Esercitazione #4

Antonio Brogi

Dipartimento di Informatica
Università di Pisa

Finora in aula

1. Lun 21/09/2015 16:00-18:00 (*2:0 h*) lezione: Introduzione al corso. Introduzione alle reti. (Antonio Brogi)
2. Mar 22/09/2015 09:00-11:00 (*2:0 h*) lezione: Introduzione alle reti. (Antonio Brogi)
3. Lun 28/09/2015 16:00-18:00 (*2:0 h*) lezione: Livello application: concetti generali, WWW. (Antonio Brogi)
4. Mar 29/09/2015 09:00-11:00 (*2:0 h*) lezione: Livello application: WWW, FTP, email. (Antonio Brogi)
5. Lun 05/10/2015 16:00-17:00 (*1:0 h*) lezione: Livello application: DNS. (Antonio Brogi)
6. Lun 05/10/2015 17:00-18:00 (*1:0 h*) esercitazione: Soluzione homework assignment #1. (Antonio Brogi)
7. Mar 06/10/2015 09:00-11:00 (*2:0 h*) lezione: Livello transport: concetti generali, UDP, Stop&Wait. (Antonio Brogi)
8. Lun 12/10/2015 16:00-18:00 (*2:0 h*) lezione: Livello transport: GBN, SR. (Antonio Brogi)
9. Mar 13/10/2015 09:00-10:00 (*1:0 h*) lezione: Livello transport: TCP. (Antonio Brogi)
10. Lun 19/10/2015 16:00-18:00 (*2:0 h*) lezione: Livello transport: TCP. (Antonio Brogi)
11. Mar 20/10/2015 09:00-10:30 (*2:0 h*) lezione: Livello transport: TCP. (Antonio Brogi)
12. Lun 26/10/2015 16:00-18:00 (*2:0 h*) esercitazione: Esercitazione su ritardi e livelli application e transport. (Antonio Brogi)
13. Mar 27/10/2015 09:00-11:00 (*2:0 h*) esercitazione: Esercitazione su livelli e livello transport. (Antonio Brogi)

<table>
<tr><th></th><th>Lezioni in aula</th><th>Esercitazioni in aula</th></tr>
<tr><td>Introduzione al corso e alle reti</td><td>4 h</td><td rowspan="3">5 h</td></tr>
<tr><td>Livello application</td><td>5 h</td></tr>
<tr><td>Livello transport</td><td>9 h</td></tr>
<tr><td></td><td colspan="2">23 h</td></tr>
</table>

Modulo di RETI DI CALCOLATORI/A (a.a. 2015/2016)
Programma svolto

Introduzione al corso
Obiettivi, contenuti, testo di riferimento, organizzazione del corso, modalità d'esame.
→ *pagina Web del corso*

Introduzione alle reti di calcolatori
Una panoramica di Internet, ritardi nelle reti a commutazione di pacchetto, organizzazione dei protocolli in livelli.
→ *par. 1.1, 1.2 (escluso 1.2.3), 1.4 di [FM]; lucidi su ritardi e perdite.*

Livello application
Concetti generali, paradigma client-server, applicazioni (Web, FTP, email, TELNET), DNS.
→ *par. 2.1, 2.2, 2.3 (escluso 2.3.5) di [FM]*

Livello transport
Concetti generali, protocolli di livello transport: UDP, Stop&Wait, GBN, SR, TCP.
→ *cap. 3 di [FM]; lucidi su Stop&Wait, GBN, SR, TCP Reno.*

Il livello network
Concetti generali, protocolli di livello network (formato datagram e indirizzi IP, DHCP, NAT, forwarding, ICMP), routing unicast (routing DV, LS, gerarchico, RIP, OSPF, BGP), IPv6.
→ *par. 4.1.1, 4.1.2, 4.2, 4.3, 4.5 di [FM]; lucidi su NAT.*

Il livello link
Concetti generali, protocolli ad accesso multiplo, indirizzi a livello link, LAN cablate, dispositivi di interconnessione.
→ *cap. 5 (escluso 5.2) di [FM]*

LAN wireless
Concetti generali, IEEE 802.11.
→ *par. 6.1.1, 6.1.2 di [FM]*

Introduzione alle applicazioni peer-to-peer
Directory centralizzata, query flooding, copertura gerarchica, BitTorrent, Chord.
→ *par. 2.4 di [FM]; par. IV of [Sto]*

Introduzione alla sicurezza nelle reti
Concetti generali, cifratura a chiave simmetrica, cifratura a chiave asimmetrica, MAC e firme digitali, protocolli di autenticazione, IPsec.
→ *par.9.1, 9.2 (escluso 9.2.8), 9.3.3 di [FM]*

Riferimenti

- *[FM] B.A. Forouzan & F. Mosharraf. Reti di calcolatori. McGraw-Hill, 2013.*
- *[Sto] I. Stoica et al.. Chord: A Scalable Peer-to-peer Lookup Protocol for Internet Applications. IEEE/ACM Transactions on Networking, 11(1), 2003.*
- *Alcuni dei lucidi utilizzati dal docente, disponibili sulla pagina Web del corso.*

RCL: Reti di Calcolatori e Laboratorio *(Corso di Laurea in Informatica - corso A - a.a. 2015/2016)*

Antonio Brogi

Informazioni generali

- L'insegnamento di "Reti di Calcolatori e Laboratorio (RCL)" è strutturato in due moduli didattici: "Reti di Calcolatori" e "Laboratorio di Programmazione di Rete".
- Modalità d'esame
- Suddivisione in corsi A e B
- Informazioni per gli studenti ancora iscritti all' ordinamento "classe 26"

Modulo "Reti di calcolatori"

- **Avvisi.** Nessuno.
- **Obiettivo.** Introdurre gli aspetti fondamentali delle reti di calcolatori illustrando applicazioni e protocolli relativi al modello TCP/IP.
- **Breve descrizione dei contenuti.** Introduzione alle reti, protocolli (dal livello application al livello link), introduzione alla sicurezza nelle reti e ai sistemi P2P.
- **Materiale didattico.**
  - I testo adottato:
    - *B.A. Forouzan & F. Mosharraf. Reti di calcolatori. McGraw-Hill.* [integrazioni ed errata corrige]
    - *I. Stoica et al.. Chord: A Scalable Peer-to-peer Lo[illegible]o Protocol for Internet Applications. IEEE/ACM Transactions on Networking, 11(1), 2003.*
  - Lucidi: ritardi e perdite, S&W, GBN, SR, Reno.
  - Homework assignment: HA1, HA2, HA3, HA4, HA5
  - Esercitazioni svolte: E1, E2, E3.
  - Approfondimenti suggeriti.
  - Programma svolto.
  - Registro delle lezioni.
  - [Altri link: RFC, cookiecentral, NSLookup, Traceroute, PGP.]
- **Testi delle prove scritte** (con traccia delle soluzioni):
  - a.a. 2014/2015: 2015.06.11, 2015.07.14, 2015.09.11, 2015.11.03 STR.
  - a.a. 2013/2014: 2014.06.10, 2014.07.15, 2014.09.10, 2014.11.07 STR, 2015.01.21, 2015.02.12, 2015.04.09 STR.
  - a.a. 2012/2013: 2013.06.05, 2013.06.26, 2013.07.17, 2013.09.05, 2013.11.05 STR, 2014.01.16, 2014.02.06, 2014.04.01 STR.

Modulo "Laboratorio di Programmazione di Rete"

RCL: Reti di Calcolatori e Laboratorio *(Corso di Laurea in Informatica - corso A - a.a. 2015/2016)*

Antonio Brogi

Informazioni generali

- L'insegnamento di "Reti di Calcolatori e Laboratorio (RCL)" è strutturato in due moduli didattici: "Reti di Calcolatori" e "Laboratorio di Programmazione di Rete".
- Modalità d'esame
- Suddivisione in corsi A e B
- Informazioni per gli studenti ancora iscritti all' ordinamento "classe 26"

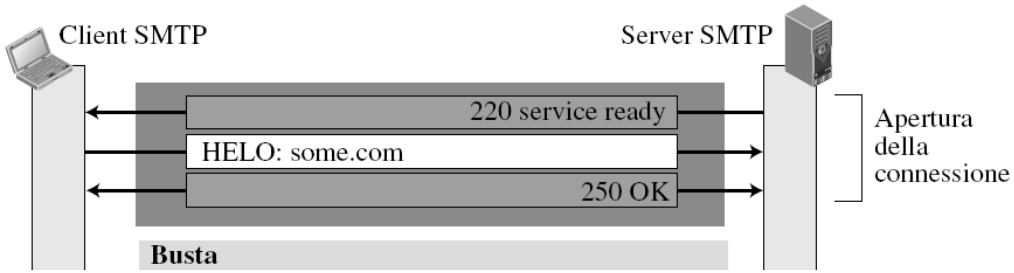
Modulo "Reti di calcolatori"

- **Avvisi.** Nessuno.
- **Obiettivo.** Introdurre gli aspetti fondamentali delle reti di calcolatori illustrando applicazioni e protocolli relativi al modello TCP/IP.
- **Breve descrizione dei contenuti.** Introduzione alle reti, protocolli (dal livello application al livello link), introduzione alla sicurezza nelle reti e ai sistemi P2P.
- **Materiale didattico.**
  - I testo adottato:
    - *B.A. Forouzan & F. Mosharraf. Reti di calcolatori. McGraw-Hill.* [integrazioni ed errata corrige]
    - *I. Stoica et al.. Chord: A Scalable Peer-to-peer Lookup Protocol for Internet Applications. IEEE/ACM Transactions on Networking, 11(1), 2003.*
  - Lucidi: ritardi e perdite, S&W, GBN, SR, Reno.
  - Homework assignment: HA1, HA2, HA3, HA4, HA5.
  - Esercitazioni svolte: E1, E2, E3.
  - Approfondimenti suggeriti.
  - Programma svolto.
  - Registro delle lezioni.
  - [Altri link: RFC, cookiecentral, NSLookup, Traceroute, PGP.]
- **Testi delle prove scritte** (con traccia delle soluzioni):
  - a.a. 2014/2015: 2015.06.11, 2015.07.14, 2015.09.11, 2015.11.03 STR.
  - a.a. 2013/2014: 2014.06.10, 2014.07.15, 2014.09.10, 2014.11.07 STR, 2015.01.21, 2015.02.12, 2015.04.09 STR.
  - a.a. 2012/2013: 2013.06.05, 2013.06.26, 2013.07.17, 2013.09.05, 2013.11.05 STR, 2014.01.16, 2014.02.06, 2014.04.01 STR.

Modulo "Laboratorio di Programmazione di Rete"

HA5/Livello transport

Q. Vasco (il cui indirizzo di email è vascor@universal.com) invia un email a Francesco (il cui indirizzo di email è francescodg@sony.com) contenente il testo “ci vediamo alle 11 al Roxy Bar”. Supponendo che durante la fase di handshake il TCP dell’host di Vasco invii 432 come valore iniziale del numero di sequenza, indicare -giustificando la risposta- quale valore conterrà il campo *ackNum* del secondo segmento che il TCP dell’host di Vasco riceverà, supponendo che non si verifichino perdite di pacchetti.

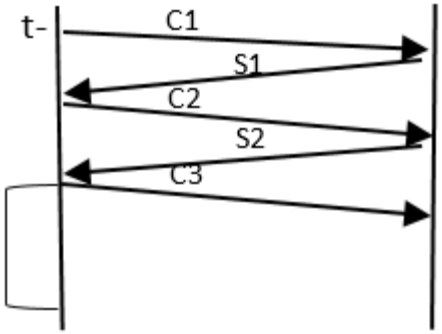


Poiché il primo messaggio SMTP (“220 service ready”) verrà inviato dal server, il secondo segmento inviato dal TCP dell’host di Vasco non conterrà alcun dato e non “consumerà” quindi il numero di sequenza 433. Di conseguenza il campo *ackNum* del secondo segmento ricevuto dal TCP dell’host di Vasco conterrà lo stesso numero di sequenza del precedente segmento ricevuto (“SYN+ACK”), ovvero 433.

Livello transport

Q. Un client C ha stabilito una connessione TCP con un server web S per scaricare una pagina web che consiste di tre oggetti. Al tempo t, subito dopo avere inviato la richiesta per il terzo oggetto, l'host di C invia a S un segmento con il flag FIN a true. Indicare – giustificando la risposta – il tempo minimo necessario al TCP di C per chiudere definitivamente la connessione supponendo che la dimensione del terzo oggetto sia 1,5 MSS, che RTT sia costantemente 700 msec, che il *maximum segment lifetime* sia 1100 msec, che tutti i segmenti vengano ricevuti corretti ed in ordine, e che il valore di *cwnd* del TCP di S sia 1MSS quando esso riceve il FIN inviato dall'host di C. Trascurare i tempi di preparazione e di trasmissione dei segmenti e assumere che S abbia già a disposizione tutti i dati da inviare.

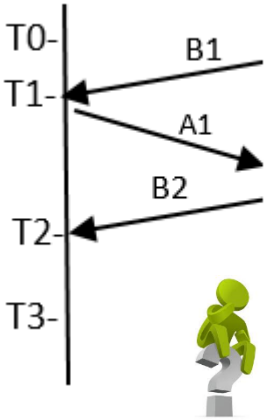
Q. Il TCP di C riceve al tempo t + RTT il riscontro S1 del segmento FIN da lui inviato. Se S1 trasporta in piggybacking il primo MSS dei dati del terzo oggetto, il TCP di C invierà* un riscontro C2 per tali dati e quindi il TCP di S invierà un segmento S2 con il flag FIN a true contenente la seconda parte dei dati del terzo oggetto. Il TCP di C invierà un riscontro C3 di S2 e considererà chiusa la connessione dopo avere atteso 2MSL, ovvero al tempo t + 2 RTT + 2 MSL = t + 3600 msec.



* Supponiamo per semplicità che il TCP di A non ritardi l'invio dei riscontri quando riceve segmenti "in ordine".

Livello transport

E. Al tempo T0 il TCP di un host A ha già stabilito una connessione con il TCP di un altro host B, ha 2 MSS di dati in volo (spediti in due segmenti *full-sized*), 2 MSS di dati non ancora spediti, il numero di sequenza del segmento più vecchio in volo è X, la dimensione della finestra di congestione *cwnd* è 6 MSS e il valore di *ssthresh* è 5,5 MSS. Al tempo T1>T0 riceve da B un riscontro B1 non duplicato e non contenente dati e in conseguenza di ciò invia un segmento A1 contenente 1 MSS di dati. Al tempo T2>T1 riceve da B un altro riscontro B2. Sapendo che nell'intervallo [T0,T3] non si verifica alcun altro evento oltre quelli sopra menzionati, indicare –giustificando la risposta– i possibili valori dei campi *ackNum* e *rwnd* contenuti in B1 e B2 e di *cwnd* al tempo T3.

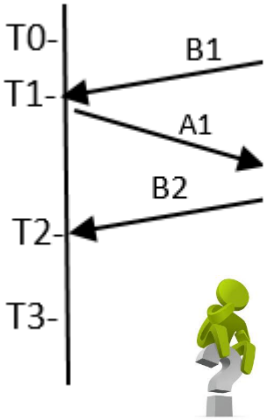


E2. Osserviamo che in TO il TCP di A si trova nello stato di congestion avoidance dato che $ssthresh_{T0} < cwnd_{T0} < ssthresh_{T0}+3$. Subito dopo avere ricevuto B1, TCP rimane in congestion avoidance e cwnd diventa (6+1/6) MSS = 37/6 MSS. Analizziamo i vari casi possibili:

- Se B1.ackNum=X+2MSS allora, dopo avere ricevuto B1, min(cwnd,B1.rwnd)=1MSS dato che viene spedito 1 solo MSS di nuovi dati, quindi B1.rwnd=1MSS.
  - Se B2.ackNum=X+3MSS allora, dopo avere ricevuto B2, min(cwnd,B2.rwnd)=0 MSS dato che non vengono spediti nuovi dati, quindi B2.rwnd=0MSS e cwndT3 = (37/6+6/37) MSS =1405/222 MSS.
  - Se B2 è un riscontro duplicato il valore di B2.rwnd non è significativo, dato che B2 viene scartato, e cwndT3 = 37/6 MSS.

Livello transport

E. Al tempo T0 il TCP di un host A ha già stabilito una connessione con il TCP di un altro host B, ha 2 MSS di dati in volo (spediti in due segmenti *full-sized*), 2 MSS di dati non ancora spediti, il numero di sequenza del segmento più vecchio in volo è X, la dimensione della finestra di congestione *cwnd* è 6 MSS e il valore di *ssthresh* è 5,5 MSS. Al tempo T1>T0 riceve da B un riscontro B1 non duplicato e non contenente dati e in conseguenza di ciò invia un segmento A1 contenente 1 MSS di dati. Al tempo T2>T1 riceve da B un altro riscontro B2. Sapendo che nell'intervallo [T0,T3] non si verifica alcun altro evento oltre quelli sopra menzionati, indicare –giustificando la risposta– i possibili valori dei campi *ackNum* e *rwnd* contenuti in B1 e B2 e di *cwnd* al tempo T3.



- Se B1.ackNum=X+1MSS allora, dopo avere ricevuto B1, min(cwnd,B1.rwnd)=2MSS dato che viene spedito 1 solo MSS di nuovi dati, quindi B1.rwnd=2MSS.
  - Se B2.ackNum=X+3MSS allora, dopo avere ricevuto B2, min(cwnd,B2.rwnd)=0 MSS dato che non vengono spediti nuovi dati, quindi B2.rwnd=0MSS e cwndT3 = 1405/222 MSS.
  - Se B2.ackNum=X+2MSS allora, dopo avere ricevuto B2, min(cwnd,B2.rwnd)≤1MSS dato che non vengono spediti nuovi dati, quindi B2.rwnd≤1MSS e cwndT3 = 1405/222 MSS.
  - Se B2 è un riscontro duplicato il valore di B2.rwnd non è significativo, dato che B2 viene scartato, e cwndT3 = 37/6 MSS.

Introduzione alle reti di calcolatori

- Ritardi HA1/Q1 + ES2/Q
- Livelli ES3/esempio

Livello application

- HTTP HA1/E1
- FTP HA1/E2 + AS1
- Email HA1/Q2 + AS2
- DNS HA2/Q3 + AS3

Livello transport

HA5

- UDP HA3/Q1
- S&W AS4
- GBN HA3/E1
- SR HA3/Q2 + AS5
- TCP HA4/Q1 + HA4/E1 **+ ES4/Q + ES4/E**