

بسمه تعالی

دانشکده مهندسی برق و کامپیوتر - دانشگاه صنعتی اصفهان

حل تمرین 2 - شبکه های کامپیوتری

-1

در این مسئله نرخ هر اتصال 10 مگابیت داده شده است

حال به طور معمول درخواست اتصال TCP، ارسال ACK، تایید ACK و درخواست فایل اصلی و همچنین ارسال فایل را داریم لذا

$$250ms + 250ms + 250ms + 250ms + \frac{300kb}{10Mb} + \frac{400kb}{10Mb} = 1070ms$$

-2

$$T_i = \frac{L_i}{R} = D_{transaction}$$

به دلیل اینکه اتصال موازی مجاز است

$$4RTT + T_0 + \max_{i \in \{1, \dots, n\}} T_i$$

-3

اگر فرض کنیم هر کلاینت هر کدام بصورت مجزا با یک اتصال TCP به سرور متصل میشوند بنابراین به هر کلاینت $\frac{1}{10} R_s$ از پهنای باند سرور میرسد در این سوال 9 کلاینت هر کدام با یک اتصال TCP به سرور متصل میشوند و یک کلاینت با 9 اتصال TCP به سرور متصل میشود بنابراین 18 اتصال TCP به سرور خواهیم داشت که سهم هر یک $\frac{1}{18}$ از پهنای باند سرور است .

کلاینتی که از Download Manager استفاده میکند از $\frac{9}{18}$ پهنای باند سرور استفاده میکند و خواهیم داشت:

$$\frac{1}{10} R_s * X = \frac{9}{18} R_s \rightarrow X = 5$$

-4

- به ازاء همه شیء ها. حافظه پنهان وب محتوای مورد نظر کاربران را cache کرده و به کاربران نزدیک میکند (برای مثال به همان LAN که کاربر به آن متصل شده)، میتواند تاخیر را برای همه object ها حتی object هایی که cache نشده اند کاهش دهد زیرا web caching باعث کاهش ترافیک در لینک ها میشود.

- طبق گفته مسئله hit rate برابر 40 درصد است بنابراین 60 درصد hit نمیشود و همچنین ظرفیت لینک محلی web cache برابر 1Gbps است بنابراین خواهیم داشت:

$$R_{average} = 0.4 * 1Gbps + 0.6 * \min\{1Gbps, 100Mbps, 30Mbps\}$$

-5

- پروتکل SMTP

- پروتکل SMTP برای پایان متن پیام از یک خط فقط حاوی نقطه استفاده می کند.

پروتکل HTTP برای نشان دادن طول متن پیام از "Content-Length header field" استفاده میکند

خیر، HTTP نمی تواند از روش استفاده شده توسط SMTP استفاده کند زیرا پیام HTTP می تواند داده های باینری باشد، در حالی که در SMTP، متن پیام باید در قالب ASCII 7 بیتی باشد.

-6

این پروتکل وظایف یک پروتکل انتقال را در کمترین سطح ممکن انجام میدهد. UDP به پیام های لایه کاربرد شماره پورت مبدا و مقصد، سرآیند بعلاوه داده و checksum را میچسباند و بلافاصله قطعه حاصل را به لایه شبکه میدهد.

این پروتکل connectionless است همچنین unreliable data transfer است و کنترل ازدحام ندارد.

نداشتن زمان setup اولیه و همچنین اطلاعات اتصال را نگه نمیدارد. وقتی برنامه از UDP استفاده میکند سرور میتواند همزمان تعداد بیشتری کلاینت فعال را پشتیبانی کند.

پس (1) در UDP تاخیر اولیه ایجاد اتصال TCP را نداریم.

(2) برای اینکه کنترل جریان و کنترل ازدحام TCP محدودیتی برای ما ایجاد نکند.

ارسال پیام برنامه از طریق پروتکل لایه انتقال را در نظر بگیرید با استفاده از TCP برنامه اطلاعات را در بافر ارسال آن اتصال مینویسد و بایت ها را بدون اینکه یک پیام واحد را در سگمنت قرار دهد جدا میکند. TCP ممکن است بیشتر یا کمتر از یک پیام واحد را در یک سگمنت قرار دهد. از طرف دیگر UDP هر

بخشی را که برنامه به آن میدهد در یک سگمنت قرار میدهد یعنی اگر برنامه پیامی را به UDP بدهد این پیام قسمت داده سگمنت UDP خواهد بود بنابراین یک برنامه کنترل بیشتری روی داده های ارسال شده در یک سگمنت دارد.

موفق باشید.