

Il Livello Network

Obiettivi:

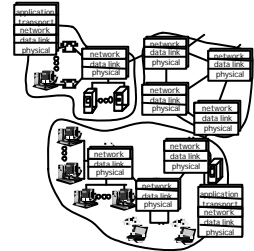
- Comprendere i principi sottostanti i servizi del livello network:
 - routing (selezione del percorso)
 - gestione della dimensione della rete
 - funzionamento del router
 - IPv6, multicast
- Instanziazione e implementazione in Internet

Panoramica:

- Servizi del livello network
- Principi di routing: selezione del percorso (path selection)
 - Routing gerarchico
 - IP
- Protocolli di trasferimento affidabile del routing in Internet
 - intra-domain
 - inter-domain
- Cosa c'è in un router?
 - IPv6
 - multicast routing

Funzioni del Livello Network

- trasporto di pacchetti dagli host mittenti ai destinatari
- I protocolli di livello network sono presenti in *tutti* gli host e router



Tre funzioni importanti:

- path determination:** percorso seguito dai pacchetti dalla sorgente alla dest. *Algoritmi di routing*
- switching:** nel router spostare i pacchetti dall'input all'output appropriato
- call setup:** architetture di rete che richiedono che il router stabilisca l'intero percorso all'inizio

Modelli di servizio di Network

Q: Che *modello di servizio* per il "canale" che trasporta i pacchetti dai mittenti ai destinatari?

LA più importante astrazione fornita dal livello network:

service abstraction

- Largh. di banda garantita?
- L'inter-packet timing viene preservato (no jitter)?
- Consegna senza perdite?
- Consegna ordinata?
- Feedback sulla congestione al mittente?

Virtual circuit diagram?

Circuiti Virtuali (VC)

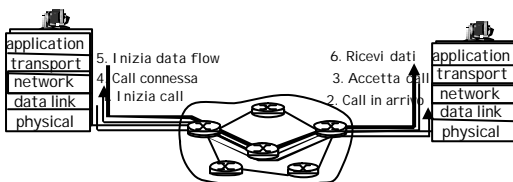
"il percorso mittente-destinatario si comporta come un circuito telefonico"

- orientato alla performance
- azioni della rete lungo il percorso mitt.-dest.

- Fase di call setup per ciascuna trasmissione prima che inizi il flusso dati
- Ciascun pacchetto contiene un identificatore di VC (e non un indirizzo di host)
- Tutti* i router sul percorso mitt.-dest. Mantengono "stato" per ciascuna connessione
- Le risorse del link e del router (largh. di banda, buffer) possono essere *reallocati* al VC
 - Per ottenere una prestazione simile ad un circuito fisico

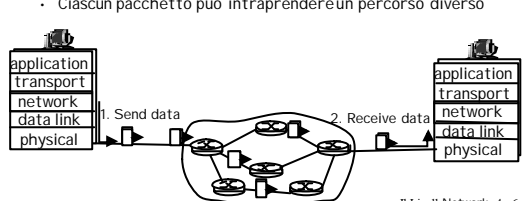
VC: protocolli di segnalazione

- Usati per stabilire, mantenere e abbattere i VC
- Usati in ATM, frame-relay, X.25
- Non usati nell'attuale Internet



Reti Datagram : il modello Internet

- Non c'è call setup a livello network
- router: non c'è stato delle connessioni end-to-end
 - Non esiste il concetto di "connessione" al livello network
- pacchetti tipicamente instradati usando ID del destinatario
 - Ciascun pacchetto può intraprendere un percorso diverso



Modelli Servizio Livello Network :

Architettura Network	Modello Servizio	Garanzie?	Garanzie?			Feedback Congestione
			Banda	Perd.	Ordin. Temp.	
Internet	best effort	Nessuna garanzia	no	no	no	no (inferita dalle perdite)
ATM	CBR	costante	si	si	si	Non c'è congestione
ATM	VBR	garantita	si	si	si	Non c'è congestione
ATM	ABR	minimo garantito	no	si	no	si
ATM	UBR	Nessuna garanzia	no	si	no	no

- Il modello Internet è in corso di estensione: Intserv, Diffserv

Il LivelloNetwork 4a-7

Reti di tipo Datagram o VC?

Internet

- Scambio dati tra computer
 - Servizio "elastico", non è richiesta tempif. forte
- Gli end systems sono "smart" (computer)
 - Possono adattarsi, effettuare controlli, recupero errori
 - La rete è semplice, la complessità è ai "bordi"
- Molti tipi di link
 - differenti caratteristiche
 - Difficoltà uniforme

ATM

- Nasce dalla telefonia
- Conversazione umana:
 - Tempif. forte, esigenze di affidabilità
 - Necessità di servizi garantiti
- Gli end systems sono "dumb"
 - telefoni
 - La complessità è all'interno della rete

Il LivelloNetwork 4a-8

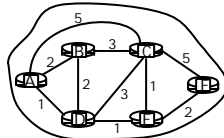
Routing

Routing protocol

Goal: stabilire sulla rete un "buon" path dal mitt. al dest. (sequenza di router)

Gli algoritmi di routing utilizzano i grafi :

- I nodi sono i router
- Gli archi sono i link fisici
 - Costo del link: ritardo, costo \$, o livello di congestione



- "buon" path:
 - Tipicamente indica il path a costo minimo
 - Sono possibili altre definizioni

Il LivelloNetwork 4a-9

Classificazione algoritmi di Routing

Informazione globale o decentralizzata?

- Globale:
 - Tutti i router conoscono la topologia completa, il costo del link
 - Algoritmi "link state"
- Decentralizzata:
 - Un router conosce i vicini fisicamente connessi, il costo del link verso i vicini
 - processo iterativo di calcolo, scambio di info con i vicini
 - Algoritmi "distance vector"

Statici o dinamici?

- Statici:
 - I percorsi cambiano lentamente nel tempo
- Dinamici:
 - I percorsi cambiano velocemente
 - aggiornamento periodico
 - in risposta al cambiamento del costo del link

Il LivelloNetwork 4a-10

Un algoritmo di Routing Link-State

Algoritmo di Dijkstra

- Topologia della rete, costo link noti a tutti i nodi
 - Ottenuti tramite un "link state broadcast"
 - Tutti i nodi hanno le stesse info
- Calcola i percorsi a costo minimo da un nodo ("source") a tutti gli altri nodi
 - Da la tabella di routing per quel nodo
- iterativo: dopo k iterazioni, conosco i percorsi a costo minimo verso k destinazioni

Notazione:

- $c(i,j)$: costo del link dal nodo i a j , costo infinito se non sono vicini diretti
- $D(v)$: valore corrente del costo di un percorso dalla sorgente alla destinazione v
- $p(v)$: nodo predecessore lungo il percorso dalla sorgente a v , ovvero il prossimo v
- N : insieme dei nodi i cui percorsi minimi sono definitivamente noti

Il LivelloNetwork 4a-11

L'Algoritmo di Dijkstra

1 Inizializzazione

- $N = \{A\}$
- for tutti i nodi v
- if v adiacente ad A
- then $D(v) = c(A,v)$
- else $D(v) = \text{infinito}$
-
-

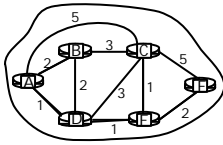
8 Loop

- Trova w not in N tale che $D(w)$ è un minimo
- aggiungi w a N
- aggiorna $D(v)$ per tutti v adiacenti a w e not in N :
 $D(v) = \min(D(v), D(w) + c(w,v))$
- /* il nuovo costo di v è il vecchio costo di v o il più breve
- Percorso noto di w più il costo da w a v */
- until tutti i nodi in N

Il LivelloNetwork 4a-12

Algoritmo di Dijkstra: esempio

Step	start N	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
→0	A	2,A	5,A	1,A	infinito	infinito
→1	AD	2,A	4,D		2,D	infinito
→2	ADE	2,A	3,E			4,E
→3	ADEB		3,E			4,E
→4	ADEBC					4,E
5	ADEBCF					



Il LivelloNetwork 4a-13

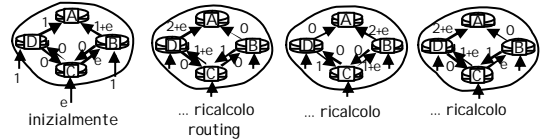
Algoritmo di Dijkstra, discussione

complessità algoritmo : n nodi

- Ogni iterazione: deve controllare tutti i nodi "w not in N"
- n*(n+1)/2 comparazioni: O(n*²)
- Implementazione possibile a massima efficienza: O(nlogn)

Possibili Oscillazioni:

- Per es., costo del link = ammontare di traffico gestito



Il LivelloNetwork 4a-14

Algoritmo di Routing Distance Vector

iterativo:

- continua finché ci sono scambi di info tra nodi
- *self-terminating* non c'è un "segnale" di stop

asincrono:

- I nodi non devono scambiare le info in passi predeterminati!

distribuito:

- Ogni nodo comunica solo con i vicini diretti

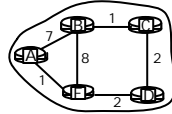
struttura dati Distance Table

- Ogni nodo ha una sua propria riga per ciascuna destinazione
- Una colonna per ogni vicino diretto del nodo
- esempio: nel nodo X, per dest. Y via vicino Z:

$$D^X(Y,Z) = \begin{matrix} \text{distanza da X a} \\ \text{Y, via Z come next hop} \end{matrix} \\ = c(X,Z) + \min_w \{D^Z(Y,w)\}$$

Il LivelloNetwork 4a-15

Distance Table: esempio



$$D^E(C,D) = c(E,D) + \min_w \{D^D(C,w)\} \\ = 2+2 = 4$$

$$D^E(A,D) = c(E,D) + \min_w \{D^D(A,w)\} \\ = 2+3 = 5 \text{ loop!}$$

$$D^E(A,B) = c(E,B) + \min_w \{D^B(A,w)\} \\ = 8+6 = 14 \text{ loop!}$$

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

Il LivelloNetwork 4a-16

distance table e routing table

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

Distance table → Routing table

		link in uscita da usare, costo
destinazione	A	A,1
	B	D,5
	C	D,4
	D	D,4

Il LivelloNetwork 4a-17

Routing Distance Vector : panoramica

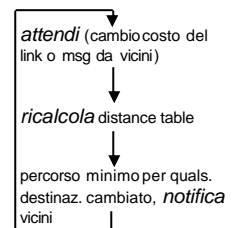
Iterativo, asincrono: ogni iterazione locale causata da:

- cambio costo del link locale
- Messaggio dai vicini: il suo percorso a costo minimo è cambiato

Distribuito:

- Ogni nodo notifica i vicini solo se ci sono cambiamenti
 - I vicini poi notificano, se necessario, i propri vicini

Ogni nodo:



Il LivelloNetwork 4a-18

Algoritmo Distance Vector :

per tutti i nodi , X:

- 1 Inizializzazione:
- 2 Per tutti i nodi adiacenti v:
- 3 $D^X(*,v) = \text{infinito}$ /* l'operatore* sta per "tutte le righe" */
- 4 $D^X(v,v) = c(X,v)$
- 5 Per tutte le destinazioni, y
- 6 invia $\min_w D^X(y,w)$ a ogni vicino /* w su tutti i vicini di X */

Il LivelloNetwork 4a-19

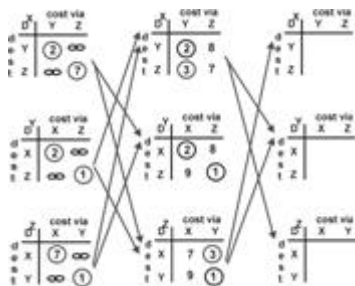
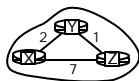
Algoritmo Distance Vector (cont.):

```

8 loop
9 wait (finchè rilevi un cambio di costo del link per il vicino V
10 o finchè ricevi un aggiornamento dal vicino V)
11
12 if (c(X,V) cambia di d)
13 /* cambia di d i costi di tutte le dest. attraverso il vicino v */
14 /* nota: d può essere sia positivo che negativo */
15 per tutte le destinazioni y:  $D^X(y,V) = D^X(y,V) + d$ 
16
17 else if (ricevuto aggiornamento da V verso la destinazione Y)
18 /* il percorso minimo da V a un dato Y è cambiato */
19 /* V ha inviato un nuovo valore per il suo  $\min_w D^V(Y,w)$  */
20 /* chiamiamolo "newval" */
21 per la singola destinazione y:  $D^X(Y,V) = c(X,V) + \text{newval}$ 
22
23 if esiste un nuovo  $\min_w D^X(Y,w)$  per una destinazione Y
24 Invia il nuovo valore di  $\min_w D^X(Y,w)$  a tutti i vicini
25
26 forever
    
```

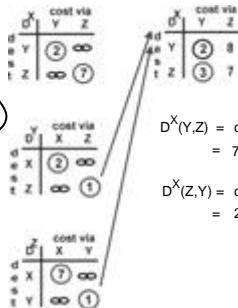
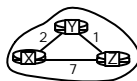
Il LivelloNetwork 4a-20

Algoritmo Distance Vector: esempio



Il LivelloNetwork 4a-21

Algoritmo Distance Vector: esempio



$$D^X(Y,Z) = c(X,Z) + \min_w \{D^Z(Y,w)\} = 7 + 1 = 8$$

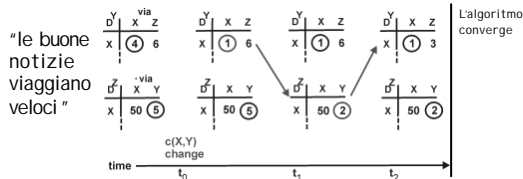
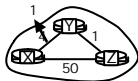
$$D^X(Z,Y) = c(X,Y) + \min_w \{D^Y(Z,w)\} = 2 + 1 = 3$$

Il LivelloNetwork 4a-22

Distance Vector: cambio costo di un link

Il costo di un link cambia:

- Un nodo rileva il cambiamento
- aggiorna la distance table (linea 15)
- Se cambia un percorso a costo minimo, notifica i vicini (linee 23,24)



"le buone notizie viaggiano veloci"

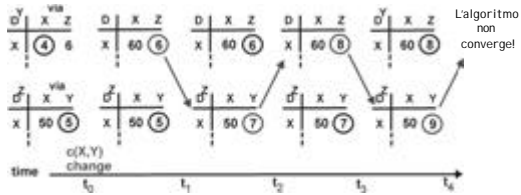
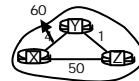
L'algoritmo converge

Il LivelloNetwork 4a-23

Distance Vector: cambio costo di un link

Il costo di un link cambia:

- Le buone notizie vanno veloci
- Le cattive piano - problema del "conteggio all'infinito"!



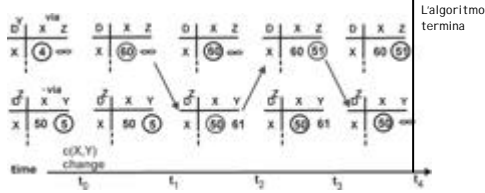
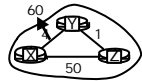
L'algoritmo non converge!

Il LivelloNetwork 4a-24

Distance Vector: poisoned reverse

Se Z passa per Y per arrivare a X :

- Z dice a Y che la sua distanza da X è infinita (così Y non vorrà instradare a X via Z)
- Questo risolve completamente il problema del conteggio all'infinito?



Il LivelloNetwork 4a-25

Paragone tra gli algoritmi di LS e DV

Complessità del Messaggio

- LS: con n nodi, E link, $O(nE)$ messaggi inviati
- DV: scambio solo tra i vicini
 - Tempo di convergenza variabile

Velocità di Convergenza

- LS: algoritmo $O(n^2)$ richiede $O(nE)$ messaggi
 - può avere oscillazioni
- DV: il tempo di converg. varia
 - vi possono essere dei loop nel routing
 - problema del conto all'infinito

Robustezza: che succede se il router si guasta?

LS:

- Il nodo può segnalare un costo del *link* sbagliato
- Ogni nodo calcola solo la sua tabella

DV:

- Il nodo può segnalare un costo del *path* sbagliato
- La tabella di ciascun nodo è usata dagli altri
 - L'errore si propaga attraverso la rete

Il LivelloNetwork 4a-26

Routing Gerarchico

Abbiamo usato finora delle ipotesi ideali

- Tutti i router sono uguali
- La rete è "flat"

... che *non* sono vere nella pratica

dimensioni: con 50 milioni di destinazioni:

- Non è possibile registrare tutte le destinazioni nelle tabelle di routing!
- Lo scambio delle tabelle di routing saturerebbe i link!

autonomia amministrativa

- internet = rete di reti
- Ciascun amministratore di rete vuole controllare il routing nella sua specifica rete

Il LivelloNetwork 4a-27

Routing Gerarchico

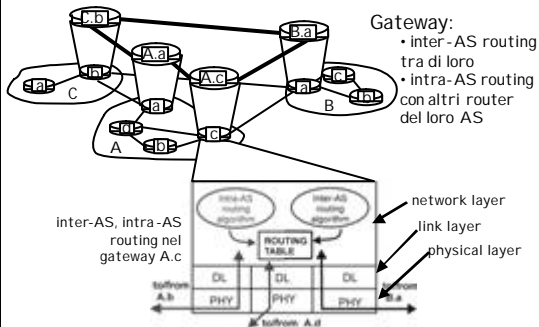
- I router sono aggregati in regioni, "autonomous systems" (AS)
- I router nello stesso AS hanno lo stesso protocollo
 - Protocollo di "intra-AS" routing
 - router in AS differenti possono avere protocolli intra-AS routing diversi

gateway routers

- router speciali negli AS
- Eseguono il protocollo di intra-AS routing con tutti gli altri router dell'AS
- Anche responsabile per il routing per le destinazioni esterne AS
 - Eseguono un protocollo *inter-AS routing* con altri gateway routers

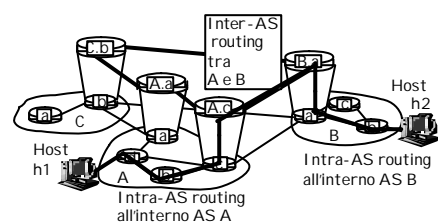
Il LivelloNetwork 4a-28

Routing Intra-AS e Inter-AS



Il LivelloNetwork 4a-29

Routing Intra-AS e Inter-AS

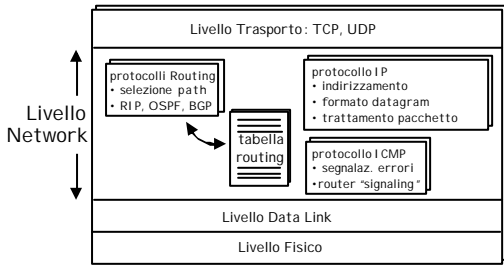


- Tra breve esamineremo esempi di specifici protocolli di routing inter-AS e intra-AS di Internet

Il LivelloNetwork 4a-30

Il Livello Network di Internet

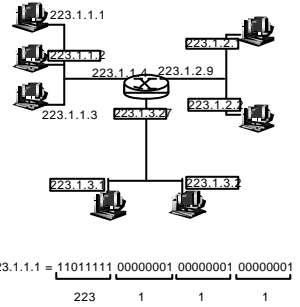
Funzioni del livello network negli Host e router:



Il LivelloNetwork 4a-31

Indirizzamento IP : introduzione

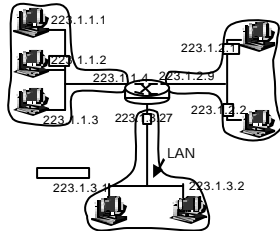
- Indirizzo IP: 32-bit di identificatore per l'interfaccia di host e router
- interfaccia: connessione tra host, router ed il link fisico
 - Tipicamente i router hanno interfacce multiple
 - Gli host possono averle
 - Gli indirizzi IP sono associati con le interfacce e non con host e router



Il LivelloNetwork 4a-32

Indirizzamento IP

- Indirizzo IP:
 - parte rete (high order bits)
 - parte host (low order bits)
- Cos'è una rete? (dal punto di vista dell'indirizzo IP)
 - Dispositivi di interfaccia che hanno la stessa parte rete dell'indirizzo IP
 - Possono raggiungersi l'un l'altro senza l'intervento del router



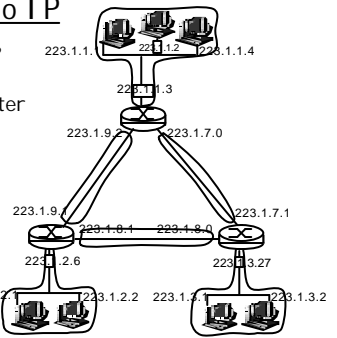
Rete consistente di 3 reti IP (per gli indirizzi IP che iniziano con 223, i primi 24 bit sono indirizzo di rete)

Il LivelloNetwork 4a-33

Indirizzamento IP

Come trovare le reti ?

- Staccando ogni interfaccia dal router o dall'host
- Si creano "isole di reti isolate"

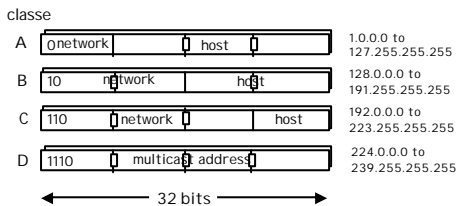


Sistema Interconnesso consistente di sei reti

Il LivelloNetwork 4a-34

Indirizzi IP

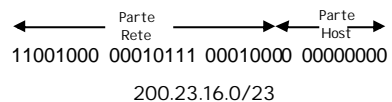
Data la nozione di "rete", riesaminiamo gli indirizzi IP: indirizzamento "a classi" (class-full):



Il LivelloNetwork 4a-35

Indirizzamento IP : CIDR

- Indirizzamento a classi:
 - uso inefficiente dello spazio indirizzi, esaurimento spazio indirizzi
 - Per es., una rete di classe B alloca indirizzi per 65K host, anche se ce ne sono solo 2K in quella specifica rete
- CIDR: Classless InterDomain Routing
 - Parte indirizzo destinata alla rete di lunghezza arbitraria
 - formato indirizzo: a.b.c.d/x, dove x è # bit della parte rete dell'indirizzo



Il LivelloNetwork 4a-36

Indirizzi IP: come ottenerli?

Network (parte rete):

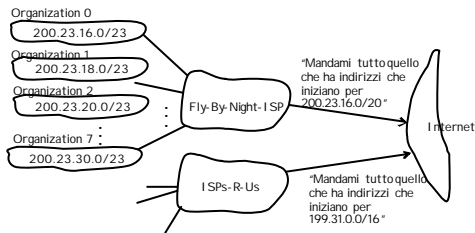
- Parti allocate spazio indirizzi per ISP:

ISP's block	11001000_00010111_00010000	00000000	200.23.16.0/20
Organization 0	11001000_00010111_00010000	00000000	200.23.16.0/23
Organization 1	11001000_00010111_00010010	00000000	200.23.18.0/23
Organization 2	11001000_00010111_00010100	00000000	200.23.20.0/23
...
Organization 7	11001000_00010111_00011110	00000000	200.23.30.0/23

Il LivelloNetwork 4a-37

Indirizzam. gerarchico: percorsi aggregati

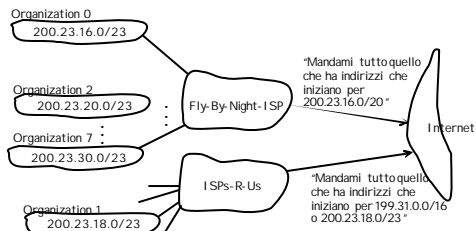
Un indirizzamento gerarchico consente una segnalazione efficiente delle info di routing:



Il LivelloNetwork 4a-38

Indirizzam. gerarchico: percorsi specifici

ISPs-R-Us ha un percorso specifico per Organization 1



Il LivelloNetwork 4a-39

I principali campi di un datagram IP



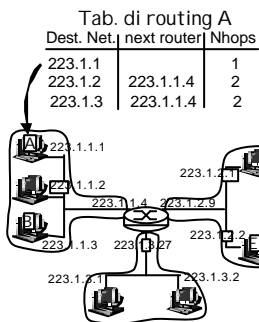
Il LivelloNetwork 4a-40

Viaggio di un datagram da mitt. a dest.

datagram IP :

misc fields	source IP addr	dest IP addr	data
-------------	----------------	--------------	------

- Il datagram non cambia durante il viaggio
- L'unica parte utilizzata sono i campi indirizzo



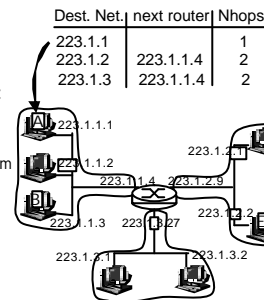
Il LivelloNetwork 4a-41

Viaggio di un datagram da mitt. a dest.

misc fields	223.1.1.1	223.1.1.3	data
-------------	-----------	-----------	------

Partendo da A, un certo IP datagram indirizzato a B:

- Cerca l'indirizzo rete di B
- Trova B sulla stessa rete di A
- Il livello link invierà il datagram direttamente a B in un pacchetto di livello link
 - B e A sono connessi direttamente



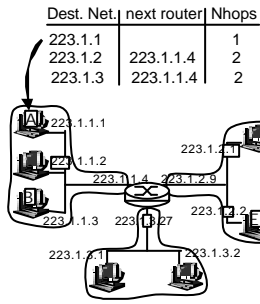
Il LivelloNetwork 4a-42

Viaggio di un datagram da mitt. a dest.

misc fields	223.1.1.1	223.1.2.2	data
-------------	-----------	-----------	------

Partendo da A, dest. E:

- Cerca l'indirizzo rete di E
- E su rete *differente*
 - A, E non dirett. connessi
- Tabella di routing: il prossimo router per E è 223.1.1.4
- Il link layer invia il datagram al router 223.1.1.4 in un pacchetto di livello link
- Il datagram arriva a 223.1.1.4
- continua.....



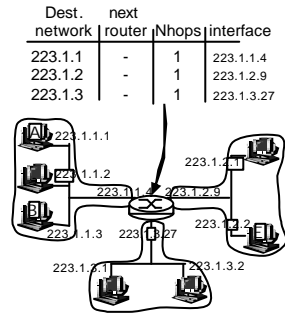
Il LivelloNetwork 4a-43

Viaggio di un datagram da mitt. a dest.

misc fields	223.1.1.1	223.1.2.2	data
-------------	-----------	-----------	------

Arrivo a 223.1.4, destinazione 223.1.2.2

- Cerca l'indirizzo rete di E
- E stessa rete dell'interfaccia 223.1.2.9 del router
 - router ed E connessi dir.
- Il link layer invia il datagram a 223.1.2.2 in un pacchetto di livello link attraverso l'interfaccia 223.1.2.9
- datagram arriva a 223.1.2.2!!! (hurra!)



Il LivelloNetwork 4a-44

Indirizzam. IP : un ultima cosa...

Q: Come fa un ISP ad ottenere un blocco di indirizzi?

A: ICANN: Internet Corporation for Assigned Names and Numbers

- Alloca gli indirizzi
- Gestisce i DNS
- Assegna i nomi di dominio arbitra eventuali dispute

Il LivelloNetwork 4a-45

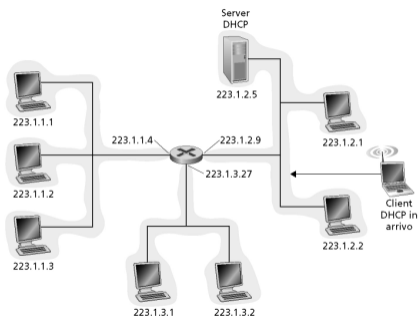
Indirizzi IP : come ottenerli?

Per gli Host (parte host):

- scritti in un file di amministr. di sistema
- DHCP: Dynamic Host Configuration Protocol: indirizzi ottenuti dinamicamente: "plug-and-play"
 - L'host invia in broadcast messaggio "DHCP discover"
 - Il server DHCP risponde con messaggio "DHCP offer"
 - L'host richiede un indirizzo IP: messaggio "DHCP request"
 - Il server DHCP invia l'indirizzo: messaggio "DHCP ack"

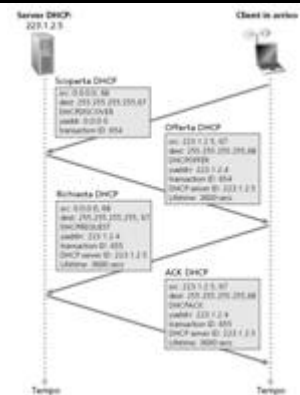
Il LivelloNetwork 4a-46

Scenario client-server DHCP



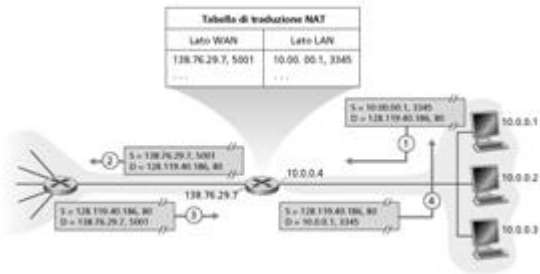
Il LivelloNetwork 4a-47

Interazione client-server DHCP



Il LivelloNetwork 4a-48

Network Address Translation



Il LivelloNetwork 4a-49