

RETI DI CALCOLATORI - Prima verifica in itinere a.a. 2019/2020 - compito A

*****Soluzioni*****

Q1 (3 punti) I record DNS contengono 4 campi. Il significato del valore di alcuni di questi campi dipende dal valore del campo Type.

Descrivere il formato dei record DNS di tipo NS e MX e specificare il significato dei possibili valori assunti dai campi del record compilando le tabelle sottostanti.

Soluzione

Record di tipo NS:

Nome campo	Possibili valori
Type	NS
Name	Nome di dominio (es. unipi.it)
Value	l'hostname dell'autoritative name server per quel dominio
TTL	Tempo di vita del record: indica quando il record deve essere rimosso dalla cache

Record di tipo MX:

Nome campo	Possibili valori
Type	MX
Name	Nome di dominio (es. di.unipi.it)
Value	Nome canonico del server di posta (es. mail.di.unipi.it)
TTL	Vedi sopra

Q2 (5 punti). Si supponga che la finestra di congestione congwin sia 16 KB e avvenga un timeout. Dire qual è la dimensione della finestra di congestione e lo stato del TCP dopo 4 RTT. Si supponga che l'invio dei dati avvenga con successo, $\text{MSS} = 1\text{KB}$ e venga riscontrato ciascun segmento inviato.

Soluzione

Dopo un timeout il valore della soglia va a $\text{congwin}/2$ e congwin cresce in modo esponenziale (stato slow start), finché $\text{congwin} \geq \text{soglia}$. Da quel punto l'andamento è di incremento lineare (congestion avoidance). Si veda l'evoluzione in tabella

Tempo	Congwin	Soglia	Stato
0	1 MSS	8 MSS	SS
RTT	2 MSS	8 MSS	SS
2 RTT	4 MSS	8 MSS	SS
3 RTT	8 MSS	8 MSS	SS-> CA
4 RTT	9 MSS= 9KB	8 MSS	CA

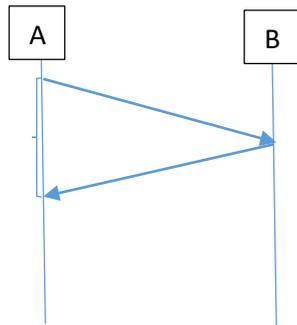
Q3 (8 punti) Sapendo che un pacchetto impiega 100mS per giungere dall'host A (client) all'host B (server), calcolare, giustificando la risposta, il tempo minimo necessario per soddisfare una richiesta DNS nei seguenti casi:

- 1) si usi UDP
- 2) si usi TCP (in questo caso includere anche il tempo di chiusura della connessione - lato host A).

Si ipotizzi che il server B abbia il record di risposta e che non ci siano perdite.

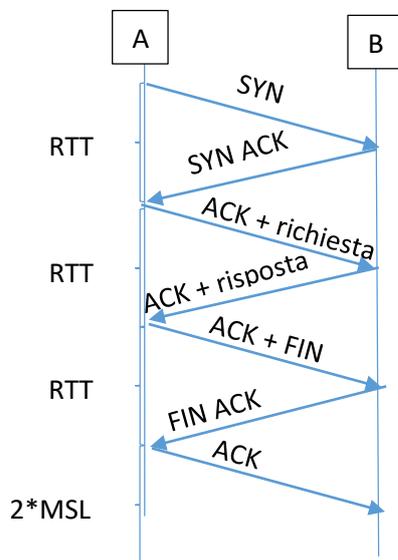
Soluzione

a) UDP - tempo tra invio richiesta e ricezione risposta (trascurando tempo di elaborazione e



ipotizzando che il tempo da B ad A sia 100 ms): 200 ms

b) TCP - 400 ms (incluso tempo di handshake) + (200 ms + 2 MSL) chiusura connessione, ipotizzando



che la richiesta sia inviata in piggybacking nel terzo messaggio dell'handshake.¹

Q4. (8 PUNTI) Supponiamo che un server FTP in esecuzione su un host B riceva da un cliente in esecuzione su un host A un segmento S con sourcePort=U, destinationPort=L, seqNumber=X, ackNumber=Y, flag ack a 1 e come dati la stringa "STOR file.PDF". Indicare i valori dei campi sourcePort, destinationPort, seqNumber, ackNumber, eventuali flag a 1 e contenuto della parte dati dei primi due segmenti che B invia ad A dopo avere ricevuto S. Si ipotizzi che il client abbia precedentemente comunicato al server di essere in ascolto sulla porta D per la connessione dati (active mode).

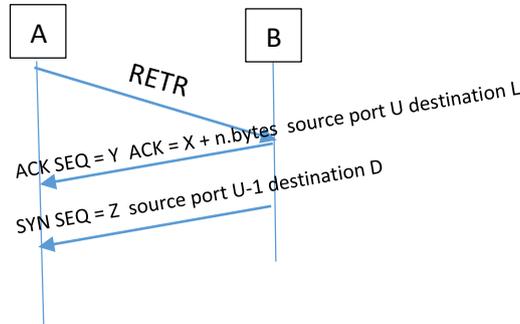
Soluzione

Il primo segmento sarà un riscontro di S e quindi conterrà: sourcePort=L, destinationPort=U, seqNumber=Y², flag ack a 1 e ackNumber=X+Δ, dove Δ è il numero di byte corrispondenti alla stringa "STOR file.PDF". La parte dati del primo segmento conterrà inoltre "piggybacked" la risposta FTP inviata dal server in risposta al comando ricevuto.

¹ E' possibile ipotizzare che il FIN sia inviato nel terzo messaggio (l'applicazione invia questo messaggio e chiede di chiudere la connessione) e analogamente il secondo FIN sia inviato con la risposta. In quel caso si ha 400 ms+2 MSL

² Assumendo che Y-1 sia l'ultimo byte di dati che B ha spedito sulla connessione di controllo.

Se il server può ricevere il file indicato³ il secondo segmento sarà un "SYN" inviato dall'host di B per aprire la connessione dati e quindi conterrà: `sourcePort=L-1`, `destinationPort=D`, `seqNumber=Z` (dove Z è il numero di sequenza scelto dall'host di B per la nuova connessione), flag `syn` a 1 e nessun dato.



Q5 (6 punti) Illustrare, tenendo conto della stratificazione, cosa si intende per pseudo header e quale sia la sua utilità.

Soluzione

Lo pseudoheader è usato nel livello di trasporto (protocolli TCP e UDP) e serve per il calcolo della checksum (in UDP opzionale in TCP obbligatorio) ai fini del controllo degli errori.

Il formato è descritto in tabella. Contiene IP sorgente e destinazione, il campo Protocol (vale 17 per UDP e 6 per TCP), lunghezza del messaggio (header TCP/UDP più payload).

0	8	16	31
Indirizzo IP di Provenienza			
Indirizzo IP di Destinazione			
Zero	Proto	Lunghhezza UDP/TCP	

Ha valenza logica, queste informazioni non vengono trasmesse a livello di trasporto. Queste informazioni vengono poi inserite effettivamente e trasmesse dal livello sottostante (rete).

Lato mittente, si tratta il contenuto del segmento/datagramma UDP (incluso pseudoheader) come una sequenza di interi da 16 bit. La checksum viene calcolata come la somma (complemento a 1) dei contenuti del segmento. Il mittente pone il valore della checksum nel campo checksum del segmento/datagramma UDP. Il destinatario calcola la checksum del segmento/datagramma UDP ricevuto, e confronta il risultato con il valore del campo checksum, se non è uguale il segmento viene scartato.

³ Ovvero se il server considera valida la richiesta del cliente e il cliente ha i diritti per inviarlo.