



دانشکده مهندسی کامپیوتر

بسمه تعالی
دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)
دانشکده مهندسی کامپیوتر



دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

درس شبکه های کامپیوتری، نیم سال دوم سال تحصیلی ۱۴۰۱-۱۴۰۲

پایخ تمرین سری چهارم

سوال ۱:

پروتکل *FTP* از دو اتصال موازی *TCP* استفاده می کند که یکی از اتصال ها برای ارسال اطلاعات کنترلی (مانند درخواست انتقال فایل و برقراری ارتباط) و دیگری برای ارسال و دریافت فایل استفاده می شود. بنابراین چون اطلاعات کنترلی روی اتصالی که برای ارسال و دریافت پیام (فایل) استفاده نمی شود می گوئیم *FTP* اطلاعات کنترلی را به صورت خارج از باندی ارسال می کند. اگر برای ارسال اطلاعات کنترلی و ارسال داده اصلی از یک اتصال استفاده شود می گوئیم اطلاعات کنترلی به صورت داخل باند ۱ ارسال می شود. پروتکل های *HTTP* و *SMTP* اطلاعات کنترلی را به صورت داخل باند ارسال می کنند.

سوال ۲:

پس از آن که ارسال کننده ایمیل (کاربر) با *Mail server* یک اتصال ایجاد کرد، بین کاربر و *Mail server* پیام های *MAIL*، *HELO*، *FROM*، *RCPT TO*، *DATA*، بدنه ی پیام و *QUIT* از کاربر به *mail server* ارسال می شود، با ارسال هر یک از این پیام ها *mail server* یک پاسخ مناسب به کاربر می دهد و پس از آن اتصال ایجاد شده خاتمه می یابد. از آنجایی که اندازه پیام کوچک است بنابراین ارسال بدنه پیام در یک *round trip* صورت می گیرد و برای هر کدام از پیام های *HELO*، *MAIL FROM*، *RCPT TO*، *DATA* و پاسخ های آن ها نیز یک *round trip* نیاز داریم که در مجموع ۶ *round trip* لازم است.

سوال ۳:

با دانلود و حذف، پس از اینکه کاربر پیام های خود را از سرور *POP* بازیابی کرد، پیام ها حذف می شوند. این یک مشکل برای کاربری که بخواهد به پیام ها از دستگاه های مختلفی (مانند: کامپیوتر اداری، کامپیوتر خانگی و ...) دسترسی پیدا کند، بوجود می آورد. در پیکربندی دانلود و نگه داشتن، پیام ها پس از بازیابی توسط کاربر حذف نمی شوند. این همچنین می تواند ناخوشایند باشد، زیرا هر بار که کاربر پیام های ذخیره شده را از یک دستگاه جدید بازیابی می کند، همه پیام های حذف نشده از جمله پیام های بسیار قدیمی، به دستگاه جدید منتقل می شوند.

سوال ۴:

برای محاسبه حداقل زمان توزیع در معماری مشتری-سرویس دهنده از رابطه زیر استفاده می کنیم:

$$D_{CS} = \max \left\{ \frac{NF}{u_s}, \frac{F}{d_{min}} \right\}$$



پایخ تمرین سری چهارم

و برای محاسبه حداقل زمان توزیع در معماری نظیر به نظیر از رابطه زیر استفاده می‌شود:

$$D_{P2P} = \max \left\{ \frac{F}{u_s}, \frac{F}{d_{min}}, \frac{NF}{(u_s + \sum_{i=1}^N u_i)} \right\}$$

طبق صورت سؤال داریم:

$$F = 1 \text{ Gbits} = 1024 \text{ Mbits}$$

$$u_s = 24 \text{ Mbps}$$

$$d_{min} = d_i = 3 \text{ Mbps}$$

$$(\text{توجه کنید: } 400 \text{ Kbps} = \frac{400}{1024} \text{ Mbps})$$

N	مشتری-سرویس دهنده			نظیر به نظیر		
	10	100	1000	10	100	1000
$u = 400K$ $= 0.39M$	426.67	4266.67	42666.67	366.94	1623.78	2469.70
$u = 800K$ $= 0.78M$	426.67	4266.67	42666.67	341.33	1002.69	1271.65
$u = 3M$	426.67	4266.67	42666.67	341.33	341.33	341.33

سوال ۵:

(الف)

بله، این امکان وجود دارد. تا زمانی که *peer*های کافی برای مدت طولانی در *swarm* باقی بمانند، امید همیشه می‌تواند داده‌ها را از طریق *optimistic unchoking* توسط سایر *peer*ها دریافت کند.

(ب)

بله، درست است. او می‌تواند یک *Client* را روی هر *Host* اجرا کند و به هر *Client* اجازه دهد که از اطلاعات به صورت رایگان استفاده کند (*free-riding*) و تکه‌های جمع‌آوری‌شده از *Host*های مختلف را در یک فایل ترکیب کند. او حتی می‌تواند یک برنامه زمان‌بندی کوچک بنویسد تا *Host*های مختلف، بخش‌های مختلف فایل‌ها را درخواست کنند. این در واقع نوعی حمله *Sybil* در شبکه‌های *P2P* است.

(ج)

در ارتباط‌های *P2P* سرویس‌دهنده و سرویس‌گیرنده می‌توانند تغییر نقش داده و ثابت نباشد، از این مشخص کردن مرجع فایل‌ها در این روش دشوار است. مثلاً شما می‌توانید یک فایل را با دوست خود به اشتراک بگذارید و دوست شما فایل را با سایرین به اشتراک بگذارد. در روش‌های *Client* و *Server* نقش *Server* به صورت مشخص به تعدادی سیستم تخصیص می‌شود و از این رو می‌توان آن‌ها را شناسایی کرده و جلوی سرویس دادن آن‌ها را گرفت.



پایخ تمرین سری چهارم

به صورت خلاصه شناسایی مرجع یک فایل در سیستم $P2P$ از سیستم $Client-Server$ دشوارتر می باشد.

سوال ۶:

(الف)

u را به صورت زیر تعریف می کنیم:

$$u = u_l + \dots + u_N$$

با توجه به فرضیات داریم (رابطه اول):

$$u_s \leq (u_s + u)/N$$

فایل را به N بخش تقسیم کنید که اندازه ی بخش i ام برابر با $\left(\frac{u_i}{u}\right)F$ سرویس دهنده بخش i ام را با نرخ $r_i = \left(\frac{u_i}{u}\right)u_s$ به عضو i ام ارسال می کند. توجه داشته باشید که $r_1 + r_2 + \dots + r_N = u_s$ ، بنابراین مجموع نرخ سرویس دهنده از نرخ لینک آن بیشتر نخواهد بود. همچنین نظیر i ام بیت های دریافتی را با نرخ r_i به $N-1$ عضو دیگر ارسال می کند. مجموع نرخ ارسالی توسط نظیر i ام برابر است با: $(N-1)r_i$.

همچنین داریم:

$$(N-1)r_i = (N-1)\left(\frac{u_i}{u}\right)u_s$$

از رابطه اول داریم:

$$(N-1)u_s \leq u$$

$$(N-1)\left(\frac{u_i}{u}\right)u_s \leq u_i$$

بنابراین: $(N-1)\left(\frac{u_i}{u}\right)u_s \leq u_i$ یعنی مجموع نرخ ارسالی عضو i ام، کمتر از پهنای باند آپلود آن است (u_i) است. در این حالت، نرخ بیت دریافتی نظیر i ام برابر است با:

$$r_i + \sum_{j < i} r_j = u_s$$

در این حالت نظیر نرخ r_i را از سرویس دهنده و $\sum_{j < i} r_j$ را از بقیه نظیرها دریافت کرده است. بنابراین هر نظیر فایل را در زمان F/u_s دریافت می کند.

(ب)

u دوباره به صورت زیر تعریف می شود:

$$u = u_l + \dots + u_s$$

همچنین داریم:

$$u_s \geq (u_s + u_l + \dots + u_s)/N$$

عبارت های زیر را نیز تعریف می کنیم:

$$r_i = \frac{u_i}{N-1}$$



پایخ تمرین سری چهارم

$$r_{N+1} = \frac{\left(u_s - \frac{u}{N-1}\right)}{N}$$

در این توزیع فایل به $N+1$ بخش تقسیم می شود. سرویس دهنده بیت های مربوط به بخش i ام را با نرخ r_i به نظیر i ام ارسال می کند ($i = 1, \dots, N$). نظیر i ام بیت های دریافتی را با نرخ r_i به $N-1$ نظیر دیگر ارسال می کند. همچنین سرویس دهنده بیت های مربوط به بخش $(N+1)$ ام را با نرخ r_{N+1} به N نظیر دیگر ارسال می کند. نظیرها بیت های مربوط به بخش $N+1$ ام را ارسال نمی کنند.

مجموع نرخ ارسالی سرویس دهنده برابر است با:

$$r_1 + \dots + r_N + N r_{N+1} = u/(N-1) + u_s - u/(N-1) = u_s$$

بنابراین نرخ ارسالی سرویس دهنده بیشتر از نرخ مربوط به لینک آن نخواهد بود. نرخ ارسالی مربوط به نظیر i ام برابر است با:

$$(N-1) r_i = u_i$$

بنابراین نرخ ارسالی هر نظیر بیشتر از نرخ مربوط به لینکش نخواهد بود.

در این توزیع، نرخ دریافت برای نظیر i ام برابر است با:

$$r_i + r_{N+1} + \sum_{j < i} r_j = u/(N-1) + (u_s - u/(N-1))/N = (u_s + u)/N$$

بنابراین هر نظیر فایل را در زمان $NF/(u_s + u)$ دریافت می کند.

(ج)

با توجه به رابطه $D_{P2P} \geq \max\{F/u_s, NF/(u_s + u)\}$ و ترکیب بخش های a و b نتیجه ی مورد نظر حاصل خواهد شد.

سوال ۷:

(الف)

پایگاه داده *whois* یک فهرست آنلاین در دسترس عموم به هدف ارائه شفافیت و پاسخگویی در سیستم نام دامنه است. که به کاربران اجازه می دهد تا اطلاعات مربوط به یک دامنه خاص و مالک آن را دریافت کنند.

دیتابیس *whois* به ازای هر *domain name* اطلاعات مربوط به آن، از جمله آدرس *IP* و *DNS server* و اطلاعات مالک را نگهداری می کند.

(ب)

Domain:

cam.ac.uk



دانشگاه مهندسی کامپیوتر

درس شبکه های کامپیوتری، نیم سال دوم سال تحصیلی ۱۴۰۱-۱۴۰۲



دانشگاه صنعتی امیرکبیر
(پاسی تکنیک تهران)

صفحه: ۵ از ۶

پایخ تمرین سری چهارم

Registered For:

The University of Cambridge

Domain Owner:

University of Cambridge

Registered By:

Jisc Services Limited

Servers:

auth0.dns.cam.ac.uk	131.111.8.37
auth0.dns.cam.ac.uk	2001:630:212:8::d:a0
dns0.cl.cam.ac.uk	128.232.0.19
dns0.cl.cam.ac.uk	2a05:b400:110::d:a0
dns0.eng.cam.ac.uk	129.169.8.8
ns1.mythic-beasts.com	
ns2.ic.ac.uk	
ns3.mythic-beasts.com	

Registrant Contact:

Cambridge Hostmaster

Registrant Address:

Cambridge University Information Services
Roger Needham Building
7 JJ Thomson Avenue
Cambridge
Cambridgeshire
CB3 0RB
United Kingdom
+44.1223748440 (Phone)
hostmaster@cam.ac.uk



دانشگاه مهندسی کامپیوتر

درس شبکه های کامپیوتری، نیم سال دوم سال تحصیلی ۱۴۰۱-۱۴۰۲



دانشگاه صنعتی امیرکبیر
(پاسی تکنیک تهران)

صفحه: ۶ از ۶

پایخ تمرین سری چهارم

Renewal date:

Tuesday 13th Aug 2024

Entry updated:

Thursday 23rd March 2023

Entry created:

Wednesday 17th September 2003

(ج)

مهاجم می تواند از پایگاه داده *whois* و ابزار *nslookup* برای تعیین محدوده آدرس *IP*، آدرس های سرور *DNS* و ... برای موسسه مورد نظر استفاده کند.

(د)

با تجزیه و تحلیل آدرس منبع *packet* های حمله، قربانی می تواند از *whois* برای به دست آوردن اطلاعات در مورد دامنه ای که از آن حمله می شود استفاده کند و احتمالاً به مدیران دامنه مبدا اطلاع دهد.