interconnessione IP, dotata degli stessi servizi dell'INTERNET come server www, server di posta, ecc.

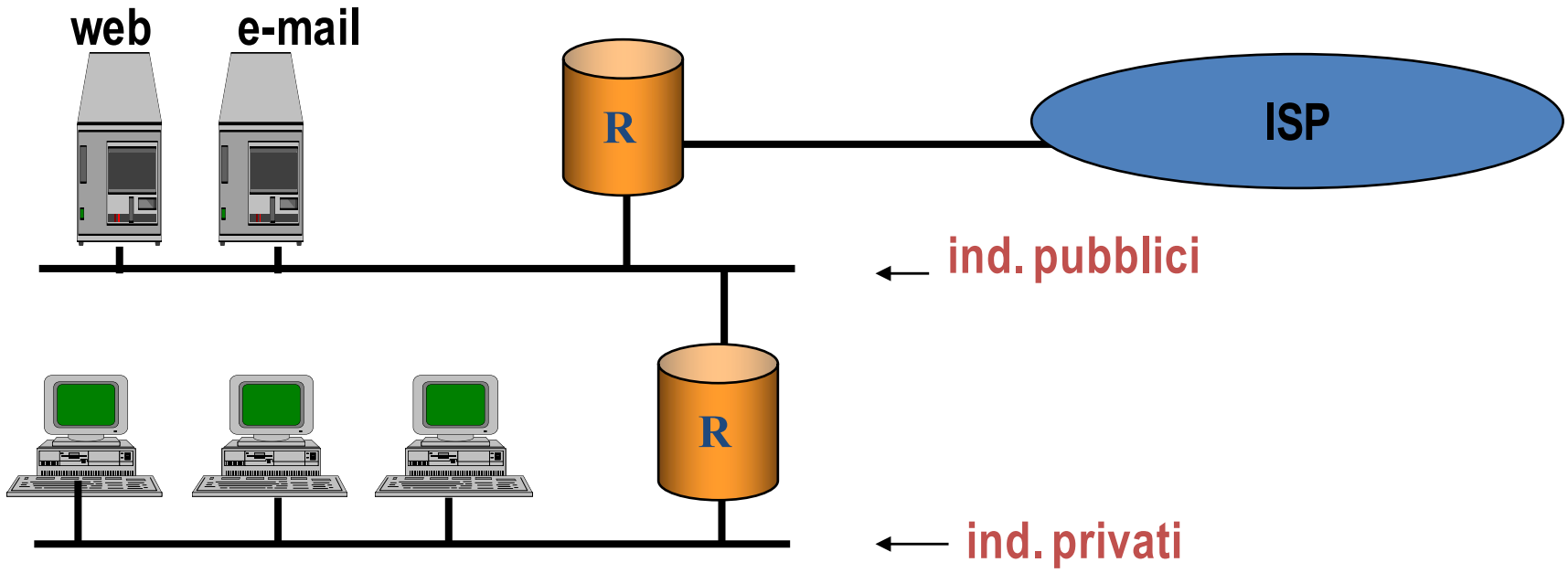


Caratteristiche Intranet

- L'evoluzione di servizi e protocolli ha però reso le *Intranet* strutturalmente differenti dalle reti pubbliche
  - Problemi di sicurezza
  - Problemi di gestione degli indirizzi
  - Problemi di distinzione tra servizi offerti ai soli utenti della *Intranet* e servizi offerti anche agli utenti di INTERNET

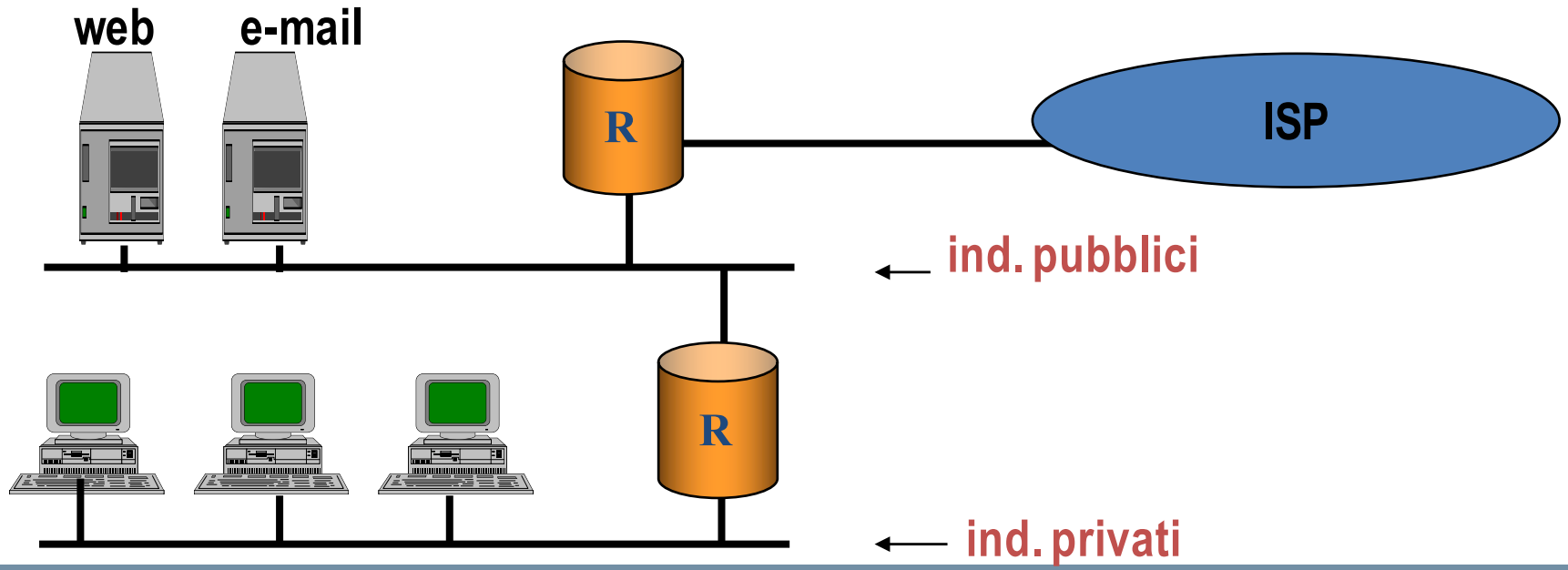
Indirizzamento privato

- Una rete privata ha normalmente una serie di servizi che sono accessibili dalla rete pubblica
- I *server* per questi servizi devono avere un indirizzo pubblico mentre gli *host* interni alla rete possono avere un indirizzo privato



Indirizzamento privato

- E' chiaro comunque che in questo modo si impedisce agli *host* della rete privata di aver accesso a tutti servizi di INTERNET
- Prima o poi sorge l'esigenza di consentire lo scambio di pacchetti tra *host* con indirizzo pubblico e *host* con indirizzo privato
- I metodi più comunemente usati per consentire il colloquio sono il *NAT e i Proxy*



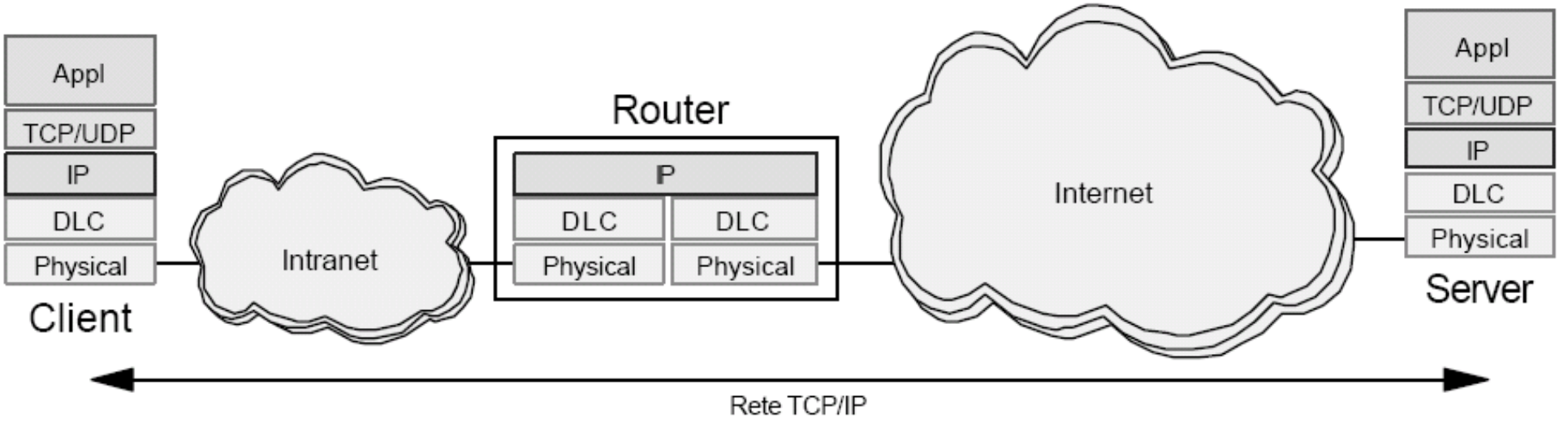
Connessione *Intranet/Internet*

- Intranet che adotta indirizzamento pubblico
  - Proxy applicativi
  - Router semplice (soluzione classica)

- Intranet che adotta indirizzamento privato
  - NAT
  - Proxy applicativi

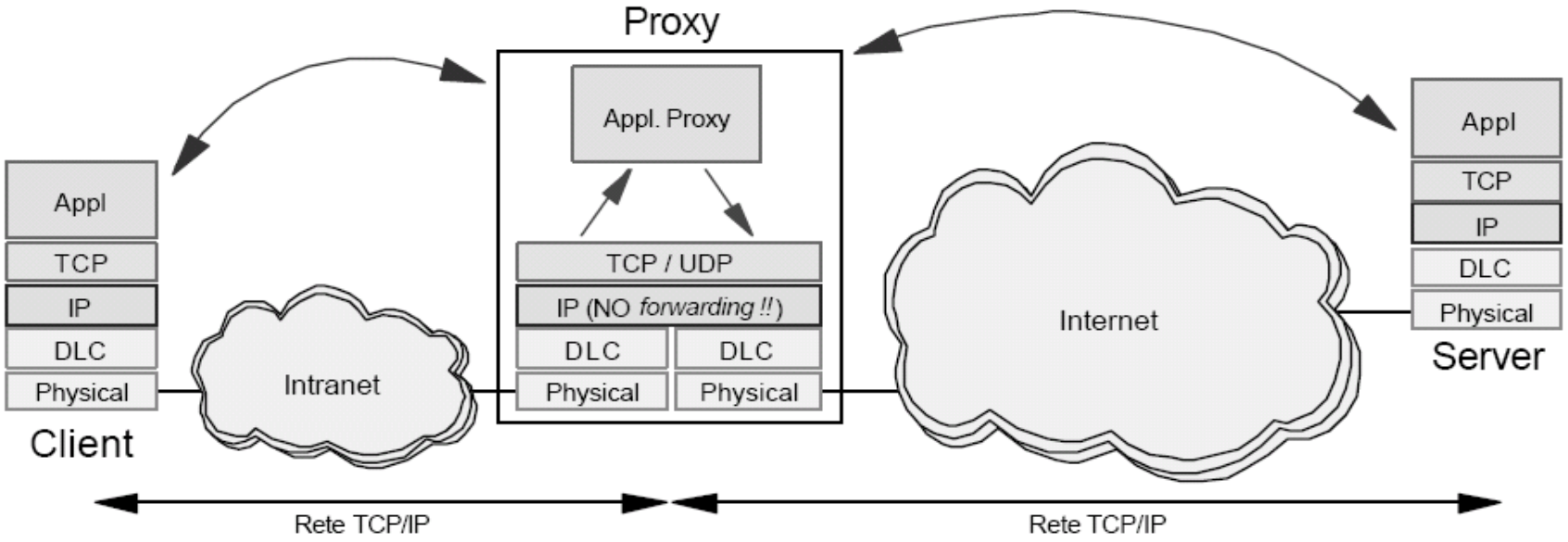
Connessione con Router Semplice

- L'intranet usa indirizzi IP pubblici
- Di fatto l'*intranet* scompare (unica rete IPv4 con l'INTERNET)
- Possibili comunicazioni da e verso l'INTERNET
- Scarsa sicurezza

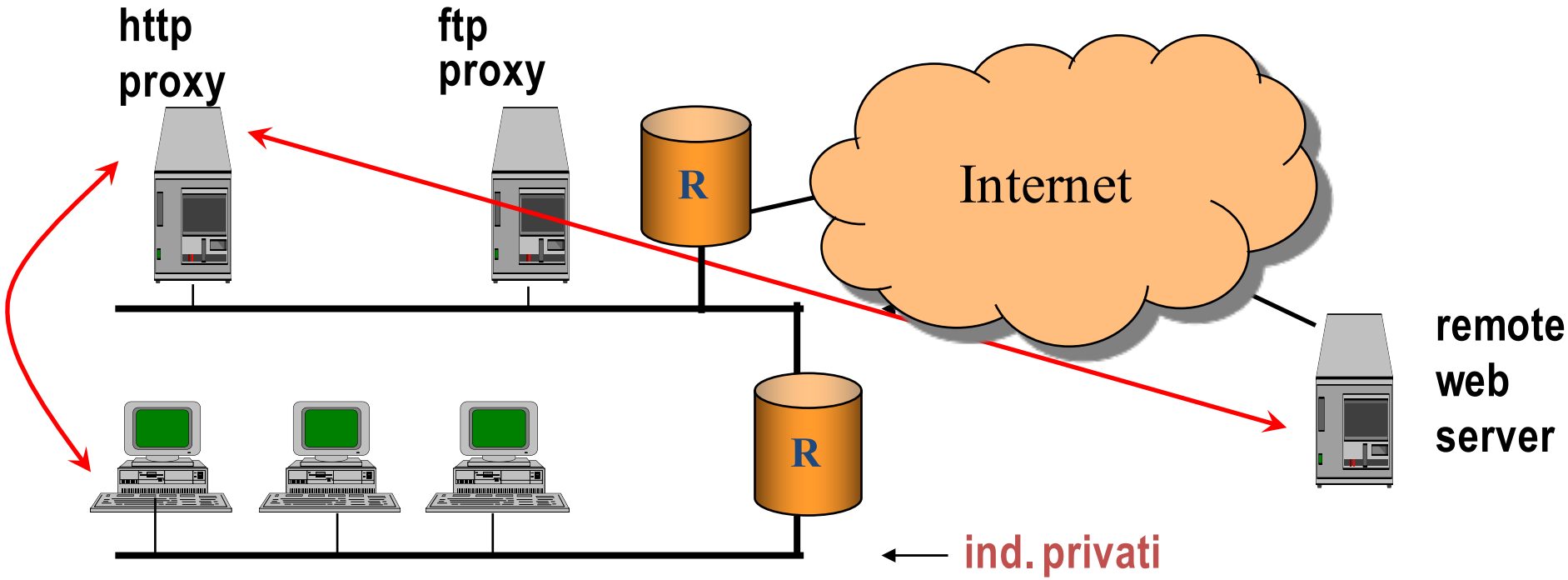


Connessione tramite Proxy Applicativo

- Funziona sia con indirizzamento pubblico che privato
- Intranet e INTERNET sono scollegate a livello IP
- Qualunque richiesta viene inviata al *proxy* che la inoltra con il proprio IP *address* pubblico
- Occorre avere un *proxy* per tutte le applicazioni



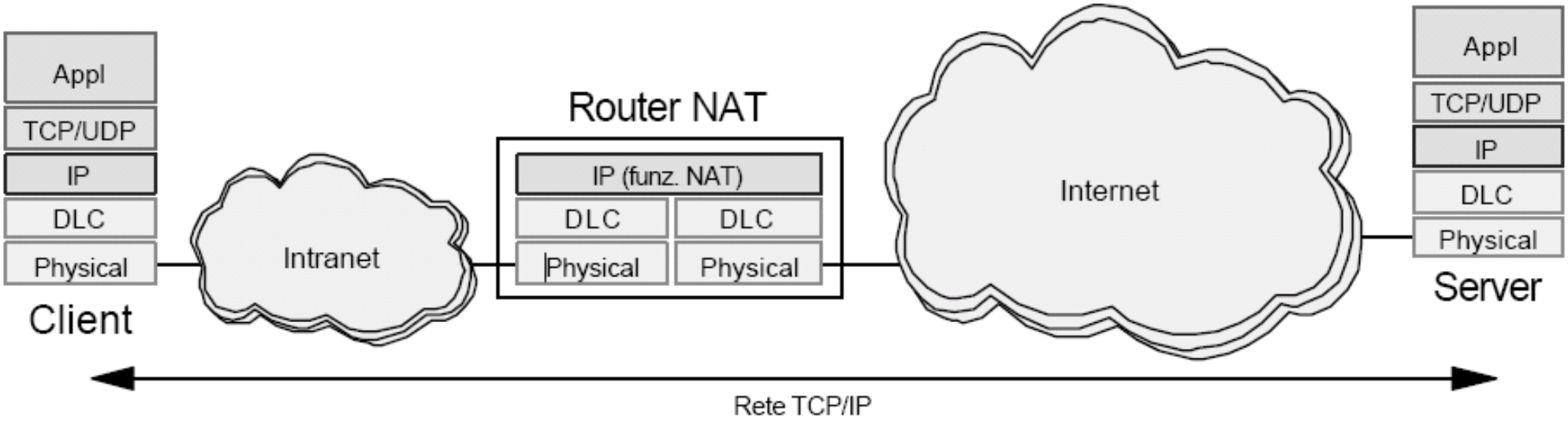
Application Proxy



- I proxy sono *application gateway*
- Qualunque richiesta viene inviata al *proxy* che la inoltra con il proprio *IP address* pubblico
- Occorre avere un *proxy* per tutte le applicazioni

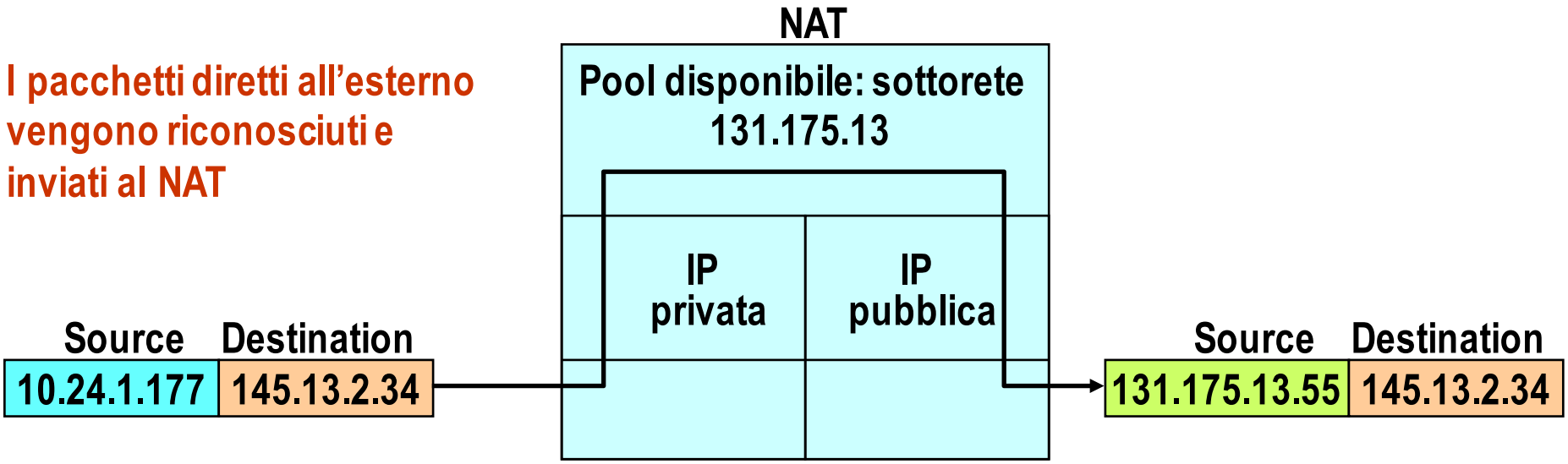
Network Address Translation (NAT)

- I NAT (*Network Address Translation*) hanno tutte le funzionalità dei router classici
- In più sanno gestire anche il *mapping* di uno spazio di indirizzamento (privato) in un altro spazio di indirizzamento (pubblico)



Network Address Translator (NAT)

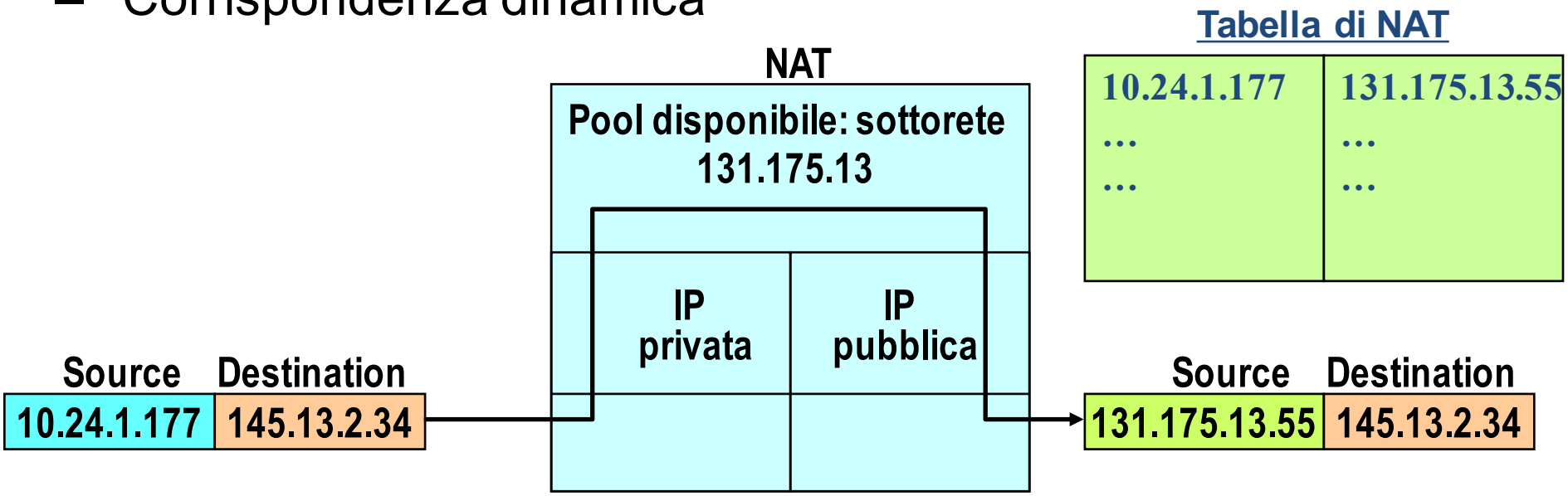
- E' un meccanismo reso disponibile su un *router/gateway*
- Consente di associare, anche temporaneamente, un ridotto numero di indirizzi pubblici, ai numeri della numerazione privata



Possibilità di blocco

NAT – Tabella di NAT

- Perché il colloquio sia bidirezionale occorre mantenere l'associazione tra indirizzo privato e pubblico un una tabella di NAT
  - Corrispondenza statica
  - Corrispondenza dinamica



NAT: Diversi Approcci

- *Traditional* NAT
  - *Basic* NAT
  - *Network Address Port Translation* (NAPT)

- *Bi-directional* NAT

Caratteristiche Comuni

- *Transparent Address Translation*
  - Associazione (binding/unbinding) trasparente alle stazioni
  - Due modalità di associazione:
    - Statica (facile ma inefficiente)
    - Dinamica (efficiente ma complessa)
- *Transparent Routing*
  - Il routing deve essere gestito in maniera coerente all'indirizzamento
- *ICMP Packet Translation*
  - Porzioni di messaggi ICMP contengono indirizzi IP, quindi vanno mappate

NAT – Associazione Dinamica (1)

- L'assegnamento dinamico si basa sul concetto di *sessione*
- Quando il NAT vede il primo pacchetto di una *sessione* crea l'associazione tra indirizzo privato e pubblico
- Al termine della sessione l'indirizzo viene rilasciato
- Cos'è una *sessione*?
  - Dipende dal protocollo utilizzato
  - Per TCP e UDP una sessione viene identificata dall'indirizzo di *socket*
  - Per ICMP dalla terna (IP sorgente, IP destinazione, *Identifier)*
  - Per direzione di una sessione si intende il verso di percorrenza del primo pacchetto

NAT – Assegnamento Dinamico (2)

- Definita la sessione occorre capire quando inizia e quando finisce
- Inizio sessione:
  - TCP: pacchetto di SYN
  - UDP, ICMP: sono connectionless, non vi è un metodo unico
- Fine sessione:
  - TCP: pacchetti di FIN per entrambe i lati (però possono non arrivare mai …)
  - Altri prot.: non vi è un metodo univoco
  - Occorrono sempre dei time-out per recuperare situazioni d'errore o perdita di pacchetti

NAT – Application Level Gateway

- Alcune applicazioni trasportano nel Payload dei loro messaggi indirizzi IP (in formato ASCII o binario) e numeri di porta
- Gli *Application Level Gateway* (*ALG*) sono funzionalità aggiuntive che servono per un corretto funzionamento del NAT
- Sulla base del tipo di applicazione e del tipo di messaggio si preoccupano di modificare i messaggi applicativi in transito e, se del caso, adattare i segmenti TCP
- Simili ai *proxy*, con la differenza che sono trasparenti alle stazioni

Traditional NAT (1)

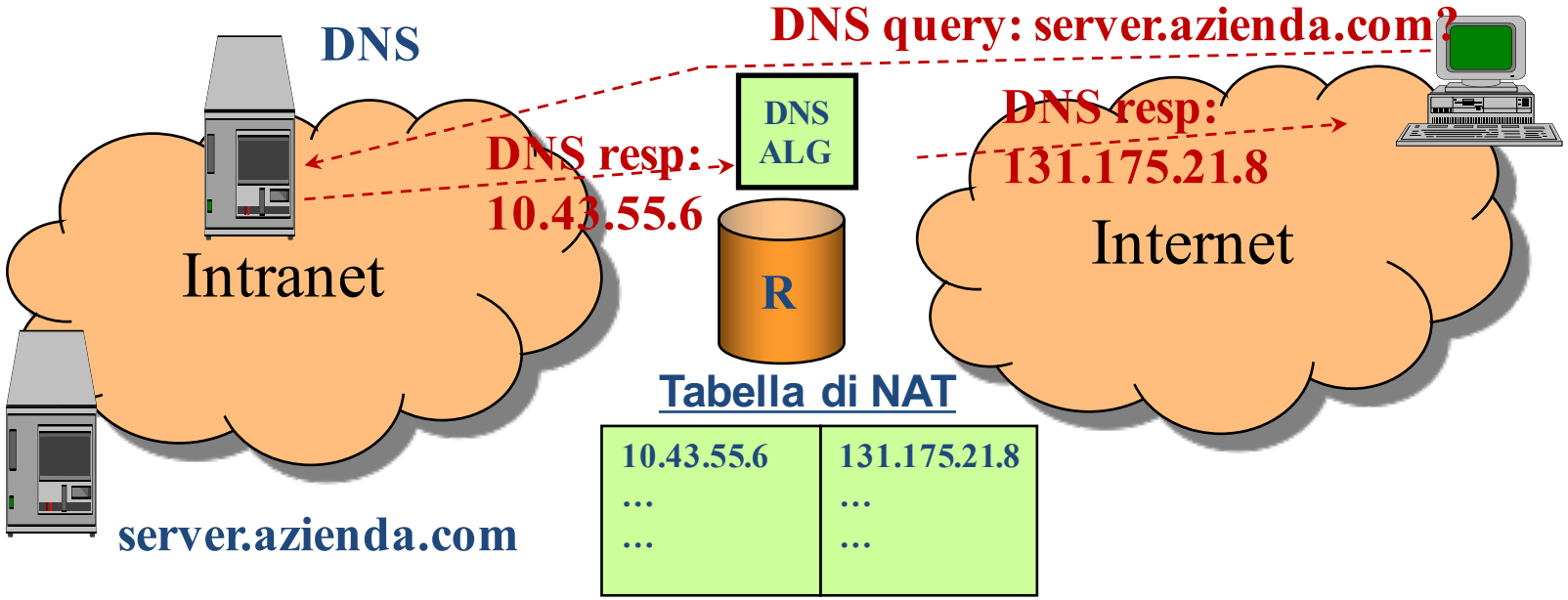
- Detto anche *Outbound* NAT
- Permette solo sessioni iniziate dall'interno (verso della sessione dall'interno verso l'esterno)
- Le informazioni di *routing* possono essere distribuite dall'esterno verso l'interno ma non viceversa
- 2 sotto-tipi
  - Basic NAT
  - NAPT (*Network Address and Port Translator*)

Traditional NAT (2)

- Basic NAT
  - Viene traslato il solo indirizzo IP
  - C'è una corrispondenza uno-a-uno nell'assegnamento degli indirizzi durante una sessione e due host non possono usare lo stesso indirizzo contemporaneamente
  - Ci può essere blocco a causa del numero scarso di indirizzi pubblici quando il traffico (numero di sessioni attive) è elevato
- NAPT
  - Viene traslata la coppia (indirizzo,porta)
  - Molti indirizzi interni possono usare lo stesso indirizzo esterno
  - Ci sono problemi con flussi diversi da UDP e TCP (per ICMP si può usare il campo Identifier)

Bi-Directional NAT

- Si può iniziare una sessione in entrambe i versi
- Problema:
  - Come fa un host pubblico ad iniziare un sessione con un host privato senza avere un indirizzo pubblico a cui raggiungerlo?
  - Occorre usare dei nomi simbolici e il servizio DNS che deve usare un unico spazio dei nomi
  - Corrispondenza statica tra indirizzo pubblico/privato del DNS privato



NAT – Alcune Considerazioni

- Il cambio di indirizzo non è un'operazione indolore
- Esso impone:
  - Il ricalcolo del *Header Checksum*
  - Sostituzione degli indirizzi dei messaggi ICMP e ricalcolo *header checksum*
  - Il ricalcolo dei *checksum* di TCP o UDP con il nuovo *pseudo-header*
- Sorgono poi dei problemi con alcuni ALG per via del trasporto degli indirizzi e porte nei messaggi di livello applicativo
- Chi crea problemi al NAT?
  - Applicazioni che trasferiscono indirizzi IP
  - IPsec e applicazioni di sicurezza