

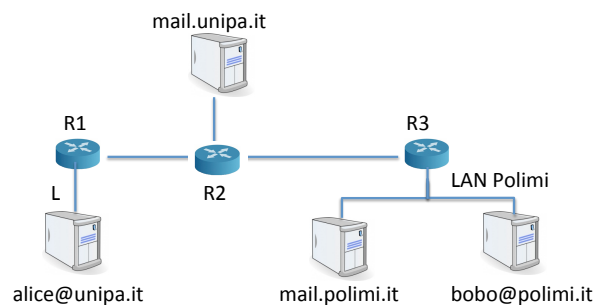
RETI DI CALCOLATORI A.A. 2016/2017

Docente: Ing. Alessandra De Paola
(Durata 2:30h)

Quesito 1

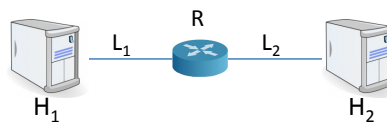
Data la configurazione schematizzata in figura, si assuma che l'utente `alice@unipa.it` invii una email all'utente `bobo@polimi.it` contenente un testo di 10 KB e un allegato di 3 MB. Assumendo che il destinatario utilizzi POP3 in modalità *scarica e cancella*, si calcoli il tempo necessario alla ricezione della email, date le seguenti caratteristiche:

- il link L , che collega Alice al suo router di confine, è caratterizzato da ampiezza di banda $R_L = 20$ Mbps e tempo di propagazione pari a 1 ms;
- la LAN Polimi, che collega Bobo e il server di posta di Polimi al loro router di confine, è caratterizzata da ampiezza di banda $R_L = 100$ Mbps e tempo di propagazione pari a 0.1 ms;
- i restanti collegamenti sono caratterizzati da ampiezza di banda $R = 50$ Mbps e tempo di propagazione pari a 0.1 ms;
- per tutti i collegamenti si consideri $MSS=1024$ Byte e overhead di 40 Byte.



Quesito 2

Dati due host connessi come indicato in figura, si supponga che il primo host inizi a spedire un file di 10 kB verso l'altro mediante il protocollo TCP, con valore iniziale della soglia sulla finestra di congestione pari a $8MSS$. Si assuma che i canali abbiano ampiezze di banda rispettivamente pari a 200 Mbps e 100 Mbps, che $RTT=0.4$ ms, $MSS=512$ byte e che gli overhead siano trascurabili. Si determini il throughput della trasmissione, assumendo che non si verifichi congestione. Si evidenzii inoltre l'andamento della finestra di congestione.



[Continua sul retro]

Quesito 3

Avendo a disposizione il range di indirizzi 147.163.124.0/22 si proponga uno schema di indirizzamento per la configurazione indicata nella figura a sinistra che minimizzi lo spreco di indirizzi per ciascuna sottorete e risulti coerente con la tabella di inoltro data per il router R_0 .

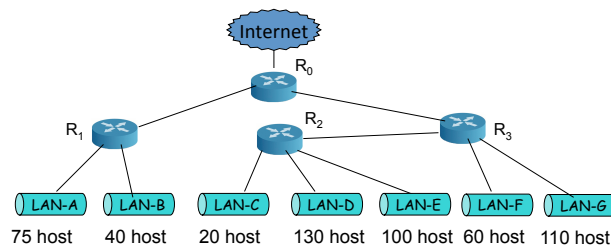


Tabella di Inoltro di R_0

Prefisso	Interfaccia
147.163.124.0 /24	R1
147.163.124.0 /22	R3
0.0.0.0 /0	Internet

Quesito 4

Completare il codice fornito per realizzare il server di una semplice applicazione DNS tramite protocollo UDP. L'applicazione prevede che il client legga un testo dallo standard input una stringa contenente un nome simbolico e la invii al server; il server riceve il messaggio, estrae la stringa rappresentante il nome simbolico da tradurre, interroga il proprio DB attraverso opportune funzioni di libreria, e invia al client l'indirizzo IP corrispondente; il client legge la risposta ottenuta e la manda sullo standard output.

```
#include <stdio.h>
#include <stdlib.h>      /* exit() */
#include <strings.h>     /* bzero(), bcopy() */
#include <unistd.h>
#include <netinet/in.h>
#include <time.h>        /* time_t e ctime() */
#include <sys/types.h>   /* tipi di dati di sistema */
#include <sys/socket.h> /* definizioni utili per le socket() */

#define MAXLINE 4096

void error(char *msg) {
    perror(msg);
    exit(1);
}

int main(int argc, char *argv[]) {
    int sockfd;
    int server_port = 2053;
    int clilen, bytesread;
    struct sockaddr_in serv_addr, cli_addr;
    char buffer1[MAXLINE], buffer2[MAXLINE];
```

[Continua ...]



```
.....  
.....  
.....  
  
bzero((char *) &serv_addr, sizeof(serv_addr));  
portno = atoi(argv[1]);  
serv_addr.sin_family = AF_INET;  
serv_addr.sin_addr.s_addr = INADDR_ANY;  
serv_addr.sin_port = htons(portno);  
  
while(1){  
  
.....  
.....  
.....  
.....  
.....  
.....  
  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
  
}  
}
```

Note:

Per le dimensioni relative ai file si considerino le grandezze come potenze di 2 e quindi in particolare:

1 MB = 1.024 kB 1 kB = 1.024 byte

Per le dimensioni relative ai tassi di trasmissione e alle ampiezze di banda si considerino le grandezze come potenze di 10 e quindi in particolare:

1 kbps = 1.000 bps 1 Mbps = 1.000.000 bps