CS305 HW-2 王奕萱 11910104

~~1. rcvd from: command of SMTP protocol~~
~~from: a part of message body~~

~~2. SMTP: using a "."~~
~~HTTP~~

1. mail from: an important identifier indicating the receiver's mail server in SMTP
from: just a part of mail message body

2. SMTP: a line containing only a "."
HTTP: use content-length in the header field.

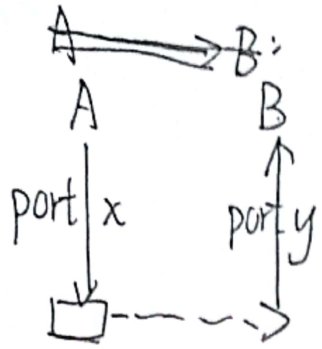
No. HTTP stores data in the binary form, while SMTP uses ASCII (human-readable) to save data. Thus they cannot use same method to save data.

3. We can select a period randomly. Then for the period, we can collect the frequency of ~~users~~ web servers among users. For those servers with higher frequency, they are more popular among users.

~~4. $F = 15Gb$, $U_s = 30Mb/s$, $D_i = 2Mb/s$~~
~~$N = 10,100$~~

5) a. in this case, we need to store $N*N = N^2$ files.
b. in the synchronized case, there are $N+N = 2N$ files to store.

6. a. This is trying to make a TCP connection with a server which is not existing. Thus the connection cannot be built.
b. UDP does not require connection. Thus the UDPClient can execute and input a sentence.
c. ~~Error will occur~~ This will cause Exception. Since their port number does not match. the client will find a wrong process.

7. 

Here we can see: host A communicates through port x, while host B communicate through port y.

Thus for the travelling from host B to host A, source port: y. destination port x

8. Here are several reasons:
① TCP has congestion control, while UDP not. This may cause the decrease of ~~transi~~ data transmission rate.
② Some applications do not need reliable data transmission.

9 sum of these 8-bit bytes:

(a) 01010011
01100110
+ 01110100
100101101
↓
overflow, omitted

⇒ result is 00101110
complement: 11010001

(b) This is to detect the transmission error.
If only use the sum. then some error cases cannot be detected.
e.g. 01010011, 01100110, 0111 0100
transmission error
00010011, 01000110, 0111 0100
Thus the data is incorrect, but sum remains the original

(c) Sum should not contain 0; Thus for those bits with 0 in the sum there are no errors

(d) 1-bit error can be detected

(e) 2-bit error may not be detected

4. $F = 15G$, to N peers.

$u_s = 30Mb/s$, $D_i = 2Mb/s$

(1) For the client-server model,

$$D_{cs} \geq \max\left\{\frac{NF}{u_s}, \frac{F}{d_{min}}\right\}$$

$$\frac{F}{d_{min}} = \frac{15 \times 1024M}{2M} = 7680s$$

④ ~~$N=10$, $u=$~~

① $N=10$, $u=300Kb/s$

$$\frac{NF}{u_s} = \frac{10 \times 15 \times 1024M}{30M/s} = 5120s, \quad D_{cs} = 7680$$

② $N=100$, $u=300Kb/s$.

$$\frac{NF}{u_s} = 51200s. \quad D_{cs} = 51200s$$

③ $N=1000$, $u=300Kb/s$

$D_{cs} = 512000s$

Since D_{cs} is not relevant to u, the chart is given

N / u	10	100	1000
300K	7680	51200	512000
700K	7680	51200	512000
2M	7680	51200	512000

(2) For the P2P model

$$D_{cs} \geq \max\left\{\frac{F}{u_s}, \frac{F}{d_{min}}, \frac{NF}{u_s + \sum_{i=1}^{n} u_i}\right\}$$

① $u = 300K$

$$\frac{F}{u_s} = \frac{15 \times 1024M}{30M} = 512s$$

$$\frac{F}{d_{min}} = \frac{15 \times 1024M}{2M} = 7680s$$

$\mu_i = 300$ Kb/s

for $N=10$, $\frac{NF}{\mu_s + N\times\mu_i} = \frac{10\times 15\times 1024M}{30M+10\times 300K} = 4654s$, $D_{p2p} = 7680s$

for $N=100$, $\frac{NF}{\mu_s+N\times\mu_i} = 25606s$ ~~25606s~~ $25903s$, $D_{p2p} = 25903s$

for $N=1000$, $\frac{NF}{\mu_s+N\times\mu_i} = 47559s$, $D_{p2p} = 47559s$

② $\mu_i = 700$ Kb/s

for $N=10$, $\frac{NF}{\mu_s+N\times\mu_i} = 4169s$ ~~4169s~~ $4170s$, $D_{p2p} = 7680s$

for $N=100$, $\frac{NF}{\mu_s+N\times\mu_i} = 15616s$, $D_{p2p} = 15616s$

for $N=10000$, $\frac{NF}{\mu_s+N\times\mu_i} = 21525s$, $D_{p2p} = 21525s$

③ $\mu_i = 2$Mb/s

for $N=10$, $\frac{NF}{\mu_s+N\times\mu_i} = 3072s$. $D_{p2p} = 7680s$

for $N=100$, $\frac{NF}{\mu_s+N\times\mu_i} = 6678s$., $D_{p2p} = 7680s$

for $N=1000$, $\frac{NF}{\mu_s+N\times\mu_i} = 7567s$. $D_{p2p} = 7680s$

μ \ N	10	100	1000
300Kb/s	7680	25903	47559
700Kb/s	7680	15616	21525
2Mb/s	7680	7680	7680