



UNIVERSITA' DEGLI STUDI - L'AQUILA

Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica

Compito di Reti di Calcolatori

05-02-2014

1. Un' azienda ha una rete di classe C, 200.1.1.x , e vuole creare sottoreti per quattro dipartimenti, con i seguenti host: A [72 host], B [35 host], C [20 host] e D [18 host]. (6 punti)
 - a. Descrivete una possibile configurazione di maschere di sottorete che risolva il problema.
 - b. Sugerite cosa può fare l'azienda se il dipartimento D cresce fino ad avere 34 host.
2. Il fatto che i dati presenti in una tabella ARP scadano dopo 10-15 minuti è un tentativo di ottenere un compromesso ragionevole. Descrivete i problemi che si possono verificare se tale durata è troppo breve o troppo lunga. (6 punti)
3. Supponete che agli host A e B sia stato assegnato lo stesso indirizzo IP sulla stessa Ethernet, all'interno della quale viene usato ARP. B inizia a funzionare dopo A. (4 punti)
 - a. Cosa accadrà alle connessioni di A già esistenti?
 - b. Fornite una soluzione plausibile a questo problema.
4. Spiegare il problema degli "Indiani": ovvero Little-Endian verso Big-Endian (4 punti)
5. Supponiamo che un router abbia la tabella di routing mostrata sotto. Il router può consegnare i pacchetti attraverso le interfacce 1 e 0, o può inoltrarli ai routers R2, R3, R4. (10 punti)

Subnet Number	Subnet Mask	Next Hop
128.96.39.0	255.255.255.128	Interface 0
128.96.39.128	255.255.255.128	Interface 1
128.96.40.0	255.255.255.128	R2
192.4.153.0	255.255.255.192	R3
<default>		R4

Descrivere il comportamento del router quando arrivano dei pacchetti indirizzati alle seguenti destinazioni:

- a. 128.96.39.10
- b. 128.96.40.12
- c. 128.96.40.151
- d. 192.4.153.17
- e. 192.4.153.90



Soluzione Esercizio 1::

a) L'indirizzo di classe C può ospitare al più 256 host, quindi dobbiamo suddividere i 256 host disponibili in gruppi potenze di 2 per ciascun dipartimento.

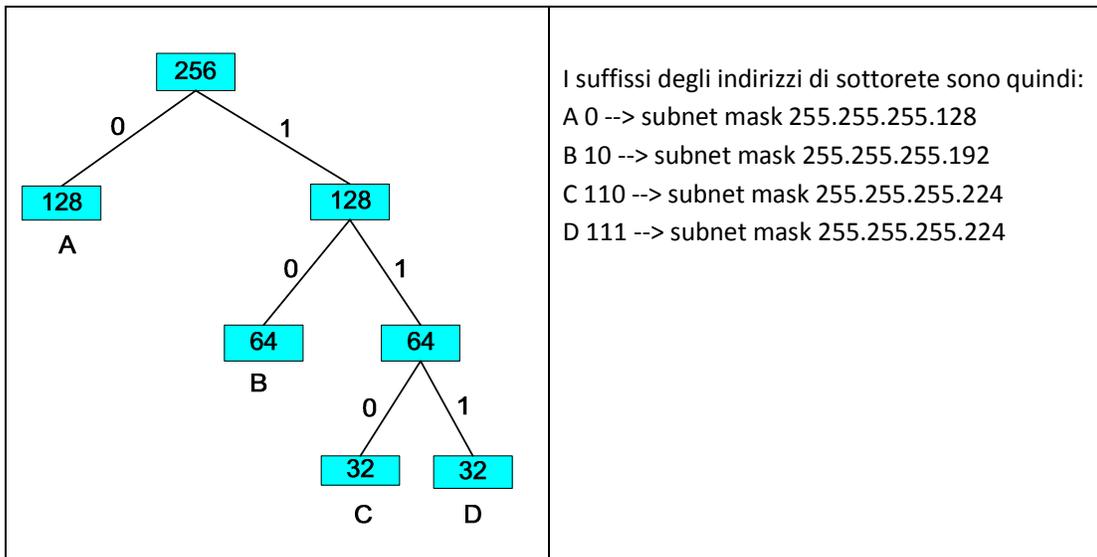
La più piccola potenza di 2 maggiore o uguale a 72 è $2^7 = 128$

Similmente:

$35 \Rightarrow 2^6 = 64$

$20 \Rightarrow 2^5 = 32$

$18 \Rightarrow 2^5 = 32$

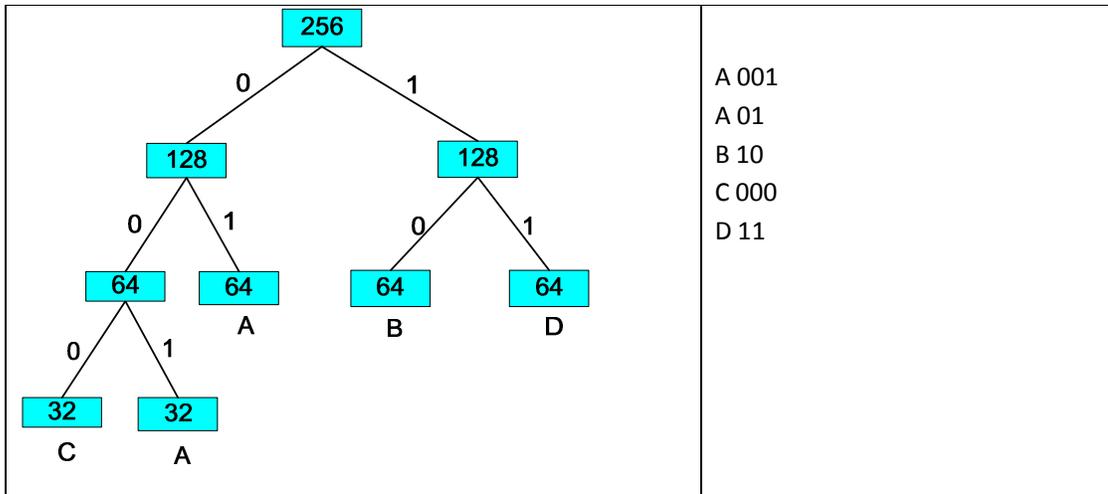


b) Se il dipartimento D passa a 34 host bisogna ridistribuire gli indirizzi.

Dobbiamo dividere il dipartimento A in 2 sottoreti (A infatti non utilizzava tutti gli indirizzi a sua disposizione: $72 < 2^7$): si può scomporre A in 2 sottoreti da 26 e 25 ottenendo così altri 25 indirizzi "disponibili" per D.

Si noti che $2^7 - 2^5 = 2^6 + 2^5$.

La nuova organizzazione è visibile nella figura sottostante





Soluzione Esercizio 2:

- Una durata eccessivamente breve delle entry nella ARP-table può portare al circolare di un elevato volume di messaggi ARP-request, il che sappiamo essere una cosa piuttosto pesante per la rete, dato che ARP si basa sul broadcast.. Il vantaggio di un simile approccio è sicuramente quello di avere tavole sempre (quasi sempre) corrispondenti al vero e comunque sempre aggiornate; non è tuttavia pensabile usare la rete per far circolare solo questo tipo di messaggi, che comunque sono e rimangono “di servizio” e quindi inutili ai fini dell’effettiva trasmissione dati.
All’opposto, usare “timeOut” troppo lunghi potrebbe causare un’errata corrispondenza nelle ARP-table fra indirizzi IP e indirizzi fisici; se vogliamo un effetto ancor più nefasto del precedente. Consideriamo infatti il caso in cui un host si rompa: potrebbe capitare che subito dopo un nuovo host si connetta ed, eseguendo DHCP, gli venga assegnato l’indirizzo IP dell’host guasto. Tutti gli altri host non sono assolutamente a conoscenza di queste dinamiche, e questo può causare incomprensioni in quegli host con tabelle ARP contenenti ancora un’informazione non aggiornata, come invii a indirizzi fisici non più esistenti o invii a host non desiderati.

Soluzione Esercizio 3:

- Gli host che hanno già una riga relativa ad A nella ARP table continuano a consegnare correttamente i pacchetti ad A. Però consegnano ad A anche i pacchetti destinati a B. L’auto-ARP risolve il problema: B si accorge subito che il suo indirizzo è già usato da A e agisce di conseguenza (ad esempio richiedendo un nuovo indirizzo IP), prima di instaurare connessioni con altri host.

Soluzione Esercizio 4:

- Big-endian e little-endian sono due metodi differenti usati dai calcolatori per immagazzinare in memoria dati di dimensione superiore al byte (es. word, dword, qword). La differenza tra i due sistemi è data dall’ordine con il quale i byte costituenti il dato da immagazzinare vengono memorizzati:
 - big-endian è la memorizzazione che inizia dal byte più significativo per finire col meno significativo; è utilizzata dai processori Motorola e nei protocolli usati in Internet
 - little-endian è la memorizzazione che inizia dal byte meno significativo per finire col più significativo; è utilizzata dai processori Intel

Soluzione Esercizio 5:

- per risolvere l’esercizio non basta fare altro che un semplice AND logico tra gli indirizzi dati e la SubnetMask. Dopo tale operazione, si confronta il risultato ottenuto con il SubnetNumber e si vede il NextHop a cui inoltrare il pacchetto:

$255.255.255.128 \wedge 128.96.39.10$

```
1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 . 1 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 . 0 1 1 0 0 0 0 0 . 0 0 1 0 0 1 1 1 . 0 0 0 0 1 0 1 0
1 0 0 0 0 0 0 0 . 0 1 1 0 0 0 0 0 . 0 0 1 0 0 1 1 1 . 0 0 0 0 0 0 0 0
il risultato è 128.96.39.0, quindi il pacchetto va inoltrato a Interface 0
```



UNIVERSITA' DEGLI STUDI - L'AQUILA

Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica

255.255.255.128 ^ 128.96.40.12

```
1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 . 1 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 . 0 1 1 0 0 0 0 0 . 0 0 1 0 1 0 0 0 . 0 0 0 0 1 1 0 0
1 0 0 0 0 0 0 0 . 0 1 1 0 0 0 0 0 . 0 0 1 0 1 0 0 0 . 0 0 0 0 0 0 0 0
```

il risultato è 128.96.40.0, quindi il pacchetto va inoltrato a R2

255.255.255.128 ^ 128.96.40.151

```
1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 . 1 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 . 0 1 1 0 0 0 0 0 . 0 0 1 0 0 1 1 1 . 1 0 0 1 0 1 1 1
1 0 0 0 0 0 0 0 . 0 1 1 0 0 0 0 0 . 0 0 1 0 0 1 1 1 . 1 0 0 0 0 0 0 0
```

il risultato è 128.96.40.128, quindi il pacchetto va inoltrato a R4

255.255.255.128 ^ 192.4.153.17

```
1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 . 1 0 0 0 0 0 0 0
1 1 0 0 0 0 0 0 . 0 0 0 0 0 1 0 0 . 1 0 0 1 1 0 0 1 . 0 0 0 1 0 0 0 1
1 1 0 0 0 0 0 0 . 0 0 0 0 0 1 0 0 . 1 0 0 1 1 0 0 1 . 0 0 0 0 0 0 0 0
```

il risultato è 192.4.153.0, quindi il pacchetto va inoltrato a R3

255.255.255.128 ^ 192.4.153.90

```
1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 . 1 0 0 0 0 0 0 0
1 1 0 0 0 0 0 0 . 0 0 0 0 0 1 0 0 . 1 0 0 1 1 0 0 1 . 0 0 0 1 0 0 0 1
1 1 0 0 0 0 0 0 . 0 0 0 0 0 1 0 0 . 1 0 0 1 1 0 0 1 . 0 0 0 0 0 0 0 0
```

il risultato è 192.4.153.0, quindi il pacchetto va inoltrato a R3