سوال ۱: فرض کنید دریافت کننده سگمنت UDP، checksum را محاسبه کرده و دقیقا مطابق فیلد checksum در بسته رسیده باشد. آیا می توان مطمئن بود که هیچ خطایی رخ نداده است؟ اثبات کنید یا مثال نقض بیاورید.

خیر نمی‌توان مطمئن بود، امکان دارد تغییرات به نحوی باشد که checksum تغییر نکرده باشد ولی بسته تغییر کرده باشد. به طور مثال فرض کنید کلمات ۱۶ بیتی زیر در بسته اولیه موجود باشند:

W1: 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1

W2: 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0

و بسته‌ی دریافتی شامل کلمات زیر باشد:

RW1: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

RW2: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

محاسبه‌ی checksum خطای موجود در بسته را نمایش نمی‌دهد.

سوال ۲: در مورد روش Hybrid ARQ مطالعه کرده و به طور مختصر آن‌ را توضیح دهید.

در روش HARQ از روش‌های ARQ و FEC به طور ترکیبی استفاده می‌شود. در حقیقت در این روش از FEC برای رفع خطاهای قابل تصحیح و از ARQ برای رفع خطاهای غیر قابل تصحیح با FEC استفاده می‌شود.

سؤال ۳: یک مودم را در نظر بگیرید که برای اتصال یک کامپیوتر شخصی به یک سرویس‏دهنده استفاده می‏شود. سرعت مودم 56kbps و تأخیر انتشار یک‌طرفه 100ms است.

الف) اگر از روش کنترل خطای Stop-and-Wait استفاده شود، با فرض نرخ خطای بیت 10-4، برای طول فریم 256 بایت و 512 بایت، بهره‏وری را به دست آورید.

ب) اگر از روش کنترل خطای Go-Back-N استفاده شود، با فرض استفاده از شماره ترتیب 3بیتی و نرخ خطای بیت 10-4، برای طول فریم 256 بایت و 512 بایت، بهره‏وری را محاسبه کنید.

الف)

ب)

سؤال ۴: فرض کنید میزبان S یک سرویس‌دهنده وب روی پورت 80 اجرا می‌کند. این سرویس‌دهنده وب که از اتصال‌های پایا استفاده می‌کند در حال دریافت درخواست‌های دو میزبان مختلف A و B است. آیا همه‌ی این درخواست‌ها به یک سوکت واحد در میزبان S تحویل داده می‌شوند؟ اگر درخواست‌ها از سوکت‌های مختلف عبور می‌کنند، آیا هر دو سوکت شماره‌ی پورت 80 دارند؟ توضیح دهید.

برای هر ارتباط پایا، وب سرور یک سوکت جداگانه ایجاد می‏کند. هر سوکت ارتباطی با چهار شناسه‏ی آدرس IP مبدأ، شماره پورت مبدأ، آدرسIP مقصد و شماره پورت مقصد معرفی می‏شود. زمانی کهیک بسته‏ی IP توسط سرور دریافت می‏شود، با بررسی این چهار شناسه، داده‏های دریافتی به سوکت مورد نظرتحویل داده می‏شود. واضح است که این دو درخواست به سوکت‏های متفاوت تحویل داده می‏شوند . پورت مقصد هر رو درخواست، 80 است و آدرس IP مبدأ و شماره پورت مبدأ برای این دو درخواست متفاوت اند.

سوال ۵: درستی یا نادرستی موارد زیر را مشخص کنید و برای هر کدام دلیل بیاورید.

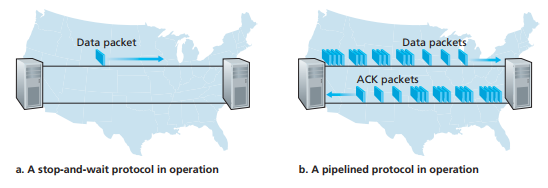
الف) در پروتکل GBN ممکن است فرستنده شماره ACK ای دریافت کند که خارج از پنجره ارسال است.

ب) در پروتکل SR ممکن است فرستنده شماره ACK ای دریافت کند که خارج از پنجره ارسال است.

هر دو قسمت صحیح هستند. برای هر دو قسمت می‌توان از سناریو زیر استفاده کرد:

فرض کنید فرستنده بسته‌های ۱، ۲ و ۳ را در زمان t0 ارسال می‌کند. گیرنده آن‌ها در در زمان t1 دریافت کرده و برای آن‌ها ACK ارسال می‌کند. فرستند در زمان t2 هنوز ACKها را دریافت نکرده است و Timeout می‌کند. در این زمان فرستنده این بسته‌ها را بازارسال می‌کند. گیرنده این بسته‌ها را در زمان t3 دریافت کرده و دوباره ACK می‌دهد. فرستنده ACKهایی که در زمان t1 ارسال شده بودند را در زمان t4 دریافت کرده و پنجره خود را به جلو می‌برد. حال در زمان t5 بسته‌های ACKای را دریافت می‌کند که در زمان t3 توسط گیرنده فرستاده شده بودند. این بسته‌های ACK مربوط به پنجره قبلی می‌باشند.

سؤال ۶: شکل زیر را در نظر بگیرید که در آن تاخیر رفت‌وبرگشت بین دو میزبان برابر با 30 میلی‌ثانیه است. فرض کنید نرخ انتقال داده کانالی که این دو میزبان را به یکدیگر متصل کرده Gbps 1 و اندازه‌ی بسته (سرایند+داده) 1500 بایت است. همچنین بسته‌های ACK آن‌قدر کوچک هستند که می‌توان از زمان انتقال آن‌ها صرف‌نظر کرد. برای داشتن بهره‌وری کانال بزرگ‌تر از 95 درصد، بزرگی اندازه پنجره ارسال چقدر باید باشد؟



سوال ۷: مقایسه SR و GBN: (اندازه پنجره را برابر w در نظر بگیرید.)

الف) در سمت فرستنده GBN و SR به چه تعداد زمان‌سنج یا تایمر نیاز است؟

ب) اگر w یک عدد زوج باشد و بسته با شماره ترتیب w/2 از بین برود، کدام بسته ها در هر کدام از پروتکل ها مجددا ارسال می شوند؟

ج) اگر محدودیت حافظه داشته باشیم و پهنای باند اضافی داشته باشیم، کدام پروتکل را ترجیح می دهید؟ چرا؟

د) اگر تاخیر دانلود برای کاربران بسیار مهم باشد، کدام پروتکل را ترجیح می دهید؟ چرا؟

الف) در GBN به یک زمان‌سنج نیاز است چرا که در صورتی که پنجره ارسال timeout شود کل پنجره باز ارسال می‌شود و این در حالی است که در SR برای هر بسته از پنجره به یک زمان‌سنج نیاز است بنابراین در GBN به w زمان‌سنج نیاز است.

ب)

SR: W/2

GBN: W/2, W/2 + 1, …, W + W/2

ج) استفاده از GBN نیاز به حافظه‌ی کمتری دارد.

د) استفاده از SR در زمان‌هایی که loss وجود دارد تاخیر را کاهش می‌دهد، می‌توان دو دلیل برای این امر ذکر کرد. اول به دلیل اینکه گیرنده بافر دارد بسته‌های خارج از ترتیب را حفظ می‌کند و به محض دریافت بسته‌ی loss شده همه را تحویل لایه بالاتر می‌دهد. در صورتی که در پروتکل GBN بسته‌های خارج از ترتیب حفظ نمی‌شوند و گیرنده بسته loss شده، باید منتظر دریافت بسته‌های خارج از ترتیب دریافتی حفظ نشده باشد. دوم در پروتکل SR علاوه بر مکانیزم timeout برای تشخیص loss یک بسته مکانیزم تشخیص زودهنگام (NACK یا Fast Retransmission) وجود دارد که فرستنده زودتر از مقدار timeout متوجه loss شدن یک بسته می‌شود و ارسال مجدد سریعتر صورت می‌گیرد.

سوال ۸: انواع روش‌های کنترل خطا (تشخیص و تصحیح خطا) را شرح دهید و همچنین توضیح دهید که در یک کاربرد خاص با مشخص بودن تاخیر، نرخ خطا و ... کدامیک از روش‌های FEC و ARQ مناسب‌تر هستند.

در روش‌های تشخیص خطا، خطا تشخیص داده شده اما تصحیح نخواهد شد. در روش‌های تصحیح اطلاعات اضافی در جهت تصحیح خطا وجود دارند. مثلا در ارتباط یک طرفه روش‌های FEC می‌بایست استفاده شوند. سربار FEC از سربار ARQ در خطای کم بیشتر است اما در صورتی که نرخ خطا بیشتر شود روش FEC موثرتر واقع می‌شود.

سوال ۹: با توجه به پروتکل S&W استفاده پاسخ دهید.

الف) فرض کنید فرستنده می‌خواهد 10 فریم ارسال کند. اگر فریم دوم و پنجم و پیام‌های ACK فریم سوم و ششم در اولین ارسال از بین بروند، ترتیب ارسال فریم‌ها را مشخص کنید. توجه‌ کنید که فریم‌هایی که بیش از یک بار ارسال می‌شوند باید به همراه تعداد ارسال مشخص باشند.

ب) با ارائه یک مثال نشان دهید اگر در این پروتکل پیام‌های ACK دارای Sequence number نبودند چه مشکلی پیش‌ می‌آمد.

الف)

1 2 2 3 3 4 5 5 6 6 7 8 9 10

ب)

- ارسال فریم صفر

- Time out و ارسال مجدد فریم صفر

- دریافت ACK مربوط به اولین ارسال فریم صفر

- ارسال فریم یک که در مسیر از بین می‌رود.

- دریافت ACK مربوط به دومین ارسال فریم صفر

- ارسال فریم دو

در این حالت پیام ACK صفر به اشتباه به عنوان ACK برای فریم یک در نظر گرفته شده است در حالی که این بسته اصلا به مقصد نرسیده است.

سؤال ۱۰: در مورد پروتکل‏های TCP و UDP به سؤالات زیر پاسخ دهید:

الف) هر دو پروتکل TCP و UDP از شماره‌ی پورت جهت شناسایی فرآیند مقصد استفاده می‌کنند. اگر بخواهیم از شناسه فرآیند در سیستم‌عامل به‌جای شماره‌ی پورت استفاده کنیم چه مشکلاتی پیش می‌آمد؟ دو مورد را بیان کنید.

ب) TCP و UDP از مکمل یک برای محاسبه checksum استفاده می‌کنند. فرض کنید داده شما از 64 بیت زیر تشکیل شده است:

1001000000101010001010101011110001000100100010010100011111100011

مکمل یک حاصل جمع این چهار کلمه‏ی 16بیتی چیست؟ تمامی مراحل را بنویسید.

ج) چرا از مکمل یک حاصل جمع استفاده می‌شود و از همان حاصل جمع استفاده نمی‌شود؟ اگر از مکمل یک استفاده نشود چه اتفاقی می‌افتد؟

د) آیا امکان دارد خطای یک بیتی وجود داشته باشد که تشخیص داده نشود؟ خطای دوبیتی چطور؟ مثال بزنید.

الف) دلیل اول:‌ شناسه‌های فرآیندها وابسته به سیستم‌عامل است و استفاده از آن‌ها باعث می‌شود پروتکل‌ها به سیستم‌عامل وابسته شوند.

دلیل دوم: یک فرآیند ممکن است بخواهد چند کانال ارتباطی ایجاد کند و به این ترتیب با استفاده از شناسه‌ فرآیند نمی‌توان این کانال‌های ارتباطی را به صورت یکتا مشخص کرد.

دلیل سوم: فرآیندها می‌توانند بر روی پورت‌های شناخته شده گوش دهند ولی این امر زمانی که بخواهیم از شناسه‌های فرآیندها استفاده کنیم غیر ممکن می‌شود.

ب) پاسخ نهایی 1011100010101100

ج) گیرنده برای تشخیص خطا Checksum کل بسته دریافت شده را محاسبه می‌کند. در صورتی که داده بدون خطا باشد تمام بیت‌های نتیجه برابر ۱ شده که این عدد در مکمل ۱ برابر با صفر می‌باشد. در غیر این صورت خطا رخ داده است. استفاده از مکمل ۱ حاصل جمع، محاسبات را ساده‏تر می‏کند. ضمن اینکه مکمل۱، به دلیل سادگی، به مکمل ۲ ترجیح داده می‏شود. در مکمل ۱ عدد صفر به دو صورت 11…11 و 0…0 نمایش داده می‌شود، از این نکته برای این منظور استفاده می‌شود که فرستنده در صورتی که checksum برابر با صفر باشد 11…11 را ارسال کرده و در صورتی که نخواهد checksum را محاسبه کند 0…0 را اراسال می‌کند.

د) این روش تمام خطاهای یک بیتی را تشخیص می‌دهد ولی ممکن است دو بیت خطا وجود داشته باشد که تشخیص داده نشود.

مثال

1001000000101010001010101011110101000100100010000100011111100011

Checksum ای برابر با Checksum اولیه دارد (2 بیت با فاصله 16 بیت را معکوس کنید).