شبكات الحاسوب (1486)

الوحدة الاولى: الاتصالات الرقمية وشبكات الحاسوب والانترنت Digital communication, computer

1-4 تراسل البيانات Data Transmission

1-5 مفهوم الانترنت The Internet

1-6اطراف الشبكة The Network Edge

1-7 اساس (نواة) الشبكة The Network Core

1-8 التأخير والفاقد والانتاجية في شبكات تبديل الحزمة

switched networks delay loss and throughput in packet

1-9 طبقات البروتوكولات ونماذج خدمتها Protocol Layers and their service models

1-10 الاجزاء الختامية

1-2 الاشارات التناظرية والرقمية (Analog and digital signals)

شهد مجال تراسل البيانات في الفترة الأخيرة تطورا هائلا، وازدادت الحاجة الى استخدام التقنيات والأدوات والوسائط المرتبطة بذلك، وتلبية الاحتياج المتزايد فقد ركزت الأبحاث والاستثمارات على تطوير وسائل تراسل البيانات بترقية الشبكات التقليدية السابقة وإدخال أنواع جديدة من الشبكات بتقنيات تتناسب مع السرعة العالية المطلوبة؛ والسعة العالية لأوساط النقل، وضمان خلو هذه الأوساط من معيقات التراسل؛ كالضجيج والتخامد، والتداخل مع الإشارات الأخرى. كما تطورت وسائل التحكم بهذه الوسائط من شبكات ومراكز معالجة

تعريف

يعرف تراسل البيانات بأنه الثقة التي تسمح بتبادل البيانات الرقمية كالصوت، والفيديو، والصور بين نقطتين تحويان تجهيزات مناسبة لاستقبالها ومعالجتها من خلال إشارات حاسوبية وإشارات تحكم وغيرها

1-2-1 أنظمة تراسل البيانات

يتضمن مفهوم نظام تراسل البيانات كل التجهيزات الفيزيائية والإلكترونية و البرمجية التي تقوم بنقل البيانات، وفيما يأتي نوضح أهم المفاهيم المرتبطة بأنظمة تراسل البيانات:

1. اوساط النقل (التجهيزات الفيزيائية)، وهي نوعان سلكية كالكوابل والألياف الضوئية، ولاسلكية عبر الأثير.

2. التجهيزات الإلكترونية: الدارات والعناصر الإلكترونية التي تحول الإشارة من شكل إلى آخر، من مستوى إلى آخر (زيادة الطاقة، التخلص من الإشارات المعيقة، التحويل من مجال ترددي إلى آخر،تغيير شكل الرموز الممثلة، نوع التضمين)

3. البرمجيات : التحكم في تراسل البيانات وتحصينها ضد الأخطاء وتحديد شكل رسائل البيانات وحجمها وأطرها وتزامنها.

4. بنية نظام تراسل البيانات: يتكون نظام تراسل البيانات من مرسل ومستقبل وقناة تراسل البيانات كما

يبين الشكل (1-1). ص٤

5) إجراءات تراسل البيانات: وهي الخطوات التي تمر بها البيانات من لحظة إرسالها إلى لحظة استقبالها، كما يبين الشكل (1-2) ص٤

1-2-2 أنواع الإشارات (Signals)

الإشارة هي الشكل الذي تنتقل بوساطته البيانات ضمن وسط النقل، وهي التعبير المباشر عن المعلومات و البيانات التي يتم توليدها لتستطيع الانتقال عبر وسط النقل بشكل مناسب، فالإشارات الكهربائية

Electrical Signal

تنتقل عبر أوساط النقل المعدنية، والإشارات الضوئية Light Signals تنتقل عبر الألياف الضوئية أو عبر الفراغ، والإشارات الكهرومغناطيسية Electromagnetic signals تنتقل عبر الأثير. ويمكن لأي من هذه الإشارات أن تحمل إشارات أخرى كما هو الحال في التضمين (Modulation)، حيث تحمل الإشارات الجيبية التي تسمى إشارات الحامل Carrier Signals بإشارات البيانات لأسباب تقنية، أو تتحول الإشارات من شكل إلى آخر لتلاءم مع مكونات وأجزاء نظام نقل البيانات، كان يتم تحويل الإشارات الكهربائية إلى إشارات كهرومغناطيسية أو ضوئية وبالعكس، وبشكل عام هناك نوعان أساسيان من الإشارات، الإشارات التناظرية، والإشارات الرقمية، كما سنوضح فيما يلي

1. الإشارات التناظرية (Analog Signals): إشارات طبيعية كالصوت والفيديو، وقد تكون الإشارة التناظرية مستمرة مثلt) s1)، أو منقطعة في القيمة (المطال Amplitude) مستمرة في الزمن مثل s2(t) او منقطعة في الزمن مستمرة في القيمة مثل (S3 (t ، كما هو موضح في الشكل (1-3)، وهناك ثلاث قيم أساسية تحدد شكل الإشارة

أ. المطال Amplitude: ويعبر عنه كقيم لحظية أو كقيم تابعة للزمن.

ب. الطيف الترددي Frequency spectrum ويعطي فكرة واضحة عن المركبات الترددية للإشارة، ويؤدي دورا أساسيا في تحديد إمكانية إرسال الإشارة ضمن قناة تراسل معينة

ج. خصائص الطور Phase Characteristics وتحديد الطور الابتدائي المركبات الترددية

2 الإشارات الرقمية (Digital Signals): مع تطور أنظمة الاتصالات، وزيادة متطلبات استخدامها، تم الاستعاضة عن الإشارات التناظرية بالإشارات المتقطعة في بعض التطبيقات، وهي إشارات منقطعة في الزمن مستمرة في القيمة، كما يبين الشكل (1-4).ص6

اهم انواع الاشارات المتقاطعة هو الإشارات الرقمية، وتنتج عند الاستعاضة عن قيم عينات الإشارة بسلسلة من الرموز تشكل شيفرة معينة، تعبر بشكل او بأخر عن القيم اللحظية للعينات، ويتم التعبير عن هذه السلسلة عادة بأرقام ثنائية ، كما بين الشكل (1-5) ص٦

زاد استخدام الإشارات الرقمية في الآونة الأخيرة بشكل مضطرد، نتيجة للتطور الهائل في أساليب المعالجة الرقمية وأجهزتها، وتتحول الإشارة المتقطعة إلى رقمية بتكميم قيم العينات (Quantization)، أي بحصر قيم العينات بعدد محدود من القيم، فتنتج اشارة منقطعة في الزمن منقطعة في القيمة " ولمعالجة الإشارات التناظرية الناتجة عن

حوادث فيزيائية طبيعية باستخدام الحاسوب، فلا بد من تحويلها إلى رقمية ليتمكن من معالجتها وتخزينها ونقلها باستخدام الشبكات على شكل ملفات ورسائل رقمية

مفاهيم مهمه

1. تقاس جودة الإشارة بنسبة الإشارة إلى الضجيج بالديسي بل Signal-to-Noise Ratio SNR=S/N dB

2. معدل الخطأ(Error Rate):هو تأثير الإشارات المعيقة ويساوي متوسط الأخطاء في وحدة الزمن

3. نسبة احتمال الخطأ (Probability of Error):هي عدد البنات الخاطئة المستقبلية في كل عدد من البنات

4. يمكنك إنقاص الأخطاء باختيار نوع مناسب من تعديل الإشارة (Modulation)والتركيز وكشف الأخطاء

(Digital Signal Encoding) ترميز الإشارات الرقمية | 3-1|

ترسل البيانات الرقمية الثنائية عن طريق ترميز كل بت بيانات إلى عناصر الإشارة، وفي ابسط الحالات، هناك علاقة واحد لواحد بين البتات وعناصر الإشارة، فمثلا كما يبين الشكل (6-1)، يتم تمثيل الثنائي 1 بمستوى جهد منخفض، والثنائي 0 بمستوى جهد عال، وسنعرض في هذا القسم مجموعة متنوعة من أنظمة الترميز

شكل 1-6: تطابق واحد لواحد بين البتات وعناصر الإشارة ص7

في البداية، دعنا، عزيزي الطالب، نعرف بعض المصطلحات

1. الإشارة احادية القطبية (Unipolar): إذا كانت جميع عناصر الإشارة لها نفس العلامة الجبرية، أي أنها موجبة أو سالبة

2. الإشارة المستقطبة (Polar) يتم تمثيل حالة واحد المنطقي بمستوى جهد موجب، والأخرى بمستوى جهد سالب

3. معدل بيانات الإشارة(Data signaling rate) هو المعدل الذي ترسل به البيانات مقدرا بالبت في الثانية

4. مدة أو طول البت (Bit duration or length) هي مقدار الوقت الذي يستغرقه المرسل في ارسال البت، فاذا كان معدل إرسال البيانات R، فإن مدة البت في 1 / R

5. معدل التضمين أو التباين (Modulation rate): هو المعدل الذي يتغير عند مستوى الإشارة، ويعتمد

ذلك على طبيعة الترميز، ويعبر عنه بالبود (Baud)، أي عنصر إشارة في الثانية

(Signal elements per second)

6. العلامة والفراغ (Mark and Space):يمثلان الارقام الثنائية 0،1 على التوالي

يعتبر نظام ترميز الإشارات الرقمية احد العوامل التي تؤثر في أداء مستقبل الإشارة دون حدوث أخطاء بالإضافة إلى نسبة الإشارة إلى الضجيج، ومعدى الإرسال، وعرض الحزمة، ونظام الترميز هو ببساطة طريقة

تحويل بتات البيانات إلى عناصر الإشارة، وهناك العديد من طرق الترميز، وسنصف في هذا القسم أكثرها شيوعا، كما هو مبين في الجدول (1-1) وفي الشكل (7-1).

جدول 1-1 : تعريف اشكال ترميز الإشارات الرقمية

شكل ترميز الإشارة الرقمية	التعريف

اللاعودة إلى مستوى الصفر Nonreturn to Zero-Level (NRZ-L)	0 = مستوى جيد عالي 1= مستوى جهد منخفض
اللاعودة إلى الصفر معكوس Nonreturn to Zero Inverted (NRZI)	0 = لا يرد القتال في بداية الفترة (في زمن البيت) 1 = الانتقال في بداية الفترة
ثنائي القطبية مع قلب العلامة بالتناوب Bipolar-AMI	0 = لا توجد إشارة خط 1 = مستوي جهد موجب او سالب ، بالتناوب للوحدات المتتالية
الترميز شبة الثلاثي Pseudoternary	0 = مستوي جهد موجب او سالب، بالتناوب للأصفار المتتالية 1 = لا توجد إشارة خط
ترميز مانشستر Manchester	0 = الانتقال من مستوى جهد عالي إلى منخفض في منتصف الفترة | = الانتقال من مستوى جيد منخفض الى عالي في منتصف الفترة
ترميز مانشستر التفاضلي Differential Manchester	الانتقال دائما في منتصف الفترة 0 = الانتقال في بداية الفترة = لا يوجد انتقال في بداية الفترة
B8ZS	كما هو الحال في ثنائي القطبية AMI باستثناء استبدال أي سلسلة من 8 أصفار بسلسلة فيها رمزين مخالفين
HDB3	كما هو الحال في ثنائي القطبية AMI ، باستثناء استبدال اي سلسلة من 8 أصفار سلسلة فيها رمز مخالف

|4-1| تراسل البيانات (Data Transmission)

عزيزي الطالب، دعنا نبدأ هذا القسم بتعريف بعض المصطلحات المتعلقة بتراسل البيانات. كما تعلم، فإن البيانات تنتقل عبر أوساط النقل بين المرسل والمستقبل على شكل موجات كهرومغناطيسية، وفيما يأتي تعريف أهم المصطلحات التي ستحتاج إليها

1. الأوساط الموجهة (Guided media) تنقل الإشارات عبر ممرات محددة كالكوابل المجدولة (Twisted pair)والمحورية والألياف الضوئية (سلكية Wired)

2. الأوساط غير الموجهة(Unguided media) تنقل الإشارات عبر الأثير أو الفضاء (لاسلكية Wireless)

3 .الخط المباشر (Direct link): مسار التراسل بين جهازين تنتقل فيه الإشارة من المرسل إلى المستقبل مباشرة دون أجهزة وسيطة باستثناء المضخمات أو المكررات اللازمة لتقوية الإشارة

4. تراسل نقطة إلى نقطة (point-to-point): التراسل عبر خط مباشر بين جهازين هما الوحيدين اللذين يتشاركان الوسيط

5. التراسل البسيط (Simplex): ترسل الإشارات بين محطتين في اتجاه واحد فقط، إحداهما المرسل والأخرى في المستقبل

6.التراسل أحادي الاتجاه (Half-duplex): قد ترسل كلتا المحطتين بالاتجاهين، ولكن ترسل واحدة منهما فقط في كل مرة

7. التراسل باتجاهين (Full duplex): قد ترسل كلنا المحطتين إشاراتهما معا عبر الوسيط في كلا الاتجاهين

1-4-1 التردد والكيف وعرض الحزمة (Frequency, Spectrum and Bandwidth)

سنهتم في بالإشارات الكهرومغناطيسية المستخدمة كوسيلة النقل البيانات، ويتم توليد الإشارة في المرسل وتنتقل عبر الوسيط على شكل دالة بالنسبة للزمن، ولكن يمكن التعبير عنها أيضا كدالة بالنسبة للتردد frequency ويعرف التردد (f) بانه معدل تكرار الإشارة مقدرا بالهيرتز(دورة في الثانية)، أي أن الدورة (T=1/f period) كما هو مبين في الشكل (8-1).

أما طيف الإشارة (Spectrum) فهو مجموعة الترددات التي تحتويها، وعرض النطاق الترددي المطلق (Absolute Bandwidth)يعبر عن عرض هذا الطيف، والعديد من الإشارات لها عرض نطاق ترددي لانهائي، ولكن طاقة الإشارة يتضمنها نطاق ضيق من الترددات يطلق عليه عرض النطاق الترددي (Bandwidth

عزيزي الطالب، في الإشارة الرقمية يستعاض عن مفهوم عرض النطاق الترددي (Bandwidth) بمفهوم معدل البيانات (Date Rate)أو معدل الارسال (Transmission Rate) ويرمز له بالرمز R ويقاس بالبت في الثانية bits per second (bps)

ولكنهما مفهومان مرتبطان بشكل وثيق، حيث تحتاج الإشارة الرقمية إلى عرض حرمة يتناسب مع معدل ارسالها من أجل إرسالها في قناة تناظرية، كما تحتاج الإشارة التناظرية إلى معدل ارسال ادنى (بعد رقمنتها) ليتم ارسالها بشكل صحيح في قناة رقمية لندرس الإشارة المبينة في الشكل (5-1)، ونلاحظ تكرار رموزها (0, 1) في فترات زمنية متساوية كل منها T=1/R ثانية أو بمعدلR/Sec، ويسمى الزمن T الفاصل الثنائي (Binary Interval) ، ويعطي منبع المعلومات R بت (bit) في الثانية

1-5| مفهوم الانترنت (The Internet)

عزيزي الطالب سنناقش في هذا القسم مفهوم الإنترنت من ناحيتين، المكونات المادية والبرمجية، والخدمات التي تقدمها

المكونات المادية والبرمجية (Hardware and Software Components)

الإنترنت هي شبكة حواسيب تربط مئات الملايين من الأجهزة المحوسبة حول العالم، وكانت هذه الأجهزة عبارة عن أجهزة حاسوب مكتبية تقليدية، ومحطات عمل لينكس تدعي الخوادم لتخزين المعلومات ونقلها مثل صفحات الويب ورسائل البريد الإلكتروني. وتضاعفت هذه الأجهزة تدريجيا لتشمل أجهزة غير تقليدية مثل أجهزة الحاسوب المحمول، والهواتف الذكية والأجهزة اللوحية، وأجهزة التلفاز، ووحدات تحكم الألعاب، وكاميرات الويب، والسيارات، وأجهزة استشعار البيئة، والأنظمة الكهربائية وأنظمة الحماية المنزلية. وفي لغة الإنترنت، تسمى جميع هذه الأجهزة المضيفين أو الأنظمة الطرفية. ويبين الشكل (1-9) عينة من الأجهزة المكونة لشبكة الإنترنت

ترتبط الأنظمة النهائية ببعضها عن طريق شبكة من خطوط الاتصال (Communication Links) ومقسمات الحزم (Packet Switches)وتستخدم خطوط الاتصال أنواعا مختلفة من الأوساط المادية، كالكوابل المحورية، والأسلاك النحاسية، والألياف الضوئية، وطيف الراديو. وتنتقل البيانات فيها بمعدلات إرسال مختلفة

تقاس بالبت/ثانية، ولإرسال البيانات يقوم المرسل بتقسيمها إلى شرائح واضافة بايت لمقدمة كل شريحة لإنتاج الحزم (Packets)وإرسالها عبر الشبكة إلى المستقبل، وهناك يتم إعادة تجميعها لإنتاج البيانات الأصلية يأخذ مقسم الحزم الحزمة الواردة عبر أحد خطوط الاتصال ويعيد توجيهها عبر أحد خطوط الاتصال الصادرة وتأتي مقسمات الحزم في أشكال عدة، ولكن أبرزها في هذه الأيام الموجهات (Routers) و مقسمات مطبقة الارتباط (Link-Layer Switch) وكلاهما يعمل على توجيه الحزم نحو وجهتها النهائية عادة تستخدم الارتباط القمات في شبكات النفاذ، بينما تستخدم الموجهات في الشبكة الأساسية

تعريف

المسار(Rout or Path)

تسلسل خطوط الاتصال والمقسمات التي تعبرها الحزمة عبر الشبكة من نظام الإرسال النهائي إلى نظام الاستقبال النهائي

تتيح الأنظمة النهائية النفاذ إلى الإنترنت من خلال مزودي خدمة الإنترنت (Internet Service Providers ISPs) ، مثل مزودي خدمات الإنترنت المنزلي من خلال الكوابل المحلية أو الهاتف ومزودي خدمات الإنترنت للشركات ومزودي خدمات الإنترنت اللاسلكي (Wi-Fi) في المطارات والفنادق والمقاهي والمتاجر والأماكن العامة كل مزود خدمة إنترنت هو بحد ذاته شبكة من المقسمات وخطوط الاتصال، ويوفر عدة أنواع من النفاذ عبر الشبكة إلى الأنظمة النهائية، بما في ذلك النفاذ إلى النطاق العريض المنزلي مثل المودم السلكي أو DSL ، والنفاذ إلى الشبكة المحلية عالية السرعة، والنفاذ اللاسلكي، ومودم الطلب الهاتفي بسرعة 56 كيلوبت في الثانية كما يوفر مزودو خدمات الإنترنت النفاذ مباشرة إلى الإنترنت لمزودي المحتوى ومواقع الويب، ويرتبط مزودو خدمات الإنترنت الذين يوفرون النفاذ إلى الأنظمة النهائية معا على المستويين الوطني والدولي وتدار كل شبكة من مزودي خدمات الإنترنت بشكل مستقل، وتشغل بروتوكول الإنترنت IP ، وتتبع معاهدات تسمية وعنونة محددة. تعمل الأنظمة النهائية والمقسمات وغيرها من أجزاء الإنترنت من خلال تشغيل بروتوكولات تتحكم بإرسال المعلومات واستقبالها عبر الإنترنت، وأهمها بروتوكول التحكم بالنقل (TCP) و بروتوكول الإنترنت (IP) الذي يحدد تنسيق الحزم التي يتم إرسالها واستقبالها بين الموجهات والأنظمة النهائية، وتعرف برتوكولات الإنترنت الرئيسة مجتمعة باسم)TCP/IP(، ونظرا لأهميتها فإنها تتبع معايير مدة تحدد وظيفة كل بروتوكول وآلية عمله، حتى يتمكن المطورون من إنشاء أنظمة ومنتجات تتفق في آلية عملها

شكل 1-9: عيلة من المكونات المادية لشبكة الإنترنت

1-5-2خدمات الانترنت (Internet Services)

عزيزي الطالب، يمكننا وصف الإنترنت من زاوية مختلفة تماما على أنها بنية تحتية توفر خدمات للتطبيقات مثل البريد الإلكتروني، وتصفح الويب، والشبكات الاجتماعية، والرسائل الفورية، وتدفق الفيديو، والألعاب الموزعة، ومشاركة الملفات من نظير إلى نظير (P2P)، والتلفاز عبر الإنترنت، والنفاذ عن بعد، وغيرها الكثير. وعلى الرغم من أن المقسمات تعمل على تسهيل تبادل البيانات بين الأنظمة النهائية، إلا أنها لا تهتم بالتطبيق الذي يصدر البيانات. ولنفترض أن لديك فكرة مثيرة لتطبيق إنترنت تود تطويرها إلى منتج فعلى ستحتاج إلى كتابة برامج، ولنقل باستخدام لغة Java أو C أو Python ، تعمل على أنظمة نهائية مختلفة تتطلب تبادل البيانات فيما بينها، وهذا يقودنا إلى وصف آخر للإنترنت كمنصة ل لتطبيقات، أي كيف لبرنامج يعمل على أحد الأنظمة أن يوعز للإنترنت بنقل البيانات إلى برنامج آخر يعمل على نظام آخر؟ توفر الأنظمة المتصلة بالإنترنت واجهة برمجة تطبيقات (API) تؤدي هذه المهمة عبر البنية التحتية ل لإنترنت، وهي مجموعة من القواعد التي يجب أن يتبعها البرنامج المرسل حتى تتمكن الإنترنت من توصيل بياناته إلى البرنامج الوجهة. عزيز الطالب، لقد قدمنا للتو وصفا للإنترنت من حيث المكونات المادية والبرمجية، والبنية التحتية لتقديم الخدمات للتطبيقات الموزعة، وسنناقش واجهة برمجة تطبيقات الإنترنت وخدمات الإنترنت بالتفصيل في الوحدة الثانية

مفهوم البروتوكول (What Is a Protocol?)

يتولى بروتوكول الشبكة إدارة تبادل الرسائل واتخاذ الإجراءات اللازمة بين المكونات المادية أو البرمجية المتصلة بالشبكة (مثل الحاسوب أو الهاتف الذكي أو الجهاز اللوحي أو الموجه أو أي جهاز(، فجميع أنشطة الإنترنت التي تتضمن اثنين أو أكثر من الكيانات المتصلة تحكمها برتوكولات محددة. مثلا، تتحكم البروتوكولات بتدفق البيانات بين بطاقتي الشبكة في جهازي حاسوب متصلين؛ كما تتحكم بروتوكولات التحكم بالازدحام بمعدل نقل الحزم بين المرسل والمستقبل؛ وتحدد البروتوكولات في الموجهات مسار الحزمة من

يحدد البروتوكول شكل الرسائل المتبادلة وترتيبها بين كيانين متصلين أو أكثر، كما يحدد الإجراءات المتخذة بشان إرسال و/أو استلام رسالة معينة

المصدر إلى الوجهة، وتعمل البروتوكولات في كل جزء من أجزاء الإنترنت.

وكمثال مألوف على بروتوكول الشبكة، دعنا، عزيزي الطالب، ننظر فيما يحدث عند تقديم طلب إلى خادم الويب، أي عند كتابة عنوان صفحة ويب (URL) في مستعرض الويب. يوضح الشكل (10-1) السيناريو كما يأتي

أ. يرسل حاسوبك رسالة طلب اتصال إلى خادم الويب وينتظر الرد.

ب. عندما يتلقى الخادم رسالة الطلب يرد برسالة موافقة

ج. يطلب حاسوبك مستند الويب بإرسال اسم صفحة الويب التي ترغب بها من خادم الويب في رسالة GET.

د. يقوم خادم الويب بإرجاع صفحة الويب (الملف) إلى حاسوبك

أطراف الشبكة (The Network Edge)

عزيزي الطالب، سنتعمق الآن في مكونات شبكة الحاسوب (وخاصة الإنترنت)، ولنبدأ في هذا القسم بأطراف الشبكة ونظر في المكونات المعروفة أكثر من غيرها، أي الحواسيب والهواتف الذكية والأجهزة التي نستخدمها يوميا، وسننتقل في القسم التالي إلى قلب الشبكة، ونناقش دور المقسمات والموجهات في شبكات الحاسوب. تسمى الأجهزة المتصلة بالإنترنت أنظمة نهائية لأنها تقع على طرف شبكة الإنترنت كالحواسيب والخادمات والأجهزة المتنقلة، كما يطلق عليها مضيفات لأنها تستضيف وتشغل البرامج التطبيقية كمستعرض الإنترنت، وخادم الويب أو البريد الإلكتروني، وتقسم إلى عملاء (Clients) كالحواسيب المكتبية والمتنقلة والهواتف الذكية، وخادمات (Servers) بمواصفات أعلى لأداء وظائف محددة، مثل تخزين صفحات الإنترنت والفيديو والبريد الإلكتروني ونشرها وتوزيعها. يبن الشكل (1-11) تفاعل الأنظمة النهائية المتصلة بالإنترنت.

شبكات النفاذ (Access Networks)

شبكة النفاذ هي الشبكة التي تربط نظاما نهائيا بأول موجه (الموجه الطرفي) على المسار بين نظام نهائي وآخر بعيد. يبين الشكل (1-12) أنواع مختلفة من شبكات النفاذ ممثلة بالخطوط العريضة المظللة، والأوضاع التي تستخدم فيها (المنزلية والشركات واللاسلكية الواسعة المتنقلة)

النفاذ المنزلي (Home Access)

توضح في هذا القسم طرق النفاذ المنزلي للإنترنت، وهي كثيرة سنذكرها، ونركز في الشرح على أشهر طريقتين هما، خط المشترك الرقمي DSL: Digital Subscriber Line والكوابل (Cable)

خط المشترك الرقمي(DSL: Digital Subscriber Line)

عادة يكون النفاذ المنزلي إلى الإنترنت سلكيا عبر مودم DSL من شركة الهاتف المحلية، وفي هذه الحالة، فإن شركة الاتصالات هي أيضا مزود خدمة الإنترنت. يبين الشكل (1-13) أن المودم الخاص بكل عميل يستخدم خط الهاتف التقليدي (سلك نحاسي مزدوج مجدول(لتبادل البيانات مع مجمع نفاذ خط المشترك الرقمي (DSLAM) الموجود في المكتب المركزي المحلي لشركة الاتصالات، يستقبل المودم البيانات الرقمية ويترجمها إلى درجات عالية التردد لنقلها عبر اسلاك الهاتف إلى شركة الاتصالات، ثم تعاد ترجمة الإشارات

التناظرية من هذه المنازل إلى تنسيق رقمي في DSLAM. يحمل خط الهاتف المنزلي البيانات وإشارات الهاتف معا، ويتم ترميزها على ترددات مختلفة

قناة تنزيل عالية السرعة، ويتراوح نطاقها من 50 kHz إلى 1 MHz .

قناة تحميل متوسطة السرعة، ويتراوح نطاقها من 4 إلى 50 kHz .

قناة هاتف عادية باتجاهين، يتراوح نطاقها من 0 إلى 4kHz

بهذه الطريقة، يظهر خط DSL الواحد كثلاثة خطوط منفصلة، حيث يمكن إجراء مكالمة هاتفية واتصال بالإنترنت مشتركة معا. على جانب العميل، يقوم جهاز الفصل (Splitter بفصل البيانات عن إشارات الهاتف التي تصل إلي المنزل ويوجه إشارة البيانات إلى مودم DSL، وعلى جانب شركة الاتصالات، يفصل DSLAM البيانات عن إشارات الهاتف ويرسل البيانات عبر الإنترنت، وقد تتصل مئات أو آلاف المنازل بخط DSLAM واحد [Dischinger 2007].

كابل النفاذ إلى الإنترنت (Cable Internet Access)

يستخدم كابل النفاذ البنية التحتية لكوابل شركات التلفاز، فيحصل المنزل على الإنترنت عبر الكابل من الشركة نفسها التي توفر خدمة التلفاز. كما هو موضح في الشكل (1-14)، تقوم الألياف الضوئية بتوصيل طرف رأس الكابل بالتقاطعات على مستوى الحي، والتي تستخدم الكابل المحوري التقليدي للنفاذ إلى المنازل، وكل تقاطع يدعم عادة 500 إلى 5000 منزل. ونظرا لاستخدام كل من الألياف الضوئية والكوابل المحورية في هذا النظام، المحورية الهجينة (HFC). يستخدم مودم كابل خارجي خاص يتصل بالحاسوب المنزلي عبر منفذ إيثرنت.

وهناك طرق أخرى ل للوصول إلى الإنترنت المنزلي نذكر منها الألياف الضوئية ل لمنازل

FTTH: Fiber To The Home والطلب الهاتفي)Dialup(، والأقمار الصناعية)Satellite(، والإيثرنت (Ethernet)، والاتصال ال لاسلكي)Wi-Fi(، وكذلك الجيل الثالث عبر الأجهزة الخلوية) 4G and 5G,G3 ,

التناظرية من هذه المنازل إلى تنسيق رقمي في DSLAM. يحمل خط الهاتف المنزلي البيانات وإشارات الهاتف معا، ويتم ترميزها على ترددات مختلفة

قناة تنزيل عالية السرعة، ويتراوح نطاقها من 50 kHz إلى 1 MHz .

. قناة تحميل متوسطة السرعة، ويتراوح نطاقها من 4 إلى 50 kHz .

قناة هاتف عادية باتجاهين، يتراوح نطاقها من 0 إلى 4kHz

بهذه الطريقة، يظهر خط DSL الواحد كثلاثة خطوط منفصلة، حيث يمكن إجراء مكالمة هاتفية واتصال بالإنترنت مشتركة معا. على جانب العميل، يقوم جهاز الفصل (Splitter بفصل البيانات عن إشارات الهاتف التي تصل إلي المنزل ويوجه إشارة البيانات إلى مودم DSL، وعلى جانب شركة الاتصالات، يفصل DSLAM البيانات عن إشارات الهاتف ويرسل البيانات عبر الإنترنت، وقد تتصل مئات أو آلاف المنازل

بخط DSLAM واحد [Dischinger 2007].

(Cable Internet Access) كابل النفاذ إلى الإنترنت

يستخدم كابل النفاذ البنية التحتية لكوابل شركات التلفاز، فيحصل المنزل على الإنترنت عبر الكابل من الشركة نفسها التي توفر خدمة التلفاز. كما هو موضح في الشكل (1-14)، تقوم الألياف الضوئية بتوصيل طرف رأس الكابل بالتقاطعات على مستوى الحي، والتي تستخدم الكابل المحوري التقليدي للنفاذ إلى المنازل، وكل تقاطع يدعم عادة 500 إلى 5000 منزل. ونظرا لاستخدام كل من الألياف الضوئية والكوابل المحورية في هذا النظام، المحورية الهجينة (HFC). يستخدم مودم كابل خارجي خاص يتصل بالحاسوب المنزلي عبر منفذ إيثرنت.

وهناك طرق أخرى ل للوصول إلى الإنترنت المنزلي نذكر منها الألياف الضوئية ل لمنازل):Fiber To The Home، FT TH والطلب الهاتفي)Dialup(، والأقمار الصناعية)Sat tellite(، والإيثرنت (Ethernet(، والاتصال ال لاسلكي)-Wi Fi(، وكذلك الجيل الثالث عبر الأجهزة الخلوية (3G5) G, 4G and

شكل 141 : شبكة النفاذ عبر الألياف المحورية الهجينة

(Physical Media) الأوساط المادية

عزيزي الطالب، في القسم الفرعي السابق، فتحنا نظرة عامة على بعض أهم تقنيات النفاذ إلى شبكة الإنترنت كما وصفنا هذه التقنيات و اشرنا إلى الأوساط المادية المستخدمة فعلى سبيل المثال، ذكرنا أن HFC تستخدم مزيجا من الألياف الضوئية والكوابل المحورية، وأن DSL و الإيثرنت تستخدم الأسلاك النحاسية، وأن شبكات مول عبر الهاتف النقال تستخدم طيف الراديو في هذا القسم الفرعي، تقدم نظرة عامة موجزة عن أوساط الأرسال و الأوساط الأخرى الأكثر شيوعا في الإنترنت

لتحديد مفهوم الوسط المادي، دعنا نفترض انتقال بت من نظام نهائي إلى اخر من خلال سلسلة من خطوط الاتصال والموجهات. يقوم نظام المصدر بنقل الجزء الأول، وبعد ذلك بفترة وجيزة، يستقبل الموجه الأول في سلسلة المواجهات هذا البت ثم يقوم بنقله، بعد ذلك بوقت قصير يتلقاه الموجه الثاني، وهكذا. فعندما ينتقل البت من المصدر إلى الهدف، يمر عبر سلسلة ازواج من اجهزة الإرسال والاستقبال، وفي كل زوج، يتم إرسال البت بواسطة موجات كهرومغناطيسية منتشرة أو نبضات ضوئية عبر وسط مادي قد يتخذ العديد من الأشكال والنماذج، وليس بالضرورة أن يكون من نفس النوع بين كل زوج من أجهزة الإرسال والاستقبال على طول المسار، ومن الأمثلة على الأوساط المادية ازواج الأسلاك النحاسية المجدولة Twisted-Pairs، أو الكوابل المحورية Coaxial Cables أو الألياف الضوئية، أو طيف الراديو الأرضي أو الفضائي. وتقع الأوساط bEI b1 (guided media) (Unguided media) الموجهة، تنتقل الموجات عبر وسط صلب، كالالياف الضوئية أو الأسلاك النحاسية المجدولة، أو الكوابل المحورية، أما في الأوساط غير الموجهة، فتنتشر الموجات في الغلاف الجوي وفي الفضاء الخارجي، كما هو الحال في الشبكات المحلية اللاسلكية أو القنوات الفضائية الرقمية

غالبا ما تكون الكلفة الفعلية للشركات التي تستخدم الأوساط المادية (كالأسلاك النحاسية والألياف الضوئية قليلة تنسية مقارنة بتكلفة الشبكات الأخرى، وتكون كلفة العمالة المرتبطة بتركيبها أعلى بكثير من كلفة المادة

الأزواج المجدولة (Twisted-Pair)

تعتبر الأزواج المجدولة من أوساط النقل الموجهة الأكثر شيوعا واستخداما، وخاصة في شبكات الهاتف، إذ يستخدم أكثر من 99 ٪ من التوصيلات السلكية بين سماعة الهاتف والهاتف المحلي السلك النحاسي مجدول، كما يبين الشكل)1-15-a(يتكون الزوج المجدول من سلكين نحاسيين معزولين سمك كل منها حوالي 1 مم، مرتبة بنمط حلزوني منتظم، وذلك للحد من التداخل الكهربائي، وعادة، يتم تجميع عدد من الأزواج معا في غلاف واق، ويشكل زوج الأسلاك خط اتصال واحد. تستخدم شبكات الحاسوب الداخلية المحلية)مثل الإنترنت المنزلي(الزوج المجدول غير المحمي)Unshielded، UTP Pair)Twisted: يتراوح معدل البيانات في الشبكات المحلية التي تستخدم الزوج المجدول من 10 ميغابت في الثانية إلى 10 جيجابت في الثانية لمسافة تصل إلى 100 متر بين المرسل والمستقبل.

الكوابل المحورية(Coaxial Cable)

تتكون الكوابل المحورية من موصلين نحاسيين بشكل متحد المركز لا متواز، كما يبين الشكل (1-15 - b)، بهذه البنية بالإضافة إلى العزل والتغليف تحقق هذه الكوابل معدلات نقل بيانات عالية، وهي شائعة جدا في أنظمة التلفاز. وكما أشرنا سابقا، دمج نظام تلفاز الكابل بأجهزة المودم لتوفير الإنترنت المنزلي بمعدل نقل يصل إلى عشرات الميغابايت في الثانية. يرحل المرسل الإشارة الرقمية إلى نطاق ترددي معين، وترسل الإشارة التناظرية الناتجة من المرسل إلى مستقبل واحد أو أكثر. وقد تستخدم الكوابل المحورية كوسط موجه مشترك، أي يمكن توصيل عدد من الأنظمة النهائية بالكابل مباشرة، بحيث يستقبل كل نظام نهائي ما يتم إرساله من الأنظمة النهائية الأخرى

الألياف الضوئية (Fiber Optics)

الألياف الضوئية هي عبارة عن وسط رفيع ومرن ينقل البيانات على شكل نبضات ضوئية كما يبين الشكل (1 15-0)، كل بت تمثله نبضة، وتدعم الألياف الضوئية معدلات نقل هائلة تصل إلى عشرات أو حتى مئات الجيغابت في الثانية فهي محصنة من التداخل الكهرومغناطيسي، وتوهين الإشارة فيها منخفض يصل إلى 100 كيلو متر ، ويصعب جدا التنصت عليها هذه الخصائص جعلت من الألياف الضوئية أوساط النقل الموجهة المفضلة للمسافات البعيدة، وخاصة للخطوط الخارجية لذا، شاع استخدامها في العمود الفقري للإنترنت. مع ذلك، فإن الكلفة العالية لأجهزتها، كالمرسل والمستقبل والمقسمات، أعاقت انتشارها للمسافات القصيرة في الشبكات المحلية أو المنزلية. تتراوح السرعة القياسية لخطوط الناقل الضوئي (Optical Carrier :OC)

من 51.8 Mbps إلى 39.8 Gbps ، ويشار إلى هذه المواصفات بالرمز OC-n ، حيث تساوي سرعة الخط 51.8. nx Mbps

قنوات الراديو الأرضية والفضائية (Terrestrial and Satellite Radio Channels)

تحمل قنوات الراديو الإشارات في الطيف الكهرومغناطيسي، وتعتبر وسطا جاذبا لأنها لا تحتاج إلى تركيب أسلاك، ويمكنها اختراق العوائق كالجدران، وتوفر خطوط اتصال لمستخدمي الهاتف المحمول، وتستطيع نقل الإشارة لمسافات بعيدة. وتعتمد خصائص قناة الراديو بشكل كبير على بيئة الانتشار والمسافة التي ينبغي أن

تحمل إليها الإشارة، فقد تتسبب بيئة الانتشار بما يأتي

1. فقدان المسار (Path loss) وتضاؤل الطيف (Shadow fading)مما يقلل من قوة الإشارة عندما تنتقل مسافات طويلة، وكذلك حول أو عبر الأجسام المعيقة

، بسبب انعكاس الإشارة عن الأجسام المسببة للتداخل Multipath fading 2. التضاؤل متعدد المسار (

3. التداخل (Interference)، الذي ينتج عن الإشارات الكهرومغناطيسية، والإشارات المرسلة الأخرى

يمكن تصنيف قنوات الراديو الأرضية بشكل عام إلى ثلاث مجموعات

1. قنوات قصيرة تعمل لمسافة قصيرة جدا من متر إلى مترين، وتستخدم في الأجهزة الشخصية مثل سماعات الأذن، ولوحات المفاتيح، والأجهزة الطبية اللاسلكية .

2.قنوات لاسلكية محلية: تعمل على المستوى المحلي من عشرة إلى بضع مئات من الأمتار، وتستخدم في تقنيات الشبكات المحلية اللاسلكية.

3. قنوات لاسلكية واسعة النطاق: تعمل لمسافات طويلة تمتد إلى عشرات الكيلومترات، وتستخدم في تقنيات النفاذ الخلوي

في قنات الراديو الفضائية، يربط القمر الصناعي بين اثنتين أو أكثر من المحطات الأرضية، أي أجهزة الإرسال والاستقبال الأرضية بالموجات الميكروية. ويستقبل الاتصالات على نطاق ترددي واحد، ويعيد توليد الإشارة باستخدام مكرر، ثم ينقل الإشارة على تردد آخر. ويستخدم نوعان من الأقمار الصناعية في الاتصالات: أقمار مستقرة بالنسبة إلى الأرض(Geostationary satellites) وأقمار تدور حول الأرض في مدارات منخفضة

(Low-Earth Orbiting: LEO satellites)

تبقى الأقمار الصناعية المستقرة بشكل دائم فوق نفس النقطة على الأرض، ويتحقق ذلك بوضع القمر في مداره على ارتفاع 36,000 Km، وهي مسافة طويلة عند الاتصال عبر القمر الصناعي بين محطتين أرضيتين، فينتج عنها تأخير كبير للإشارة يبلغ 280 MS. وغالبا، تستخدم الاتصالات الفضائية، التي قد تصل سرعتها الشكل (16-1) أنواع الاتصال عبر الأقمار الصناعية مئات الميجابت في الثانية، في المناطق التي لا تصلها خدمة DSL أو النفاذ إلى الإنترنت عبر الكوابل، ويبين

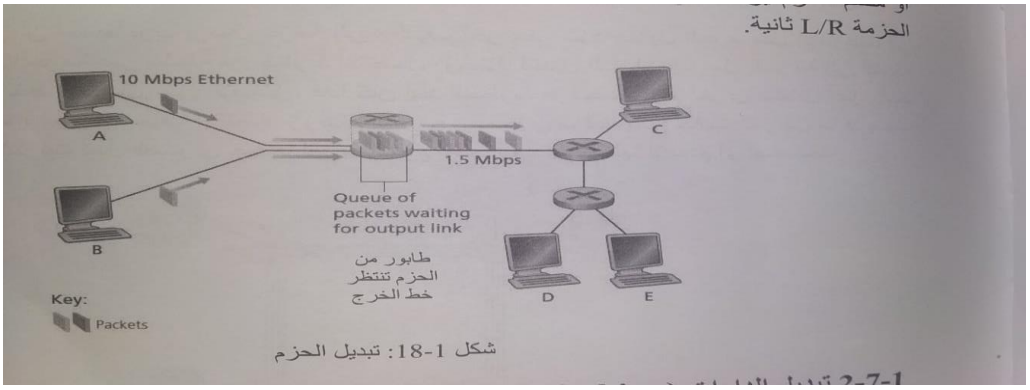
أما الأقمار الصناعية المتحركة (LEO)، فيتم وضعها أقرب بكثير، وتدور حول الأرض كالقمر، ويمكنها أن تتواصل مع بعضها البعض، ومع المحطات الأرضية. ولتوفير تغطية مستمرة لمنطقة معينة، يجب وضع العديد من الأقمار الصناعية في المدار، ويتم حاليا تطوير العديد من هذه الأنظمة، وقد يستخدم هذا النوع من الأقمار الصناعية في النفاذ إلى الإنترنت مستقبلا

1-7 أساس (نواة) الشبكة (The Network Core) |

عزيزي الطالب، بعد دراسة أطراف شبكة الإنترنت في القسم السابق، دعنا ننتقل الآن نتعمق أكثر في نواة الشبكة(Network Core أي شبكة تبديل الحزم والخطوط التي تربط بين أنظمة الإنترنت النهائية. يميز الشكل (17-1) نواة الشبكة بخطوط سميكة مظلة.

تبديل الحزم (Packet Switching)

في تطبيقات الشبكة، تتبادل الأنظمة النهائية الرسائل فيما بينها، وقد تحتوي الرسائل على ما يريده مصمم التطبيق، فقد تؤدي وظيفة تحكم، وقد تحتوي على بيانات كرسالة بريد إلكتروني أو صورة بتنسيق JPEG أو ملف صوت MP3 . ولإرسال رسالة يقوم المصدر بتجزئتها إلى أجزاء أصغر من البيانات تسمى الحزم (Packets) كما يبين الشكل (18-1). ويبين المصدر والوجهة، تنتقل كل حزمة عبر خطوط الاتصال ومقسمات الحزم (مثل الموجهات والمقسمات(. ترسل الحزم عبر كل خط اتصال بمعدل الإرسال الكامل. فإذا كان المصدر أو مقسم الحزم يرسل حزمة طولها L بت عبر خط اتصال بمعدل إرسال R بت/ثانية، يكون زمن إرسال الحزمة L/R ثانية



شكل 1-18: تبديل الحزم

تبديل الدارات (Circuit Switching)

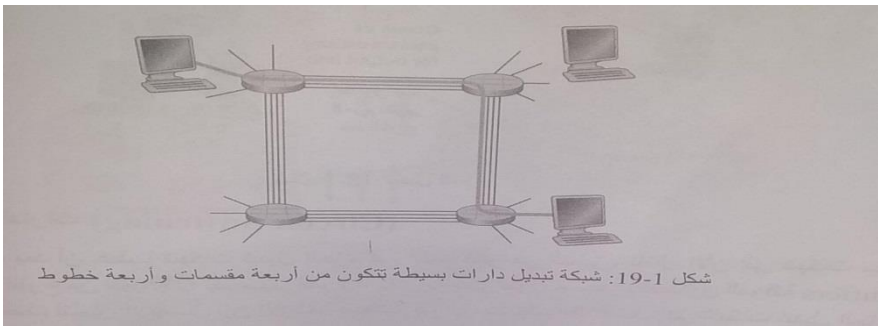
عزيزي الطالب، بعد أن غطينا شبكات تبديل الحزم في القسم الفرعي السابق، ننتقل الآن إلى شبكات تبديل الدارات. في هذا النوع من الشبكات، يتم حجز الموارد اللازمة على طول المسار المخازن المؤقتة Buffers ، ومعدل إرسال الخط لتأمين الاتصال بين الأنظمة النهائية طوال مدة جلسة الاتصال. في شبكات تبديل الحزم، لا يتم حجز هذه الموارد، بل تستخدم عند الطلب، ونتيجة لذلك، قد تضطر إلى الانتظار في قائمة الانتظار للحصول على خط اتصال.

وتعتبر شبكات الهاتف التقليدية مثالا على شبكات تبديل الدارات، فعندما يريد شخص إرسال معلومات (صوت أو فاكس) إلى شخص آخر، وقبل أن يتمكن المرسل من إرسال المعلومات، يجب إنشاء اتصال بين المرسل والمستقبل، وتحافظ المقسمات في المسار بين المرسل والمستقبل على حالة الاتصال. في لغة الاتصال الهاتفي، يسمى هذا الاتصال دارة، فعندما تنشئ الشبكة هذه الدارة، فإنها تحتفظ أيضا بمعدل إرسال ثابت في خطوط الشبكة (يبين حصة كل خط من سعة الإرسال طوال مدة الاتصال. وبما أنه قد تم حجز معدل إرسال معين لهذا الاتصال بين المرسل والمستقبل، نضمن نقل البيانات بمعدل ثابت.

يوضح الشكل (19-1) شبكة تبديل الدارات. في هذه الشبكة، ترتبط المقسمات الأربعة من خلال أربعة خطوط يحتوي كل خط على اربع دوائر، ليتمكن كل خط من دعم اربعة اتصالات متزامنة. يتم ربط كل مضيف (مثل أجهزة الحاسوب ومحطات العمل مباشرة بأحد المقسمات، فعندما يرغب مضيفان بالاتصال، تنشئ الشبكة اتصالا متكاملا بين المضيفين. ليتمكن المضيف A من الاتصال بالمضيف B ، يجب على الشبكة أولا حجز دارة واحدة على كل خط في هذا المثال، يستخدم الاتصال الدارة الثانية في الخط الأول والدارة الرابعة في الخط الثاني. ولأن كل خط يحتوي على أربع دوائر، يحصل كل خط يستخدمه الاتصال على ربع

سعة الإرسال الإجمالية للخط طوال مدة الاتصال. فمثلا، إذا كان معدل إرسال كل خط بين مقسمين متجاورين ميجا بت في الثانية، فإن كل خط تبديل يحصل على 250 كيلو بت في الثانية من معدل الإرسال الإجمالي المخصص

في المقابل، اعتبر أن مضيفا يريد إرسال حزمة إلى مضيف آخر عبر شبكة تبديل الحزم مثل الإنترنت، في هذه الحالة ستنتقل الحزمة عبر سلسلة من خطوط الاتصال، وخلافا لتبديل الدارات، ترسل الحزمة إلى الشبكة دون حجز موارد خط الاتصال على الإطلاق، فإذا كان أحد الخطوط مزدحما بحزم أخرى ستنتقل عبر الخط ذاته في نفس الوقت، فعلى المضيف الانتظار ووضع الحزمة في مخزن مؤقت على جانب المرسل ما قد يسبب التأخير. أي أن الإنترنت تبذل قصارى جهدها لتسليم الحزم باسرع وقت، ولكنها لا تقدم أي ضمانات.



شكل 1-19: شبكة تبديل دارات بسيطة تتكون من أربعة مقسمات وأربعة خطوط

1-8 | التأخير والفاقد والإنتاجية في شبكات تبديل الحزمة

Delay, Loss, and Throughput in Packet-) (Switched Networks

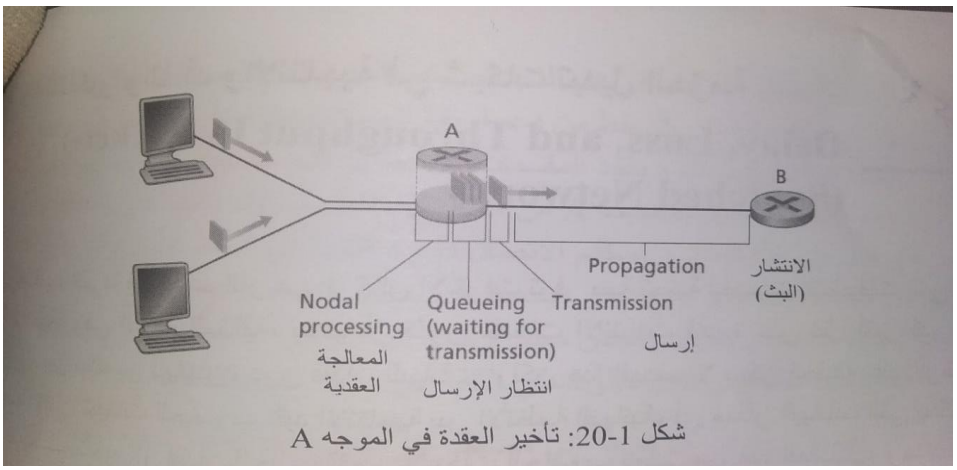
عزيزي الطالب، اشرنا في القسم الفرعي 1-5 أن الإنترنت توفر بنية تحتية لخدمات التطبيقات التي تعمل على الأنظمة النهائية، وفي الحالة المثالية، يجب أن تكون خدمات الإنترنت قادرة على نقل أكبر قدر ممكن من البيانات فورا بين نظامين نهائيين، دون فقدان للبيانات. ولكن هذا الهدف لا يمكن تحقيقه على أرض الواقع، وبدلا من ذلك، فإن شبكات الحاسوب تقيد الإنتاجية بين الأنظمة النهائية، أي مقدار البيانات التي يمكن نقلها في الثانية، وقد يحدث تأخير أو فقدان الحزم بالفعل. فشبكات الحاسوب تعاني من هذه المشاكل، وهناك العديد من طرق التعامل معها في هذا القسم، سنبدأ بتحديد مقدار التأخير والفاقد والإنتاجية في شبكات الحاسوب.

1-8-1 التأخير في شبكات تبديل الحزمة (Delay in Packet-Switched Networks)

تبدا حزمة البيانات رحلتها من مضيف(المصدر)، وتمر عبر سلسلة من الموجهات، وتنتهي في مضيف آخر الوجهة. عندما تنتقل الحزمة من عقدة (المضيف او الموجه) إلى العقدة اللاحقة (المضيف أو الموجه على طول المسار، فإنها قد تعاني من عدة أنواع من التأخير في كل عقدة، وأهم هذه الأنواع هو تأخير معالجة العقدة (Nodal Processing Delay) وتأخير الارسال (Transmission Delay (Queuing) تأخير الانتشار)Propagation، Delay)، وتتراكم هذه الأنواع لتعطي التأخير الكلي، ويمكن التعبير عن التأخير الكلي للعقدة رياضيا بالمعادلة

$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

ونشير هنا إلى الفاقد في الحزم Packet loss) الناتج عن امتلاء الطابور وارتباطه بتأخير الطابور. يتأثر أداء العديد من تطبيقات الإنترنت كالبحث وتصفح الويب والبريد الإلكتروني والخرائط والرسائل الفورية و اتصالات الصوت عبر بروتوكول الإنترنت بتأخير الشبكة، ولفهم تبديل الحزم وشبكات الحاسوب، يجب أن نفهم طبيعة هذه التأخيرات وأهميتها. يبين الشكل (1-20) التأخير في المسار بين المصدر والوجهة حيث تنتقل الحزمة من الموجه A إلى الموجه B ، وهدفنا هو تحديد تأخير العقدة في الموجه A ، لاحظ أن الموجه A يحتوي على خط صادر يؤدي إلى الموجه B ، يسبق هذا الخط طابور (Queue) يسمى المخزن المؤقت (Buffer). عندما تصل الحزمة إلى الموجه A ، فإنه يتفحص مقدمة الحزمة لتحديد الخط الصادر المناسب للحزمة، ثم يوجه الحزمة إلى هذا الخط في هذا المثال، يؤدي الخط الصادر للحزمة إلى الموجه B ، ولا يمكن إرسال الحزمة على هذا الخط إلا إذا لم يكن هناك حزمة أخرى قيد الإرسال حاليا أو تسبقها في الطابور؛ وإلا ستقوم الحزمة التي وصلت حديثا بالانضمام الى قائمة الانتظار في الطابور



شكل 1-20: تأخير العقدة في الموجه A

وهنا تتعرض الحزمة الأنواع عدة من التأخير، نوضحها بإيجاز فيما يأتي:

تأخير المعالجة Processing delay ويشمل الزمن اللازم لفحص مقدمة الحزمة للتعرف على وجهتها، وكذلك الزمن ال لازم لفحص الخطأ على مستوى البت في الحزمة المرسلة، ولا يتجاوز تأخير المعالجة بضع ميكرو ثوان.و بعد معالجة العقدة يحول الموجه الحزمة إلى الطابور الذي يسبق الخط المؤدي إلى الموجه .B

2.تأخير الطابور Queuing delay :في الطابور، تتعرض الحزمة إلى تأخير ناتج عن الانتظار على الخط وتعتمد فترة تأخير الحزمة في الطابور على عدد الحزم التي تصل قبلها إلى الطابور بانتظار الخط فإذا كان الطابور فارغا ولم يكن هناك حزم أخرى قيد الإرسال، فسيكون تأخير الطابور صفرا وإذا كانت الطابور مزدحما بالحزم التي تنتظر، سيكون التأخير طويلا، ويتراوح تأخير الطابور من ميكرو ثنية إلى 1 ميلى ثانية.

تأخير الارسال (Transmission delay): هو الزمن اللازم لنقل جميع بتات الحزمة عبر خط الاتصال، اذا افترضنا أن الحزمة تنتقل بأسلوب "من يأتي أولا يخدم أولا")-First-come-first served) الشائع في شبكات تبديل الحزم، فلا يمكن إرسال الحزمة إلا بعد كل الحزم التي وصلت قبلها، فإذا كان طول الحزمة (Lبت)، ومعدل إرسال الخط من جهاز التوجيه A إلى جهاز التوجيه B (R بت ثانية)، فإن تأخير الإرسال هو (L/R ثانية)، ويتراوح من 1ميكرو ثانية إلى 1ميلي ثانية

تأخير الانتشار او البت (Propagation delay):هو الزمن اللازم لانتشار الحزمة من بداية الخط الى الموجة B ويساوي d/s اي المسافة بين الموجه A والموجه B) (d مقسومة على سرعة انتشار الخط s ينتشر البت بسرعة انتشار خط الاتصال وهي تعتمد على نوع الوسط المادي ويتراوح بين 10^8×2 و 10^8×3 متر /ث اي اقل من سرعة الضوء بقليل.

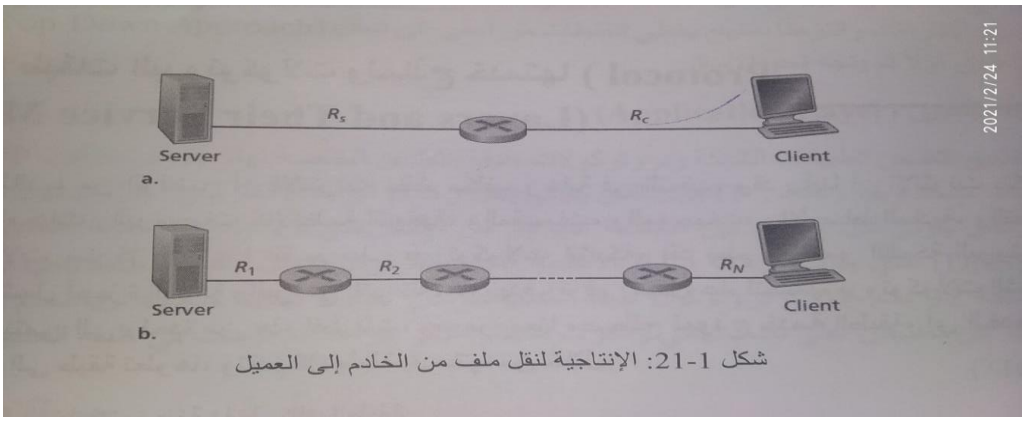
الفاقد في الحزم (Packet Loss) عزيزي الطالب، يعتبر تأخير الطابور الأكثر تعقيدا، إذ يتباين من حزمة إلى أخرى اعتمادا على موقعها في الطابور حتى لو ارسلت هذه الحزم معا، لا يقل إحصائيا بالمتوسط والتباين والاحتمال من ناحية أخرى، فلي طول الطابور محدود وقد يمتلئ في أي لحظة، فإذا وصلت الحرمة إلى الموجه وهو مقلى ولم تجد حيز لتخزينها في المخزن المؤقت، عندها يقوم الموجه بإهمالها (Drop)، أي تضيع الحزمة فينتج الفاقد (Loss) في الحزم، ويطلق على هذا السيناريو الفيضان (Overflow).

من وجهة نظر النظام النهائي، يبدو فقدان الحزم وكأنه تم نقل حزمة إلى نواة الشبكة ولكنها لم تصل أبدأ من الشبكة إلى وجهتها، وقد يضطر المرسل إلى إعادة إرسال الحزم المفقودة لضمان نقل جميع البيانات في نهاية المطاف من المصدر إلى الوجهة. وتزداد نسبة الفاقد في الحزم بزيادة كثافة حركة المرور، لذا، غالبا ما يقاس أداء العقدة باحتمال فقدان. الحزم إضافة الى زمن التأخير.

الانتاجية في شبكات الحاسوب (Throughput in Computer Networks)

عزيزي الطالب، بالإضافة إلى التأخير والفاقد في الحزم، هناك مقياس آخر مهم لأداء الشبكات هو الإنتاجية من نظام نهائي لآخر. لتحديد سرعة النقل، سنأخذ بعين الاعتبار نقل ملف كبير من المضيف A إلى المضيف B عبر الشبكة، وقد يكون هذا الملف مقطع فيديو كبير من نظير لآخر عبر نظام مشاركة الملفات P2P . الإنتاجية اللحظية هي المعدل الذي يستقبل فيه المضيف B الملف بالبت/ثانية، فإذا كان الملف يتكون من F بت ويستغرق نقله T ثانية، فإن متوسط الإنتاجية في نقل الملف F/T بت/ثانية. بعض التطبيقات مثل الاتصال الهاتفي عبر الإنترنت، تتطلب أن يكون التأخير منخفضا ومتوسط الإنتاجية اللحظية أكثر من قيمة محددة تسمى العتبة (أكثر من24 kbps لبعض تطبيقات الاتصال الهاتفي عبر الإنترنت، وأكثر من kbps 256 لبعض تطبيقات فيديو الوقت الحقيقي .لتوضيح مفهوم الإنتاجية، دعنا نأخذ بعض الأمثلة يوضح الشكل (1-21 - a)

نظامين نهائيين، خادم وعميل، متصلان عبر خطي اتصال وموجه، لنحسب الإنتاجية لنقل ملف من الخادم إلى العميل، ولنفرض أن Rs معدل التنقل بين الخادم والموجه؛ وأن Rc معدل النقل بين الموجة والعميل، ولنفرض أن البنات الوحيدة التي يتم إرسالها عبر الشبكة بالكامل هي التي ترسل من الخادم إلى العميل. فما هو معدل نقل البيانات من الخادم إلى العميل؟ سنعتبر البتات هي السائل وخطوط الاتصال هي الأنابيب، فلا يمكن للخادم ضخ البنات عبر خطه بمعدل أسرع من ، bps Rs ولا يستطيع الموجه إعادة توجيه البتات بمعدل أسرع من Rs bps.



شكل 1-21: الإنتاجية لنقل ملف من الخادم إلى العميل

فإذا كان ,Rs،Rc> فإن البتات التي يخضها الخادم ستدفق مباشرة عبر الموجه وتصل إلى العميل بمعدل Rs bps، أي أن الإنتاجية أيضا ,Rs، bps وإذا كان Rc<Rsمن ناحية أخرى، فلن يتمكن الموجه من إعادة توجيه البنات بالسرعة التي يتلقاها بها، وفي هذه الحالة، ستغادر البتات الموجه فقط بمعدل R c، أي أن الإنتاجية Rc. bps)لاحظ: في حال استمرار وصول البتات إلى الموجه بمعدل ,Rsومغادرته بمعدل ، Rc فسينمو تراكم البتات على الموجه في انتظار الإرسال إلى العميل، وهو وضع غير مرغوب به إلى حد كبير!). بذلك، يكون معدل النقل في هذه الشبكة البسيطة المكونة من خطين {min، Rc .Rs} أي بمعدل نقل خط الاختناق(Bottleneck) بعد تحديد سرعة النقل، يمكننا الآن احتساب الوقت المستغرق لنقل ملف كبير حجمه F bits من الخادم الى العميل Rs Rc F/ min.

مثال (1-2)

لنفترض أنك تقوم بتنزيل ملف MP3 بحجم 32 مليون بت، ومعدل إرسال الخادم Rs Mbps = 2، ولديك خط نفاذ بمعدل Rc= 1Mbps

الحل:

الوقت اللازم لنقل الملف = حجم الملف / معدل النقل =

= 32 ثانية

يوضح الشكل(b-21-1) شبكة ذات خطوط اتصال بين الخادم والعميل عددها N، ومعدلات نقلها R1. R2. ...RN

وعند تطبيق نفس التحليل الخاص بشبكة الخطين، نجد أن معدل نقل الملفات من الخادم إلى العميل هو min، {RR2 ,{R], ،.. و هو نفسه معدل نقل خط الاختناق مرة أخرى على طول المسار بين الخادم والعمبل توضح الأمثلة الواردة في الشكل (21-1) أن الإنتاجية تعتمد على معدلات إرسال الخطوط التي تتدفق فيها البيانات، وقد رأينا أنه عندما لا يكون هناك حركة مرور أخرى متداخلة، ويمكن ببساطة تعريف الإنتاجية على أنها الحد الأدنى لسرعة الإرسال على طول المسار بين المصدر والوجهة ستفحص الإنتاجية في شبكات الحاسوب في التدريبات وفي الفصول اللاحقة.

1-9 | طبقات البروتوكولات ونماذج خدمتها (Protocol (Layers and Their Service Models |

عزيزي الطالب، من الواضح أن الإنترنت نظام مكلف وغاية في التعقيد، وقد رأينا أن الإنترنت يتكون من العديد من المعدات والبرمجيات كالأنظمة النهائية، والمقسمات، والموجهات، والأوساط المادية، والتطبيقات، والبروتوكولات ستركز في هذا القسم على بروتوكولات الشبكة، فقد نظم مصممو الشبكة البروتوكولات في طبقات تشمل أجهزة الشبكة والبرامج التي تنفذها، بهدف توفير هيكل عام لتصميم بروتوكولات الشبكة، كل بروتوكول ينتمي إلى واحدة من هذه الطبقات، وسيمر معنا مصطلح نموذج خدمة الطبقة، أي الخدمات التي تقدمها طبقة إلى طبقة تعلوها، وتقدم كل طبقة خدماتها من خلال

1. تنفيذ إجراءات معينة داخل تلك الطبقة
2. استخدام خدمات الطبقة التي تحتها مباشرة.

على سبيل المثال، قد تتضمن الخدمات المقدمة من طبقة n تسليما موثوقا للرسائل من طرف لآخر، وقد ينفذ ذلك عن طريق خدمة تسليم غير موثوقة للرسائل من طرف الآخر للطبقة 1- ، وإضافة وظيفة للطبقة n لكشف الرسائل المفقودة وإعادة إرسالها.

يمكن تنفيذ طبقة البروتوكول في البرمجيات أو في الأجهزة أو كليهما، فبروتوكولات طبقة التطبيقات مثل HTTP و SMTP تنفذ دائما في برمجيات النظم النهائية؛ وكذلك الأمر في بروتوكولات طبقة النقل. ولأن الطبقة المادية الفيزيائية وطبقة ربط البيانات مسؤولتان عن التعامل مع الاتصالات عبر خط معين، فيتم تنفيذهما في بطاقة الشبكة المرتبطة بخط اتصال معين، سواء كانت Ethernet أو Wi-Fi ، وغالبا ما تنفذ بروتوكولات طبقة الشبكة في الأجهزة والبرمجيات معا. وتوزع بروتوكولات الطبقة n ما بين الأنظمة النهائية والمقسمات، ومكونات الشبكة الأخرى، أي أنه يوجد غالبا جزء من بروتوكول الطبقة n في كل مكون من هذه المكونات.

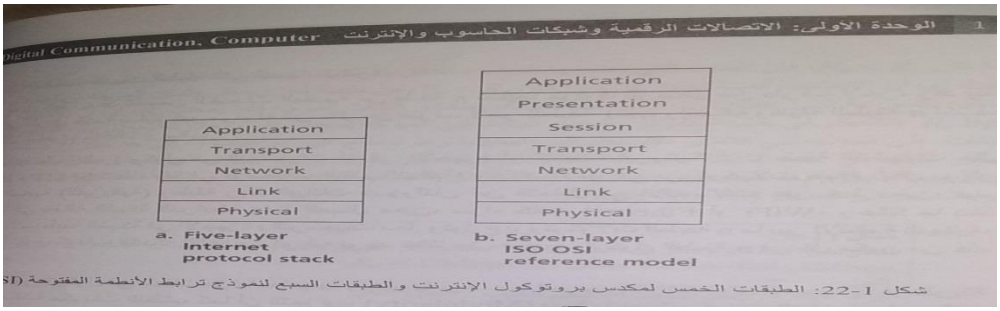
إن لتنظيم البروتوكول في طبقات مزايا مفاهيمية وتنظيمية يوضحها المرجع [RFC 3439]، حيث يوفر طريقة منظمة لمناقشة مكونات النظام، كما أن التجزيء (Modularity) يجعل تحديث مكونات النظام أسهل، ولكن هناك بعض العيوب مثل التكرار في وظائف الطبقات، كما أن وظائف الطبقة قد تحتاج إلى معلومات تتوفر في طبقة أخرى فقط، ما يخالف الهدف من فصل الطبقات. وعند النظر إليها كرزمة واحدة، تسمى بروتوكولات الطبقات المختلفة بمكدس البروتوكول(Protocol Stack).

طبقات مكدس بروتكول الانترنت (Internet Protocol Stack Layers)

يتكون مكدس بروتوكول الإنترنت (Internet Protocol Stack) من خمس طبقات: المادية Physical)، والارتباط (Link)، والشبكة (Network)، والنقل (Transport)، والتطبيق (Application)، كما هو موضح في الشكل (1-22 a -)، وقد قمنا، عزيزي الطالب، بتنظيم هذا الكتاب استنادا إلى طبقات مكدس بروتوكول الإنترنت، والتزمنا بمنهج يغطي الطبقات من أعلى إلى أسفل. (Top-Down A proach) فيغطي طبقة التطبيق أولا ثم يتجه نحو الأسفل.

1. طبقة التطبيق (Application Layer)

طبقة التطبيق تتضمن تطبيقات الشبكة وبروتوكولات طبقة التطبيق الخاصة بها، مثل بروتوكول HTTP الذي يتولى طلب مستند ويب ونقله، و SMTP الذي يتولى بنقل رسائل البريد الإلكتروني، وبروتوكول نقل الملفات (FTP) الذي يتولى نقل الملفات بين الأنظمة النهاية. من الأمثة على بروتوكولات طبقة التطبيق نظام اسم النطاق (DNS) الذي يتولى ترجمة اسماء الأنظمة النهائية للإنترنت (مثل العنوان www.qou.edu) إلى عنوان شبكة 32 بت. يتوزع بروتوكول طبقة عبر أنظمة نهائية متعددة، ويستخدم لتبادل حزم المعلومات بين تطبيقين على نظام نهائي وآخر، وسنطلق على حزمة المعلومات في طبقة التطبيق "رسالة"(message).



2. طبقة النقل (Transport Layer)

تتولى هذه الطبقة نقل رسائل طبقة التطبيق بين طرفي التطبيق، وهناك بروتوكولا نقل في الإنترنت TCP و UDP ، وكلاهما يمكنه نقل رسائل طبقة التطبيق، حيث يوفر TCP خدمة اتصال موجهة لتطبيقاته، وهي خدمة تضمن تسليم رسائل طبقة التطبيق إلى الوجهة، والتحكم بالتدفق (أي، المواءمة بين سرعة المرسل والمستقبل(، كما يقوم بتقسيم الرسائل الطويلة إلى أجزاء أقصر، ويوفر آلية للتحكم باحتقان الشبكة من خلال تخفيض معدل الإرسال لدى المصدر، أما بروتوكول UDP ، فيوفر خدمة بدون اتصال لتطبيقاته، وهي خمة الا توفر أيا من الوظائف التي يقدمها TCP . في هذا الكتاب، سنطلق على حزمة طبقة النقل شريحة"(Segment)

3. طبقة الشبكة (Network Layer)

تتولى طبقة الشبكة مسؤولية نقل حزم طبقة الشبكة" مخططات البيانات" (Datagrams) من مضيف إلى آخره ويقوم بروتوكول طبقة النقل (TCP أو UDP(في المضيف المصدر بتمرير شريحة طبقة النقل وعنوان الوجهة إلى طبقة الشبكة، فتوفر طبقة الشبكة خدمة توصيل هذه الشريحة إلى طبقة النقل في المضيف الوجهة اول طبقة الشبكة بروتوكول IP المشهور، الذي يحدد حقول مخطط البيانات وكيفية تصرف الأنظمة النهائية والموجهات بهذه الحقول. هناك بروتوكول IP واحد فقط، وعلى جميع مكونات الإنترنت التي تحتوي - طبقة شبكة تشغيله، كما تحتوي طبقة الشبكة على بروتوكولات التوجيه التي تحدد المسارات التي تتخذها وحدات البيانات بين المصدر والوجهة ورغم أن طبقة الشبكة تحتوي على عدد من بروتوكولات التوجيه إضافة إلى IP ، إلا أنه يشار إليها بطبقة IP ، لأن IP هو الذي يربط الإنترنت ببعضها البعض.

4. طبقة الارتباط (Link Layer)

تقوم طبقة الارتباط بنقل حزمة البيانات من عقدة (مضيف او موجه التالية في المسار، في كل عقدة تمرر طبقة الشبكة مخطط البيانات إلى أسفل نحو طبقة الارتباط التي تنقل بدورها مخطط البيانات إلى العقدة التالية على طول المسار، وفي هذه العقدة)التالية(، تقوم طبقة الارتباط بتمرير مخطط البيانات إلى أعلى نحو طبقة الشبكة. تعتمد الخدمات التي طبقة الارتباط على البروتوكول المستخدم فقد توفر بعض بروتوكولات طبقة خدمة التسليم الموثوق من العقدة المرسلة عبر خط اتصال واحد الى العقدة المستقبلة .وهي تختلف عن خدمة التسليم الموثوق التي يوفرها TCP من نظام إلى اخر ومن الأمثلة على بروتوكولات طبقة الارتباط الإيثرنت وبروتوكول Wi-Fi وبروتوكول DOCSIS الخاص عبر الكوابل. ولأن مخططات البيانات قد تحتاج إلى اجتياز عدة خطوط للانتقال من المصدر إلى الوجهة، يتم التعامل معها من خلال بروتوكولات مختلفة عند الخطوط المختلفة على طول مسارها فقد تتم معالجة مخطط البيانات من خلال Ethernet على أحد الخطوط ومن خلال PPP على الخط التالي . لذا قد تتلقى طبقة الشبكة خدمة مختلفة من كل بروتوكول. في هذا الكتاب سنطلق على حزم طبقة الارتباط اطارات(Frames)

الطبقة المادية (Physical Layer)

تتلخص مهمة الطبقة المادية بنقل كل بت ضمن الإطار المرسل من عقدة إلى أخرى، البروتوكولات في هذه الطبقة مرة أخرى على خط الاتصال وعلى وسط الإرسال الفعلي، مثل الأزواج المجدولة والألياف الضوئية، وغيرها فعلى سبيل المثال، يحتوي الإيثرنت على العديد من بروتوكولات الطبقة المادية أحدها للزوج المجدول النحاسي، وآخر المحورية، للألياف الضوئية، وهكذا، وفي كل حالة يتم نقل خط الاتصال بطريقة مختلفة .

2-9-1 نموذج ترابط الأنظمة المفتوحة The OSI Model

عزيزي الطالب، بعد أن ناقشنا طبقات مكدس بروتوكول الإنترنت بالتفصيل، ينبغي أن نذكر أنه ليس المكدس الوحيد، ففي أواخر السبعينيات، اقترحت المنظمة الدولية للمقاييس (ISO) نموذجا ينظم شبكات الحاسوب في سبع طبقات أطلقت عليه نموذج ترابط الأنظمة المفتوحة OSI 2012)(ISO تبلور نموذج SI0 عندما كانت بروتوكولات الإنترنت في مهدها، وكان مجرد واحد من مجموعات البروتوكولات قيد التطوير. كما يبين الشكل (1-22 - b)، الطبقات السبع من النموذج المرجعي (OSI) هي: طبقة التطبيق، وطبقة العرض، وطبقة الجلسة، وطبقة النقل، وطبقة الشبكة، وطبقة ربط البيانات، والطبقة المادية. وظيفة خمسة من هذه الطبقات نفس نظائرها المسماة بالمثل في برتوكول الإنترنت تقريبا. لذا، سنناقش الطبقتين الإضافيتين في نموذج OSI المرجعي.

- **طبقة العرض:** توفر هذه الطبقة خدمات تسمح للتطبيقات المتصلة بتفسير معنى البيانات المتبادلة، مثل ضغط البيانات وتشفيرها)واضحة التفسير(، بالإضافة إلى وصف البيانات التحرير التطبيقات من التنسيق الداخلي لتمثيل البيانات أو تخزينها الذي يختلف من حاسوب لأخر).

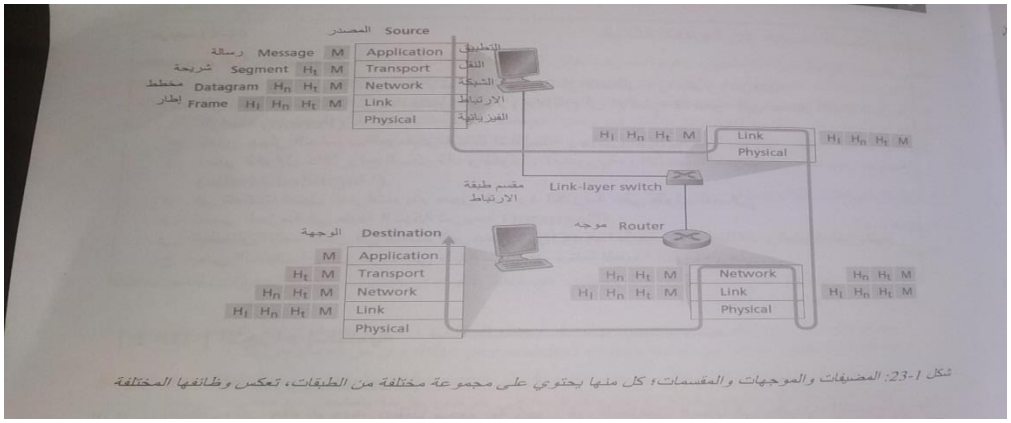
- **طبقة الجلسة:** تتيح طبقة الجلسة تحديد تبادل البيانات وتزامنه، بما في ذلك وسائل بناء نظام التحقق من البيانات واستعادتها .

إن حقيقة افتقار الإنترنت لطبقتين يوفرهما النموذج المرجعي (OSI) يطرح سؤالين مهمين: هل الخدمات اللاتي تقدمها هاتان الطبقتان غير ضرورية؟ وهل تحتاج التطبيقات إلى هذه الخدمات؟ إن إجابة الانترنت واضحة وهي نفسها دائما، فالأمر متروك لمطور التطبيق ليقرر ما إذا كانت الخدمة مهمة، وإذا كانت كذلك فان الأمر متروك لمطور التطبيق لبناء هذه الوظيفة.

2-9-1 التغليف (Encapsulation)

عزيزي الطالب، من المفاهيم المهمة في نقل الحزم غير الشبكات مفهوم التغليف، كما يوضح الشكل (231 في المضيف المرسل، يتم تمرير رسالة (Message طبقة التطبيقات (M) إلى طبقة النقل، وفي ابسط الحالات، تأخذ طبقة النقل الرسالة وتضيف إليها معلومات مقدمة طبقة النقل (Ht) التي تستخدمها طبقة النقل في جانب المستقبل تشكل رسالة طبقة التطبيق ومعلومات مقدمة طبقة النقل معا شريحة (Segment) طبقة النقل.

وبالتالي، فإن شريحة طبقة النقل غلفت رسالة طبقة التطبيقات قد تحتوي المعلومات المضافة على معلومات تسمح لطبقة النقل في جانب المستقبل بتسليم الرسالة إلى التطبيق المناسب في الأعلى، وقد تحتوي على بنات كشف الأخطاء التي تسمح للمستقبل بتحديد ما إذا كان قد تم تغيير محتوى الرسالة أثناء انتقالها عبر المسار ثم تمرر طبقة النقل الشريحة إلى طبقة الشبكة التي بدورها تضيف معلومات مقدمة طبقة الشبكة (Hn) وتشمل عنواني النظام النهائي المصدر والوجهة، لتشكل مع مخطط بيانات)Datagram(طبقة الشبكة. ثم يتم تمرير مخطط البيانات إلى طبقة الارتباط والتي تضيف معلومات مقدمة طبقة الارتباط وإنشاء إطار (Frame)الارتباط.



شكل 1-23: المضيفات والموجهات والمقسمات؛ كل منها يحتوي على مجموعة مختلفة من الطبقات، تعكس وظائفها المختلفة

ملاحظة

عزيزي الطالب، تحتوي الحزمة في كل طبقة على نوعين من الحقول: حقول المقدمة وحقل الحمولة (Payload) أي حزمة البيانات الفعلية من الطبقة الأعلى. من جانب آخر، قد يكون التغليف أكثر تعقيدا، فمثلا، قد تقسم الرسالة الكبيرة إلى عدة شرائح، وقد تقسم كل شريحة إلى عدة وحدات بيانات، وفي جانب المستقبل، يجب إعادة بناء الشريحة من مخططات البيانات المكونة لها.

مسرد مصطلحات:

احادية القطب Unipolar

احتمال الخطأ probability of Error

ازواج الاسلاك النحاسية المجدولة Twisted pairs

اطار ،frame

اقمار تدور حول الارض في مدارات منخفضة low earth :LEO satellite

اقمار مستقرة بالنسبة الى الارض Geostationary satellite

الاتصال اللاسلكي Wi-Fi

الاختناق Bottleneck

الازواج المجدولة Twisted pair

الاشارات signals

الاشارات التناظرية analog signals

الاشارات الرقمية digital signals

الاشارات الضوئية light signals

الاشارات الكهربائية electrical signals

الاشارات الكهرومغناطيسية electromagnetic signals

الاقمار الصناعية satellite

الالياف الضوئية fiber optics

الالياف الضوئية للمنازل FTTH throughput in computer networks

الاوساط المادية physical media

الاوساط الموجهة guided media

الاوساط غير الموجهة unguided media

الايثرنت Ethernet

التأخير في شبكات تبديل الحزمة delay in packet switched networks

التجزيء modularity

التداخل interference

التراسل احادي الاتجاه Half duplex

التراسل البسيط simplex

التراسل باتجاهين full duplex

التردد والطيف وعرض الحزمة (Frequency, Spectrum and Bandwidth)

الترميز شبه الثلاثي

التضاؤل متعدد المسار (Multipath fading)

التضمين (Modulation)

التغليف (Encapsulation)

الحزم (Packets

الخط المباشر (Direct link)

الزوج المجدول غير الحامي (Unshielded Twisted Pair: UTP

الطبقات من اعلى الى اسفل (Top-Down Approach)

الطبقة المادية (Physical Layer

الطلب الهاتفي (Dialup

الطيف التردد Frequency spectrum

العلامة والفراغ (Mark and Space)

الفاصل الثنائي (Binary Interval)

الفاقد في الحزم (Packet loss

الفاقد في الحزم (Packet Loss)

الفيضان (Overflow)

الكوابل المحورية (Coaxial Cable

الكوابل المحورية (Coaxial Cables

اللاعودة إلى الصفر معكوس

اللا عودة إلى مستوى الصفر

المخازن المؤقتة Buffers

المحزن الموقت Buffer

المستقطبة(Polar

المطالAmplitude

Hardware and Software (مكونات مادية وبرمجية

المنظمة الدولية للمقاييس ISO)

(Routers) الموجهات

(Home Access النفاذ المنزلي

بروتوكول الإنترنت TP)

بروتوكول التحكم بالنقل

(Transport Control Protocol: TCP)

(User Datagram Protocol: UDP بروتوكول مخطط بيانات المستخدم

(Transmission delay)تأخير الارسال

(Transmission Delay)تأخير الارسال

(Propagation delay)تأخير الانتشار

(Queuing delay)تأخير الطابور

(Processing delay تأخير المعالجة

(Nodal Processing Delay تأخير معالجة العقدة

(Packet Switching تبديل الحزم

(Circuit Switching تبديل الدارات

(point-to-point تراسل نقطة الى نقطة

ترميز مانشستر

ترميز مانشستر التفاضلي

(Shadow fading- تضاؤل الطيف

(Quantization تكميم قيم العينات

ثنائي القطبية مع قلب العلامة بالتناوب

جهاز الفصل (Splitter)

(Servers خادمات

(Internet Services خدمات انترنت

Phase Characteristics خصائص الطور

(Cable كوابل

(Digital Subscriber Line: DSL) خط المشترك الرقمي

(Communication Linksخطوط الاتصال ى

(Optical Carrier: OC خطوط الناقل الضوئي

(Message رسالة

(Access Networks شبكات النفاذ

(Segment شريحة

(Queue)طابور

(Link Layer .طبقة الارتباط

(Application Layer طبقة التطبيق

(Network Layer طبقة الشبكة

(Transport Layer طبقة النقل

(Spectrum طبقة الاشارة

(Bandwidth عرض النطاق الترددي

(Absolute Bandwidth) عرض النطاق الترددي المطلق

(Clients) عملاء

(Signal elements per second عنصر اشارة في الثانية

(Path loss) فقدان المسار

(Terrestrial and

(Satellite Radio Channels) قنوات الراديو الارضية والفضائية

(Cable Internet Access)كابل النفاذ الى الانترنت

(Bit duration or length) مدة او طول البت

(Datagram) مخطط البيانات

(Internet Service Providers "ISPS") مزودي خدمة الانترنت

(Transmission Rate)معدل الارسال

(Modulation rate) معدل التضمين او التباين

(Error Rate)معدل الخطأ

(Data signaling rate) معدل بيانات الاشارة

(Packet Switches) مقسمات الحزم

Link-Layer) Switches مقسمات طبقة الارتباط

(Internet Protocol Stack) مكدس بروتوكول الانترنت

Data Transmission System (DTS)lنظام تراسل البيانات

(The OSI Model) نموذج ترابط الانظمة المفتوحة

2-2 مبادئ تطبيقات الشبكة Principles of Network Applications

لنفترض أن لديك فكرة تطبيق شبكة جديد سيقدم خدمة كبيرة للبشرية، أو يجلب لك ثروة كبيرة، أيا كان الدافع سندرس في هذه الوحدة كيف يمكنك تحويل الفكرة إلى تطبيق حقيقي. تعتبر كتابة البرامج لتعمل على أنظمة مختلفة وتتصل بعضها البعض عبر الشبكة في صميم تطوير تطبيقات الشبكة، فعلى سبيل المثال، في تطبيقات الويب هناك نوعان من البرامج المميزة التي يتصل بعضها ب بعض برنامج المتصفح (Browser) قيد التشغيل على جهاز المستخدم)سطح المكتب، أو الحاسوب المحمول، أو الجهاز اللوحي، أو الهاتف الذكي)؛ وبرنامج خادم الويب قيد التشغيل على الخادم. مثال آخر، في نظام تبادل الملفات النظير ا(peer to-peer :p2p)هناك برنامج على كل جهاز مشارك في الملفات، في هذه الحالة، تكون البرامج متشابهة أو متطابقة على مختلف الأجهزة.

لذا، عند تطوير تطبيقك الجديد، تحتاج إلى كتابة برنامج يعمل على أنظمة متعد دة، ويمكنك كتابة هذا البرنامج باستخدام أي لغة برمجة مثل C ، او جافا، أو بايثون، والأهم من ذلك، أنك لن تحتاج إلى كتابة البرنامج ليعمل على أجهزة الشبكة مثل الموجهات (Routers) أو المقسمات (Switches)، حتى لو كنت تريد أن تكتب البرنامج لهذه الأجهزة الأساسية للشبكة، فلن تكون قادرا على ذلك كما تعلم عزيزي الطالب، فإن أجهزة الشبكة الأساسية لا تعمل في طبقة التطبيقات، بل تعمل في طبقات أدني مثل طبقة الشبكة، وهذا التصميم سهل وسرع تطوير البرامج التطبيقية، ونشر مجموعة واسعة من تطبيقات الشبكة كما هو مبين في الشكل 2-1

2-2-1 (Network Application Architectures) بنية تطبيقات الشبكة

قبل الغوص في كتابة كود البرامج، يجب أن يكون لديك خطة معمارية موسعة للتطبيق الخاص بك، واضعا بالحسبان أن بنية التطبيق تختلف بشكل واضح عن بنية الشبكة، فمن وجهة نظر مطور التطبيق، معمارية الشبكة ثابتة وتوفر مجموعة محددة من الخدمات للتطبيقات. من جانب آخر، فإن معمارية التطبيق يصممها مطور التطبيق والتي تفرض بنية التطبيق على مختلف الأنظمة، مستندا إلى أحد النماذج المعمارية السائدة المستخدمة في تطبيقات الشبكة الحديثة: معمارية النظير ل لنظير Peer-to-Peer:P2P أو العميل الخادم (Client-Server)

في معمارية النظير للنظير (P2P)، هناك حد أدنى من الاعتمادية على خوادم مخصصة في مراكز البيانات، بل يستغل التطبيق الاتصال المباشر بين أزواج من المضيفين المتصلين بشكل متقطع يدعى كل منها النظير ، وهي أجهزة غير مملوكة من مزود الخدمة، بل هي أجهزة حاسوب مكتبية أو محمولة يملكها المستخدمون، ومعظم النظراء يقيمون في المنازل والجامعات والمكاتب، ويتواصلون مع نظرائهم دون المرور بخالم مخصص. والعديد من التطبيقات الأكثر شيوعا هذه الأيام والتي تشهد حركة كثيفة تستند إلى هذه المعمارية، وتشمل مشاركة الملفات (مثل Bit Torrent)، ومسرعات تحميل الملفات (مثل Xunlei)، والمهاتفة عبر الإنترنت (مثل Skype)، والتلفاز عبر بروتوكول الإنترنت (مثل Kankan PPstream). ويوضح الشكل (2-2) معمارية النظير للنظير

ومن أهم سمات معمارية النظير للنظير **قابلية التوسع الذاتي**، فعلى الرغم من أن كل النظراء يضيفون عبئا إضافية عند طلب الملفات في تطبيق مشاركة الملفات، فإن كل نظير يضيف طاقة جديدة للنظام لدى توزيع الملفات على نظرائهم، كما أن هذه المعمارية فعالة من حيث الكلفة، كونها عادة لا تتطلب بنية تحتية وعرض نطاق ترددي كبير للخادم (على عكس تصميم العميل الخادم ومراكز البيانات المستندة إليه)، ومع ذلك، فإن تطبيقات النظير للنظير ستواجه ثلاثة تحديات رئيسة في المستقبل :

1. الألفة مع مزود خدمة الإنترنت (ISP(: Friendly: معظم مقدمي خدمات الإنترنت يستخدمون عرض النطاق الترددي لغير المتماثل"، والذي يرتكز على أن تلقي البيانات)المصب .2 Downstream) أكبر بكثير من رفعها (المنبع: Upstream)، ولكن تدفق الفيديو وتوزيع الملفات في تطبيقات النظير ل لنظير ترحل حركة المرور على المنبع Traffic( )Upstream من الخوادم إلى مزودي خدمات الإنترنت المنزلي، مما يضيف ضغطا كبيرا على مزودي خدمات الإنترنت، لذا، فإن تصميم تطبيقات النظير للنظير المستقبلية يتطلب التركيز على الألفة مع مزودي خدمة الإنترنت [Xie 2008].
2. . لأمن (Security): بسبب طبيعتها المفتوحة والموزعة إلى حد كبير، فإن تطبيقات النظير للنظير تواجه تحديا كبيرا لتأمينها

{Dhungel 2008; LeBlond2011 Doucer 2002; Yu 20 06; Liang 2006 ;Naoumov 2006}

3. الحوافز(Incentive): يعتمد نجاح تطبيقات النظير ل لنظير في المستقل إلى إقناع المستخدمين بالتطوع بعرض النطاق الترددي، وذاكرة التخزين، وموارد الحوسبة لهذه التطبيقات، ما يتطلب تصميم مزيد من الحوافز

[;Feldman 2005 ; Piatek2008 Aperjis 2008 Liu2010]

أما في معمارية العميل الخادم، فهناك مضيف يدعى الخادم، يستجيب لطلبات الخدمة الواردة من المضيفين الآخرين، أي العملاء، مثل تطبيق الويب الذي يطلب خدمات من خلال المتصفح الذي يعمل على مضيف العميل - من خادم (ملقم) الويب، فعندما يتلقى خادم الويب طلبة من العميل، فإنه يستجيب بإرسال الكائن المطلوب إلى مضيف العميل، لاحظ ان العملاء لا يتواصلون مع بعضهم مباشرة؛ ففي تطبيق الويب، لا يتصل متصفحان معا بشكل مباشر، وهناك سمة أخري المعمارية العميل الخادم، فالخادم لديه عنوان ثابت ومعروف، يدعى عوان الإنترنت (IP Protocol،Internet) : لذا، يمكن ل لعميل دائما الاتصال بالخادم عن طريق إرسال حزمة الى عنوانه. ومن التطبيقات المعروفة في معمارية العميل الخادم .e-Mail، Telnet ، Web،FTP

في كثير من الأحيان لا يستطيع خادم واحد تلبية جميع طلبات العملاء، فعلى سبيل المثال، قد تنهار مواقع الشبكات الاجتماعية الشائعة إذا كان لديها خادم واحد فقط يتعامل مع جميع طلباتها، لذا، فإن مركز البيانات، الذي يستضيف عددا كبيرا من الأجهزة، يعمد إلى خلق خادم افتراضي قوي، كما أن المواقع الشائعة التي تقدم خدمات الإنترنت مثل محركات البحث Bing, Google ومواقع التجارة الإلكترونية e-Bay , Amazon ومواقع البريد الإلكترونيYaho Mail Gmail والشبكات الاجتماعية Twitter Facebook توظف مركز بيانات أو أكثر، فعلى سبيل المثال، لدى شركة غوغل 30 إلى 50 مركز بيانات موزعة في جميع أنحاء العالم، تتعاون جميعا لخدمة محرك البحث، واليوتيوب، والبريد الإلكتروني Gmail ، ويحتوي كل مركز بيانات مئات الآلاف من الخوادم ومن الجدير بالذكر، أن بعض التطبيقات ذات معمارية هجينة أي تجمع بين ممارستي النظير ل لنظير والعميل. الخادم ، فالعديد من تطبيقات المراسلة الفورية تستخدم الخوادم لتتبع عناوين الإنترنت IP للمستخدمين ولكن يتم تبادل رسائل المستخدمين مباشرة بين النظراء (دون المرور بخوادم وسيطة.)

2-2-2 الاتصال بين العمليات (Processes Communicating)

قبل إنشاء تطبيق الشبكة، نحتاج إلى فهم كيفية تواصل البرامج التي تعمل في أنظمة نهائية (end systems) متعددة بعضها ببعض. في الواقع، واستنادا إلى أنظمة التشغيل، لا تصل البرامج بحد ذاتها بل العمليات، ويمكن التفكير بأي عملية كبرنامج يعمل في نظام نهائي، وعند تشغيل العمليات على نفس النظام النهائي، تتواصل مع بعضها البعض من خلال الاتصال الداخلي بين العمليات باستخدام قواعد يحكمها نظام التشغيل، ولكننا ستركز على الاتصال بين العمليات التي تعمل على اجهزة (مضنين) مختلفة ضمن أنظمة تشغيل مختلفة وتواصل العمليات التي تنفذ على نظامين مختلفين بتبادل الرسائل عبر شبكة الحاسوب، حيث تخلق العملية المرسل رسائل وترسلها عبر الشبكة؛ تتلقى العملية المستقبل هذه الرسائل وتستجيب بالرد عليها وكما احنا في الشكل (2-1)، فإن العمليات التي تتصل بعضها ببعض تقيم في طبقة التطبيقات ضمن مكنس البروتوكول ذي الطبقات الخمس.

عمليات العميل والخادم

يتكون تطبيق الشبكة من أزواج من العمليات التي تتبادل الرسائل عبر شبكة، فمثلا، في تطبيق الويب تتبادل علبة متصفح العميل الرسائل مع عملية خادم الويب، أما في نظام مشاركة الملفات P2P ، ينتقل الملف من عملية في أحد النظراء إلى عملية في نظير آخر. ولكل زوج من العمليات المتصلة، نطلق على إحداهما العميل والأخرى الخادم، ففي تطبيق الويب، يكون المستعرض هو عملية العميل وخادم الويب هو عملية الخادم، وفي مشاركة الملفات P2P ، يكون النظير الذي يحمل Download الملف يوصف بأنه العميل، والنظير الذي يرقع (Upload) الملف يوصف بأنه الخادم عزيزي الطالب، قد تكون لاحظت انه في بعض التطبيقات، مثل مشاركة الملفات P2P ، قد تكون العملية عميلا وخادما على حد سواء. في الواقع يمكن للعملية في نظام مشاركة الملفات P2P تحميل أو رفع الملفات، ذلك، مع وفي سياق جلسة الاتصال بين زوج من العمليات، يمكنك أن تصف إحداهما كعميل والأخرى كخادم

تعريف

عمليات العميل والخادم: في سياق جلسة اتصال بين زوج من العمليات، **العميل** هو العملية التي تبدأ الاتصال (أي تبدا الاتصال بالعملية الأخرى في بداية الجلسة)أما **الخادم** فهو العملية التي تنتظر الاتصال بها لبدء الجلسة

الواجهة بين العملية وشبكة الحاسوب The Interface between the Process and the Computer network

تتكون معظم التطبيقات من أزواج من العمليات المتصلة، في كل زوج تتبادل العمليتان الرسائل، وأي رسالة يجب أن تمر عبر الشبكة الأساسية، ويتم تبادل الرسائل عبر الشبكة من خلال واجهة برنامج يدعى المأخذ أو المقبس (Socket)، ولفهم العمليات و المقابس، إذا اعتبرنا العملية هي المنزل فإن المقبس يمثل الباب، فعندما تريد عملية ارسال رسالة إلى عملية أخرى على مضيف آخر، فإنها توجه الرسالة من بابها (أي مقبسها)، وهنا تفترض العملية)المرسل(أن هناك بنية تحتية)وسيلة نقل(على الجانب الآخر من بابها لنقل الرسالة إلى باب العملية (الهدف)، وبمجرد وصولها إلى المضيف الهدف، يمرر الرسالة من باب العملية الهدف (أي المقبس) لتقوم بدورها بالرد على الرسالة وإجراء المطلوب يوضح الشكل (3-2) اتصال المقابس بين عمليتين عبر الإنترنت، ويفترض الشكل أن النقل الأساسي بين العمليات يتم من خلال بروتوكول التحكم بالنقل TCP.

المقبس هو الواجهة بين طبقة التطبيقات وطبقة النقل داخل المضيف، ويشار إليه أيضا بأنه واجهة برنامج التطبيق API) بين التطبيق والشبكة، لأنه يمثل واجهة البرمجة التي تبنى عليها تطبيقات الشبكة. فعطور التطبيق لديه سيطرة كاملة على كل شيء من المقبس من جهة طبقة التطبيقات، ولكن لديه سيطرة طفيفة من جهة طبقة النقل، غير يسيطر فقط على اختيار بروتوكول النقل، وضبط قليل من معاملات طبقة النقل مثل الحد الأقصى لحجم المخزن المؤقت (buffer) وحجم القطاع (segment). وعند اختيار بروتوكول النقل)إذا كان متاحة)، يقوم المطور ببناء التطبيق باستخدام الخدمات التي يوفرها البروتوكول. لمزيد من التفصيل، ستتطرق إلى المقابس في القسم 2-8 .

عمليات العنونة (Addressing Processes)

كما في البريد العادي، إذا أرادت عملية قيد التشغيل على مضيف معين إرسال الحزم إلى عملية قيد التشغيل على مضيف آخر، يجب أن يكون للعملية)المستقبل(عنوان محدد، ولتحديد العملية)المستقبل(، يلزم تحديد تون من المعلومات: عنوان المضيف، ومعرف يحدد العملية المستقبل في المضيف الهدف. في عالم الإنترنت، يتم تعريف المضيف بعنوان الإنترنت IP الخاص به، كما ستناقش في الوحدة الرابعة، وكل ما نحتاج إلى معرفته الآن هو أن عنوان الإنترنت (IP Address) هو معرف فريد للمضيف طوله 32 بت في الإصدار الرابع (IPv4)، بعد التعرف على عنوان المضيف الذي تم توجيه رسالة إليه، يجب على العملية المرسل) أن تحدد العملية)المستقبل(، وخاصة المقبس المستقبل قيد التشغيل على المضيف، لأنه قد يتم تشغيل أكثر من تطبيق على مضيف واحد، ورقم المنفذ (Port number) هو الذي يعالج هذه المسالة، حيث يتم تعيين رقم منفذ لكل تطبيق. فمثلا، يعرف خادم الويب بالمنفذ رقم 80 ، والبريد الإلكتروني)باستخدام بروتوكول SMTP) يعرف بالمنفذ رقم 25 ، وهذا ما سيتم عرضه بالتفصيل في الوحدة الثالثة. يمكنك العثور على قائمة أرقام المنافذ المعروفة لجميع بروتوكولات الإنترنت القياسية على الرابط http://www.iana.org.

2-2-3خدمات النقل المتوفرة للتطبيقات Transport Services Available to Application

عزيزي الطالب، تذكر أن المقيس هو الواجهة بين عملية التطبيق وبروتوكول طبقة النقل، فالتطبيق من جهة المرسل يرسل الرسائل من خلال المقبس، ومن الجهة الأخرى، بروتوكول طبقة النقل هو المسؤول عن توصيل الرسائل إلى مقيم العملية الهدف توفر العديد من الشبكات، بما فيها شبكة الإنترنت، أكثر من بروتوكول طبقة نقل واحد، وعند تطوير تطبيق، عليك اختيار أحد البروتوكولات المتاحة، وذلك بدراسة الخدمات التي تقدمها بروتوكولات طبقة النقل المتاحة ومن ثم اختيار البروتوكول الذي يقدم الخدمات التي تنطبق مع احتياجات تطبيقك، ويمكن تصنيف الخدمات المملكة التي يقدمها بروتوكول طبقة النقل في أربعة محاور: النقل الموثوق للبيانات، والإنتاجية، والتوقيت، والأمن.

النقل الموثوق للبيانات (Reliable Data Transfer)

قد تضيع الحزم في شبكة الحاسوب عندما تتجاوز حجم المخزن المؤقت للموجه (Router)، أو قد يهملها المضيف أو الموجه إذا كانت بعض البنات تالفة، وفي العديد من التطبيقات مثل البريد الإلكتروني، ونقل الملفات، والوصول إلى المضيف عن بعد، ونقل وثائق الويب، والتطبيقات المالية، قد يكون لفقدان البيانات تبعات خطرة، ولدعم هذه التطبيقات، لا بد من التدخل لضمان أن البيانات المرسلة يتم تسليمها في الطرف الآخر من التطبيق صحيحة وكاملة، فإذا وفر البروتوكول مثل هذه الخدمة المضمونة لتسليم البيانات، يقال أنه يوفر **نقلا موثوقا للبيانات**، ومن الخدمات المهمة التي قد يقدمها بروتوكول طبقة النقل للتطبيق هو النقل الموثوق البيانات سن عملية إلى عملية، وفي هذه الحالة، ما على العملية)المرسل(إلا أن تمرر بياناتها إلى المقيس، ولديها الثقة الكاملة بان البيانات ستصل العملية)المستقبل(دون أخطاء، وإلا فإن البيانات التي ترسلها عملية قد لا تصل أبدا إلى العملية الهدف. قد يكون هذا مقبولا في التطبيقات التي تسمح بذلك، كمعظم الوسائط المتعددة التي قدح بفقدان بعض البيانات، إذ تؤدي البيانات المفقودة إلى خلل بسيط في الصوت أو الفيديو لا يشكل فرق واضح في الجودة.

الإنتاجية (Throughput)

عزيزي الطالب، مفهوم الإنتاجية المتاحة، في سياق جلسة اتصال بين عمليتين على مسار شبكة، هو معدل البتات التي تستطيع العملية (المرسل) تسليمها للعملية (المستقبل)، وكون الجلسات الأخرى ستتقاسم عرض النطاق الترددي على مسار الشبكة، وكونها تأتي وتذهب، فان الإنتاجية المتاحة قد تقلب مع مرور الوقت، وتقولنا هذه الملاحظات إلى خدمة طبيعية أخرى قد يوفرها بروتوكول طبقة النقل، وهي ضمان الإنتاجية المتاحة بمعدل محتد، عدها يمكن للتطبيق طلب معدل إنتاجية مضمون (٣) بت/ ثانية، وبروتوكول النقل هو الذي يتأكد أن الإنتاجية المتاحة دائما (r بت/ ثانية) على الأقل. هذه الإنتاجية المضمونة ستجنب العديد من التطبيقات، فعلى سبيل المثال، إذا كان ترميز الصوت بمعدل 32 كيلوبت في الثانية في أحد تطبيقات المهاتفة نشر الانترنت، فانه يرسل البيانات إلى الشبكة ليتم تسليمها إلى المتلقي بالمعدل ذاته، وإذا لم يتمكن البروتوكول من توفير هذه الإنتاجية، فإن التطبيق يحتاج إلى تقليل معدل الترميز (وتلقي ما يكفي من الإنتاجية للحفاظ على هذا المعدل(أو قد يضطر إلى التوقف لأن استقبال نصف الإنتاجية اللازمة، مثلا، قلما يستخدمه تطبق المهاتفة

عزيزي الطالب، يطلق على التطبيقات التي لديها متطلبات الانتاجية تطبيقات حساسة لعرض النطاق الترددي (Bandwidth-sensitive applications) على الرغم أن بعضها قد يستخدم تقنيات ترميز تكيفية الصوت الرقمي أو الفيديو بمعدل يتناسب مع الإنتاجية المتاحة، أما التطبيقات المرنة (Elastic Applications) فقد تستفد قدرا قليلا أو كبيرة من الإنتاجية جسيما هو متاح، وكلما زادت الإنتاجية افضل، مثل البريد الإلكتروني، ونقل الملفات، ونقل الويب

التوقيت (Timing)

قد يوفر بروتوكول طبقة النقل ضمانات التوقيت بأشكال عدة، كان يضمن وصول كل بت يرسله المرسل عبر المقبس إلى مقبس المستقبل بما لا يزيد عن 100 ميللي ثانية. هذه الخاصية مهمة في تطبيقات الزمن الحقيقي التفاعلية، مثل المهاتفة عبر الإنترنت، والبيئات الافتراضية، والمؤتمرات عن بعد، والألعاب متعددة اللاعبين وغيرها التي تتطلب الدقة في توقيت تسليم البيانات، فالتأخير في المهاتفة عبر الإنترنت مثلا قد يؤدي إلى انقطاع غير طبيعي في المحادثة، فيبدو التطبيق أقل فاعلية ورغم عدم وضع قيود صارمة على التأخير في غير تطبيقات الزمن الحقيقي، إلا أن زمن التأخير كلما قل كان أفضل.

الأمن (Security)

يوفر بروتوكول النقل التطبيق واحدة أو أكثر من خدمات الأمن، ففي المضيف المرسل مثلا، يمكن لبروتوكول النقل تشفير البيانات التي ترسلها العملية)المرسل(، وفك التشفير لدى المضيف المستقبل قبل تسليم البيانات ال لعملية)المستقبل(، ومن شأن هذه الخدمة أن توفر السرية (Confidentiality) بين العمليتين حتى وإن كان يمكن ملاحظة البيانات بطريقة أو بأخرى بين العمليات المرسلة والمستقبلة، وقد يوفر بروتوكول النقل خدمات آخرى كسلامة البيانات (Integrity) والمصادقة (Authentication)

خدمات النقل التي تقدمها الانترنت Transport Services Provided by the internet

عزيزي الطالب، حتى هذه اللحظة، ما زلنا نفكر في خدمات النقل التي توفرها شبكة الحاسوب بشكل عام، دعنا الأن نكون أكثر تحديدا ونختبر نوع خدمات النقل التي توفرها شبكة الإنترنت. توفر الإنترنت)أو شبكات TCP/IP بشكل عام) للتطبيقات بروتوكولي نقل هما بروتوكول مخطط بيانات المستخدم (Protocol: UDP User Datagram) وبروتوكول التحكم بالنقل (Transport Control Protocol: TCP) وعندما تقوم (كمطور تطبيق) بإنشاء تطبيق شبكة جديد على الإنترنت، عليك أولا اتخاذ القرار باستخدام بروتوكول UDP | أو TCP ، فكل منهما يقدم مجموعة مختلفة من الخامات التطبيقات التي تطلبها، ويبين الشكل (42) متطلبات الخدمة لمجموعة من التطبيقات المختارة

خدمات بروتوكول التحكم بالنقل TCP Services

عندما يستدعي أحد التطبيقات بروتوكول التحكم بالنقل (TCP)، فإنه يتلقى نوعين من الخدمة حسب نموذج خدمة TCP كما يأتي :

- الخدمة الموجه بالاتصال (Connection-oriented Services) العميل والخادم معلومات التحكم الخاصة بطبقة النقل قبل أن تبدأ رسائل التطبيق بالتدفق، وهذا ما يسمى إجراء المصافحة (handshaking) الذي ينبه العميل والخادم ويسمح لهما بتحضير دفعة من الحزم. بعد مرحلة المصافحة، يدخل اتصال TCP حيز الوجود بين مقيسي العمليتين، وهو اتصال باتجاهين (full –duplex) يسمح ل لعمليتين بإرسال الرسائل إلى بعضها البعض في نفس الوقت، وعند انتهاء التطبيق من إرسال الرسائل ينتهي الاتصال. في الوحدة الثالثة ستناقش بالتفصيل الخدمة الموجهة بالاتصال وكيفية تنفيذها.

- خدمة النقل الموثوق للبيانات (Reliable Data Transfer Services) المتصلة الاعتماد على بروتوكول TCP التسليم جميع البيانات المرسلة خالية من الخطأ وبالترتيب السليم، فعندما يمرر أحد جانبي التطبيق

دفعة من وحدات البايت في مقيس المرسل، فإنه يستطيع الاعتماد على TCP لتسليم الدفعة ذاتها إلى مقبس المستقبل دون فقدان أو تكرار.

التركيز على الامنFocus on Security E just

تأمين بروتوكول التحكم بالنقل TCP

لا يوفر بروتوكول TCP أو UDP أي تشفير، أي أن البيانات التي تمررها عملية ما من مقبسها هي نفس البيانات التي تنتقل عبر الشبكة إلى وجهتها، لذلك، على سبيل المثال، فإذا ارسلت كلمة مرور في نص واضح (اي، غير مشفرة) في مقبسها، فإنها ستنتقل عبر جميع الوصلات بين المرسل والمستقبل، وقد تكتشف في أي من هذه الوصلات، ولأن الخصوصية وقضايا الأمن الأخرى أصبحت حاسمة في كثير من التطبيقات، فقد أجرى مجتمع الإنترنت تحسينات

لتعزيز بروتوكولTCP تدعى طبقة المقابس الامنة (secure socket layer : ssl) فلم يعد بروتوكول TCP المعزز ب ssl يعمل بشكل تقليدي بل اصبح يقدم خدمات امنية حاسمة بين العمليات بما في ذلك التشفير وسلامة البيانات والمصادقة

ونؤكد هنا ان SSL ليس بروتوكول نقل ثالث بنفس مستوى TCP وUDP، بل لهو تعزيز ينقد و تحسينات في طقة عيد، وخاصة إذا كل لتطبيق يريد استخدام ختمات SSL ولتطيق ذلك يتم ادراج الكود بها في التطبيق من جانب العميل والحكم على حد سواء (يتوفر تكون على شكل مت برمجة مفتوحه) لدى SSL مس ام خاص يشبه تلك الخاص ببروتوكول TCP التقليدي فعندما يستخدم تطبيق طبقة SSL فان المرسل يمرر النص الواضح للبيانات الى مقبس SSL التي يقوم بتشفير البيانات في المضيف المرسل ، ثم يمرر البيانات المشفرة الى مقبس TCP تنتقل البيانات المشفرة عبر الانترنت الى مقبس TCP للمستقبل ويمررها الى SSL التي يقوم بدوره بفك تشفير البيانات وفي نهاية يمرر SSL النص الواضح للبيانات عبر مقبس من SSL الى العملية المستقبل

يوفر بروتوكول TCP أيضا آلية لضبط الازدحام (Congestion-control)، وهي خمة علمة نص الإنترنت وليست لمنفعة العمليات المتصلة بشكل مباشر، وتعمل آلية ضبط الازدحام كصمام يقص ق البيانات من عملية المزمل (العميل أو الخادم) ما تكون الشبكة مزدحمة بين المرسل والمستقبل وكر موضح في الوحدة الثالثة، قان ضبط الازدحام من خلال TCP يحاول أيضا أن يحدد لكل اتصال TCP العادلة من عرض النطاق الترددي للشبكة .

خدمات بروتوكول مخطط بيانات المستخدم UDP Services

بروتوكول UDP هو بروتوكول نقل خيف الوزن، ويوفر الحد الأدنى من الخت ، وهو بروتوكول تون اتصال، لنا، ليس هناك مصافحة قبل أن تبدأ العمليتان بالاتصال، كما يوقر خدمة نقل غير موثوق ل ات، أي عما ترسل عملية رسالة إلى متبس UDP ء قان تضمن أن الرسلة ستصل إلى العملية المتقبل، وانن وصلت قد تكون بغير ترتيبها الأصلي

من جانب آخر، لا يتضمن بروتوكول UDP آلية لضبط الازدحام، وبالتالي فان المرسل من خلال UDP يمكنه ضخ البيانات نحو الطبقة التي تدنو)طبقة الشبكة باي معل يحلو له، ومع ذلك، فإن الإنتاج الفعلي قد يكون أنت من هذا المعدل نظرا لمحدودية طاقة الإرسال أو نتيجة الازدحام

الخدمات التي لا تقدمها بروتوكولات النقل عبر الانترنت Transport Services Not Provided by Internets Protocols

لابد أنك لاحظت، عزيزي الطالب، أنا لم تشر في نقاشنا بشكل واضح إلى ضمان الإنتاجية أو التوقيت، قيل هذا يعني أننا لن تتمكن من تشغيل التطبيقات الحساسة ل لزمن، مثل المهاتفة عبر الإنترنت بالطبع لا، بما في تعمل منذ سنوات عدة، ولكن هذه التط ، غب، تعمل بشكل جيد لأنها مصممة لتعامل ، إلى أقصى حد ممكن، مع هذا النقص في الضمان. وباختصار، فإن الإنترنت اليوم قد تقدم خدمة مرضية للتطبيقات الحساسة للزمن في كثير من الأحيان، ولكن لا يمكنها أن توفر أي ضمانات على التوقيت أو الإنتاجية يشير الشكل (2-5) إلى بروتوكولات النقل المخية في بعض تطبيقات الإنترنت الشدة، حيث تجد أن البري الإلكتروني، والوصول للحواسيب عن بعد، وشبكة الإنترنت، ونقل الملفات جميعها TCP ، و أن TCP ، في المقام الأول، يقدم نقلا موثوقة للبيانات، كما يضمن وصول جميع الان في نهاية المطاف إلى وجهتها. ولأن تطبيقات الاتصال الهاتفي عبر الإنترنت (مثل سكايب) قد تحتمل في كثير من الأحيان بعض الخسائر، ولكنها تتطلب حد أدني لمعدل النقل لتكون فعالة، يفضل المطورون عادة تشغيل تطبيقاتهم الهاتفية على UDP للالتفاف على آلية ضبط الازدحام وبتات التحكم الإضافية في TCP ، وكون العديد من الجدران النارية تمنع معظم أنواع حركة المرور عبر UDP ، لذا فإن تطبيقات الاتصال الهاتفي عبر الإنترنت غالبا تصمم لاستخدام TCP احتياطا إذا فشل الاتصال عبر UDP.

2-2-5 بروتوكولات طبقة التطبيقات (Application-Layer Protocols)

لقد علمت للتو، عزيزي الطالب، أن عمليات الشبكة تتواصل مع بعضها البعض بإرسال الرسائل عبر المقابس ولكن كيف يتم تنظيم هذه الرسائل؟ وما معاني الحقول المختلفة في الرسائل؟ ومتى ترسل العمليات هذه الرسائل؟ كل هذه الأسئلة تعيدنا إلى بروتوكول طبقة التطبيقات، والذي يوجد د كيف ت تمكن عمليات أحد التطبيقات، الذي يعمل على أنظمة مختلفة، من تمرير الرسائل لبعضها البعض. بالتحديد، يحدد بروتوكول طبقة التطبيقات ما يأتي:

1 أنواع الرسائل المتبادلة، مثل رسائل الطلب ورسائل الاستجابة.

2 صياغة الرسائل بأنواعها المختلفة، مثل حقول الرسالة وكيفية وصفها أو رسمها.

3 دلالات الحقول، أي معاني المعلومات الواردة في الحقول.

4 قواعد لتحديد متى وكيف يمكن لعملية أن ترسل الرسائل وتستجيب لها.

عزيزي الطالب، من المهم أن نميز بين تطبيقات الشبكة وبروتوكولات طبقة التطبيقات، يمثل بروتوكول طبقة التطبيقات جزء واحدة من تطبيق الشبكة (مع ذلك، فهو جزء مهم جدا من التطبيق)

2-2-6تطبيقات الشبكة التي يغطيها المقرر(Covered Network Applications)

في هذه الوحدة نناقش خمسة تطبيقات مهمة شبكة الإنترنت (Web)، ونقل الملفات (FTP)، والبريد الإلكتروني زات Ma)، وخدمة الدليل (DNS)، وتطبيقات النظير للنظير (P2P) علينا أولا أن نيتك في الويب، لا لأنه تطبيق شائع فحسب، بل لأن بروتوكول طبقة التطبيقات، HTTP واضح و مباشر وسهل الفهم. ثم تدرس FTP بإيجاز، كونه لا يختلف كثيرا عن HTTP. ثم ننتقل لناقش البريد الإلكتروني، أول تطبيق إنترنت مهم، وهو أكثر تعقيدا من الويب، بمعنى أنه يوظف عدة بروتوكولات في طبقة التطبيقات. ثم تغطى بعد ذلك DNS ، الذي يوفر خدمة دليل الإنترنت، إذ يلاحظ أن معظم المستخدمين يتفاعلون بشكل غير مباشر مع DNS من خلال التطبيقات الأخرى (مثل شبكة الإنترنت، ونقل الملفات، والبريد الإلكتروني(، فهر يوضح كيف يمكن لوظيفة جوهرية بسيطة)ترجمة اسم الشبكة إلى عنوان الشبكة(أن تنفذ في طبقة التطبيقات في الإنترنت. وأخيرا، تناقش في هذه الوحدة تطبيقات P2P ، مع التركيز على تطبيقات مشاركة الملفات، وخدمات البحث الموزعة.

2-3 | الشبكة العنكبوتية وبروتوكول نقل النص التشعبي (The Web and HTTP)

حتى أوائل التسعينات تم استخدام الإنترنت في المقام الأول من قبل الباحثين، والأكاديمي ين، وطلاب الجامعات التسجيل الدخول إلى الحاسوب)المضيف(عن بعد، ونقل الملفات من الحاسوب المحلي إلى البعيد، والعكس بالعكس، كما استخدم في تلقي الأخبار ورسائل البريد الإلكتروني وإرسالها، وعلى الرغم من أن هذه التطبيقات كانت وما زالت مفيدة للغاية، لم يكن الإنترنت معروفا خارج الأوساط الأكاديمية والبحثية وفي بداية التسعينات ظهر تطبيق جديد على المشهد وهو الشبكة العنكبوتية العالمية

Web :WWW.World Wide)]94 19 Berners-Le وكان الويب من تطبيقات الإنترنت الأولى الملفتة ل لنظر، غير أنها تغيرت بشكل كبير ومستمر فيما يخص تفاعل الناس داخل بيئات عملهم وخارجها، ثم ارتقت الإنترنت من مجرد واحدة من العديد من شبكات البيانات إلى شبكة البيانات الأساسية الوحيدة. ولعل أكثر ما يروق للمستخدمين أن الشبكة تعمل عند الطلب على عكس وسائل الإعلام التقليدية كالإذاعة والتلفاز، بالإضافة إلى العديد من الميزات الرائعة التي أحبها الناس، فمن السهل جدا لأي فرد توفير المعلومات عبر الإنترنت ونشرها بكلفة قليلة للغاية، وتساعدنا محركات البحث على التنقل في عدد لا حصر له من مواقع الإنترنت، والرسومات، والاستمارات، وسندات الجافا (JavaScript)، وتطبيقات الجافا، والعديد من الأدوات التي تمكننا من التفاعل مع الصفحات والمواقع، كما وفر الويب منصة للعديد من التطبيقات الناشئة بعد عام 2003 ، بما في ذلك يوتيوب YouTube ، وبريد جوجل Gmail ، و الفيسبوك Facebook

2-3-1 لمحة حول بروتوكول نقل النص التشعبي

بروتوكول نقل النص التشعبي (Hyper Text Transfer Protocol)

هو بروتوكول طبقة تطبيقات الويب، في قلب الشبكة العنكبوتية، وتم تعريفه في [RFC 1945] و [RFC 2616]، ويطبق على الأنظمة المختلفة في برنامجين: برنامج العميل وبرنامج الخادم اللذان يتحادثان مع بعضهما البعض من خلال تبادل رسائل HTTP ، حيث يحدد HTTP بنية هذه الرسائل وكيفية تبادلها، وقبل الدخول في التفاصيل، لا بد لنا أن نستعرض بعض المصطلحات المهمة صفحة الويب تتكون من مجموعة من الكينونات، والكائن هو مجرد ملف مثل ملف HTML ، أو صورة JPEG ، أو برنامج جافا، أو مقطع فيديو، قابلة للعنونة بعنوان أو رابط واحد يطلق عليه موقع المعلومات العالمي (Universal Resource Locator: URL)

اساس معظم صفحات الويب تتكون من ملف HTTL اساس يشير الى كائنات متعددة، فمثلا، إذا كانت صفحة الويب تحتوي على نص HTML وخمسة صور JPEG ، فإنها تتكون من ستة كائنات : ملف HTML أساس وخمس صور، ويشير ملف

HTML إلى مراجع الكائنات الأخرى في الصفحة مع عناوينها URLs ، ويتكون كل عنوان URL من عنصرين: اسم المضيف الخادم الذي يضم الكائن، واسم مسار الكائن.

مثال (2-3)

حدد مكونات عنوان URL الاتي

http://www.someSchool.edu/someDepartment/picture.gif

الإجابة :اسم مضيف الخادم هو: www.someSchool.edu some

اسم المسار هو :Some Department/picture.gif/

ولان متصفحات الويب (مثل إنترنت إكسبلورر وفايرفوكس) تنفذ ما يخص العميل من HTTP في سياق الويب، سنستخدم الكلمات المتصفح والعميل بشكل متناوب. أما خوادم الويب، التي تنفذ ما يخص الخادم من HTTP ، فتستضيف الكائنات كل بعنوانه الخاص، وأكثر خوادم الويب شعبية أباتشي (Apache وخادم .(Internet Information Server: IIS) ayI files يحتل بروتوكول HTTP كيف يطلب العملاء صفحات الويب من الخوادم، وكيف تنقل الخوادم صفحات الويب العملاء. ويبين الشكل (2-6) الفكرة العامة، فعندما يقوم المستخدم بطلب صفحة ويب (بالنقر على ارتباط شعبي) يرسل المتصفح إلى الخادم رسائل طلب HTTP لكائنات محددة من الصفحة، فيتلقى الخادم الطلبات ويستجيب لها برسائل استجابة HTTP تحتوي الكائنات المطلوبة يستم بروتوكول HTTP بروتوكول النقل الأساس TCP ، فيبادر عميل HTTP بداية بإنشاء اتصال TCP مع الخام، وبمجرد إنشاء الاتصال، تصل عمليات المتصفح والخادم إلى TCP من خلال واجهات مقابسها ، وكما أوضحنا في القسم 2-2 ، فعلى جانب العميل تمثل واجهة المقبس الباب بين عملية العميل والاتصال بروتوكول TCP ، وعلى جانب الخادم تمثل الباب بين عملية الخادم والاتصال ببروتوكول TCP ترسل العميل رسائل طلب HTTP إلى واجهة مقبسه، ويتلقى رسائل استجابة HTTP منها، وبالمثل، يتلقى الخادم رسائل طلب HTTP من واجهة مقبسه ويرسل رسائل الاستجابة إليها، فعندما يرسل العميل رسالة إلى واجهة مقبسه ، تخرج الرسالة من العميل إلى بروتوكول TCP الذي يوفر البروتوكول HTTP خدمة نقل موثوق البيانات، وهذا يعني أن كل رسالة طلب HTTP أرسلتها عملية العميل ستصل سليمة إلى الخادم، وبالمثل، فإن كل رسالة استجابة HTTP أرسلتها عملية الخادم ستصل في نهاية المطاف سليمة إلى العميل، وهنا نلاحظ واحدة من أهم مزايا معمارية الطبقات؛ إذ لا داع للقلق لدى HTTP من فقدان البيانات أو أي تفاصيل حول كيفية استردادها أو إعادة ترتيبها داخل الشبكة، فهذه وظيفة بروتوكول TCP وبروتوكولات الطبقات الدنيا من مكدس البروتوكول

لاحظ عزيزي الطالب ، أن الخادم يرسل الملفات المطلوبة للعملاء دون تخزين أي معلومات عن حالة العميلة فإذا طلب عميل معين نص الكائن مرتين في بضع ثوان، يعيد إرسال الكائن، وكانه نسي ما فعله سابقا، وكونه لا يحتفظ بمعلومات حول حالة العملاء، يقال أن بروتوكول HTTP عديم الحالة لاحظ أيضا أن شبكة الإنترنت تستخدم معمارية العميل الخادم كما هو موضح في القسم 22 ، فخادم الويب يعمل دائما، وله عنوان انترنت IP ، كما يقدم الخدمة لجميع الطلبات من ملايين المتصفحات المختلفة.

الاتصالات الدائمة والغير الدائمة (non persistent and persistent connection)

في العديد من تطبيقات الإنترنت، يتواصل العميل والخادم لفترة من الزمن، يطلب العميل سلسلة من الطلبات او متقطعة. وعندما يأخذ تفاعل العميل الخادم مجراه عبر TCP ، يحتاج مطور التطبيق إلى اتخاذ القرار: هل ينبغي لكل زوج)طلب استجابة(أن يرسل عبر اتصال TCP مستقل، أم أن ترسل جميع الطلبات والاستجابات المتعلقة بها عبر نفس الاتصال؟ تسمى الطريقة الأولى أعلاه الاتصال غير الدائم أو Connection (Non-Persistent Connection) Persistent)، ولفهم هذه المسالة في التصميم، دعنا نختبر مزايا وعيوب الاتصال الدائم في تطبيق محدد البروتوكول HTTP ، حيث أنه يستخدم الطريقتين؛ الاتصال الدائم في الوضع الافتراضي، ويمكن إعداد العميل والخادم الاستخدام الاتصال غير الدائم

HTTP with Non-Persistent بروتوكول نقل النص التشعبي والاتصالات غير الدائمة

Connections

عزيزي الطالب، دعنا نتتبع خطوات نقل صفحة ويب من الخادم إلى العميل لحالة الاتصالات غير الدائمة فلنفترض أن الصفحة تكون من ملف HTML و10صور من نوع PEG أي 11 كائنا على نفس الخادم وافترض أن عنوان URL لملف HTML الأساس هو http://www.someSchool.edu/someDepartment/homeindex

فيما يلي أهم الخطوات من البداية إلى النهاية

1 تنشئ عملية العميل http اتصال tcp بالخادم www.someSchool.edu على المنفذ (port)رقم 80 وهو المنفذ الافتراضي لبروتكول http ويصحب اتصال tcpتفعيل مقبسي العميل والخادم .

2 يرسل عميل HTTP رسالة طلب إلى الخادم عبر مقبسها تتضمن اسم المسار على النحو الآتي some Department/home. Index

3 تتلقى عملية الخادم HTTP رسالة الطلب من مقبسها ، وتسترجع الكائن Some Department/home. Index من وسط التحزين)الذاكرة المؤقتة أو القرص(، وتغلف الكان في رسالة استجابة HTTP وترسلها إلى العميل عن طريق المقبس

٤ تبلغ عملية الخادم بروتوكول TCP ليقوم بإغلاق الاتصال، ولكن بروتوكول TCP لا ينهي الاتصال فعلا حتى يتأكد أن العميل تلقى رسالة استجابة سليمة

٥ يتلقى العميل رسالة الاستجابة، وينتهي اتصال TCP رشيد الرسالة إلى أن الكائن الملف هو ملف

HTML ، فيقوم العميل استخراج ملت HTML وفحصه ليجد مراجع لعشر كائنات JPEG ،

٦ يتم تكرار الخطوات (1-4) لكل كائن من كائنات JPEG المشار اليها

بمجرد حصول المتصفح على صفحة الويب، فإنه يعرضها للمستخدم، ولا علاقة لبروتوكول HTTP كيف يقر العميل صفحة الويب، توضح الخطوات أعلاه الاتصال غير الدائم، حي يغلق كل اتصال TCP بن إرسال كل كائن، ففي هذا المثال، عندما يطلب المستخدم صفحة الويب ، يتم إنشاء 1 1 اتصال TCP . في الواقع، يمكن للمستخدمين إعداد المتصفحات الحديثة التحكم بعد الاتصالات التي يمكن انتشارها على التوازي، في الوضع الافتراضية أكثر المتصفحات تلفن 5-10 اتصالات TCP ، كل اتصال له حركة طلبة استجابة، ويمكن للمستخدم تحديد العدد الأقصى بقيمة (1)، اي على التوالي، وكما سترى في الوحدة التالية فإن الاتصالات المتوازية تقلل من الاستجابة التقدير الوقت المستغرق من لحظة طلب العميل ملف HTML حتى استلامه، لا بد من قياس زمن الذهاب والاياب ((Round-Trip Time: RTT) الخادم ثم العودة إلى العميل، ويتضمن التأخير في انتشار الرزمة و تأخير الاصطفاف في الطابور، والتأخير في معالجة الرزم (حسبما ناقشنا في القسم 1-8) لنتأمل الأن ما يحدث عندما ينقر المستخدم على ارتباط تشحي كما هو مبين في الشكل (7-2)، يشي المتصفح اتصال TCP مع خادم الويب وتحدث مصافحة ثلاثية فيرسل العميل مقطع TCP صغير للخادم، فيقر الخادم بالاستلام ويستجيب بمقطع TCP آخر، وأخيرا، يقر العميل للخادم بالاستلام. اول ٠جزأين يحتاجان RTT واحدة، ثم يرسل العميل رسالة طلب HTTP مع الجزء الثالث من المصافحة الثلاثية)الاعتراف(في اتصال TCP ، وفور وصول رسالة الطلب إلى الخادم، يبدأ بإرسال ملف HTML عبر اتصال TCP ، وبذلك يكون مجموع زمن الاستجابة هو RTTs2 بالإضافة إلى زمن إرسال ملف HTML لدى الخادم

HTTP with Persistent Connections بروتوكول نقل النص التشعبي والاتصالات الدائمة

للاتصالات غير الدائمة بعض نقاط الضعف، أولا، يجب إنشاء اتصال جديد والاحتفاظ به لكل كائن، لكل منها يجب حجز مخازن والاحتفاظ بمتغيرات TCP لدي كل من العميل والخادم، مما يضيف عبئا كبيرا على خادم الويب الذي يخدم مئات العملاء في نفس الوقت. ثانيا، كما وصفنا التو، كل كائن يعاني من زمن تأخير بقيمة 2RRTTs

أما الاتصالات الدائمة، يبقي الخادم اتصال TCP مفتوحا بعد إرسال الاستجابة لتبادل الطلبات والاستجابات ال لاحقة مع نفس العميل عبر الاتصال نفسه، ف في المثال أعلاه، يمكن إرسالها ملف HTML و10 صور عبر اتصال TCP دائم واحد. علاوة على ذلك، يمكن إرسال صفحات ويب متعددة من نفس الخادم لنفس العميل عبر اتصال TCP دائم واحد

عزيزي الطالب، في العادة، يغلق خادم HTTP الاتصال إذا لم يستخدم لفترة معينة)timeout،interval(وفي الوضع الافتراضي يستخدم HTTP الاتصالات الدائمة مع خط الأنابيب (pipelining). سنقوم بمقارنة

الاتصالات غير الدائمة والدائمة في واجبات الوحدتين الثانية والثالثة ولمزيد من التفاصيل أشجعك على الاطلاع على المرجعين [Heidemann1997, Nielsen 1997]

2 3 - 3 - تنسيق رسائل النص التشعبي(HTTP Message Format)

تشمل مواصفات[HTTP [RFC 1945] ، و [RFC 2616] تعريفات لصيغة رسالةHTTP ، وهناك نوعان. من رسائل HTTP 1. رسائل الطلب 2. ورسائل الرد (الاستجابة)، سنناقشهما حالا

رسالة الطلب HTTP Request Message

أولا، نلاحظ أن الرسالة مكتوبة بنص ASCII العادي حيث يمكن لشخص غير خبير بالحاسوب قراءتها. ثانيا، تتألف الرسالة من خمسة أسطر، ومع ذلك يمكن أن يكون عدد الأسطر أكثر من ذلك أو أقل لغاية سطر واحد فقط، ويطلق على السطر الأول سطر الطلب(request line) ، وتسمى السطور التالية (سطور المقدمة .header lines) يحتوي سطر الطلب على ثلاثة حقول: حقل الطريقة، وحقل عنوانURL ، وحقل إصدار .HTTP قد يحمل حقل الطريقة عدة قيم مختلفة، بما في ذلك DELETE ، POST, GET , PUT, HEAD, ومعظم رسائل الطلب تستخدم طريقة GET التي تستخدم عندما يطلب المتصفح كائن معرف في حقل URL في هذا المثال، المتصفح يطلب الكائن somedir / page . html حسب الإصدار HTTP/1.1 ننتقل الى سطور المقدمة Host : Www . someschool. Edu المضيف الذي يتواجد فيه الكائن والذي تطلبه الذاكرة المخبأة لوكيل الويب (Proxy cache) ويعني السطر ويعني السطرConnection: close آن المتصفح يبلغ الخادم أن بإمكانه إغلاق الاتصال بعد إرسال الكائن المطلوب، إذ لم يعد بحاجة إليه. أما سطر المقدمة

User - agent :Mozilla/0.5 فيشير إلى وكيل المستخدم، أي نوع المتصفح الذي طلب الخدمة ،وهو في المثال متصفح موزيلا 5 . 0 من فايرفوكس، وهذا السطر مفيد لأن الخادم قد يرسل إصدارات مختلفة من نفس الكائن إلى أنواع مختلفة من الوكلاء. أخيرا، فإن السطر Accept : - language يشير أن المستخدم يفضل الحصول على النسخة الفرنسية من الكائن إن وجدت على الخادم، وإلا سيرسل الخادم النسخة الافتراضية عزيزي الطالب، دعنا ننظر الآن في الشكل العام لرسالة الطلب كما هو مبين في الشكل (8-2) ص 63 ، نلاحظ أنه يتوائم مع المثال السابق.

بالإضافة إلى نص الكيان(entity body)، ويكون فارغا مع طريقةGET ، ولكنه يستخدم مع .POST غالبا ما يستخدم عميل HTTP طريقة POST عندما يملا المستخدم نموذجة مثل إدخال كلمات مفتاحية لمحرك البحث، وهنا لا يزال المستخدم يطلب صفحة ويب من الخادم، ولكن محتواها يعتمد على ما أدخله المستخدم في حقول النموذج.

طريقة Head تشبه طريقةGET ، فعندما يتلقى الخادم طلبة بطريقةHead ، فإنه يستجيب برسالة HTTP دون الكائن المطلوب، وغالبا، يستخدم مطور التطبيق هذه الطريقة لفحص الأخطاء .)Debugging(ويقترن استخدام طريقة PUT بأدوات النشر على الويب، فتتيح للمستخدم أو التطبيق رفع كائن إلى مسار معين (Directory)على خادم الويب. وأخيرا، فإن طريقة DELETE تسمح للمستخدم أو التطبيق بحذف كائن عن خادم الويب

رسالة الاستجابة HTTP Response Massage

فيما يلي رسالة استجابة HTTP نموذجية وقد تكو ردا على رسالة الطلب في المثال السابق :

HTTP/1.1 200 OK

Connection: close

Date: Tue, 09 Aug 2011 15:44:04 GMT

(Server: Apache/2..2.3 (CentOS

Last-Modified: Tue, 09 Aug 2011 15:11:03 GMT

Content-Type: text/html

Content-Length: 6821

(... data data data data data)

تتكون رسالة الاستجابة من ثلاثة أقسام:1. سطر الحالة الابتدائي ،2. ستة سطور مقدمة، 3. ثم نص الكيان (entity body) يحتوي نص الكيان على الكائن المطلوب نفسه ممثلة بالسطر .. data data data data data ويتكون سطر الحالة من 3 حقول: إصدار البروتوكول، ورمز حالة، ورسالة الحالة المقابلة. في هذا المثل بشير سطر الحالة إلى أن الخادم يستخدم

HTTP / 1.1 وأن كل شيء على ما يرام (أي أن الخادم موجود و بر سل الكائن المطلوب).

الأن سنناقش سطور المقدمة، يستخدم الخادم السطر Connection: close ليخبر العميل بأنه سيغلق اتصل TCP بعد ارسال الرسالة، ويدل سطر التاريخ : Date على وقت وتاريخ إنشاء الاستجابة وإرسالها من الخادم ولا يتعلق بوقت إنشاء الكائن نفسه أو أخر تعديل يشير السطر :Server أن الخادم الذي أنشأ الرسالة هو خادم الويب Apache. ويشير السطر :Last=Modified إلى وقت إنشاء الكائن أو آخر تعديل، وهو أمر بالغ الأهمية في التخزين المخبأ (الكاش) للكائن، سواء

كان محلية لدى العميل أو لدي خادمات الكاش التي تسمى بروكسي servers(.)Proxy يشير سطر طول المحتوى Content-Length: إلى عدد بايتات الكائن المرسل. وأخيرا، يشير سطر نوع المحتوى Content-Type: إلى نوع الكائن الموجود في نص الكيان، وهو في هذا المثال HTML نصي.)لاحظ أن Content-Type: هو الذي يحد د نوع الكائن وليس امتداد الملف(بشير الشكل (9-2) الشكل العام لرسالة الاستجابة، وهو يتفق مع ما شرحنا في مثالنا أعلاه في البداية، دعنا تحت قليلا من رموز الحالة والعبارات المرتبطة بها والتي تشير إلى نتيجة الطلب، وفيما يأتي بعض الرموز و العبارات الشائعة

1- 200 OK تحت الطلب ويتم إرجاع المعلومات في الاستجابة

2- 301 Moved Permanently تم نقل الكائن المطلوب بشكل دائم، يتم تحديد عنوان URL الجديد في السطر Location: في مقدمة رسالة الاستجابة، ويقوم برنامج العميل باسترداده تلقائيا

3- 400 Bad Request هو رمز خطا عام يشير أن الطلب غير مفهوم للخادم

4- 404 Not Found. السند المطلوب لا يتوفر على هذا الخادم

5- 505 HTTP Version Not Supported الخادم لا يدعم الإصدار الخاص بروتوكول HTTP المطلوب

2-3-4التفاعل بين المستخدم والخادم: الكوكيز (User Server Interaction Cookies)

يستخدم بروتوكول HTTP الكوكيز (Cookies)، غالية، حتى يتمكن موقع الويب من التعرف على مستخدميه ، اما لتقييد وصول المستخدم أو لتقديم المحتوى ل لمستخدم استنادا إلى هويته. والكوكيز معرفة في [RFC 6265] بأنها السماح للمواقع بتتبع المستخدمين، ومعظم مواقع الويب التجارية الكبرى تستخدم الكوكيز هذه الأيام

كما هو مبين في الشكل (10-2)، تتكون تكنولوجيا الكوكيز من أربعة عناصر، (1) سطر كوكى فى من رسالة استجابة HTTP (2)سطر كوكي في رسالة الطلب (3) الاحتفاظ بملف كوكي على نظام المستخدم بديره المتصفح(4) قاعدة بيانات الخلفية في موقع الويب

مثال: لنفترض أن سعاد دائما تستخدم متصفح Explorer Internet للوصول إلى الويب من حاسوبها ودخلت إلى

Amazon .Com لأول مرة، ولنفترض أنها زارت موقع Bay- e سابقة، تتبع كيف ستعمل من استنادا إلى الشكل (10-2) ، عندما يصل الطلب إلى خادم الويب الخاص بالأمازون، فإنه ينشئ رقم هوية فريد وينشئ قيدا في قاعدة البيانات الخلفية مفهرسة بهذا الرقم، ثم يستجيب خادم الويب إلى متصفح سعاد باستجابة تتضمن سطر المقدمة Cookie - Set: الذي يحتوي رقم الهوية، وقد يكون السطرSet-cookie: 1678 :

عندما يتلقى متصفح سعاد رسالة الاستجابة، يقرأ السطر Cookie- Set: فيضيف سطرا يشمل اسم الخادم ورقم الهوية إلى ملف الكوكي الذي يديره. تذكر أن ملف الكوكي لديه قيد لموقع Bay- e عندما زارته سعاد سابقا. مع استمرار سعاد بتصفح موقع الأمازون، في كل مرة تطلب صفحة ويب، يرجع المتصفح إلى ملف الكوكى، ويسترجع رقم هويتها لهذا الموقع، ويضع سطر كوكي يتضمن رقم الهوية في الطلب، وعلى وجه التحديد، كل الطلبات الموجهة إلى خادم الأمازون تشمل سطر المقدمة Cookie1678

بهذه الطريقة، يتمكن خادم الأمازون من تتبع نشاط سعاد في موقع الأمازون، على الرغم من أنه ال يعرف بالضرورة اسم سعاد، بل يعرف بالضبط الصفحات التي زارها المستخدم 1678 ،ومتى وبأي ترتيب. يستخدم الأمازون الكوكيز لتوفير خدمة عربة التسوق، إذ يستطيع الموقع الإبقاء على قائمة بجميع مشتريات سعاد حتى تتمكن من دفع الكلفة الإجمالية في نهاية الجلسة.

إذا عادت سعاد إلى موقع أمازون، بعد أسبوع، سيستمر متصفحها بوضع سطر Cookie:1678 في رسائل الطلب، وسيعرض الأمازون بعض المنتجات على سعاد بناء على صفحات الويب التي زارتها سابقا. وإذا قامت سعاد بالتسجل في موقع الأمازون بتوفير الاسم الكامل، والبريد الإلكتروني، والعنوان البريدي، ومعلومات بطاقة الائتمان، يمكن للأمازون إدراج هذه المعلومات في قاعدة البيانات الخاصة بها، وربط اسم سعاد برقم هويتها)وجميع الصفحات التي زارتها في الموقع سابقا(، وبهذه الطريقة توفر مواقع التجارة الإلكترونية خدمة التسوق بنقرة واحدة .(One - click shopping)

وعلى سبيل المثال، عندما تسجل الدخول إلى البريد الإلكتروني عبر الويب (مثل Hotmail)، ي رسل المتصفح من الدخول عبر الكوكي إلى الخادم، ويسمح للخادم بالتعرف على المستخدم حتى نهاية الجلسة. ورغم ك ن من الأحيان، تسهل التسوق عبر

الإنترنت، إلا أن الكوكيز مثيرة للجدل فيما يعتبر انتهاكا الخصوصية .يمكنك الرجوع إلى[Central2012 Cookie] لمزيد من المعلومات

5-3-2 الذاكرة المخبأة في الشبكة العنكبوتية(Caching Web)

ذاكرة الويب المخبأة(cache Web) وتسمى أيضا خادم وكيل server Proxy(، (هي أحد مكونات الشبكة يعمل على تلبية طلبات HTTP نيابة عن خادم الويب، ولديه قرص تخزين خاص يحتفظ فيه بنسخ من الأشياء المطلوبة مؤخرة .كما هو مبين في الشكل (11-2)ص67 ، يمكنك إعداد متصفحك بحيث يتم توجيه الطلبات إلى خادم الوكيل أولا.

مثال: لنفرض أن المتصفح طلب الكائن http: //www.Someschool .edu /campus.gif إليك ما سيحدث

1. ينشئ المتصفح اتصال TCP مع خادم الوكيل، ويرسل له طلب HTTP للحصول على كائن
2. يتحقق خادم الوكيل فيما إذا كان لديه نسخة من الكائن مخزنة محلية، فإذا كان كذلك، يرد الوكيل برسالة استجابة تتضمن إلى المتصفح العميل
3. .إذا لم يكن الكائن لدى الوكيل، يفتح اتصال TCP مع الخادم الأصل WWW Someschool . edu . ، ثم يرسل طلب HTTP للحصول عليه، فيرد الخادم الأصل برسالة استجابة للوكيل تتضمن الكائن
4. عندما يتلقى خادم الوكيل الكائن، فإنه يخزن نسخة محلية، ويرسل نسخة ضمن رسالة استجابة HTTPإلى متصفح العميل) عبر اتصال TCP القائم بين متصفح العميل وخادم الوكيل).

.شهد التخزين المخبأ للويب) caching Web) انتشارا في الإنترنت لسببين :أسباب انتشار الإنترنت

1. يمكنه أن يقلل وقت الاستجابة الطلب العميل بشكل كبير، وخاصة إذا كان عرض النطاق الترددي بين العميل والخادم الأصل أقل بكثير منه بين العميل وخادم الوكيل، وكان الكائن المطلوب لدي خادم الوكيل
2. يمكنه أن يقلل بشكل كبير حركة المرور على خط اتصال المؤسسة بشبكة الإنترنت، وبذلك ال تضطر المؤسسة إلى ترقية عرض النطاق الترددي بسرعة، مما يخفض التكاليف.

عزيزي الطالب، للحصول على فهم أعمق لفوائد الذاكرة المخبأ cache دعنا نناقش مثال استنادا إلى الشكل (12 - 2 ص69)الذي يظهر شبكتين؛ شبكة المؤسسة وشبكة الإنترنت العامة، شبكة المؤسسة هي شبكة محلية(LAN) عالية السرعة، وهناك موجه في شبكة المؤسسة وموجه آخر في شبكة الإنترنت متصلان عبر وصلة(خط)سرعتها. 15 Mbps

تصل الخوادم الأصل بشبكة الإنترنت ولكن تقع في جميع أنحاء العالم .لنفترض أن متوسط حجم الكائن عنا1M ،وأن المتوسط معدل الطلبات من متصفحات المؤسسة إلى الخ وادم الأصل هو 15 طلبا في الثانية ولنفترض أن رسائل طلب HTTP صغيرة وال تشكل حركة المرور تذكر في الشبكات أو في خط النفاذ الرئيس، ولنفترض أيضا أن مقدار الوقت اللازم لتوجيه طلب HTTPمن موجه الإنترنت على خط النفاذ الرئيس في الشكل (2 -12 ص69) إلى أن يتلقى ردا هو ثانيتين بالمتوسط، ودعنا نطلق عليه، تأخير الإنترنت (Internet delay).

وقت الاستجابة الإجمالي، أي من لحظة طلب كائن من قبل المتصفح حتى استلامه،1) هو مجموع تأخير الشبكة المحلية (LAN 2)تأخير الوصول (أي، التأخير بين الموجهين) وتأخير الإنترنت. دعنا الآن نقوم بحسبة بسيطة لتقدير هذا التأخير.

كثافة حركة المرور على الشبكة المحلية انظر القسم (1-8)

(15 requests/sec). (1 Mbits/request)/(100 Mbps) = 0.15

في حين أن كثافة حركة المرور على خط النفاذ الرئيس (بين الموجهين)هي

(15 requests/sec). (1 Mbits/request)/(15 Mbps) = 1

عادة، ينتج عن كثافة حركة المرور على الشبكة المحلية بقيمة 15.0 ، عشرات الميلي ثانية من التأخير على الأكثر؛ أي يمكن إهماله. وكما ناقشنا في القسم 1-8 ،عند اقتراب كثافة حركة المرور إلى 1(كما في حالة خط النفاذ الرئيس) يصبح التأخير على الخط كبيرا جدا، وينمو بلا حد. وهكذا، فإن متوسط زمن الاستجابة لتلبية الطلبات سيكون بضع دقائق، وهو أمر غير مقبول لمستخدمي المؤسسة وال بد من عالج أحد الحلول الممكنة هو زيادة معدل الوصول من 15Mbps إلى، فلنقل، 100Mbps ،وهذا يقلل من كثافة حركة المرور على خط النفاذ إلى 15.0 ،وهو 92تأخير ضئيل بين الموجهين، وعليه، تكون الاستجابة الكلية ثانيتين تقريبا، وهي تأخير الإنترنت. لكن هذا الحل يعني أن على المؤسسة ترقية الخط، فهو حل مكلف هناك حل بديل بتركيب مخبأ الويب، أي خادم الوكيل (server proxy) ،)في شبكة المؤسسة كما في الشكل (13 - 2)ص 70 .وتتراوح

معدلات الإصابة (rates Hit)،)أي نسبة الطلبات التي يلبيها الوكيل، عادة بين0.7-2.0 في الواقع العملي للتوضيح، دعنا نفترض أن معدل الإصابة4.0 لهذه المؤسسة. بما أن العملاء والوكيل

يتصلان عبر شبكة محلية عالية السرعة، 40 %من الطلبات تلبي على الفور تقريبا، ولنقل، في غضون 10ميلى ثانية. مع ذلك، هناك 60 %من الطلبات يجب أن تلبيها الخادمات الأصل، ولكن فقط 60 %من الطلبات ستمر عبر خط النفاذ، فتقل كثافة حركة المرور عليه من 0.1 إلى 6.0. عادة، كثافة حركة المرور التي تقل عن 8.0 يقابلها تأخير قليل، أي عشرات الميلي ثانية على خط Mbps15 .وهذا التأخير ال يكاد يذكر مقارنة مع تأخير الإنترنت الذي يبلغ 2 ثانية. ونظرة لهذه الاعتبارات، يكون متوسط التأخير هو 0.4(2.01 seconds) +0.6 (0.01 seconds)

وهو أكبر قليلا من 2.1 ثانية، وبالتالي، فإن هذا الحل له وقت استجابة أقل من الحل الأول، وال يتطل من المؤسسة رفع سرعة خط الإنترنت، بل عليها، بالطبع، شراء خادم الوكيل وتثبيته، ولكن كلفته أقل، وخاصة عند استخدام برنامج نطاق عام يعمل على أجهزة حاسوب غير مكلفة.

من خلال استخدام شبكات توزيع المحتوى (Content Distribution Networks: CDN ،)بدأت مخابئ الويب بشكل متزايد تلعب دورا مهما في شبكة الإنترنت، وتعمل شركة CDN على تثبيت العز من المخابئ الموزعة جغرافية عبر الإنترنت، وبالتالي، توطين الكثير من حركة المرور. هناك شبكات توزيع مشتركة (مثل أكماي ولايملايت(وشبكات مخصصة)مثل جوجل ومايكروسوفت)

6-2-3 الدالة الشرطية (The Conditional GET)

أن ذاكرة الويب المخبأة (caching web) تقلل زمن الاستجابة الذي يشعر به المستخدم، فقد تكون ت المخبأة قديمة، أي تم تعديل نسخة الكائن في خادم الويب بعد حفظها في الذاكرة المخبأة لدى العميل، ولكن آلية للتحقق من أن هذا الكائن هو الأحدث، وتسمى الدالة الشرطية GET conditional بروتوكول HTTP يقدم الية للتحقق من أن هذا الكائن هو الأحدث، تسمى رسالة الطلب رسالة GET الشرطية إذا كانت (1)رسالة الطلب تستخدم دالة GET أو (2)تتضمن مقدمة رسالة الطلب السطر

"If - Modified – Since"

مثال :آلية عمل دالة GET الشرطية أولا يرسل مخبأ الوكيل رسالة طلب إلى خادم الويب نيابة عن المتصفح

ثانيا، برسيل خادم الويب رسالة استجابة تتضمن الكائن المطلوب لمخبأ الوكيل الذي يقوم بدوره بتحويل الكائن إلى المتصفح الذي طلبه، ويخزن نسخة محلية مع تاريخ آخر تعديل

ثالثا، إذا طلب متصفح آخر نفس الكائن عبر الوكيل بعد أسبوع، وكان ال يزال في الذاكرة المخبأة، فيقوم الوكيل بفحص تاريخ آخر تعديل للتأكد أن الكائن لم يعدل في خادم الويب عن طريق إصدار دالة GET الشرطية بإرسال الرسالة

لاحظ ن قيمة IF-Modified-since تساوي بالضبط قيمة last-Modified التي ارسلها الخادم منذ اسبوع

4 - 2 بروتوكول نقل الملفات(File Transfer Protocol FTP)

في جلسة بروتوكول نقل الملفات(FTP Protocol Transfer File) النموذجية، يطلب المستخدم نقل ملفاته من وإلى مضيف بعيد host Remote) ،)ويمكنه الوصول إليه عن طريق اسم مستخدم وكلمة مرور، وبذلك يمكن للمستخدم نقل الملفات من نظام الملفات المحلي إلى البعيد والعكس بالعكس. وكما يبين الشكل ، 74 ص(14-2)يتفاعل المستخدم مع FTP عن طريق وكيلFTP.

حيث تنشئ عملية عميل FTP اتصال TCP مع عملية خادم FTP في المضيف البعيد، فيطلب الخادم تحديد اسم المستخدم وكلمة المرور، ويتم إرسالهما كجزء من أوامر FTP عبر اتصال . TCP وعندما يأذن الخادم للمستخدم، يمكنه نسخ واحد أو أكثر من الملفات المخزنة في المضيف المحلي إلى البعيد أو العكس. يتشابه FTP و HTTP بأنهما بروتوكولان لنقل الملفات بين أنظمة الملفات المحلية والبعيدة، وكالهما يعمل عبر. TCP ومع ذلك، فإن بينهما اختلافات مهمة، الفرق الأول هو أن FTPيستخدم اتصالي TCP متوازيين

اتصال التحكم و اتصال البيانات، كما هو موضح في الشكل (2 – 15ص75) يتم استخدام اتصال التحكم الإرسال مرات التحكم مثل اسم المستخدم وكلمة المرور، وأوامر تغيير الدليل البعيد، وأوامر وضع الملفات(PUT)الحصول عليها(get) أما اتصال البيانات فيستخدم لإرسال الملفات.

عندما يبدأ المستخدم جلسة FTP مع مضيف بعيد، ينشئ العميل اتصال تحكم TCPمع الخادم)المضيف البعيد(على رقم المنفذ 21 . يرسل العميل اسم المستخدم وكلمة المرور وكذلك أوامر تغيير الدليل البعيد عبر هذا الاتصال. عندما يتلقى الخادم أمراً لنقل الملفات ينشئ اتصال بيانات TCP مع العميل، يرسل بروتوكول FTP ملف واحدة عبر اتصال البيانات ثم يغلقه، وإذا، أراد المستخدم نقل ملف آخر خلال الجلسة نفسها، عليه إنشاء اتصال بيانات آخر. الحظ أن اتصال التحكم يبقى مفتوحة طوال مدة الجلسة، بينما يتم إنشاء اتصال بيانات جديد النقل كل ملف في نفس الجلسة، أي أن اتصال البيانات غير دائم

(non -persistent).على عكس HTTP، يجب أن يحتفظ FTP بحالة(state) المستخدم طوال الجلسة، فعلى الخادم أن يقرن اتصال التحكم بحساب مستخدم معين، وأن يتتبع مسار الدليل الحالي للمستخدم طالما هو يتنقل في شجرة الدليل البعيد، وهذا بالطبع يحد بشكل كبير من عدد الجلسات التي يستطيع FTP الإبقاء عليها معا.

1-4-2 أوامر نقل الملفات وردودها (FTP Commands and Replies)

في نهاية هذا القسم، نقدم مناقشة موجزة لبعض أوامر FTP الأكثر شيوعا والردود عليها. يتم إرسال الأوامر من العميل إلى الخادم، والردود من الخادم إلى العميل، عبر اتصال تحكم بشكل . bit ASCII -7 ولفصل الأوامر المتتالية، نحتاج إلى حرف إرجاع (return carriage) نهاية كل أمر. كل أمر يتكون من أربعة أحرف ASCII كبيرة، وفيما يلي بعض الأوامر الأكثر شيوعا-:

1- USER username: يستخدم لإرسال اسم المستخدم إلى الخادم

2- PASS password: يستخدم لإرسال كلمة المرور إلى الخادم

3- LIST : يستخدم ليطلب من الخادم عرض قائمة بجميع الملفات في الدليل البعيد، ويتم إرسال قائمة الملفات عبر اتصال بيانات جديد، وليس عبر اتصال التحكم TCP.

4- RETR filename: يستخدم للحصول على (get) ملف من الدليل الحالي على المضيف البعيد، فيأمر المضيف البعيد بإنشاء اتصال البيانات وإرسال الملف المطلوب عبره.

5- STOR filename:يستخدم لتخزين(put) ملف على الدليل الحالي في المضيف البعيد .هناك تراسل واحد لواحد بين الأمر الذي يصدره المستخدم وأمر FTP الذي يرسل عبر اتصال التحكم، فكل امر يتبعه رد من الخادم الى العمل، وتكون الرد على شكل أرقام من ثالثة منازل مع رسالة اختيارية تتبع

2-5| البريد الإلكتروني عبر الإنترنت - (Mail' in the Internet -Electronic Mail'e)

ظهر البريد الالكتروني منذ بداية الإنترنت، وكان مازال الأكثر شعبية وأهمية واستخداما بين تطبيقات الإنترنت

[Segaller 1998]، وكما هو الحال في البريد العادي، فإن البريد الإلكتروني وسيلة اتصال غير متزامن الإرسال الرسائل وقراءتها في الوقت المناسب دون الحاجة إلى تنسيق بين المرسل والمستقبل، وما يميزه عن البريد العادي هو السرعة وسهولة التوزيع بكلفة قليلة إن لم تكن مجانية، كما أن البريد الإلكتروني الحديث يوفر العديد من الميزات، بما في ذلك تضمين المرفقات، والروابط، والصور، وتنسيق النصوص. عزيزي الطالب، يغطي هذا القسم بروتوكولات طبقة التطبيقات الخاصة بالبريد الإلكتروني عبر الإنترنت، وقبل التعمق في هذه البروتوكولات، سنلقي نظرة عامة على هذا النظام ومكوناته الرئيسة.

هناك ثلاثة عناصر رئيسية؛ وكلاء المستخدم، خادم البريد، وبروتوكول نقل البريد البسيط

(Transfer Protocol: SMTP Simple Mail) وفي هذا السياق سيكون المرسل "أليس" A، والمستلم "بوب" B. يسمح الوكلاء للمستخدم بقراءة الرسائل، والرد عليها، وتحويلها، وحفظها، وكتابتها، ومن الأمثلة على وكلاء المستخدم برنامج Microsoft Outlook وبرنامج Apple. Mail عندما تنتهي أليس من كتابة رسالتها، يقوم وكيلها بإرسالها إلى خادم البريد الذي يضعها في طابور الرسائل الصادرة outgoing message queue) وعندما يريد بوب قراءة الرسالة، يتلقى وكيله الرسالة من صندوق بريده في خادم البريد.

يشكل خادم البريد جوهر البنية التحتية للبريد الإلكتروني، فكل مستلم لديه صندوق بريد في أحد خوادم البريد، ويعمل صندوق البريد على إدارة الرسائل التي أرسلت له وحفظها. تبدأ رحلة أي رسالة نموذجية من وكيل المستخدم الخاص بالمرسل، وتنتقل إلى خادم البريد المرسل، ثم إلى خادم البريد المستلم، حيث يتم ايداعها في صندوق بريد المستلم

عندما يريد بوب الوصول إلى رسائله في صندوق البريد، فإنه بحاجة إلى مصادقة خادم البريد الذي يحتوي صندوق بريده من خلال اسم المستخدم وكلمة المرور، وعلى خادم البريد الخاص بأليس التعامل مع حالات الخطأ أو الفشل في خادم البريد الخاص ببوب، فإذا لم يستطع خادم أليس تسليم البريد إلى خادم بوب، يقوم خادم أليس بالاحتفاظ بالرسالة في طابور الرسائل message queue ويحاول نقل الرسالة في وقت لاحق، وغالبا عيد المحاولة كل 30 دقيقة؛ فإذا نجح خلال بضعة أيام، فإنه يحذف الرسالة ويعلم المرسل (أليس).

يعد SMTP بروتوكول طبقة التطبيق الأساسي للبريد الإلكتروني عبر الإنترنت، ويستخدم خدمة نقل البيانات الموثوقة من TCP لنقل البريد من خادم البريد المرسل (جانب العميل) إلى خادم البريد المستلم (جانب الخادم(. بعبارة أخرى، كلا الجانين، العميل والخادم، يعملان على كل خادم للبريد، فعندما يرسل الخادم بريدة إلى خادم آخر، فإنه يعمل كعميل SMTP ، وعندما يتلقى الخادم بريدا من خادم آخر، فإنه يعمل كخادمSMTP

2-5-1بروتكول نقل البريد البسيط (Simple Mail Transfer Protocol: SMTP

يعتبر بروتوكول SMTP، المعرف فيRFC5321 ، في صميم البريد الإلكتروني عبر الإنترنت، فهو بعمل على نقل الرسائل من خادم البريد المرسل إلى خادم البريد المستلم، وهو أقدم بكثير من HTTP، وعلى الرغم من العديد من المزايا، يستند SMTPإلى تقنيات وخصائص بانت قديمة. فعلى سبيل المثال، هناك قيود على محتوى الرسالة)وليس المقدمة فقط(لجميع رسائل البريد نحو استخدام الاسكي البسيط 7-ASCII bit، فقد كان هذا منطقية في وقت مبكر عندما كانت الرسائل بسيطة بال مرفقات ضخمة من صور وصوت وفيديو، أما اليوم، في عصر الوسائط المتعددة، أصبحت هذه القيود تشكل عبئه، حيث ينبغي ترميز بيانات الوسائط المتعددة الثنائية إلى اسكي قبل إرسالها عبر SMTP، ومن ثم فك الترميز إلى ثنائي في الطرف المقابل بعد النقل .لتوضيح آلية عمل SMTP الأساسية، دعنا نأخذ السيناريو الذي يلخصه الشكل 78ص(17-2)

لنفترض أن أليس تريد إرسال رسالة أسكي بسيطة إلى بوب، كما يبين الشكل (17-2)ص78 ،عندها تنتقل الرسالة بالخطوات الآتية:

1. تستدعي أليس وكيل المستخدم الخاص بها للبريد الإلكتروني، وتدخل عنوان البريد الإلكتروني لبوب مثلا bob@someschool.edu)، وتنشئ رسالة ثم تطلب من وكيل المستخدم إرسال الرسالة.

2. يرسل وكيل أليس الرسالة إلى خادم بريدها، حيث يقوم بوضعها في طابور الرسائل.

3.عندما يجد عميل SMTP الذي يعمل على خادم بريد أليس في طابور الرسائل، إلى خادم SMTP الذي يعمل على خادم بريد بوب.

4 بعد المصافحة الأولية، يرسل عميل SMTP يفتح اتصال TCP رسالة أليس عبر اتصال TCP

5. يتلقى خادم SMTP الرسالة على خادم بريد بوب، ثم يضع الرسالة في صندوق بريد بوب

6. يستدعي بوب وكيل المستخدم الخاص به لقراءة الرسالة في الوقت الذي يناسبه.

لاحظ أن SMTP لا عادة خوادم وسيطة لإرسال حتى لو كان الخادمان على طرفي العالم، فمثلا، إذا كان خادم أليس في هونغ كونغ وخادم بوب في سانت لويس، يكون اتصال TCP مباشرة وبالتحديد، إذا كان خادم بريد بوب معطلا، تبقى الرسالة لدي خادم أليس منتظرة محاولة جديدة، ولا ترسل الى خادم وسيط.

الرسالة إلى الخادم دون أخطاء، ثم يكرر العميل هذه

العملية على نفس الاتصال إذا كان لديه رسائل بعد ترسل إلى خادم وسيط أخرى لإرسالها إلى الخادم؛ وإلا فإنه يطلب من TCP إغلاق الاتصال. مثال: يتبادل العميل الرسائل مع الخادم من خلال بروتوكول اسم مضيف العميل crepes.fr واسم مضيف الخادم hamburger.edu. سطور نص التي تسبقها : C هي التي يرسلها العميل عبر مقبس TCP الخاص به، وسطور نص أسكي التي تسبقها هي التي يرسلها الخادم إلى مقبس TCP الخاص به، ويبدأ تنفيذ النص التالي فور إنشاء اتصال TCP :

S: 220 hamburger.edu

C: HELO crepes.fr

S: 250 Hello crepes.fr, pleased to meet you

C: MAIL FROM: <alice@crepes.fr> S: 250 alice@crepes.fr ... Sender ok

<C: RCPT TO: <bob@hamburger.edu

S: 250 bob@hamburger.edu ... Recipient ok

C: DATA

?S: 354 Enter mail, end with "." on a line by itself C: Do you like ketchup

?C: How about pickles

. :C

S: 250 Message accepted for delivery

C: QUIT

S: 221 hamburger.edu closing connection

يرسل العميل الرسالة

("?Do you like ketchup? How about pickles") من خادم البريد crepes.fr الى خادم البريد burger.edu

وضمن الحوار، أصدر العميل خمسة أوامر : HELO

(اختصار مرحبا)QUIT , DATA,RCPT TO, MAIL FROM وهي اوامر واضحة كما يرسل العميل سطرا يتكون من نقطة (.) وتعني للخادم نهاية الرسالة. في لغة اسكي، كل رسالة تنتهي ب CRLF.CRLF، حيث الاختصار CR حرف إرجاع ت(curriage return) والاختصار LF سطر التغذية Line feed يرد الخادم على كل أمر، لكل رد رمز وبعض الشرود روحات الاختيارية. تذكر أن SMTP يستخدم الاتصال الدائم: إذا كان خادم البريد المرسل سيرسل عدة رسائل إلى نفس خادم البريد المستقبل، يمكنه إرسالها كافة على نفس اتصال TCP . لكل رسالة، تبدأ العملية لدى العميل بسطر جديد FROM:crepes.fr ويشير إلى نهاية الرسالة بنقطة معزولة (.)،ولا يصدر أمر الخروجQUIT إلا بعد ارسال الرسائل كافة

2-5-2 تنسيق رسالة البريد (Mail Message Formats)

تم تعريف ووصف رسالة البريد الإلكتروني في RFC 5322 ، ويحدد الشكل الدقيق لسطور المقدمة وتفسيراتها الدلالية. يتم فصل سطور المقدمة أو نص الرسالة بسطر فارغ (CRLF)، وكل سطر في المقدمة يحتوي نصا مقروءا يتألف من كلمة متبوعة بنقطتين ثم قيمة محددة، بعض الكلمات أساسية وبعضها اختياري. تحتوي المقدمة على سطر "من" (: From) وسطر "إلى")To:(، وقد تحتوي سطر "الموضوع" (: Subject) وغيرها من السطور الاختيارية. لاحظ أن هذه السطور تختلف عن الأوامر التي درسناها في القسم 2-5-1 ، فالأوامر هي جزء من بروتوكول مصافحة SMTP ، بينما سطور المقدمة هي جزء من رسالة البريد الإلكتروني نفسها يتبع مقدمة الرسالة سطر فارغ، ثم نص الرسالة (ASCII)، وتبدو مقدمة الرسالة كما يأتي

From: alice@crepes.fr

To: bob@hamburger.edu

Subject: Searching for the meaning of life. حاول، عزيزي الطالب، استخدام Telnet لإرسال رسالة إلى خادم البريد تحتوي بعض سطور المقدمة مثل سطر الموضوع كما فعلت في القسم 2-5-1

2-5-3 بروتوكولات الوصول إلى البريد Mail Access Protocols

في هذه الأيام، تستخدم معمارية العميل لخادم للوصول إلى البريد الإلكتروني، وذلك من خلال عميل ينفذ على النظام النهائي للمستخدم، كحاسوبه المكتبي أو المحمول، أو هاتفه الذكي، فالمستخدم العادي يمكنه التمتع بمجموعة غنية من الميزات، بما في ذلك قراءة البريد الإلكتروني وعرض الوسائط المتعدد والمرفقات. على اعتبار أن بوب (المستلم) يشغل وكيل المستخدم الخاص به على حاسوبه الشخصي، فمن الطبيعي وضع خادم البريد على حاسوبه الشخصي أيضا، وبذلك، يتم الحوار مباشرة بين خادم البريد الخاص بأليس وحاسوب بوب. ولكن هناك مشكلة في هذا النهج، فخادم البريد يدير صناديق البريد ويشغل جانبي العميل والخادم لبروتوكول SMTP ، فإذا كان خادم البريد متواجدة على الحاسوب الشخصي المحلي لبوب، يجب أن يكون دائما متصلا بالإنترنت، للحصول على البريد الجديد الذي قد يصل في أي وقت. وهذا غير عملي لكثير من المستخدمين، لذا، فإن الحل هو أن يشغل المستخدم العادي وكيلا على حاسوبه المحلي ويتصل بصندوق بريده المخزن على خادم بريد قيد التشغيل دائما، يشترك فيه مع المستخدمين الآخرين، ويتم الاحتفاظ به لدى مزود خدمة الإنترنت. تم تصميم SMTP لدفع البريد الإلكتروني إلى خادم البريد من خلال وكيل المستخدم الخاص بالمرسل، الذي لا يمكنه الحوار المباشر مع خادم البريد المستلم، بل خادم بريد المرسل هو المسؤول عن التحاور مع خادم بريد المستلم وترحيل رسائل البريد الإلكتروني إليه، كما هو مبين في الشكل (18-2). فإذا كان خادم المستقبل معطلا يستمر الخادم المرسل بالمحاولة مرارا لإرسال الرسالة كل 30 دقيقة، إلى أن يتم تشغيله.

لاحظ أن وكيل المستخدم لا يمكنه استخدام SMTP السحب (pull) الرسائل بل لدفعها (push)، فلا بد من بروتوكول آخر يمكن المستلم من تلقي الرسائل من صندوق البريد على خادم بريده إلى جهازه الشخصي المحلي وهناك عدد من البروتوكولات الشائعة للوصول إلى البريد الإلكتروني، بما في ذلك بروتوكول مكتب البريد الاصدار 3

(Post Office Protocol-Version 3: POP3)

وبروتكول الوصول الى البريد عبر الانترنت (Internet Mail Access Protocol: IMAP)

ويلخص الشكل (18-2) البروتوكولات التي تستخدم في بريد الإنترنت

بروتوكول مكتب البريدPOP3

POP3 هو بروتوكول وصول إلى البريد تم تعريفه في [RFC 1939]،و هو في غاية البساطة، ووظائفه محدودة، ويبدا عمله عندما يفتح وكيل المستخدم (العميل)اتصال TCP إلى خادم البريد (الخادم) على المنفذ 110، وبتقدم العمل في ثالث مراحل:

1 المصادقة أو التقوي)Authorization): يرسل وكيل المستخدم اسم المستخدم وكلمة المرور (نص واضح) للمصادقة

2 الحركة او المعاملات(Transaction): يتلقى وكيل المستخدم الرسائل، وفي هذه المرحلة، يمكنه الحديد الرسائل للحذف، وإزالة عالمات الحذف، والحصول على إحصائيات حول البريد .

3 التحديث(Update): ويحدث عندما يصدر العميل أمر التوقف) quit) وإنهاء جلسة بروتوكول POP3عندها ، يقوم خادم البريد بحذف الرسائل التي تم تحديدها للحذف.

في معاملة POP3، يصدر وكيل المستخدم الأوامر، ويستجيب الخادم بالرد على كل أمر، وهناك نوعان من الردود المحتملة: الأول OK+)أحياناً، تليها بيانات من الخادم إلى العميل(، ويستخدمه الخادم للإشارة إلى أن

الأمر السابق كان على ما يرام. والثاني، يستخدمه الخادم للإشارة إلى حدوث خطأ في الأمر السابق. - ,<pass <password> user <username> Y1:ll al Jar و عزيزي الطالب، لتوضيح الأمرين، حاول إصدارهما من خلال Telnet مباشر إلى خادم POP3 ، باستخدام المنفذ 110 ، لنفترض أن اسم خادم البريد الخاص بك هو mail Server،ستحصل على شيء من هذا القبيل:

telnet mail Server 110

+OK POP3 server ready

user bob

+OK

pass hungry

+OK user successfully logged on

فإذا أخطأت في كتابة الأمر، فإن رد خادم POP3 يكون بالرسالة ERR- في مرحلة المعاملات، يمكن للمستخدم غالبا تغيير إعدادات وكيل المستخدم POP3 إلى وضعين أساسيين: "وضع التنزيل والحذف" download-and-delete- mode أو "وضع التنزيل والإبقاء" download and-Keep- mode ويعتمد تسلسل الأوامر التي يصدرها وكيل المستخدم POP3 على هذين الوضعين. ف في الوضع الأول يصدر الأوامر:.dele, list، retr.

من خلال POP3 ، بمجرد أن يتلقى بوب رسائله ويحفظها في حاسوبه المحلي، يمكنه إنشاء مجلدات بريدية ونقل الرسائل إليها، كما يمكنه حذف الرسائل، ونقلها عبر المجلدات، والبحث عن رسائل (عن طريق اسم المرسل أو الموضوع(. ولكن هذا النموذج، أي المجلدات والرسائل على الحاسوب المحلي، يشكل عقبة للمستخدم المتنقل، إذ يفضل حفظها على الخادم البعيد ليتمكن من الوصول إليها من أي مكان وفي أي زمان، وهذا ما لا يقدمه بروتوكول POP3 . لذا لا بد من البحث عن بروتوكول آخر يقدمها، كما سنوضح حالا.

بروتوكول الوصول إلى البريد عبر الإنترنت IMAP

الحل المشاكل التي يعاني منها بروتوكول POP3 ، كما ذكرنا آنفا، جاء بروتوكول IMAP الذي تم تعريفه في [RFC 3501] كبروتوكول وصول إلى البريد. ولديه العديد من المميزات التي لا يوفرها POP3 ، ولكنه بشكل ملحوظ أكثر تعقيدا، ما يعني أن تنفيذ برمجة العميل والخادم أكثر تعقيدا أيضا.

يربط خادم IMAP كل رسالة بمجلدها؛ فعندما تصل الرسالة إلى الخادم لأول مرة، ترتبط بمجلد البريد الوارد للمستلم، الذي يمكنه بعد ذلك نقلها إلى مجلد جديد أنشاه، أو قراءتها أو حذفها. ويسمح بروتوكول IMAP للمستخدمين بإنشاء المجلدات ونقل الرسائل من مجلد إلى آخر، كما يسمح بالبحث عن الرسائل على المجلدات البعيدة. لاحظ أنه، على عكس POP3 ، يحتفظ خادم IMAP بمعلومات عن حالة المستخدم طوال الجلسة، مثل أسماء المجلدات والرسائل التي ترتبط بها ميزة .

هامة أخرى من IMAP هي أنه يسمح لوكيل المستخدم بالحصول على مكونات الرسائل، مثل مقدمة الرسالة فقط أو جزء من رسالة MIME متعددة الأجزاء. وهي ميزة مفيدة عندما يكون الاتصال ضعيفة (مودم بطيئة) بين وكيل المستخدم وخادم البريد، فقد لا ترغب في تحميل الرسائل كافة في صندوق ولا سيما الرسائل الطويلة التي قد تحتوي على مقاطع صوت أو فيديو.

البريد الإلكتروني عبر الإنترنت Web-Based E-Mail

أصبح البريد الإلكتروني المعتمد على الإنترنت أكثر شيوعا هذه الأيام، إذ يتم توفير البريد الإلكتروني من قبل هوتميل، وجوجل، وياهو، فضلا عن الجامعات والشركات وغيرها. في هذه الحالة، وكيل المستخدم هو متصفح ويب عادي، ويصل المستخدم إلى صندوق بريده البعيد عبر HTTP ، فعندما يريد المستلم، مثل بوب، الوصول إلى رسالة في صندوق بريده، يتم إرسال رسالة البريد الإلكتروني من خادم البريد إلى متصفحه باستخدام بروتوكول HTTP بدلا من POP3 أو IMAP. وعندما يريد مرسل، مثل أليس، إرسال رسالة البريد الإلكتروني، يتم إرسالها من متصفحها لخادم بريدها عبر HTTP بدلا من SMTP ، مع ذلك، يبقى خادم البريد الخاص بأليس، يرسل الرسائل ويتلقاها من وخوادم البريد الأخرى باستخدام SMTP.

نظام اسم النطاق (DNS _The internets

(Directory Service

تماما كما هو الحال لدي البشر، يمكن التعرف على كل مضيف على شبكة الإنترنت بطرق عدة، واحدة من هذه المعرفات مثل هي اسم المضيف (hostname)مثل . www .edu . qou ، Www. yahoo .com,cnn.com ولكن اسم المضيف لا يقدم إلا قليلا من المعلومات عن موقع المضيف على مثال، اسم المضيف . www. eurecom fr . ، الذي ينتهي برمز الدولة fr ،يبين أن المضيف هو على فرنسا لا أكثر)، وعلاوة على ذلك، فإن أسماء المضيفين تتكون من حروف متغيرة الطول، فيصعب اليه جهات معالجتها. لهذه الأسباب، يتم تعريف المضيف بما يسمى عنوان الانترنت IP.Address

نناقش عناوين الإنترنت بشيء من التفصيل في الوحدة الرابعة، ولكن دعنا نذكر بعض المعلومات المهمة هنا بإيجاز، يتكون عنوان الإنترنت، بإصداره الرابع (IPv4)، من 4 بايت تفصل بين كل بيت منها نقطة، وتمثل بالنظام العشري من 0 إلى 255 ، مثل الرقم: 121.7.106.83 . وكلما انتقلنا في قراءته من اليسار إلى اليمين، تحصل على المزيد من المعلومات المحددة حول موقع المضيف على الإنترنت (أي في أي شبكة ضمن الشبكات)

1-6-2 الخدمات التي يقدمها نظام اسم النطاق(Services Provided by DNS)

يفضل المستخدم معرف اسم المضيف للتذكر، بينما الموجهات تفضل معرفات ذات طول ثابت، أي عناوين الإنترنت المنظمة بشكل هرمي. ومن أجل التوفيق بينهما، فإننا بحاجة إلى خدمة الدليل الترجمة اسماء المضيفين إلى عناوين إنترنت، وهذه هي الخدمة الرئيسة التي يقدمها نظام اسم النطاق (: Domain -Name System: DNS) فهو: (1)يوفر قاعدة بيانات موزعة ضمن تسلسل هرمي من خوادم DNS، (2) يسمح للمضيفين بالاستعلام في قاعدة البيانات الموزعة.

غالبا، تعمل خوادم DNS على أجهزة UNIX وتشغل برنامج نطاق اسم الإنترنت من بيركلي

(Internet Name Domain: BIND Berkley) [BIND 2012] ويستخدم المنفذ 53 .وعادة، يستخدم DNS من قبل بروتوكولات أخرى في طبقة التطبيقات، بما في ذلك FTP، SMTP، HTTP لترجمة أسماء المضيفين التي يضعها المستخدم إلى عناوين إنترنت. فمثال، عندما يقوم المتصفح اي عميل HTTP بطلب العنوان

WWW. Someschool . edu/ index . html ،وحتى يتمكن تضيق المستخدم من إرسال طلب HTTP إلى خادم الويب WWW . Someschool.edu ، عليه الحصول أولا على عنوان الإنترنت الخاص بخادم الويب كما يأتي:

1. يعمل عميل تطبيق DNS على جهاز المستخدم نفسه

2. يستخرج المتصفح اسم المضيف www.someschool.edu من عنوان URL ويمر ره إلى جانب العميل

3. يرسل عميل DNS استعلامه يحتوي اسم المضيف إلى خادم DNS

4. يتلقى عميل DNS الرد، والذي يتضمن عنوان الإنترنت الخاص باسم المضيف.

5.عندما يحصل المتصفح على عنوان الإنترنت من DNS ، فإنه ينشئ اتصال TCP مع خادم HTTP عن طريق المنفذ 80 على ذلك العنوان تلاحظ من هذا المثال آن ختمة DNS تسبب تأخيرا إضافية، قد يكون طويلا، لتطبيقات الإنترنت التي تستخدمها تستخدم ذاكرة الكاش المخبأة لتقليل حركة المرور ومتوسط التأخير الناجمة عن DNS بالإضافة إلى الترجمة يوفر DNS عدة من الخدمات المهمة الأخرى.

*المضيف المستعار عندما تكون أسماء المضيفين معقدة وطويلة، يمكن تسميتها باسم واحد أو أكثر من الأسماء المستعارة مثلا، اسم المضيف relayl-west-coast.enterprise.com ، قد يكون له اسمين www.enterprise.com, enterprise.com وفي هذه الحالة يسمى relayl.west-coast.enterprise.com يكون أسهل الذكر من الاسم الأساسي، ويمكن التطبيقات الحصول على اسم المضيف الأساسي و عنوان الإنترنت (IP) لاسم المضيف المستعار عبر خادم DNS

*خادم البريد المستعار: يستحسن أن يكون عنوان البريد الإلكتروني سهل التذكر، فعلى سبيل المثال، قد يكون عنوان البريد الإلكتروني تبسيط مثل bob@hotmail.com، مع ذلك، فإن الاسم الأساسي لخلع البريد قد يكون أكثر تعقيدا وصعب التذكر من مجرد hotmail.com، فقد يكون الاسم - relayl.west coast.hotmail.com أو شيء من هذا القبيل. ومن الفوائد الأخرى التي يمكن الحصول عليها في هذا الإطار، أن سجل MX يسمح باستخدام نفس الاسم لأكثر من خاتم من خلال الأسماء المستعارة فمثلا قد يحمل خادم البريد وخادم الويب اسما واحدا enterprise.com.

*توزيع الحمل: يستخدم DNS أيضا لتوزيع الحمل بين الخوادم المستنسخة، مثل خواتم الويب، فالمواقع المترنحة، مثل cnn.com ، مستنسخة على خوادم متعددة، وكل خاتم يعمل على منظومة مختلفة ولكل منها عنوان إنترنت IP مختلف. وفي هذه الحالة ترتبط مجموعة من عناوين الإنترنت IP باسم مضيف أساسي واحد، وتحتوي قاعدة بيانات DNS هذه المجموعة من العناوين، فعند الاستعلام عن الأسد الأساسي، يستجيب خادم DNS بالمجموعة الكاملة من العناوين، ولكن بترتيب نوراني متباين في كل رد، لأن العميل عادة يرسل رسالة طلب HTTP إلى العنوان الذي يتم مرده أولا، وهذا الدوران يعمل على توزيع حركة المرور بين الخادمات المستنسخة. ويستخدم نوران DNS في البريد الإلكتروني ومواقع توزيع المحتوى، بحيث يكون للخوادم المتحدة نفس الاسم المستعار [2002] Dilley.

حددت مواصفات DNS في المرجعين RFC 1034 و RFC 1035 ، وحثت في عدة مراجع إضافية و هو نظام معقد، وهنا سنسلط الضوء على الجوانب الرئيسة فقط، وننصح المهتمين بالعودة إلى هذه المراجع، وكذلك كتاب البيتز وليو [1993 Albila] لمزيد من المعلومات.

2-6-2لمحة حول الية عمل DNS

عزيزي الطالب، سنقدم في هذا البند نظرة عامة حول الية عمل نظام DNS ، وسنركز على خدمة الترجمة من اسم المضيف إلى عنوان الإنترنت IP لنفترض أن أحد التطبيقات (مستعرض ويب أو البريد الإلكتروني) في مضيف المستخدم يحتاج إلى ترجمة اسم المضيف إلى عنوان الإنترنت IP سيستدعي التطبيق جانب العميل من DNS ، وبالتحديد أسم المضيف المنوي ترجمته (في العديد من الأجهزة المستندة إلى UNIX تستخدم التطبيقات الدالة ()gethostbyname لطلب الترجمة(، وعندها يتولى DNS في المضيف المستخدم إرسال رسالة استعلام إلى الشبكة يتم إرسال جميع وسائل الاستعلام و الرد عبر مخططات UDP إلى المنفذ 53 بد تأخير، قد يصل إلى ثانية، يتلقى DNS في مضيف المستخدم رسالة الرد التي توفر الترجمة المطلوبة ليتم تمريرها التطبيق الذي استدعاها من منظور التطبيق المستدعي يعتبر DNS صندوق أسود يوفر خدمة ترجمة بسيطة وواضحة في الواقع، هذا الصندوق معقد، ويتألف من عدد كبير من خوادم DNS الموزعة في جميع انحاء العالم، كما أن هناك بروتوكول طبقة تطبيقات يحدد كيفية تواصل خوادم DNS والمضيفين الذين يطلبون الخدمة قد يكون التصميم مركزيا، أي خادم DNS واحد يحتوي على جميع السجلات، وترسل جميع استعلامات العملاء مباشرة إلى هذا الخادم، الذي يستجيب مباشرة الاستعلامات العملاء، وعلى الرغم من بساطة هذا التصميم، قاله لا يناسب الإنترنت هذه الأيام، مع العدد الكبير والمتزايد من المضيفين، وتتضمن مشاكل التصميم المركزي:

** نقطة واحدة للفشل: إذا تعطل خادم DNS ، تعطلت شبكة الإنترنت بالكامل

**حجم حركة المرور. لن يتمكن خادم DNS واحد من التعامل جميع الاستعلامات من طلبات HTTP ورسائل البريد الإلكتروني الصادرة من مئات الملايين من الأجهزة أو المضيفين

** قاعدة بيانات مركزية بعيدة. خادم DNS واحد لا يمكن أن يكون قريبا من جميع العملاء، فإذا كان خادم DNS واحد في مدينة رام الله، وجاءت جميع الاستعلامات من نيوزلندا فإن عليها أن تنتقل إلى الجانب الآخر من الأرض، عبر وصلات قد تكون بطيئة ومزدحمة، ما قد يؤدي إلى تأخير كبير

** الصيانة احتفاظ خادم DNS واحد بسجلات جميع المضيفين على الإنترنت لا يشكل قاعدة بيانات مركزية ضخمة فحسب، بل لا بد من تحديثها باستمرار لكل مضيف جديد. وباختصار نتيجة لذلك، يتم تصميم DNS بشكل قاعدة بيانات موزعة على الإنترنت

قاعدة بيانات هرمية موزعةHierarchical Distributed

من اجل التعامل مع مسالة حجم الطلبات، يستخدم DNS عدة كبيرة من الخوادم الموزعة بطريقة هرمية في جميع أنحاء العالم، إذ لن يتمكن خادم DNS واحد من احتواء جميع المضيفين في شبكة الإنترنت، بل توزع على هذه الخوادم. هناك ثلاث فئات من خوادم DNS خوادم الجذر، وخوادم النطاق عالي المستوى (TLD)والخوادم الموثوقة نظمت جميعا بتسلسل هرمي كما يبين الشكل 19-2 ص87

كيف يمكن لهذه الفئات الثلاث من الخوادم أن تتفاعل، لنفترض أن عميل DNS يريد تحديد عنوان IP للمضيف www.amazon.com البداية يتصل العميل بأحد خوادم الجذر، الذي يرد بعناوين IP لخوادم TLD للنطاق عالي المستوى من نوع com ، فيتصل العميل بأحد خوادم TLD ، الذي يرد بعنوان IP لخادم موثوق لدى amazon.com . وأخيرا، يتصل العميل بأحد الخوادم الموثوقة لدى amazon.com ، الذي يرد بعنوان IP ل لمضيف www.amazon.com. عزيزي

الطالب، سنفحص عملية البحث DNS lookup بمزيد من التفاصيل في الأقسام الفرعية التالية، ولكن دعنا الآن نلقي نظرة فاحصة على فئات خوادم DNS الثلاث :

*خوادم الجذر Root. DNS servers في الانترنت هناك 13 نوع من خوادم الجذر من (A إلى M)، كل خادم هو في الواقع شبكة من الخوادم المتكررة، لأغراض الأمن والموثوقية على حد سواء، وفي خريف 2011 بلغ مجموع خوادم الجذر 247 خادما.

*خوادم النطاق عالي المستوى Top-Level Domain TLD servers هذه الخادمات مسؤولة عن المستوى الأعلى ل لنطاقات مثل (com, org، edu .Gov ,net, كما أنها مسؤولة عن نطاقات الدول مثل ca jp uk ,fr,

*الخوادم الموثوقة المخولةAuthoritative. DNS servers كل مؤسسة لديها مجموعة متاحة من المضيفين على شبكة الإنترنت (مثل خوادم الويب والبريد الإلكتروني(يجب أن تتيح سجلات DNS التي تترجم اسماء هؤلاء المضيفين إلى عناوين إنترنت IP ، ويضم الخادم الموثوق للمؤسسة هذه السجلات والمؤسسة أن تختار ما بين حفظ هذه السجلات على خادم موثوق خاص بها، أو أن تدفع مقابل حفظها في خوادم موثوقة لدى مزود الخدمة.

هناك نوع آخر مهم من خوادم DNS يدعى الخادم المحلي أو الافتراضي local، DNS server ورد غران لا ينتمي للتسلسل الهرمي إلا أنه مركزي في بنية DNS لدى مزود الخدمة، فعندما يتصل مضيف بمزود الخدمة، يزوده بعناوين الإنترنت لواحد أو أكثر من خوادمه المحلية)عادة من خلال DHCP)، وقد يكون خادم DNS المحلي قريبا من المضيف، أي على نفس الشبكة المحلية LAN ، فعندما يقوم المضيف باستعلام DNS فإنه يرسل إلى خادم DNS المحلي الذي يعمل، بدوره كوكيل، على توجيه الاستعلام إلى خادم DNS ضمن التسلسل الهرمي.

مثال: لنفترض أن المضيف cis.poly.edu يرغب بالحصول على عنوان gaia.cs.umas s.edu، وأن خادم DNS المحلي لبوليتكنيك يدعى dns.poly.edu، وأن الخادم الموثوق لدى gaia.cs.umass.edu يسمى dns.umas s.edu، كما هو مبين في الشكل (20-2). بين الية الحصول على العنوان المطلوب. يقوم المضيف cis.poly.edu أولا بإرسال رسالة استعلام DNS إلى خادمه المحلي dns.poly.edu تحتوي اسم المضيف المراد ترجمته، أي gaia.cs.umass.edu ، فيقوم الخادم المحلي بتوجيه رسالة الاستعلام إلى الخادم الجذر الذي يستجيب بدوره للخادم المحلي ويزوده بقائمة من عناوين الإنترنت لخوادم النطاق عالي المستوى (TLD المسؤولة عن (edu)، فيعيد الخادم المحلي إرسال رسالة الاستعلام إلى واحد من خوادم TLD الذي يقوم بدوره بتسجيل المقطع النهائي umass.edu ، ويستجيب بعنوان الإنترنت الخادم الموثوق الجامعة ماساشوستس، أي dns.umas s.edu. واخيرا، يعيد الخادم المحلي إرسال رسالة الإستعلام مباشرة إلى dns.umass.edu ، الذي يستجيب بعنوان الإنترنت للمضيف gaia.cs.umass.edu . في هذا المثال، تم ارسال 8 رسائل DNS4 : رسائل استعلام و 4 رسائل استجوابية! وسترى في الفقرات التالية كيف يمكن تقليص حركة مرور الاستعلامات من خلال التخزين المؤقت DNS caching

يستخدم مثالنا المبين في الشكل كلا من استعلامات الإعادة Iterative Queries S a dns.poly.edu cis.poly.edu a sa (Recursive Queries) Yli lai doll, تكرار، إذ أنه يطلب من dns.poly.edu الحصول على الترجمة نيابة عنه، بينما الاستعلامات الثلاثة التالية الهي استعلامات إعادة، لأن كل الردود تعود مباشرة إلى inspoly.du من الناجية النظرية، يمكن لأي استعلام DNS أن يكون استعلام إعادة أو تكرار فعلى سبيل المثال، فإن سلسلة استعلامات DNS المبينة في الشكل

(21-2) كلها استعلامات تكرار. أما في الممارسة العملية، فتتبع الاستعلامات النمط المبين في الشكل (20-2)، أي أن الاستعلام عن المضيف الطالب إلى خادم DNS المحلي هو استعلام تكرار، والبقية هي استعلامات إعادة.

الذاكرة المخبأة (التخزين المؤقت)DNS Caching

عزيزي الطالب، لغاية الآن لم نتطرق إلى مفهوم الذاكرة المخبأة في نظام DNS ، إذ أنها ميزة بالغة الأهمية حيث تستغل الذاكرة المخبأة على نطاق واسع لتحسين الأداء وللحد من التأخير، وتقليص عدد رسائل DNS المرتدة عبر شبكة الإنترنت. وفكرة الذاكرة المخبأة هي في غاية البساطة، فعندما يتلقى الخادم في سلسلة الاستعلام استجابة DNS)قد تحتوي على ترجمة من اسم المضيف إلى عنوان الإنترنت IP)، فإنه يمكنه تخزين هذه الترجمة في الذاكرة المحلية، فمثلا، في الشكل (20-2)، في كل مرة يتلقى خادم DNS المحلي dns.poly.edu الرد من أحد خوادم DNS ، فإن بإمكانه تخزين أي من المعلومات الواردة في الرد. فإذا تم تخزين الزوج (اسم المضيف، عنوان الإنترنت في خادم DNS ، ووصل استعلام آخر لخادم DNS النفس المضيف، فإن هذا الخادم يوقر عنوان الإنترنت IP المطلوب، حتى لو لم يكن هو الخادم الموثوق لهذا المضيف. ولأن ترجمة أسماء المضيفين إلى عناوين إنترنت هي عملية دائمة، فإن خوادم DNS تتجاهل المعلومات المخزنة في الذاكرة المخبأة بعد فترة تحدد غالبا بيومين

وكمثال على ذلك، افترض أن المضيف apricot.poly.edu يستعلم من dns.poly.edu عن عنوان الإنترنت المضيف cnn.com ، ولنفرض أن مضيفا آخر من جامعة بوليتكنيك kiwi.poly.fr ء استعلم أيضا من dns.poly.edu عن نفس المضيف بعد بضع ساعات، ونتيجة للتخزين في الذاكرة المخبأة، يستطيع خادم DNS المحلي فورا استرجاع عنوان المضيف cnn.com في المرة الثانية دون الحاجة إلى الاستعلام من أي خوادم DNS أخرى، كما يستطيع الخادم المحلي تخزين عناوين خوادم TLD ، مما يتيح لخادم DNS المحلي تجاوز خوادم الجذر في سلسلة الاستعلام (وهذا ما يحدث في كثير من الأحيان).

2-6-3 سجلاتDNS ورسائله (DNS Records and Messages)

عزيزي الطالب، تعمل خوادم DNS التي تشكل مع قاعدة بيانات موزعة على الاحتفاظ بسجلات الموارد

Resource Records (RRs) بما في ذلك الترجمة من اسم المضيف إلى عنوان الإنترنت IP،وكل رسالة استجابة تحمل واحدة أو أكثر من سجلات الموارد. في هذا القسم، نقدم لمحة موجزة عن سجلات الموارد DNS ورسائله ويمكنك الاستزادة من[Abitz 1993] أو من مراجع .[DNS1034، RFC 1035]يتكون سجل المورد من أربعة أجزاء تحتوي على حقول (الاسم، القيمة، النوع، زمن الحياة ,) (Value, TTL Name, Type). ويحدد زمن الحياة (TTL :Time to Live) الفترة الزمنية المتاحة قبل إزالة سجل المورد من الذاكرة المخبأة. وتبين النقاط الآتية أمثلة على سجل المورد، حيث تجاهلنا حقل زمن الحياة (TTL)كما أن معنى كل من الاسم (Name) والقيمة(Value) يعتمد على النوع(Type):

- اذا كان النوع Type= A ، سيكون الاسم (Name) هو اسم المضيف، والقيمة(Value) هي عنوان الإنترنت IP ،وكمثال على ذلك، السجل (relay1. bar. Foo.com,145.37.93.126, A) هو سجل من نوع type A

- ذا كان النوع Type = NS ، يمثل الاسم (Name) نطاقا (domain) ,مثل Foo.com وتمثل القيمة(Value) اسم المضيف لخادم DNS موثوق يعرف كيف يحصل على عناوين المضيفين في هذا النطاق، ويستخدم هذا السجل التوجيه استعلامات DNS في سلسلة الاستعلام، فعلى سبيل المثال، السجل(foo.com, dns.foo.com,NS) وهو سجل من نوع Type NS,

- اذا كان النوع Type= CNAME تمثل القيمة تمثل القيمة (Value) اسم المضيف الأساسي, ويمثل الاسم (name) اسم المضيف المستعار، ويمكنك هذا السجل من الاستعلام عن الاسم الأساسي للمضيف مثال على ذلك (foo.com, relay1.bar.foo.com, CNAME) وهو سجل CNAME

- إذا كان النوع Type = MX, تمثل القيمة تمثل القيمة (Value) ,ا لاسم الأساسي لخادم البريد الإلكتروني الذي له اسم مستعار ويمثل الاسم (name) وكمثال على ذلك، السجل (, foo.com MX foo.com .mail. Bar) من نوع MX ، وتتيح سجلات MX للمضيفين في خوادم البريد أن يكون لهم اسماء مستعارة بسيطة، كما يمكنك سجل MX من استخدام نفس الاسم المستعار لخادم البريد وخوادم أخري)مثل خادم الويب(. للحصول على الاسم الأساسي لخادم البريد، يستعلم عميل DNS عن سجل MX وللحصول على الاسم الأساسي للخوادم الأخرى، يستعلم عميل DNS عن سجل CNAME

رسائل اسم النطاق DNS Messages

شرنا سابقا، عزيزي الطالب، إلى رسائل استعلام DNS ورسائل الاستجابة، وهناك نوعان فقط من رسائل DNS، وسواء كانت رسائل استعلام أو استجابة فإن لها نفس الشكل، كما هو مبين في الشكل(2-2 2) ص92

تكون صيغة الحقول المختلفة في رسالة DNS كما يأتي:

- مقدمة الرسالة (Header section) بايت الأولى، ويحتوي هذا المقطع عددا من الحقول، الحقل الأول هو رقم طوله 16 بت يعرف الاستعلام، حيث يتم نسخ هذا المعرف في رسالة الرد على استعلام مما يتيح للعميل ربط الردود المستقبلة بالاستعلامات المرسلة، وهناك عدد من الرايات في حقل الراية (Flag)، راية الاستعلام / الرد" طولها 1 بت، وتحدد فيما إذا كانت الرسالة هي استعلام (0) أو رد (1)، راية "الخادم الموثوق" طولها 1 بت، وتحدد في رسالة الرد إذا كان خادم DNS موثوقة للاسم المستعلم عنه، راية الإعادة مرغوبة" طولها 1 بت، وتحدد عندما يرغب العميل)مضيف أو خادم DNS) بأن يقوم خادم DNS بإعادة الطلب عندما لا يكون السجل متوفرة لديه. راية حقل الإعادة متوفر " طولها 1 بت، ويحدد في رسالة الرد إذا كان خادم DNS يدعم الإعادة. في المقدمة، هناك أيضا اربعة حقول رقمية، وتشير هذه الحقول إلى عدد مرات حدوث الأنواع الأربعة من مقاطع البيانات التي تتبع المقدمة
- المسألة أو الاستعلام (Question section) يحتوي هذا المقطع معلومات حول الاستعلام الحالي، ويتضمن (1) حقل الاسم، ويحتوي على الاسم الذي يجري الاستعلام عنه (2) حقل النوع، ويشير إلى البريد لاسم من نوع (Type). نوع

السؤال المطروح حول الاسم، مثلا عنوان مضيف يرتبط بالاسم من نوع(A Type) أو خادم البريد لاسم من نوع (type mx)

- الإجابة أو الرد Answer section يحتوي هذا المقطع سجلات المورد الاسم الذي تم الاستعلام عنه أصلا تذكر أنه في كل سجل مورد هناك النوع(CNAME، ، NS،A Type وMX)، والقيمة Value ، ووقت الحياة TTL الرد قد يعود بسجلات موارد متعددة في الجواب، إذ أن المضيف قد يكون له عناوين إنترنت IP متعددة
-
- لسلطة أو التحويل Authority section يحتوي هذا المقطع سجلات الخوادم الموثوقة الأخرى. مقطع إضافي Additional section يحتوي هذا المقطع سجلات أخرى مفيدة. على سبيل المثال، فإن حقل الجواب في الرد على استعلام MX يحتوي سجل مورد يزودك باسم المضيف الأساسي لخادم البريد، وهنا يحتوي المقطع الإضافي سجلا نوعه Type A يزودك بعنوان IP للمضيف الأساسي الخادم البريد.

السؤال الذي يدور في ذهنك عزيزي الطالب، كيف ترسل رسالة استعلام DNS مباشرة من المضيف الذي تعمل عليه إلى خادم DNS ؟ يتم ذلك باستخدام برنامج NSLOOKUP الذي يتوفر في معظم أنظمة تشغيل ويندوز ويونيكس. للاختبار من نظام Windows ، افتح موجه الأوامر ثم قم باستدعاء NSLOOKUP ببساطة عن طريق كتابة الأمر " NSLOOKUP "، عندها يمكنك إرسال استعلام DNS إلى أي خادم DNS بعد تلقي رسالة الرد من خادم DNS ، سوف يعرض لك برنامج NSLOOKUP السجلات المدرجة في الاستجابة سيتناول الجانب العملي الخاص بهذه الوحدة خادم DNS بالتفصيل من خلال مختبر وير شارك.DNS Wireshark lab

Inserting Records into the DNS Database ادراج سجلات DNS في قاعدة البيانات

ر كزنا في المناقشة اعلاه كيفية استرجاع السجلات قاعدة بيانات DNS ربما تتساءل، عزيزي لطالب، كيف يتم إدراج السجلات في قاعدة البيانات في المقام الأول؟ دعنا نجيب على التساؤل بمثال
مثال : مثال افترض أنك أنشأت شركة جديدة أطلقت عليها اسم شبكة يوتوبيا)Network، Utopia(في البداية عليك

تسجيل اسم النطاق netWorkutopia.com لدى أحد المسجلين أو مزودي الخدمة المسجل هو كيان تجاري يتحقق من تفرد اسم النطاق، ويدخل اسم النطاق في قاعدة بيانات DNS مقابل رسوم رمزية (كما هو مبين أدناه(، وهناك العديد من المسجلين المعتمدين من مؤسسة الإنترنت ل لأسماء والأرقام « the Internet Corporation for Assigned Names and Numbers (ICANN) aail| والقائمة الكاملة للمسجلين المعتمدين متوفرة على الموقع .http://www.internic.net

وعند تسجيل اسم النطاق networkutopia.com تحتاج أيضا إلى تزويد المسجل بأسماء وعناوينIP لخوادم DNS الموثوقة الأساسية والثانوية الخاصة بك، ولنفترض أن الأسماء والعناوين هي dns.networkutopia.com, 212.212.212.2 ,212.212.212.1 ,dns2.networkutopia.com ولكل من هذين الخادمين الموثوقين يتأكد المسجل من إدخال سجل من نوع NS وآخر من نوع A إلى خوادم TLD com وبالتحديد لخادم networkutopia.com الموثوق الأساسي، فإن المسجل يدرج سجلي الموارد لآتيين إلى نظام DNS :

(networkutopia.com, dns.networkutopia.com, NS)

(dns1.networkutopia.com, 212.212.212.1, A) كما عليك التأكد أيضا من إدراج سجل مورد لخادم الويب الخاص بك www.networkutopia.com من نوع A ، وسجل مورد من نوع MX لخادم بريدك الالكتروني mail.networkutopia.com إلى خوادم DNS الموثوقة الخاصة بك. بمجرد الانتهاء من هذه الخطوات، يصبح بإمكان المستخدمين زيارة موقع الويب الخاص بك و إرسال بريد إلكتروني للموظفين شركتك. دعنا نلخص مناقشتنا لنظام DNS

مثال لنفترض أن أليس في أستراليا تريد استعراض صفحة الويب WWW.networkutopia.com ، عندها يرسل مضيفها استعلام DNS إلى خادم DNS المحلي الخاص بها، والذي يقوم بدوره بالاتصال بخادم TLD.com وإذا لم يتم العثور على عنوان الخادم TLD com يقوم خادم DNS المحلي بالاتصال بخادم DNS جذر، وخادم TLD يحتوي سجل مورد من نوع NS وآخر من نوع A ، لأن المسجل أدرج هذه السجلات في جميع خوادم TLD. com يرسل خادم TLD com الرد الذي يحتوي سجلي المورد إلى خادم DNS المحلي الخاص بها، ثم يرسل خادم DNS المحلي استعلام DNS إلى 212.212.212.1، طالياً السجل من نوع A الذي يقابل المضيف W W W.networkutopia.com، هذا السجل يزود أليس بعنوان الإنترنت IP لخادم الويب المرغوب، مثل 212.212.71.4 ، الذي يعيده خادم DNS المحلي إلى جهاز أليس. عندها يستطيع متصفح أليس إنشاء اتصال TCP مع المضيف 212.212.71,4 وإرسال طلب HTTP عبر هذا الاتصال.

2-7 تطبيقات النظير للنظير (Peer-to-Peer "P2P Applications |

لا تعتمد بنية النظير للنظير (Peer-to-Peer P2P) أو تعتمد بالحد الأدنى، على خوادم البنية التحتية التي تعمل دائما)always-on(، بل يقوم أزواج من المضيفين المرتبطين بشكل متقطع، يطلق عليهم النظراء، بالاتصال المباشر مع بعضهم البعض. والنظراء غير مملوكين لمزود الخدمة، بل هم عبارة عن أجهزة حاسوب مكتبية ومحمولة يديرها المستخدمون.

في هذا القسم سنقوم بدراسة أحد التطبيقات المناسبة تماما لتصاميم P2P ، وهو توزيع الملفات، حيث يوزع التطبيق ملفا من مصدر واحد إلى عدد كبير من النظراء، فتوزيع الملفات هو أفضل بداية لتحقيقنا في P2P ، إذ بعرض بوضوح قابلية التوسع الذاتي في بنية P2P ، وسنأخذ نظام تورنت الشائع كمثال على توزيع الملفات

(2-7-1 P2P File Distribution توزيع ملفات النظير للنظير

تبدا p2p بالنظر في تطبيق طبيعي جدا، وهو توزيع ملف كبير (ملف موسيقى أو فيديو مثلا) من خادم واحد إلى عدد كبير من المضيفين، هم النظراء في بنية العميل الخادم، يرسل الخادم نسخة من الملف إلى كل نظرائه، فيضع عبئا هائلا على كاهل الخادم، ويستهلك كمية كبيرة من عرض النطاق الترددي.. أما في بنية P2P يستطيع كل النظراء إعادة توزيع أي جزء تلقوه من الملف إلى أي من نظرائه، وبالتالي مساعدة الخادم في عملية التوزيع. اعتبارا من عام 2012 ، أصبح تورنت أكثر بروتوكولات P2P شعبية لتوزيع الملفات، وهناك العديد من عملاء تورنت المستقلين تتوافق مع بروتوكول تورنت. في هذا القسم الفرعي، سندرس او قابلية التوسع الذاتي لبنية P2P في سياق توزيع الملفات، ثم نصف تورنت وأبرز خصائصه وميزاته

Scalability of P2P Architectures قابلية التوسع في معمارية النظير-للنظير

لمقارنة بين بنيتي العميل الخادم والنظير للنظير، وتوضيح قابلية التوسع الذاتي في P2P ، سنختبر نموذجا كمية بسيطة لتوزيع ملف إلى مجموعة ثابتة من النظراء باستخدام المعماريتين. كما يبين الشكل (2-23)ص95، يتصل الخادم والنظراء إلى الإنترنت من خلال روابط النفاذ.

ولنعبر عن معدل التحميل من رابط النفاذ للخادم بدلالةus ، ومعدل تحميل رابط النفاذ للنظير بدلالة ui ، و معدل نزيل رابط النفاذ للنظير بدلالة d , . ولنعتبر حجم الملف الذي سيتم توزيعه F (بت)، و عدد النظراء الذين

يرغبون بالحصول على نسخة من الملف N. يعرف زمن التوزيع Distribution time بأنه الز من الليزر الحصول جميع النظراء N على نسخة من الملف. في تحليلنا لزمن التوزيع أدناه، سنستخدم فرضية تبسيط أن خط الإنترنت دو عرض نطاق ترددي وفير، ما يعني أن أي اختناقات هي في شبكات النفاذ، ولنفترض أيضا أن الخادم والعملاء لا يشاركون في أي تطبيق شبكة أخر، حتى يتسنى تكريس النطاق الترددي لنفاد التحميل التوزيع البنية العميل الخادم D ، وفي هذه الحالة لن يشارك أي نظير في توزيع والتنزيل كاملا لتوزيع هذا الملف

دعنا أولا نحدد زمن التوزيع لبينة العميل الخادم Dcs وفي هذه الحالة يشارك أي نظير في توزيع الملف ونلاحظ ما يأتي

- على الخادم نقل نسخة واحدة من الملف إلى كل نظير من النظراء N ، بذلك يجب نقل NF بت. ولأن معدل التحميل ل لخادم ,، يكون وقت توزيع الملف ,NF/ ث على الأقل
- لنعتبر dmin معدل التنزيل للنظير بحده الأدني، أي أن { dx , d , mind = dmin . لن يتمكن النظير هو أدنى معدل تنزيل من الحصول على جميع بتات F من الملف في أقل من F/dmin ث وهكذا يكون الحد الأدني لزمن التوزيع F/dy على الاقل

بوضع هاتين الملاحظتين معا، نحصل على المعادلة (2-1):

Dcs>=max {NF\Us,F\Dmin}

و هذا يوفر الحد الأدني لزمن التوزيع في معمارية العميل الخادم. دعنا نعتبر الحد الأدنى الوارد أعلاه زمن التوزيع الفعلي. تبين المعادلة 2-1 أنه إذا كان عدد النظراء N كبير بما فيه الكفاية، يعطى زمن التوزيع في بنية عد د النظراء، فإذا زاد العدد خلال أسبوع الف ضعف

العميل الخادم بالقيمة NF / 1 ، أي يتناسب طرديا مع إلى مليون(، يزداد زمن توزيع الملف 1000 مرة

من دعنا الأن نجري تحليلا مماثلا لبنية P2P ، حيث يسهم كل نظير في توزيع الملف، فعندما يتلقى نظير بعض الملف لنظرائهم الآخرين لهذه الغاية، علينا أولا إبداء الملاحظات التالية

بيانات الملف، يمكنه استخدام قدرة التحميل الخاصة به لإعادة توزيع البيانات إلى نظير آخر. لكن حساب زمن التوزيع لبنية P2P إلى حد ما أكثر تعقيدا من بنية العميل الخادم، إذ يعتمد على كيفية توزيع كل نظير لأجزاء

في بداية التوزيع، يجب أن يرسل الخادم كل بنت من الملف مرة واحدة على الأقل عبر خط النفاذ الخاص به. وهكذا، فإن زمن التوزيع بحده الأدنى , F u على الأقل. (خلافا لنظام العميل الخادم)

كما هو الحال في بنية العميل الخادم، لا يستطيع النظير بأدنى معدل تنزيل الحصول على جميع البنات من الملف في أقل من F/d ث. أي الحد الأدنى لزمن التوزيع هو F/d على الأقل إجمالي قدرة تحميل النظام ككل يساوي معدل تحميل الخادم بالإضافة إلى معدلات تحميل كل نظير، وعلى النظام تسليم (تحميل) F بت إلى كل نظير، وبالتالي تسليم ما مجموعه NF بلت، أي أن اس NF/(us + u1+ .. un وبذلك يكون الحد الأدني لزمن التوزيع) الملاحظات الثلاث معا، نحصل على الحد الأدني من زمن توزيع D بالمعادلة 2-2

$$Dp2p>=max\{F/Us,F/Dmin,NF/Us+\sum_{i=1}^{N} ui\}$$

يقارن الشكل (242) الحد الأدني لزمن التوزيع بين بنيتي العميل الخادم والنظير للنظير على افتراض أن جميع النظراء لهم نفس معدل التحميل it . وقد تم تثبيت hour 1 10 u F/u us وهكذا، يستطيع النظير تقل ملف كامل في ساعة واحدة، ومعدل نقل الخادم 10 أضعاف معدل تحميل النظير، ول لتبسيط، تم تثبيت معدلات تنزيل النظراء بحيث تكون كبيرة بما فيه الكفاية حتى لا تؤثر. نلاحظ من الشكل (24-2)، في بنية العميل الخادم، أن زمن التوزيع يزيد خطية بلا حدود كلما ازداد عدد النظراء، أما في بنية النظير للنظير، الحد الأدني لزمن التوزيع دائما أقل من زمن التوزيع في بنية العميل الخادم، بل أقل من ساعة واحدة لأي عدد من النظراء N لذلك، تطبيقات P2P قابلة للتوسع ذاتية كنتيجة مباشرة لإسهام نظرائهم كموزعين ومستهلكين للبنات في الوقت ذاته.

تطبيق بت تورنت (سيل الثنائيات) Bit Torrent

بت تورنت هو بروتوكول P2P شائع لتوزيع الملفات [Chao[201 1 وتسمى مجموعة النظراء المشاركين في توزيع ملف معين تورنت، وينزل نظراء التورنت قطعا (chunks) متساوية الحجم من الملف فيما بينهم كل منها 256 كيلوبايت، وعندما ينضم نظير لأول مرة، لا يكون لديه قطع، و تراكم بمرور الوقت. فبينما يقوم بتنزيل قطع، يعمل على تحميل قطع لنظرائه الآخرين. وبعد حصوله على الملف بأكمله، قد يترك التورنت بأنانية(، أو يستمر في تحميل أجزاء لنظرائه. كما يمكن مغادرة التورنت والانضمام مرة أخرى لاحقاً. يعنا الأن نلقي نظرة فاحصة على كيفية عمل تورنت. كل تورنت لديه عقدة بنية تحتية تسمى متعقب tracker فعندما ينضم نظير التورنت، فإنه يسجل نفسه مع متعقب يبلغه بشكل دوري أنه لا يزال في تورنت. بهذه الطريقة، يتابع المتعقب النظراء المشاركين في تورنت، الذين قد يتراوح عددهم في اللحظة الواحدة من أقل من عشرة إلى أكثر من ألف مشارك .

كما يبين الشكل (25-2)، عندما ينضم نظير جديد، اليس، إلى تورنت، يختار المتعقب عشوائيا مجموعة فرعية من النظراء المشاركين (ولنقل 50)، ويرسل عناوين IP الخاصة بهم إلى أليس. تحاول أليس تأسيس اتصال TCP متزامن مع جميع النظراء على هذه القائمة، ولنطلق على جميع النظراء الذين نجحت أليس في الاتصال بهم "نظراء مجاورون" Neighboring peers ، (وقد يزيد أو ينقص النظراء المجاورون بمرور الوقت. كل نظير سيكون لديه قطع من ملفات متعددة، فتطلب اليس من كل نظير من جيرانها قائمة القطع التي لديهم عبر اتصال TCP)، فإذا كان لدي أليس L من الجيران، ستحصل على L من قوائم القطع، ثم تصدر أليس طلبات للحصول على قطع ليست بحوزتها حاليا، على أليس اتخاذ قرارين مهمين؛ الأول، أي من القطع تطلب من جيرانها أولا؟ والثاني، لأي من جيرانها عليها أن ترسل القطع المطلوبة؟ لاتخاذ القرار الأول تستخدم اليس تقنية تسمى الأكثر ندرة أولا rarest first ، فتطلب أولا القطع التي لا تملكها والأكثر ندرة بين جيرانها (أي أقل عدد من النسخ المتكررة بين جيرانها(. بهذه الطريقة، توزع القطع النادرة بسرعة أكبر، فيتساوى عدد نسخ كل قطعة في التورينت

لتحديد أي طلب تستجيب له، يستخدم تورنت خوارزمية ذكية، فتعطي أليس الأولوية لجيرانها الذين يزودونها بالبيانات حاليا بمعدل أعلى، أي تقيس باستمرار المعدل الذي تحصل فيه على بنات من كل نظير، وتحدد أفضل أربعة نظراء (يطلق عليهم unchoked) وتتبادل إرسال القطع معهم. ثم تقوم باحتساب المعدلات كل 10 ثواني، وقد تستبدل النظراء الأربعة في ضوء ذلك. وكل 30 ثانية، تختار جارا إضافيا عشوائيا (ولنقل "بوب") وترسل له القطع. ولأن أليس هي من ارسلت البيانات لبوب، فقد تصبح واحدة من أكثر أربعة محملين لبوب، فيبدأ بوب بإرسال البيانات إلى اليس. إذا كان المعدل الذي يرسل به بوب البيانات إلى أليس عالية بما فيه الكفاية فقد يصبح بوب، بدوره، واحدة من أكثر أربعة محملين لأليس

2-8 برمجة المقابس: إنشاء تطبيقات الشبكة | Socket Programming: Creating Network Application

عزيزي الطالب، بعد اطلاعك على عدد من تطبيقات الشبكة المهمة، حان دورك لإنشاء برامج تطبيقية الشبكة على أرض الواقع. كما ذكرنا سابقا في القسم 2-2 ، يتكون تطبيق الشبكة النموذجي من زوج من البرامج العميل والخادم، وينفذان على نظامين مختلفين، فعندما يتم تنفيذ هذين البرنامجين، يتم إنشاء عمليتين تتواصل مع بعضهما البعض من خلال القراءة من

المقابس والكتابة عليها. فعند بناء تطبيق شبكة، تكمن المهمة الرئيسة للمطور في كتابة التعليمات البرمجية لكل من العميل والخادم. وهناك نوعان من تطبيقات الشبكة

أي تطبيق الشبكة المفتوح: يتم تحديد العملية في معيار البروتوكول، مثل مراجع RFC أو أي معايير موثقة أخرى، أي أن قواعد عملها معروفة للجميع لمثل هذا التنفيذ، فيجب أن تتوافق برامج العميل والخادم مع قواعد مرجع RFC . فإذا عمل مبرمجان الأول على برنامج العميل والآخر على برنامج الخادم، واتبعا قواعد RFC بدقة، سيتمكن البرنامجان من التعامل فيما بينهما ب. تطبيق الشبكة المملوك: تستخدم برامج العميل والخادم بروتوكول طبقة التطبيقات التي قد لا يتم نشرها علنا في مراجع RFC ، وفي هذه الحالة يقوم مطور واحد (أو فريق تطوير)ببناء برنامجي العمل

والخادم، ويكون لديهم سيطرة كاملة على محتوى التعليمات البرمجية، لذا لن يتمكن المطورون المستقلون من تطوير تعليمات برمجية تتوافق مع التطبيق.

عزيزي الطالب، في هذا القسم، سنقوم بدراسة القضايا الرئيسية اللازمة لتطوير تطبيقات العميل الخادم. خلال مرحلة التطوير، يجب على المطور اتخاذ القرار ما إذا كان التطبيق موجه للعمل على TCP أو UDP. تذكر أن TCP مهيأ للاتصال ويوفر قناة موثوقة تتدفق فيها البيانات بين نظامين نهائيين، أما UDP فهو بدون اتصال ويرسل حزم مستقلة من البيانات من نظام نهائي إلى آخر دون أي ضمانات بشأن التسليم. تذكر أيضا أنه عندما يطبق برنامج العميل أو الخادم بروتوكولا محددة في مرجع RFC ينبغي أن يستخدم رقم المنفذ المعروف والمرتبط بالبروتوكول، وعلى العكس، عند بناء تطبيق مملوك، يجب الحذر من استخدام هذه المنافذ المعروفة سنستخدم لغة بايثون في برمجة المقابس لتطوير تطبيقين بسيطين، أحدهما على UDP والآخر على TCP ، وقد اخترنا هذه اللغة لأنها تكشف بوضوح المفاهيم الرئيسية المتعلقة بالمقبس باستخدام سطور أقل من التعليمات البرمجية، ويمكن لمبرمج مبتدئ تفسير كل سطر بسهولة، ومن لديه تجربة في أي لغة أخرى كذلك يستطيع المتابعة بسهولة.

2-8-1 برمجة المقابس في بروتوكول المخطط البياني للمستخدم (Socket Programming with UDP)

في هذا القسم الفرعي، سنقوم بكتابة برامج عميل خادم بسيطة تستخدم UDP ، وفي القسم التالي، سنكرر ذلك باستخدام TCP. دعنا نلقي نظرة فاحصة على التفاعل بين اثنتين من العمليات التي تستخدم مقايس UDP ، قبل أن تتمكن العملية المرسلة من إرسال حزمة بيانات خارج المقبس باستخدام UDP ، يجب أولا إرفاق عنوان الوجهة بهذه الحزمة وبعد مرور الحزمة من مقيس المرسل، يستخدم عنوان الوجهة التوجيه الحزمة من خلال الإنترنت إلى مقبس العملية المستقبلة، التي تعمل على استرداد الحزمة من المقبس، وتفقد محتوياتها واتخاذ الإجراء المناسب.

يتكون عنوان الوجهة المرفق بالحزمة من عنوان الإنترنت IP للمضيف الوجهة ورقم المنفذ، تستخدم أجهزة التوجيه عنوان الإنترنت لتتمكن من توجيه الحزمة إلى المضيف، ولكن قد تكون العديد من عمليات تطبيقات الشبكة قيد التشغيل على المضيف عبر واحد أو أكثر من المقابس، لذا فمن الضروري أيضا تحديد رقم المنفذ، وهو رقم معرف يسند إلى المقيس عند إنشائه في المضيف الوجهة باختصار، يرفق المرسل بالحزمة المرسلة عنوان IP ورقم منفذ المقبس للمضيف الوجهة وكذلك المضيف المصدر، ولكن إرفاق عنوان المصدر لا يتم في التعليمات البرمجية لتطبيق UDP ،بل يقوم بذلك نظام التشغيل تلقائيا.

مثال: سنستخدم تطبيق بسيطا للعميل - الخادم لشرح برمجة المقبس لكل من بروتوكولي UDP و TCP :

1يقرأ العميل سطر من الأحرف (البيانات) من لوحة المفاتيح ويرسلها إلى الخادم

2. يتلقى الخادم البيانات ويحول الأحرف إلى أحرف كبيرة

3. يرسل الخادم البيانات المعدلة إلى العميل.

4. يتلقى العميل البيانات المعدلة ويعرض السطر على شاشته بين الشكل)2-26(المهمة الرئيسة ل لعميل والخادم المتصلان عبر خدمة نقل بروتوكول UDP، والمتعلقة بالمقبس. الآن، دعنا نلقي نظرة على زوج برنامج العميل الخادم لتنفيذ UDP لهذا التطبيق البسيط، حيث ستقدم تحليلا مفصلا سطرا بسطر بعد كل برنامج. سنبدأ بعميل UDP الذي سيرسل رسالة بسيطة على مستوى التطبيق إلى الخادم، وليتمكن الخادم من استقبال الرسائل والرد على العميل، يجب أن يكون الخادم جاهزا وقيد التشغيل، أي يجب أن يتم تشغيل الخادم كعملية قبل أن يرسل العميل رسالته لنطلق على برنامج العميل UDPClient.py، وعلى برنامج الخادم UDPServer.py، ول لتأكيد على المسائل الأساسية، قدمنا عمدة الحد الأدني من التعليمات البرمجية (code). ونعلم بالتأكيد أن الكود الجيد لديه بضعة سطور مساعدة، وخصوصا لمعالجة حالات الخطأ، وفي هذا التطبيق، اخترنا 12000 عشوائية كرقم منفذ الخادم.

تبين التعليمات البرمجية (code) الآتية جانب العميل من التطبيق

UDPClient.py

```
from socket import*

serverName = "hostname

serverPort = 12000

   =clientSocket = socket(socket.AF_INET, socket.SOCK DGRAM) message = raw_input('Input lowercase sentence(':

client Socket.sendto(message (serverName, serverPort((

modifiedMessage, serverAddress = clientSocket.recvfrom( 2048(

print modified Message

client Socket.close()
```

*** from socket import ,UDPClient.py الآن، دعنا نلقي نظرة على السطور المختلفة من التعليمات البرمجية في اساس جميع اتصالات الشبكة في بايثون، وبإضافة هذا السطر ، سنتمكن من إنشاء المقابس Socket تشكل وحدة المقيس برنامجنا.**

```
serverName = "hostname?

serverPort = 12000
```

الخادم (مثلا، IP ، ويحتوي إما على عنوان serverName السطر الأول يضع اسم المضيف في متغير السلسلة النصية DNS ، فإذا أردنا استخدام اسم المضيف، سيتم البحث في(128.138.32.126cis.poly.edu) أو اسم المضيف)مثلا، . برقم المنفذ 12000 serverPort ، أما السطر الثاني فيحدد قيمة المتغير(IP تلقائيا للحصول على عنوان الإنترنت

Crea socket

clientSocket = socket(socket.AF_INET, socket.SOCK DGRAM) هذا السطر يخلق مقبس العميل ويدعى أن الشبكة تستخدم عناوين AF_INET ، يشير المعامل الأول إلى عائلة العنوان على وجه الخصوص، يشيرclientSocket ، أي SOCK_DGRAM في الوحدة 4)، أما المعامل الثاني فيشير أن المقبس من نوع IPv4 سنناقش عناوين) IPv4 لاحظ أننا لم نحدد رقم منفذ مقيس العميل عند إنشائه، بل تركنا ذلك لنظام التشغيل، والآن بعد أن أنشأنا باب عملية .UDP العميل، ستعمل على إنشاء رسالة لإرسالها عبر هذا الباب

message = raw_input('Input lowercase sentence:') الدالة raw_input() مضمنة في بايثون، فعند تنفيذ هذا الأمر، يطلب من المستخدم لدى العميل إدخال البيانات، فيستخدم لوحة المفاتيح لإدخال سطر يوضع في المتغير message ، والآن بعد أن أصبح لدينا مقبس ورسالة سنقوم بإرسال الرسالة من خلال المقبس الى المضيف الوجهة

clientSocket.send to(message,(server Name, server Port

serverName,( )serverPort في السطر أعلاه، ترفق الطريقة sendto()عنوان الوجهة

، ينتظر العميل ليتلقى البيانات إلى الرسالة ويرسل الحزمة الناتجة عبر مقبس العملية .clientSocket

، بعد إرسال الحزمة ينتظر العميل ليتلقى بيانات من الخادم

modifiedMessage, serverAddress = clientSocket.recvfrom(2048)

عند وصول حزمة الإنترنت عبر مقبس العميل، يتم وضع بيانات الحزمة في المتغير modifiedMes sage و عنوان المصدر للحزمة في المتغير serverAddress . يحتوي المتغير serverAddress كلا من عنوان الإنترنت IP ورقم المنفذ الخاصين بالخادم. لا يحتاج برنامج UDPClient حقيقة إلى معلومات عنوان الخادم، لأنه يعلم عنوان الخادم من البداية، مع ذلك يوفر هذا السطر من بايثون عنوان الخادم. أما الطريقة فتأخذ حجم المخزن المؤقت Buffer size(2048) كمدخل. (يعمل هذا الحجم لمعظم الأغراض

(print modified Message).

هذا السطر يطبع من modifiedMessage إلى شاشة المستخدم، وينبغي أن يكون السطر الأصلي الذي كتبه المستخدم، ولكن بالأحرف الكبيرة

client Socket.close ()

هذا السطر يغلق المقيس، عندئذ تنتهي العملية عزيزي الطالب، دعنا، الآن، نلقي نظرة على التعليمات البرمجية (code) الجانب الخادم من التطبيق

UDPServer.py

لاحظ أن بداية UDPServer تشبه UDPClient ، ويستورد أيضا وحدة المقبس، كما يضع في المتغير serverPort العدد الصحيح 12000 ، ويحدد المقبس من نوع SOCK_DGRAM (مقيس UDP). أما السطر الأول من التعليمات البرمجية التي تختلف اختلافا كبيرا عن UDPClient هو

serverSocket.bind(("", serverPort)) يعطي السطر أعلاه لمقبس الخادم رقم المنفذ 12000 ، وبهذه الطريقة، عندما يرسل أي شخص حزمة إلى المنفذ 12000 في عنوان IP للخادم، سيتم توجيه تلك الحزمة إلى هذا المقبس. ثم ينخل UDPServer في دورة انتظار تسمح له باستقبال الحزم من العملاء ومعالجتها إلى أجل غير مسمى، وينتظر الحزمة حتى تصل serverSocket.recyfrom(2048) = (message, clientAddress هذا السطر من التعليمات البرمجية يشبه ما رأيناه في UDPClient ، فعندما تصل حزمة في مقبس الخادم، يتم

وضع بيانات الحزمة في المتغير message ويتم وضع عنوان مصدر الحزمة في المتغير clientAddress ويحتوي من عنوان الإنترنت IP ورقم المنفذ الخاصين بالعميل هنا، يستفيد UDPServer من معلومات العنوان، إذ انه يوفر عنوان الرد على غرار عنوان الرد على المرسل

في البريد العادي، فبمعرفة معلومات عنوان المصدر، يعرف الخادم أين ينبغي أن بو جه رده = modified Message message upper() هذا السطر هو قلب تطبيقنا البسيط، فهو بأخذ السطر الذي أرسله العميل، ويستخدم الطريقة ( ) upper لتحويله إلى حروف كبيرة.

serverSocket.sendto(modified Message, clientAddress) هذا السطر الأخير يرفق عنوان العميل (عنوان IP ورقم المنفذ) بالرسالة ذات الأحرف الكبيرة، ويرسل الحزمة الناتجة عبر مقيس الخادم، ثم يتم تسليم الحزمة عبر الإنترنت إلى عنوان هذا العميل. وبعد أن يرسل الخادم الحزمة، يبقى في دورة الانتظار حتى وصول حزمة UDP أخرى)من أي عميل قيد التشغيل على أي مضيف(. ولاختيار زوج البرامج، عليك تنفيذ UDPClient.py على مضيف وتنفيذ UDPServer.py على مضيف آخر، تأكد من تضمين اسم المضيف الصحيح أو عنوان الخادم في UDPClient.py. بعد ذلك، نفذ UDPServer.py وبرنامج الخادم المجمع على مضيف

الخادم، حيث يتم إنشاء عملية في الخادم تبقى منتظرة

حتى يتصل بها عميل ما. ثم تقوم بتنفيذ UDPClient.py، وبرنامج العميل المجمع على مضيف العميل. وهذا بنت عملية في العميل. وأخيراً، لاستخدام التطبيق لدى العميل، اكتب جملة ثم اضغط مفتاح الإرجاع (Enter). Lms moe from soc srverPo

التطوير تطبيق العميل الخادم الخاص بك عبر UDP ، يمكنك البدء بتعديلات طفيفة على برنامجي العميل أو الخادم على سبيل المثال، بدلا من تحويل جميع الحروف إلى أحرف كبيرة، يعد للخادم عدد مرات ظهور الحرف s وإرجاع النتيجة، أو يمكنك التعديل على العميل بحيث يمكن للمستخدم الاستمرار في إرسال مزيد من الجمل للخادم بعد أن يتلقى العميل الجملة ذات الأحرف الكبيرة.

2-8-2 برمجة المقابس في بروتوكول التحكم بالنقل (Socket Programming with TCP)

على عكس UDP، بروتوكول TCP ما هو ل لاتصال، أي أن العميل والخادم بحاجة إلى المصافحة وإنشاء اتصال TCP قبل أن يبدأ إرسال البيانات فيما بينهما، حيث يرتبط أحد طرفي اتصال TCP بمقبس العميل ويرتبط الطرف الآخر بمقبس الخادم. عند إنشاء اتصال TCP ، يرتبط معه عنوان مقبس العميل (عنوان IP ورقم المنفذ) وعنوان مقبس الخادم)عنوان IP ورقم المنفذ(، فعندما يريد أحد الجانبين إرسال البيانات إلى الجانب الأخر، فما عليه سوى وضع البيانات لتنتقل عبر اتصال TCP عن طريق المقبس الخاص به، دون الحاجة إلى إرفاق عنوان الوجهة بالحزمة المرسلة كما كان الحال في UDP.

الآن دعنا نلقي نظرة فاحصة على التفاعل بين برنامجي العميل والخادم في TCP ، العميل بدء الاتصال بالخادم، وعلى الخادم أن يكون مستعدة للتفاعل مع العميل المتصل، وهذا يعني أمرين:

1. يجب أن يكون خادم TCP قيد التشغيل كعملية (process) قبل محاولة العميل بدء الاتصال، كما في حالة UDP

2. يجب أن يكون لدى برنامج الخادم مقبس خاص، يرحب بأي اتصال برد من عملية العميل قيد التشغيل على مضيف ما

بإمكان أي عملية في برنامج العميل بدء اتصال TCP مع عملية خادم قيد التشغيل عن طريق إنشاء مقبس (باب) TCP ، يحدد عنوان مقبس الترحيب في الخادم (أي عنوان IP ورقم منفذ المقبس(، عندئذ يبدأ العميل الخادم، يتلم client المرسل Modi تحويله ser مصافحة ثلاثية Three-way handshake في طبقة النقل، وينشئ اتصال TCP مع الخادم، لاحظ أن هذه المصافحة غير مرئية لبرنامجي العميل والخادم. في مثالنا أدناه، سمينا مقبس الترحيب serverSocket، وهو نقطة بدء الاتصال لجميع العملاء الراغبين في التواصل مع الخادم، وسمينا المقبس الذي أنشئ حديثا وخصص للعميل المتصل con nectionSocket. من منظور التطبيق، فإن مقبس العميل ومقبس اتصال الخادم يتصلان مباشرة عن طريق ما يشبه الأنبوب. وقد ترسل عملية العميل البايتات عشوائية عبر المقبس، ويضمن TCP أن تستقبل عملية الخادم (عبر مقبس الاتصال(كل بايت حسب ترتيب إرسالها، كما هو مبين في الشكل (27-2). وبالتالي يقدم TCP خدمة موثوقة بين عمليتي العميل والخادم. علاوة على ذلك، فإن عملية العميل لا ترسل البايتات إلى مقاسها فحسب، بل تتلقاها منه أيضا، كما أن عملية الخادم لا تتلقى البايتات من مقبس اتصالها فحسب، بل ترسلها إليه أيضا.

شكل 2-27 عملية TCPServer ذات المقبسين ص106

مثال: سنستخدم نفس التطبيق البسيط للعميل الخادم لعرض برمجة المقبس عبر TCP, حيث يرسل العميل سطرا واحدا من البيانات إلى الخادم، فيقوم الخادم بتغيير حالة الحروف إلى أحرف كبيرة، ثم يعيد السطر المعدل إلى العميل.

يوضح الشكل (28-2) ص107 المهمة الرئيسة للعميل والخادم المتصلين عبر خدمة نقل TCP ، والمتعلقة بالمقبس. كما تبين السطور الآتية التعليمات البرمجية (code) لجانب العميل من التطبيق

TCPClient.py

```
from socket import

serverName = 'servername' serverPort = 12000

clientSocket=socket(AF_INET,SOCKSTREAM)
clientSocket.connect((serverName,serverPort))

sentence = raw_input('Input lowercase sentence:') clientSocket.send(sentence(

modifiedSentence = clientSocket.recv(1024) print 'From Server:', modifiedSentence

clientSocket.close()
```

عزيزي الطالب، دعنا الآن تلقي نظرة في سطور التعليمات البرمجية التي تختلف بشكل كبير عنها في UDP.

يظهر أول اختلاف في سطر إنشاء مقبس العميل. = (client Socket = socket(AF INET, SOCK_STREAM ينشئ هذا السطر مقيس العميل، ويدعى clientSocket، يشير المعامل الأول أن الشبكة تستخدم عناوين

from serve IPv4

، أما المعامل الثاني فيشير أن المقبس من نوع SOCK_STREAM، أي مقبس TCP . لاحظ أننا مرة أخرى لم تحدد رقم المنفذ لمقيس العميل لدى إنشائه، بل تركنا ذلك لنظام التشغيل. السطر التالي من التعليمات

البرمجية يختلف كثيرا عما رأيناه في UDPClient client Socket.connect((server Name,server Port)) لنگر ضرورة إنشاء اتصال TCP بين العميل والخادم قبل أن يتمكن العميل من إرسال البيانات إلى الخادم أو العكس باستخدام مقبس TCP. يعمل السطر أعلاه على بدء اتصال TCP بين العميل والخادم، ويمثل معامل الطريقة () connect عنوان جانب الخادم، وبعد تنفيذ هذا السطر من التعليمات البرمجية، يتم تنفيذ المصافحة الثلاثية وإنشاء اتصال TCP بين العميل والخادم.

(':sentence = raw_input('Input lowercase sentence

كما هو الحال في UDPClient ، التعليمة أعلاه تطلب المستخدم إدخال جملة، وتواصل السلسلة النصية sentence جمع الأحرف حتى ينهي المستخدم السطر بالضغط على مفتاح Enter. أيضا يختلف السطر التالي

من التعليمات البرمجية كثيرة عن UDPClient :

(clientSocket.send(sentence

السطر أعلاه يرسل السلسلة النصية sentence من خلال مقبس العميل وعبر اتصال TCP. لاحظ أن البرنامج لا ينشئ حزمة ويرفق بها عنوان الوجهة كما هو الحال في مقبس UDP، بل يلقي برنامج العميل بايتات السلسلة النصية sentence ببساطة داخل اتصال TCP ، ثم ينتظر العميل لتلقي بايتات من الخادم.

(modified Sentence = clientSocket.recv(2048

عندما تصل الأحرف من الخادم، توضع في السلسلة modified Sentence وتتراكم حتى ينتهي السطر بالمفتاح enter، وبعد طباعة الجملة بأحرف كبيرة، نغلق مقبس العميل :

()clientSocket.close

يغلق السطر الأخير المقبس، وبالتالي يغلق اتصال TCP بين العميل والخادم، فيرسل TCP لدى العميل رسالة TCP إلى TCP لدي الخادم (انظر القسم 3-6 في الوحدة الثالثة(.

عزيزي الطالب، دعنا، الآن، نلقي نظرة على برنامج الخادم

TCPServer.py

from socket import

serverPort = 12000

serverSocket = socket(AF_INET,SOCK_STREAM(

serverSocket.bind((,serverPort)) while 1:

serverSocket.listen(1) print The server is ready to receive'

connectionSocket, addr = serverSocket.accept()

sentence = connection Socket.recv(1024) capitalizedSentence = sentence.upper()

connection Socket.send(capitalizedSentence) connection Socket.close()

هناك سطور تختلف كثيرا عن UDPServer وعن TCPClient . كما هو الحال في TCPClient، ينشئ الخادم مقبس TCP من خلال السطر الآتي:

(serverSocket-socket(AF_INET,SOCK_STREAM

على غرار UDPServer ، نربط serverPort,((serverSocket.bind)) لكن في TCP ، لدينا مقبس الترحيب ServerSocket ، وبعد إنشاء باب الترحيب هذا، سننتظر ونصت حتى

رقم منفذ الخادم serverPort بهذا المقبس

يطرق الباب عميل ما1)serverSocket.listen)ينصت

الخادم لطلبات اتصال TCP من العميل، ويحدد المعامل الحد الأقصى لعدد الاتصالات في قائمة الانتظار

على الأقل 1).

connectionSocket, addr = serverSocket.accept( ) عندما يطرق العميل هذا الباب، يستدعي البرنامج الطريقة ( ) accept للمقبس serverSocket ، وينشئ مقبسا جديدة في الخادم يدعى connectionSocket مخصص لهذا العميل بعينه. ثم يكمل العميل والخادم المصافحة، وإنشاء اتصال TCP

بين مقبس العميل clientSocket ومقبس الخادم connectionSocket ، وبعد إنشاء اتصال TCP ، عندها يستطيع العميل والخادم إرسال البايتات فيما بينهما عبر

هذا الاتصال. لا يضمن TCP وصول جميع البايتات المرسلة من جانب إلى آخر فحسب، بل يضمن وصولها بالترتيب. connection Socket.close()

بعد إرسال الجملة المعدلة إلى العميل، نغلق مقبس الاتصال، ولكن لأن مقبس الخادم serverSocket لا يزال مفتوحا، يستطيع عميل آخر أن يطرق الباب ويرسل للخادم جملة جديدة لتعديلها. بهذا ننهي المناقشة في برمجة المقابس في TCP ، ونشجعك على تشغيل البرنامجين على مضيفين منفصلين، وتعديلهما لتحقيق أهداف مختلفة قليلا عليك أيضا مقارنة برنامجي UDP ببرنامجي TCP ، لترى مدى الاختلاف بينهما، كما عليك أن تنفذ العديد من مهمات البرمجة الموصوفة في نهاية الوحدتين 2 ، 4 .

وأخيرا، وبعد أن تتقن هذه البرامج وغيرها من برامج المقابس المتقدمة، وتطور تطبيقات شبكات مشهورة لك، وتصبح غنيا ومشهورا جداً، في ذلك اليوم، أذكرنا!

مسرد المصطلحات الوحدة الثانية

(Persistent Connection)الاتصال الدائم او الثابت

الاتصال غير الدائم أو المتقطع ((Non-Persistent Connection)all all

(Security)الامن

(Throughputalayi الانتاجية

(Web caching)التخزين المخبأ للويب

DNS Caching الذاكرة المخبأة (التخزين المؤقت)

(One-click shopping)التسوق بنقرة واحدة

(Timing)التوقيت

الحركات أو المعاملات (Transaction)

(Connection-oriented Services)الخدمة الموجهة بالاتصال

Authoritative DNS servers الخوادم الموثوقة (المخولة)

الدالة الشرطية GET (The Conditional GET)

السرية (Confidentiality)

العميل او الخادم Client_server

(World Wide Web: Www الشبكة العنكبوتية العالمية

الكوكيز(Cookies)

المال أو المقبس (Socket)

المصادقة Authentication)

المصادقة أو التفويض (Authorization)

المصافحة(handshakingaae

النظير للنظير(Peer-to-Peer: P2P)

(Reliable Data Transfer)النقل الموثوق للبيانات

بت تورنت (سيل الثنائيات) BitTorrent

(Transport Control Protocol: TCP) بروتوكول التحكم بالنقل

: بروتوكول الوصول إلى البريد عبر الإنترنت) (Internet Mail Access Protocol: IMAP)

بروتوكول مخطط بيانات المستخدم (User Datagram Protocol: UDP)

برتوكول مكتب بريد الاصدار 3 (Post Office Protocol-Version 3: POP3)

(Simple Mail Transfer Protocol: SMTP برتوكول نقل البريد البسيط

برتوكول نقل الملفات(File Transfer Protocol FTP)

برتوكول نقل النص التشعبي(Hypertexts transfer protocol :http)

بنية النظير للنظير (P2P : Peer - to – Peer

تأخير الانترنت internet delay

خادم معلومات الانترنت(Internet Information Server: IIS)

خدمة النقل الموثوق للبياناتi(Reliable Data Transfer Services)

خوادم الجذرRoot DNS servers

خوادم النطاق عالي المستوىTop-Level Domain (TLD) servers

رسالة الاستجابةHTTP Response Message

رسالة الطلبHTTP Request Message

رقم المنفذ(Port number)

زمن التوزيعDistribution time))

زمن الذهاب والاياب(Round-Trip Time: RTT)

سلامة البيانات integrity

شبكات توزيع المحتوى(Content Distribution Networks: CDNS)

ضبط الازدحام (Congestion-control)

طبقة المقابس الامنة(Secure Socket Layer: SSL)

عنوان الانترنت (Internet Protocol: IP)

عنوان الانترنت ip address

قاعدة بيانات هرمية موزعةDistributed Hierarchical-Databases

مؤسسة الانترنت للأسماء والارقام المخصصة (ICANN) the Internet Corporation for Assigned Names and numbers

موقع المعلومات العالمي(Universal Resource Locator: URL)

نص الكيان entity body

نطاق اسم الانترنت من بيركلي(Berkley Internet Name Domain: BIND)

نظام اسم النطاق

3-2 خدمات طبقة النقل Transport-Layer Services

:يوفر بروتوكول طبقة النقل اتصالا منطقيا بين عمليات التطبيق التي تعمل في مضيفات مختلفة، وتعني بالاتصال المنطقي، أن المضيفات التي تشغل العمليات، من وجهة نظر التطبيق، تبدو وكأنها متصلة مباشرة، مهما كانت متباعدة ومتصلة عبر العديد من أجهزة التوجيه، وأنواع مختلفة من خطوط الاتصال أما عمليات التطبيق فتستخدم الاتصال المنطقي لتبادل الرسائل بينها دون الاكتراث بتفاصيل البنية التحتية المستخدمة لنقلها. ويوضح الشكل (3-1) ص 122مفهوم الاتصال المنطقي

تعمل بروتوكولات طبقة النقل على جانبي الإرسال والاستقبال بدلا من موجهات الشبكة، في جانب المرسل، تحول رسائل طبقة التطبيقات التي تتلقاها من عملية التطبيق المرسل إلى حزم تسمى في مصطلحات الانترنت "شرائح طبقة النقل" (transport- layer segments)وربما يتم ذلك من خلال تقطيع رسائل التطبيق إلى قطع أصغر وإضافة مقدمة طبقة النقل إلى كل قطعة لإنشاء شريحة طبقة النقل تمررها إلى طبقة الشبكة في الطرف المرسل حيث يتم تغليف الشريحة ضمن حزمة طبقة الشبكة (مخطط البيانات) وإرسالها إلى الوجهة

وفي جانب المستقبل، تستخرج طبقة الشبكة شريحة طبقة النقل من مخطط البيانات وتمررها إلى طبقة النقي، التي تعالج الشريحة المستلمة، وتتيح البيانات الموجودة فيها للتطبيق المستقبل.

3-2-1 العلاقة بين طبقتي النقل و الشبكة Relationship Between Transport and Network Layers

تذكر أن طبقة النقل تعلو طبقة الشبكة، فبينما يوفر بروتوكول طبقة النقل اتصالا منطقيا بين العمليات الي تعمل على مضيفات مختلفة، يوفر بروتوكول طبقة الشبكة اتصالا منطقيا بين المضيفات. ورغم أنه اختلاف بسيط الا انه مهم

تعمل بروتوكولات طبقة النقل في النظم النهائية، فتنقل الرسائل من عمليات التطبيق إلى طبقة الشبكة والعكس بالعكس، ولكن لا تتدخل بكيفية نقل الرسائل داخل الشبكة الأساسية. في الواقع، وكما هو مبين في الشكل (3-1)، لا تتصرف، أو تعترف، الموجهات الوسيطة باي معلومات قد تكون أضافتها طبقة النقل الى رسائل التطبيق

قد توفر شبكة الحاسوب بروتوكولات نقل عدة، يختلف كل منها في نموذج الخدمة الذي يقدمه للتطبيقات فالخدمات التي قد يوفرها بروتوكول النقل، تتقيد، غالبا، بنموذج الخدمة الخاص ببروتوكول طبقة الشبكة فإذا كان بروتوكول طبقة الشبكة لا يضمن التأخير أو عرض النطاق لشرائح طبقة النقل بين المضيفات فإن بروتوكول طبقة النقل لا يضمن زمن التأخير أو عرض النطاق الرسائل التطبيق بين العمليات مع ذلك، قد يقدم بروتوكول النقل خدمات بعينها وان لم يوفر بروتوكول الشركة خدمة مقابلة في طبقة الشبكة فمثلا، قد يقدم بروتوكول النقل خدمة موثوقة لنقل البيانات الى تطبيق معين حتى لو لم يكن بروتوكول الشبكة موثوقا (أي، يفقد أو يشوه، أو يكرر الحزم). وقد يستخدم بروتوكول النقل عملية التشفير لضمان عدم اطلاع الدخلاء على رسائل التطبيق، حتى عندما لا تستطع طبقة الشبكة ضمان سرية شرائح طبقة النقل

Overview of the Transport. Layer in the Internet نظرة عامة على طبقة النقل في الانترنت:

عزيزي الطالب، دعنا نتذكر معا أن الإنترنت، أو شبكات TCP/IP بشكل عام، تتعامل مع بروتوكولي نقل بروتوكول مخطط بيانات المستخدم UDP وهو بروتوكول غير موثوق أو مهيا للاتصال (Connectionless)، وبروتوكول التحكم بالنقل TCP وهو بروتوكول موثوق ومهيا للاتصال oriented( )connection بالتطبيق الذي يستدعيه. وعند تصميم تطبيق شبكة، على مطور التطبيق تحديد أي منهما سيوظف في هذا التطبيق، كما رأينا في القسم 2-8 في الوحدة الثانية، حيث تم اختيار UDP و TCP عند إنشاء المقابس

عزيزي الطالب، قبل البدء بالحديث عن بروتوكولي TCP UDP ، سنعرج قليلا على طبقة الشبكة في الإنترنت. يقدم بروتوكول الإنترنت Internet Protocol; TP الاتصال المنطقي بين المضيفين، فهو يبذل ما في وسعه، ولكن لا يضمن، تسليم الشرائح segments (حزم طبقة النقل(ما بينهم، لذا يطلق على هذا النموذج خدمة التسليم بأفضل سعة best-effort delivery service إذ لا يضمن تسليم الشرائح أو ترتيبها أو سلامة بياناتها، لذا يقدم بروتوكول IP خدمة غير موثوقة service،unreliable ولكل مضيف عنوان IP address واحد على الأقل في طبقة الشبكة هذا ما ستوضحه بالتفصيل في الوحدة الرابعة

تتمثل المسؤولية الأساسية لبروتوكولي UDP و TCP في التجميع وفك التجميع multiplexing and demultiplexing) على مستوى طبقة النقل، اي توسيع خدمة التوصيل التي يقدمها IP بين نظامين نهائيين (مضيفين) إلى خدمة توصيل بين عمليتين تعملان عليهما كما يوفر هذان البروتوكولان فحصا لسلامة البيانات من خلال تضمين حقول كشف الأخطاء في مقدمات الشرائح إذن، تقدم طبقة النقل خدمتين كحد ادنى؛ تسليم البيانات من عملية إلى أخرى، والتحقق من الأخطاء. وهما الخدمتان الوحيدتان اللتان يقدمهما UDP ، وكما هو الحال في IP، يقدم UDP خدمة غير موثوقة، إذ لا يضمن أن البيانات المرسلة من عملية ستصل سليمة (إن وصلت إلى العملية الوجهة سيتم مناقشة UDP بالتفصيل في القسم 3-4

من ناحية أخرى، يقدم TCP خدمات إضافية للتطبيقات، فهو أولا يوفر النقل الموثوق للبيانات reliable data transfer كما يوفر ضبط الاحتقان congestion controlفهو بالضرورة اكثر تعقيدا لذا سنغطي هذه المفاهيم بالإضافة إلى مفاهيم TCP بحد ذاته في الأقسام 53 إلى 3-8 التي تغطي المفاهيم الأساسية للنقل الموثوق للبيانات وضبط الاحتقان بشكل عام، ثم كيفية تطبيقهما في بروتوكول TCP وقبل الانتقال إلى هذه المفاهيم، دعنا نلقي نظرة في التجميع وفك التجميع في طبقة النقل في القسم التالي

/ 3-3 | التجميع وفك التجميع Multiplexing and Demultiplexing

تهدف هذه العملية إلى توسيع خدمة التسليم "مضيف الى مضيف")host-to-host(التي توفرها طبقة الشبكة، لتصبح "عملية إلى عملية")proce-to-process s(التطبيقات قيد التشغيل على المضيفات. لدى المضيف الوجهة، تتلقى طبقة النقل الشرائح من طبقة الشبكة التي تدنوها، وطبقة النقل مسؤولة عن تسليم بيانات هذه الشرائح إلى عملية التطبيق المناسبة في التشغيل في المضيف

مثال لنفرض انك تجلس أمام حاسوبك، وتعمل على تحميل صفحات ويب (جلسة HTTP) انشاء تشغل FTPوجلستي Telnet بذلك يكون لديك 4 عمليات قيد التشغيل، ماذا سيحدث؟ وكيف؟

عندما تتلقى طبقة النقل في حاسوبك بيانات من طبقة الشبكة، فإنها تعمل على توجيه البيانات الواردة الى واحدة من العمليات الأربعة. كما تعلمنا في الوحدة الثانية، القسم2 -8 . قد تحتوي العملية كجزء من تطبيق الشبكة(على مقيس واحد أو أكثر، تمر من خلالها البيانات من الشبكة إلى العملية ومن العملية الى الشبكة كما هو مبين في الشكل (2-3)، فإن طبقة النقل في المضيف المستقبل لا تقوم بالفعل بتسليم البيانات مباشرة الى العملية، بل إلى مقتبس وسيط إذ قد يكون هناك أكثر من مقيم في المضيف المستقبل ، لكل منها معرف فريد تعتمد صيغته على ما إذا كان العقد UDP أم TCP

والآن، دعنا ننظر كيف يوجه المضيف المستقبل شريحة واردة من طبقة النقل إلى المقبس المناسب لكل شريحة حقول مخصصة لهذا الغرض، فتقوم طبقة النقل، لدى المستقبل، بفحص هذه الحقول لتحديد المقبس الصحيح للمستقبل ثم توجيه الشريحة إليه، ويطلق على هذه العملية فك التجميع (demultiplexing) أما عملية التجميع (multiplexing) فهي التقاط قطع البيانات (data(chunks لدى المضيف المصدر من مقابس مختلفة، وتغليف كل قطعة من البيانات مع معلومات المقدمة لإنشاء الشرائح وتمريرها إلى طبقة الشبكة

لاحظ في الشكل (2-3)، أن طبقة النقل في المضيف الأوسط تعمل على فك تجميع الشرائح القادمة من طبقة الشبكة أدناها، إما للعملية P1 أو P2 في طبقة التطبيقات أعلاها؛ وذلك عن طريق توجيه بيانات الشرائح القادمة إلى مقبس العملية المقابلة، وعلى طبقة النقل أيضا تجميع البيانات الصادرة من هذه المقابس ومن شرائح طبقة النقل، وتمريرها إلى طبقة الشبكة

لكن كيف يتم التجميع وفك التجميع فعليا في المضيف؟ من خلال المناقشة أعلاه، نعلم أن التجميع في طبقة النقل يتطلب أن يكون (1) للمقابس معرفات فريدة، (2) لكل شريحة حقول خاصة تحدد المقبس الذي سيستلمها. هذه الحقول مبينة في الشكل)3-3(، وهي رقم منفذ المصدر source port number ورقم منفذ الوجهة) (destination port number) ويتكون رقم المنفذ من 16بت ويتراوح من 0 الى (65535

تدعي الأرقام)0 إلى 1023(أرقام المنافذ الشائعة Well-known، port numbers إذ أنها محجوزة من بروتوكولات تطبيقات معروفة، مثلاً HTTP يستخدم المنفذ 80 ، و FTP يستخدم المنفذ 21 القائمة الكاملة متوفرة في المرجع

[RFC 1700]، وتحدث على الموقع http://www.jana.org في المرجع 3232[[RFC، وعند تطوير تطبيق جديد، يجب تحديد رقم المنفذ.

3-4 | النقل بدون اتصال: بروتوكول المخطط البياني للمستخدم Connectionless Transport: UDP

في هذا القسم، سنلقي نظرة عن قرب إلى بروتوكول UDP إلى الية عمله وماذا يفعل بالضبط وهو عزيزي الطالب، استكمال لما عرض في القسم 2_2 وكذلك في القسم 2-8-1 في الوحدة الثانية كما تعلم عزيزي الطالب، فان UDF غير مهيا للاتصال دون اتصال" (connectionless)، وكما هو معرف في المرجع [RFC 768)، فهو لا يزيد عما يقوم به بروتوكول النقل، فبعيدا عن وظيفته في التجميع وفك التجميع، والتحق من بعض الأخطاء، فانه لا يضيف شيئا الي عنوان الانترنت IP فاذا اختير UDP في بناء التطبيق، فان التطبيق يتحدث مباشرة مع IP ، فيأخذ الرسائل من عملية التطبيق ويرفق بها حقلي رقم منفذ المصدر والوجهة الهدف من اجل خدمة التجميع وفك التجميع، يضيف حقلين صغيرين اخرين ويمرر الشريحة (segment) الناتجة إلى طبقة الشبكة تقوم طبقة الشبكة بتغليف شريحة طبقة النقل في مخطط بيانات IP ثم تبذل ما بوسعها لتقديم الشريحة إلى المضيف المستقبل، فإذا وصلت الشريحة الى المضيف المستقبل ، يستخدم UDP رقم منفذ الوجهة التسليم بيانات الشريحة إلى عملية التطبيق الصحيحة ولا يوجد مصافحة بين الكيان المرسل والمستقبل قبل إرسال الشريحة في بروتوكول UDP.

عزي الطالب، قد تساءل لما قد يلجأ مطورو التطبيقات الى UDP بدلا من TCP الذي يوفر خدمة نقل بيانات موثوقة الجواب هو أن العديد من التطبيقات يلائمها UDP : للأسباب الاتية

_ضبط ادق على مستوى التطبيق لما يتم إرساله من البيانات، ومتى" بمجرد أن تمرر عملية التطبيق البيانات الى UDP ، سوف يحزم البيانات داخل شريحة UDP ويمررها مباشرة إلى طبقة الشبكة بينما TCPفلديه اليه الاحتقان (congestion control) تخنق مرسل TCPفي طبقة النقل عندما يصبح خط الاتصال بين المضيفين المصدر و الوجهة مزدحما بشكل مفرط ويستمر في إعادة ارسال الشريحة إلى أن يتم الإبلاغ باستلامها من الوجهة، بغض النظر عن المدة التي يستغرقها. وبما أن تطبيقات الوقت الحقيقي غالبا تتطلب الحد الأدنى من معدل الارسال، ولا تريد ان يتأخر نقل الشريحة، وقد تسمح بفقدان بعض البيانات. فان نموذج خدمة TCP لا يلبي جيدا احتياجات هذه التطبيقات وكما هو مبين ادناه قد تستخدم هده التطبيقات UDPكجزء من التطبيق وتنفذ اي وظيفة اضافية مطلوبة خارج .UDP.

_لا داعي لإنشاء اتصال. كما سنناقش لاحقا، يستخدم TCP مصافحة ثلاثية قبل البدء بنقل البيانات. بينما UDP فينطلق دون أي تمهيد، ولا يحتمل التأخير، وربما هذا هو السبب وراء تشغيل DNS على UDP ، إذ سيكون أبطأ بكثير لو استخدم TCP

_لا تحتفظ بحالة الاتصال على عكس TCP الذي يحافظ على حالة الاتصال في الأنظمة النهائية ، وتشمل المخازن المؤقتة للاستقبال والإرسال، ومعامل ضبط الاحتقان، وغيرها من المعلومات اللازمة للنقل الموثوق للبيانات ومنع الاحتقان، فان UDP لا يحتفظ بحالة الاتصال ولا يتتبع أي من المعاملات، لذا، يستطيع خادم مخصص لتطبيق معين دعم العديد من عملائه النشطين عند تشغيل التطبيق على UDP بدلا من TCP

_كلفة مقدمة الحزمة قليلة تحتوي كل شريحة TCP على 20 بايت كلفة إضافية للمقدمة، بينما تحتوي شريحة UDP على 8 بايت فقط

الشكل (3-4) قائمة بتطبيقات الإنترنت الشائعة وبروتوكولات النقل التي تستخدمها، ومنها البريد الكتروني و الوصول عن بعد، والإنترنت، ونقل الملفات من خلال TCP لما يقدمه من خدمة موثوقة لنقل البيانات. إلا أن العديد من التطبيقات الهامة تعمل من خلال UDP ، فمثلا يستخدم UDP في بروتوكول معلومات التوجيه (Routing Information Protocol RIP)لتحديث جدول التوجيه بشكل دوري كما يستخدم UDPفي برتوكول ادارة الشبكة البسيط simple network (Management SNMP protocol) لإدارة الشبكة، ويفضل UDP في هذه الحالة، لأن تطبيقات إدارة الشبكة يجب أن تعمل احيانا عندما تكون الشبكة في حالة حرجة ويصعب الحصول على نقل موثوق للبيانات، وكما ذكرنا انفا فان DNsيعمل عبر UDP لتجنب التأخير الناجم عن إنشاء اتصال TCP

تطبيقات الانترنت الشائعة وبرتوكولات النقل الاساسية لها

Application	Application-Layer Protocol	Underlying Transport Protocol
Electronic mail	SMTP	TCP
Remote terminal access	Telnet	TCP
Web	HTTP	TCP
File transfer	FTP	TCP
Remote file server	NFS	Typically UDP
Streaming multimedia	typically proprietary	UDP or TCP
Internet telephony	typically proprietary	UDP or TCP
Network management	SNMP	Typically UDP
Routing protocol	RIP	Typically UDP
Name translation	DNS	Typically UDP

شكل 3-4: تطبيقات الإنترنت الشائعة و بروتوكولات النقل الأساسية لها

ويبين الشكل 4_3 ان كلا من UDPو TCPيستخدمان في تطبيقات الوسائط المتعددة مثل المكالمات عبر الانترنت internet phoneوالمؤتمرات المرئية video conferences في الوقت الحقيقي وتدفق (streaming الصوت والفيديو. كل هذه التطبيقات قد تسمح بفقدان الحزم بشكل طفيف، كون النقل الموثوق للبيانات غير حاسم، كما أن تطبيقات الوقت الحقيقي، كالمكالمات عبر الإنترنت والمؤتمرات المرئية ، لا تتفاعل جيدا مع آلية ضبط الاحتقان في TCP ، لذا يلجا المطورون إلى تشغيل تطبيقات الوسائط المتعددة على UDP ، مع ذلك، يتزايد استخدام TCP في تدفق الوسائط وخاصة عندما يكون معدل فقدان الحزم منخفضة، وعندما تكون حركة مرور UDP ممنوعة لأسباب أمنية

ورغم عدم إمكانية ضبط الاحتقان في UDP ، والذي يعتبر ضروريا لمنع الشبكة من الدخول في حالة اختناق نتيجة فيضان الحزم لدى الموجهات، ما يمنع وصول كثير منها من المصدر إلى الوجهة، إلا أن بالإمكان توفير نقل موثوق للبيانات عبر UDP ، من خلال تضمين ذلك في بناء التطبيق نفسه، ولكن من عيوبها إشغال مطور التطبيق لوقت أطول في تصحيح الأخطاء في التعليمات البرمجية ومعالجتها.

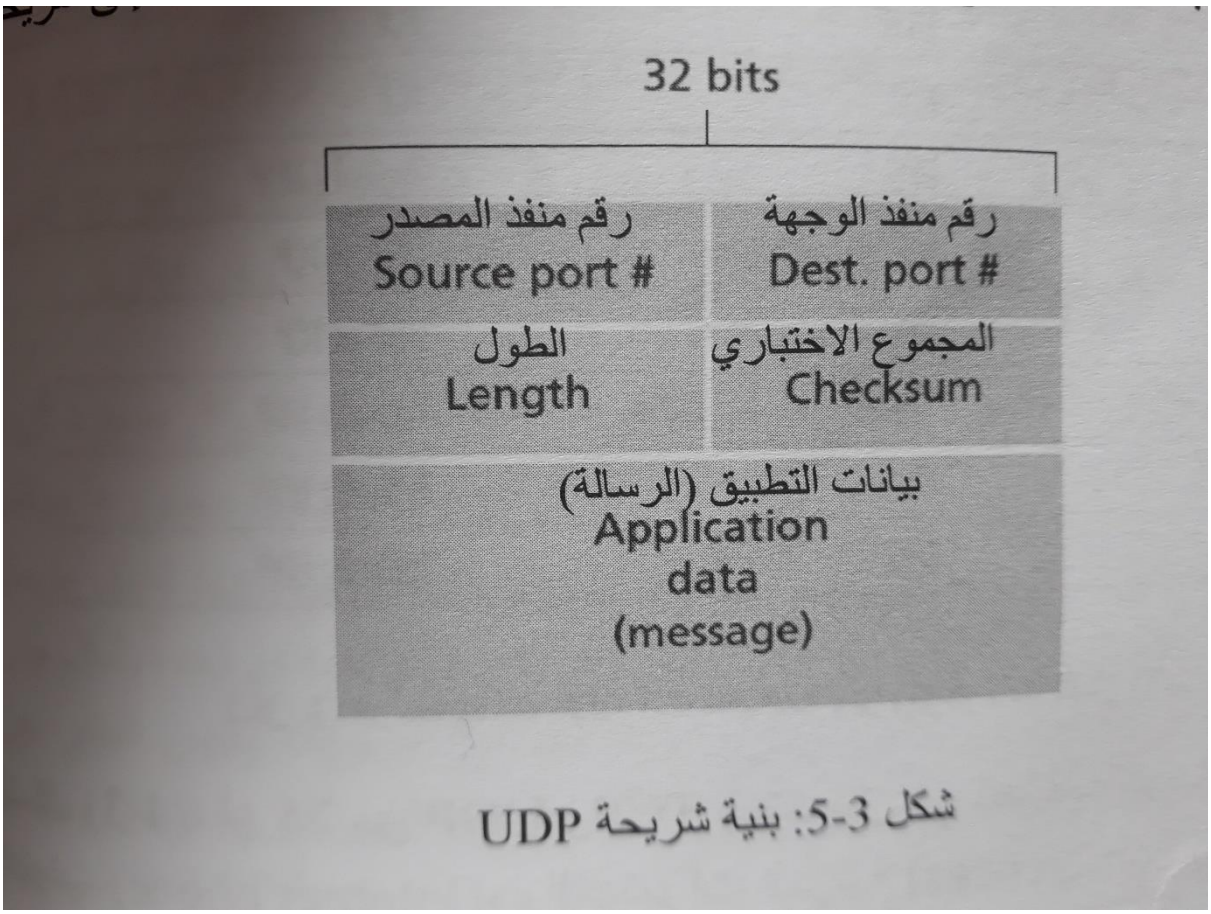
3-4-1 بنية شريحة بروتوكول مخطط بيانات المستخدم UDP Segment Structure

عزيزي الطالب، يبين الشكل (3-5) بنية شريحة UDP التي ورد تعريفها في المرجع [RFC 768]: وتحتل بيانات التطبيق "الرسالة" Application data message حقل البيانات، تتكون المقدمة من 4 حقول كل منها 2 بايت

_ حقل رقم المنفذ (# port Source(: #, Destination port يسمح بمرور البيانات إلى العملية الصحيحة قيد التشغيل لدى المضيف الوجهة (فك التجميع demultiplexing(.

-و حقل الطول (Length): يبين عدد بايتات الشريحة بما في ذلك المقدمة والبيانات .

_حقل المجموع الاختباري (Checksum(: يتحقق المستقبل من ورود الأخطاء في الشريحة في الواقع، يتم حساب هذا المجموع على بعض حقول مقدمة عنوان IP إضافة إلى شريحة UDP.



المجموع الاختباري لبروتوكول مخطط بيانات المستخدم UDP Checksum

يستخدم المجموع الاختباري للكشف عن الأخطاء، أي لتحديد ما إذا تغير أحد البتات في شريحة UDP أثناء انتقالها من المصدر إلى الوجهة (بسبب التشويش في خط الاتصال أو أثناء تخزينها في الموجه(. ويقوم UDP في جانب المرسل بإيجاد المتمم)قلب البتات) is complement لمجموع كل الكلمات ذات 16 بت في الشريحة، وأي فائض يصادف أثناء الجمع يضاف إلى الناتج، ويوضع الناتج في حقل المجموع الاختباري من الشريحة

مثال الجمع صفحة 130

3-5 أساسيات النقل الموثوق للبيانات Principles of Reliable Data Transfer |

ربما تتساءل، عزيزي الطالب، إذا كانت بروتوكولات طبقة الربط الأخرى توفر فحص الأخطاء، فلماذا ينفذ UDP المجموع الاختباري؟ والسبب هو أنه لا يوجد ضمان بأن جميع خطوط الاتصال بين المصدر والوجهة توفر فحص الأخطاء؛ أي قد تستخدم بروتوكولا لا يوفر فحص الأخطاء في طبقة الربط حتى لو تم نقل الشرائح بشكل صحيح، فقد تحدث أخطاء البنات عند تخزينها في ذاكرة الموجه ورغم أن UDP يوفر فحص الأخطاء، فإنه لا يعمل على تصحيحها، فبعض تطبيقات UDP يتجاهل الشريحة التالفة؛ وبعضها الآخر يمررها إلى التطبيق مع رسالة تحذير. سننتقل حالا إلى بروتوكول TCP الذي يوفر خدمة نقل موثوقة وغيرها من الخدمات التي لا يوفرها بروتوكول UDP. ولكن قبل الانتقال إلى مناقشة TCP ، دعنا، عزيزي الطالب، نناقش أساسيات النقل الموثوق للبيانات يعتبر النقل الموثوق للبيانات قضية جوهرية في الشبكات، وعند مناقشة TCP في القسم التالي، سنبين أنه يغطي العديد من أساسياتها. وهي قضية لا تقتصر على طبقة النقل، بل تطبق في طبقة الربط وطبقة التطبيقات في دراستنا لهذا الموضوع سنتبع الإطار المبين في الشكل (6-3).

وجود قناة موثوقة)reliable،channel(لن يكون هناك بتات تالفة)مقلوبة من 0 إلى 1 أو العكس(أو متعة في البيانات، ويتم استلامها جميعها بنفس الترتيب الذي أرسلت به. وهذا نموذج الخدمة الذي يقدمه

TCP بالتحديد لتطبيقات الإنترنت التي تستدعيه ويقع تنفيذ هذه الخدمة على عاتق بروتوكول موثوق لنقل البيانات، وتصبح المهمة صعبة إذا كانت الطبقة التي تقع أسفل بروتوكول نقل البيانات الموثوق غير موثوقة على سبيل المثال، TCP هو

بروتوكول نقل موثوق يتم تنفيذه في طبقة شبكة (IP) غير موثوقة. وبصورة أعم، الطبقة الواقعة اسفل نقطتي الاتصال الموثوقتين قد تتكون من خط اتصال منفرد (كما في حالة بروتوكول نقل البيانات على مستوى الخط(أو شبكة عالمية (كما في حالة بروتوكول على مستوى النقل(، وقد ننظر إلى هذه الطبقة السفلي ببساطة أنها قناة نقطة إلى نقطة غير موثوقة

في هذا القسم، ستقوم تدريجيا بتطوير جانبي المرسل والمستقبل لبروتوكول نقل البيانات الموثوق، مع مراعاة التدرج في نماذج أكثر تعقيدا القناة الأساسية. فمثلاً، ستراعي الآليات التي تلزم عند تلف بعض البنات أو فقدان حزم كاملة في القناة الأساسية، وستفترض هنا أن الحزم ستسلم حسب الترتيب الذي ارسلت به، مع احتمال فقدان بعض الحزم؛ أي لن تقوم القناة بإعادة ترتيب الحزم.

في هذا القسم، سنقوم تدريجيا بتطوير جانبي المرسل والمستقبل لبروتوكول نقل البيانات الموثوق، مع مراعاة التدرج في نماذج أكثر تعقيدا للقناة الأساسية. فمثلا، ستراعي الآليات التي تلزم عند تلف بعض البتات أو فقدان حزم كاملة في القناة الأساسية، وسنفترض هنا أن الحزم ستسلم حسب الترتيب الذي ارسلت

به، مع احتمال فقدان بعض الحزم؛ أي لن تقوم القناة بإعادة ترتيب الحزم بين الشكل (3-6 - b) واجهات بروتوكول نقل البيانات وسيتم استدعاء جانب المرسل من الأعلى عن طريق استدعاء ( ) rdt_send ، التي ستمرر البيانات لتسليمها للطبقة العليا لدى جانب المستقبل المختصر الى rdt يعنى reliable،data transfer أما المختصر send يعني أنه يتم استدعاء جانب المرسل وفي جانب المستقبل يتم استدعاء rcv_((rdt عندما تصل حزمة من جانب المستقبل ل لقناة. وعند رغبة البروتوكول بتسليم البيانات إلى الطبقة الأعلى يستدعي ( ) deliver_data ، وهنا استخدمنا مصطلح حزمة packet بدلا من شريحة segment لأننا نتحدث بشكل عام.

وفي حالتنا اعتبرنا نقل البيانات أحادي الاتجاه unidirectional data transfer ل لتبسيط أي من جانب المرسل إلى جانب المستقبل، ومع ذلك، نلاحظ أن جانبي المرسل والمستقبل بحاجة إلى إرسال الحزم في كلا الاتجاهين، كما هو مبين في الشكل)3-6(. سترى بعد قليل أنه يتم تبادل حزم البيانات والتحكم على حد سواء بين جانبي المرسل والمستقبل، وكلاهما يرسل إلى الجانب الأخر باستدعاء الدالة ( ) udt_send والمختصر Udt unreliable data transfer.

3-5-1Building a Reliable Data Transfer Protocol بناء برتوكول موثوق لنقل البيانات

سنتطرق الآن إلى سلسلة من البروتوكولات التي تتدرج في التعقيد حتى الوصول إلى بروتوكول موثوق النقل البيانات. سنبدأ بأبسط حالة عندما تكون القناة موثوقة تماما، فالبروتوكول الذي يمثل هذه الحالة بسيط يطلق عليه rdt1.0، ويبين الشكل)3-7(تعريفات آلة الحالات المنتهية (FSM) linite-state machine للمرسل والمستقبل، لكل منهما آلة منفصلة وحالة واحدة فقط

تشير الأسهم إلى انتقال البروتوكول من حالة إلى أخرى، وفي حالتنا ينتقل السهم)يخرج ويعود(إلى نفس الحالة، كونها الحالة الوحيدة، ويظهر الحدث الذي يسبب الانتقال فوق الخط الأفقي، بينما تظهر الإجراءات المتخذة عند حصوله تحت الخط. وسنستخدم الرمز ٨ (تحت الخط أو فوقه(تباعا للتعبير الصريح عن غياب الإجراء أو الحدث على التوالي. أما السهم المنقط في بين الحالة الابتدائية

يقبل جانب المرسل البيانات من الطبقة العليا عبر الحدث (rdt_senddata)وينشئ حزمة تحتوي هذه البيانات عبر الإجراء)make_pkt(data، ويرسلها عبر القناة. بينما يستقبل جانب المستقبل الحزمة من القناة عبر الحدث)rdt_rcv(packet، ويستخرج البيانات من الحزمة عبر الإجراء extract ,(packet)data، ويمر ر البيانات إلى الطبقة العليا عبر الإجراء data(data(.deliver في هذا البروتوكول البسيط لا فرق بين وحدة البيانات والحزمة، وليس هناك حاجة للتغذية الراجعة من جانب المستقبل إلى المرسل لأن القناة موثوقة تماما ولن يحدث خطا وهنا افترضنا أن المستقبل قادر على النقل البيانات من المرسل فور إرسالها وبنفس السرعة، أي لا حاجة بان يطلب المستقبل من المرسل أن يقلل سرعة الإرسال لكن، قد يظهر خطا في إحدى بنات الحزمة، فلا بد من إيجاد بروتوكولات تعالج حالة الخطأ، أطلقنا عليها rdt2.1, rdt2.2،rdt2.0, وقد تكون القناة نفسها غير موثوقة تماما، أي قد تضيع بعض البتات في القناة وقد أطلق على البروتوكول الذي يعالج هذه المسالة rdt3.0. في مثل هذه الحالة، تظهر الحاجة إلى إشعارات ايجابية من المستقبل إلى المرسل positive acknowledgments (ACK)(بالاستلام دون خطا، أو negative acknowledgments illay asic !Jyl isle (Ja Ka!AN وتعرف البروتوكولات التي تستند إلى إعادة الإرسال ببروتوكولات طلب الإعادة الآلي ARQ Automatic Repeat reQuest). وفي هذه الحالة سنحتاج إلى ثلاث وظائف إضافية لمعالجة أخطاء البت:

__اكتشاف الخطأ: يحتاج المستقبل إلى آلية لكشف الخطأ في البتات المستقبلة، وفي هذه الحالة سيحتاج المرسل إلى بتات إضافية غير البنات الأصلية المرسلة ليرسلها إلى المستقبل، ويتم تجميع هذه البنات في حقل المجموع الأختباري لحزمة البيانات. سنتطرق بالتفصيل إلى طرق كشف الخطأ وتصحيحه في الوحدة الخامسة.

2. إشعار من المستقبل: نظرا لتباعد المرسل والمستقبل، فإن الطريقة الوحيدة هي إشعار استلام صريح إيجابي (ACK) أو سلبي (NAK) من المستقبل إلى المرسل بأن الحزمة وصلت صحيحة أو فيها خطا، وقد يكون الإشعار حزمة طولها بت واحد (1 أو 0)

3- إعادة الإرسال: يقوم المرسل بإعادة إرسال الحزمة التي أشعره المستقبل أنها تحتوي على خطا كما أشرنا سابقاً، فإن البروتوكول الذي يعالج هذه المسائل هو rdt2.0 وإصداره اللاحقين , redt21 rdu ، وهنا سنغطي الإصدار الأخير rdt2.2 بالتفصيل، مع عرض ميزاته مقارنة بإصداريه السابقين بوقف rdt2.0 كشف الأخطاء والإشعارات الإيجابية والسلبية من المستقبل، ففي هذا البرتوكول وأمثاله فان المرسل لن يرسل أي بيانات جديدة حتى يتأكد أن المستقبل قد استلم الحزمة الحالية، ولهذا تعرف بروتوكول التوقف والانتظار)stop-and-wait(. ومن عيوب بروتوكول rdt2.0 الفادحة أنه لا يعلج احتمالية تلف حزمة الإشعار الإيجابي أو السلبي، لذا فإنه بحاجة إلى إضافة بنات المجموع الاختباري إلى فرم ARNAK لكن هذه الأخطاء

وتكمن الصعوبة، في حال تلف حزمة ACK/NAK، في أن المرسل لن يتمكن من التأكد من استلام المستقبل لأخر حزمة بيانات سواء كانت صحيحة أم لا، وهناك ثلاثة احتمالات لمعالجة الإشعارات التالفة 1

- يرسل المرسل إشعارا يطلب فيه من المستقبل إعادة إرسال ACK/NAK، فيقوم المستقبل بذلك، ولكن ماذا إذا تلف إشعار المرسل، عندها لن يكون لدى المستقبل فكرة فيما إذا كانت هذه حزمة بيانات جديدة أم إشعارا بعدم وصول حزمة ACK/NAK، وتستمر العملية على نحو أكثر تعقيدا 2

- إضافة بتات كافية للمجموع الاختياري بشكل يتيح للمرسل تصحيح الخطأ عند اكتشافه، وهو حل مناسب للقنوات التي قد تتلف البنات ولكن لا تفقدها.

3- أن يقوم المرسل ببساطة بإعادة إرسال حزمة البيانات الحالية عند استلامه إشعارات ACK/NAK تالفة، ولكن ينتج عن هذا السيناريو حزم متكررة في قناة المرسل إلى المستقبل، وتكمن الصعوبة هنا بان المستقبل لن يعرف فيما إذا كان إشعار ACK/NAK قد وصل المرسل، وبالتالي فيما إذا كانت الحزمة المرسلة هي حزمة بيانات جديدة أم إعادة إرسال.

أبسط حل للمشكلة الأخيرة هو إضافة حقل جديد لحزمة البيانات يمثل الرقم التسلسلي لدى المرسل، وعلى المستقبل فحص هذا الرقم ليتبين أنها حزمة جديدة أم معادة، وفي هذه الحالة قد يكون طول الحقل إبت كافيا. وكون القناة لا تفقد الحزم، فلا داعي لرقم تسلسلي لإشعارات ACK/NAK، حيث يعرف المرسل أن هذه الإشعارات هي استجابة لأخر حزمة بيانات أرسلها. وقد عالج الإصدار الجديد rdt2.1 هذه المشكلة أما الإصدار الأخير rdt2.2 فقد تخلص من الإشعار السلبي NAK ، ويبين الشكل (3-8) والشكل (9-3)

آلة الحالات المنتهية (FSM) لجانبي المرسل والمستقبل من بروتوكول rdt2.2 على التوالي.

إن عدد الحالات لدى المرسل والمستقبل تضاعف عما هو في rdt2.0 ، فأصبح لدى المرسل 4 حالات ولدي المستقبل حالتان، وذلك لتعكس فيما إذا كانت الحزمة المرسلة حاليا أو التي يتوقعها المستقبل لها رقم تسلسلي 0 أو 1. وعلى عكس rdt2.1 الذي يستخدم فيه المرسل إشعارا إيجابيا ACK عند استلام حزمة سليمة وإشعارا سلبيا NAK عند استلام حزمة تالفة، يحقق rdt2.2 نفس تأثير الإشعار السلبي NAK ، فبدلا من إرسال NAK يرسل ACK لأخر حزمة مستلمة سليمة، والمرسل الذي يستقبل إشعاري ACK لنفس الحزمة (اي ACK متكررة)، يعلم بأن المستقبل لم يستلم أي حزمة سليمة بعد الحزمة التي أشعرت مرتين ويتضمن المستقبل الرقم التسلسلي للحزمة التي سيتم إرسال إشعارها برسالة ACK ، وذلك بتضمين 0 , ACK أو 1 , ACK في الإجراء ()mnake_pkt في FSM المستقبل، وعلى المرسل الأن فحص الرقم التسلسلي للحزمة قيد الإشعار عبر رسالة ACK مستلمة، وذلك بتضمين 0 أو 1 في الحدث isACK0 في FSM المرسل أما البروتوكول الأخير rdt3.0 فيفترض أن القناة قد تفقد الحزم إضافة إلى تلفها (البتات الخاطئة)، و طبيعي في شبكات اليوم بما في ذلك الإنترنت. في هذه الحالة يظهر مسألتان؛ كيف يتم كشف فقدان الحزمة؟ وما العمل عندما يحدث ذلك؟ وللإجابة على ذلك، هناك عدة طرق محتملة لمعالجة فقدان الحزم

فلنفترض أن المرسل أرسل حزمة بيانات، وفقدت هذه الحزمة ذاتها أو إشعار المستقبل ACK ، في كلا الحالتين لن تصل المرسل أي استجابة من المستقبل، فما الوقت اللازم للتأكد أن الحزمة أو إشعار ها فقدت في الواقع لا يمكن الانتظار كثيرة، إذ على المرسل وضع قيمة محددة لوقت الانتظار للحكم بان الحزمة فقدت، فإذا لم يصل إشعار ACK خلال هذا الوقت يعيد ارسالها. ولكن قد تصل الحزمة بعد إعادة الإرسال فيصبح في القناة حزم بيانات متكررة، ويكمن الحل في الأرقام التسلسلية. وتحتاج إلى مؤقت تنازلي ينبه المرسل عند نفاذ الوقت، وعلى المرسل أن يكون قادرا على (1) تشغيل المؤقت عند إرسال كل حزمة

لأول مرة أو إعادة إرسالها(،)2(الاستجابة للتنبيه واتخاذ إجراء محدد، (3) وقف المؤقت يبين الشكل (103) آلة الحالات المنتهية FSM للمرسل في بروتوكول rdt3.0، ويبين الشكل (3-1) كيف يعمل دون فقدان الحزمة أو تأخيرها، وكيف يعالج الحزم المفقودة. يتقدم الزمن من أعلى المخطط إلى أسفله، ومن الطبيعي أن يتأخر زمن الاستقبال عن زمن الإرسال نتيجة للتأخير في الإرسال والبث، في الشكل (3-1) (b) إلى (d)، تشير الأقواس في جانب المرسل إلى زمني تشغيل المؤقت ونفاذة، ولأن الأرقام التسلسلية ت تناوب بين 0 و 1، يسمى بروتوكول البت المتناوب alternating-bit

عزيزي الطالب، بهذا نكون قد غطينا مجموعة من العناصر الأساسية لبروتوكول نقل البيانات الموثوق المجاميع الاختيارية، والأرقام التسلسلية، والمؤقتات، وحزم الإشعار الإيجابي والسلبي، وجميعها تلعب دورا أساسيا في آلية عمل البروتوكول، والآن أصبح لدينا برتوكول موثوق يعمل على نقل بيانات..

3-5-2 خط انتاج بروتوكولات نقل البيانات الموثوقة Pipelined Reliable Data Transfer Protocols

رغم ان بروتوكول نقل البيانات الموثوقة صحيح الا ان اداءه ضعيف وخاصة في شبكات اليوم عالية السرعة ومن اهم مشاكله انه بروتوكول توقف وانتظر

مثال ص 138

معدل استغلال المرسل او القناة هو نسبة انشغال المرسل في ارسال البتات عبر القناة يبين التحليل في الشكل (A-13-3) ان معدل استغلال المرسل في بروتكول توقف وانتظر يعبر عنه بالمعادلة التالية

U SENDER=(L/R)/(RTT+L/R)

شكل 11- 3 صفحة 139 الية عمل RTD3.0 برتوكول البت المتناوب

شكل 12-3 صفحة -140-139 يمثل بروتكول توقف-وانتظر مقارنو ببرتوكول خط الانتاج

شكل 13-3 ص 140 يمثل الية عمل بروتكول خط الانتاج

يستطيع المرسل ارسال عدة حزم دون انتظار اشعار الوصول كما يوضح الشكل 3-13 صفحة 140 فإذا اتيح للمرسل ارسال ثالث حزم دون انتظار سيزيد استغلال المرسل ثالث مرات ويبدو ذلك كما يتم ملء خط انتاج وتسمى هذه التقنية بـ خط الإنتاج Pipelining**أما تبعات استخدامها في بروتوكولات نقل البيانات الموثوقة فهي:**

-يجب زيادة عدد ارقام التسلسل ، لكل حزمة قيد الإرسال عدا عند الإرسال رقم متسلسل فريد وقد نكون هناك رزم متعددة بدون اشعار -قد يتعين على طرفي المرسل والمستقبل التخزين المؤقت الكثر من حزمة واحدة , بالحد الأدنى على المرسل ان يخزن الحزم التي ارسلت ولم يتم اصدار أي اشعار لها حتى الآن وقد يلزم التخزين المؤقت للحزم المستقبلة بشكل صحيح لدى المستقبل -يتوقف مدى ارقام التسلسل اللازمة والتخزين المؤقت على الطريقة التي يستجيب بها البروتوكول نقل البيانات للحزم المفقودة والتالفة والمتأخرة ز وهناك نهجان اساسيان لتصحيح الخطأ في خط الإنتاج - : العودة ال ن - (GO –back- N) والتكرار الانتقالي (SR)(SELECTIVE REPEAT)

3-5-3 العودة -ن(GO- Back-N)

يسمح للمرسل بإرسال عدة حزم دون انتظار اشعار الوصول مع التقييد بحد اعلى لعدد الحزم التي لم يصل اشعارها N في خط الإنتاج (يبين الشكل)3-14) مدى الأرقام التسلسلية في بروتوكول GBN من منظور المرسل حيث يمثل base الرقم التسلسلي لأقدم حزمه بدون اشعار Nextseqnum اصغر رقم تسلسلي غير مستخدم يستخدم لترقيم الحزمة اللاحقة هناك 4 فترات في مدى الأرقام التسلسلية الفترة [0, base- 1] تشمل الحزم المرسلة التي وصل اشعارها الفترة

[base, Nextseqnum - 1] تشمل الحزم المرسلة التي لم يصدر اشعارها والفترة (base, nextseqnum+N- 1) تشمل الحزم قيد الإرسال حيث ان الأرقام التسلسلية اكبر او يساوي base+ n ال يمكن استخدامها حتى تصل اشعارات الحزم الحالية في خط الإنتاج وخاصة الحزمة ذات الرقم التسلسلي base

شكل 14-3ص141 برتوكول العودة GBN:GO-BACK-N الارقام التسلسلية من منظور المرسل

يشار الى N على انها **حجم النافذة windows size** حيث تم تحديد الحجم ولم يترك مفتوحا لسببين السبب الأول لضبط التدفق flow control لفرض هذا الحد على المرسل كما سنبين في القسم 3-6 وكذلك لضبط الاحتقان في TCP كما سنوضح في القسم 3-8

في الواقع بوضع الرقم التسلسلي في حقل ثابت الطول في مقدمة الحزمة، فإذا كان k يمثل عد د البتات في الحقل، يكون مدى الأرقام التسلسلية [0 ,2K-1]، وتحسب باستخدام modulo 2K (أي الرقم الذي يلي الرقم 2K-1 هو 0)

يبين الشكل15-3 آلية عمل بروتوكول النافذة المنزلقة GBN بنافذة حجمها 4 حزم فالمرسل يرسل 0-3 وعليه انتظار وصول إشعار لحزمة أو أكثر قبل استنفاف العمل. وعن استلام كل إشعارين متتاليين (مثلا ACK0, ACK1) تنزلق النافذة قدما ويستطيع المرسل نقل حزمة جديدة(pkt3, pkt4)على التوالي، فإذا فقد المستقبل الحزمة 2 ، تعتبر الحزم 3 ، 4 ، 5 خارج الخدمة وتهمل.

لاحظ عزيزي الطالب، أن بروتوكول GBN يتضمن تقريبا كل التقنيات التي ستتعرض إليها عند دراسة مكونات نقل البيانات الموثوق في بروتوكول TCP في القسم 3-6 ، وتشمل الأرقام التسلسلية، والإشعارات التراكمية، والمجاميع الاختبارية، وانتهاء المهلة إعادة الإرسال.

شكل 15-3 ص 143 الية عمل برتوكول العودة (GBN:GO- Back-N)

3-5-4 اعادة الإرسال الانتقائية selective Repeat (SR)

يسمح برتوكول GBN للمرسل بتعبئة خط الإنتاج بالحزم ، وبالتالي تجنب مشاكل استخدام القناة التي ظهرت في بروتوكولات التوقف والانتظار مع ذلك هناك سيناريوهات يعاني فيها GBN نفسه من مشاكل في الأداء وخاصة عندما يكون حجم النافذة وتأخير عرض النطاق الترددي كالهما كبير ، قد يكون العديد من الحزم في خط الإنتاج وبالتالي قد يؤدي خطأ في حزمة واحدة إلى إعادة إرسال عدد كبير من الحزم ،كثير منها غير ضروري ،وإذا زاد احتمال الخطأ في القناة ،قد يصبح خط الإنتاج مليئاً بالحزم المعادة غير الضرورية.

كما يوحي الاسم فإن بروتوكولات الإعادة الانتقائية تتجنب إعادة الإرسال غير الضرورية بل يعيد المرسل إرسال تلك الحزم التي يشتبه بورود خطأ فيها فقط (أي فقدت او تلفت) لدى المستقبل .وستتطلب إعادة الإرسال ، عند الحاجة أن يشعر المستقبل المرسل بالحزم السليمة المستلمة بشكل فردي (انتقائي) ويستخدم حجم النافذة N لحصر أرقام الحزم العالقة دون إشعار في خط الإنتاج

مع ذلك وعلى عكس GBN فإن المرسل قد تلقى بالفعل إشعارات ACKs لبعض الحزم في النافذة ، ويبين الشكل (3_16)صفحة 145 فضاء الأرقام التسلسلية لبروتوكولSR

يصدر المستقبل إشعارات للحزم السليمة المستلمة سواء كانت بالترتيب أم ال ،فالحزم خارج الترتيب تخزن مؤقتاً حتى استلام أي حزم مفقودة) ذات رقم تسلسلي أصغر(عندها قد تسلم رزمة من الحزم بالترتيب للطبقة الأعلى .يبين الشكل (3_17 صفحة 147مثالا على آلية عمل بروتوكول SR بوجود حزم مفقودة ،حيث يعمل المستقبل مبدئياً على تخزين الحزم 3,4,5 ويسلمها مجتمعة مع الحزمة 2 عند استقبالها في نهاية المطاف إلى الطبقة الأعلى.

إن غياب التزامن بين نافذتي المرسل والمستقبل قد تكون له تبعات مهمة في ظل المدى المحدود للأرقام التسلسلية . فماذا سيحدث مثالً إذا كان المدى 4أرقام تسلسلية للحزم 0,1,2,3 وحجم النافذة 3 ?افرض أن الحزم 0,1,2قد أرسلت واستلمت لدى المستقبل وأشعر المرسل بوصولها ، في هذه اللحظة تغطي نافذة المستقبل الحزم 4,5,6 ذات الأرقام التسلسلية التوالي على 1,0,3

السيناريو الأول : كما يبين الشكل(3_18 _a)صفحة 149 فقدت إشعارات ACK الحزم الثالثة الأولى والمرسل يعيد إرسال هذه الحزم ، وعليه يستلم المستقبل الحزمة ذات الرقم التسلسلي 0 أي نسخة من الحزمة الأولى المرسلة

السيناريو الثاني :كما يبين الشكل(b_ 18_3)صفحة 149 إشعارات ACK الحزم الثالثة الأولى وصلت سليمة فيحرك المرسل نافذته قدماً ويرسل الحزم 4,5,6 ذات الأرقام التسلسلية 3,0,1 على التوالي . الحزمة ذات الرقم التسلسلي 3 فقدت ،ولكن الحزمة ذات الرقم التسلسلي 0 وصلت وهي حزمة تحتوي بيانات جديدة

من منظور المستقبل كونه ال يرى الإجراءات التي يتخذها المرسل ،كل ما يراه هو تسلسل الرسائل التي يستقبلها من الفتاة أو يرسلها ، فالسيناريو الأول والثاني متشابهان . وليس هناك طريقة للتمييز بين إعادة إرسال الحزمة الأولي وبين إرسال الحزمة الخامسة للمرة الأولي . من الواضح أن حجم النافذة الذي يقل بواحد عن مدى الأرقام التسلسلية لن يعمل .ولكن ما الحد الأدنى لحجم النافذة؟ أثبت أن حجم النافذة في بروتوكول SR يجب أن يكون أقل أو يساوي نصف مدى الأرقام التسلسلية.

3-6 | النقل الموجه بالاتصال: بروتوكول التحكم بالنقل

Connection-Oriented Transport: TCP

عزيزي الطالب، بعد أن غطينا المبادئ الأساسية للنقل الموثوق للبيانات، سننتقل الآن إلى بروتوكول التك بالنقل؛ بروتوكول طبقة النقل في الإنترنت، الموجه بالاتصال، والموثوق. في هذا القسم، سترى أن هنا البروتوكول يعتمد على العديد من المبادئ الأساسية التي نوقشت في القسم السابق، بما في ذلك الكشف ع الأخطاء، وإعادة الإرسال، والإشعارات التراكمية، والمؤقتات، وحقول المقدمة الأرقام التسلسل والإشعارات. يتوفر تعريف TCP في المراجع ,

RFC 2581, RFC 793, RFC 1122, RFC 1323, RFC 2018

3-6-1 الاتصال عبر بروتوكول التحكم بالنقل TCP Connection

كما تعلم، عزيزي الطالب، فإن بروتوكول TCP موجه بالاتصال، أي أن العمليتان تتصافحان (اي عليه إرسال بعض الشرائح الأولية لإنشاء معاملات نقل البيانات لاحقا قبل بدء الإرسال، وعند إنشاء الاتصال كلا الجانبين يعدان متغيرات عدة حول حالة TCP. يعمل هذا البروتوكول في الأجهزة الطرفية (end systems) لا الأجهزة البينية (كالموجهات والمقسمات التي لا تحتفظ بحالة الاتصال عبر TCP) فهي تعنى بمخططات البيانات لا بالاتصالات. يوفر الاتصال عبر TCP خدمة الإرسال باتجاهين معا (full- duplex), وهو دائما اتصال نقطة إلى نقطة (point-to-point), أي بين مرسل وحيد ومستقبل وحيد.

وللتعرف على آلية إنشاء الاتصال، لنفرض أن عملية ما على أحد المضيفين)عميل(تريد بدء الاتصال بعملية على مضيف آخر)خادم(، فتبلغ العملية طبقة النقل لدى العميل بأنها تنوي إنشاء اتصال بعملية الخادم، وفي مثالنا في الوحدة الثانية استخدمنا الأمر

clientSocket.connect((serverName,serverPort))

فيبدأ TCP على العميل بإنشاء اتصال مع TCP على الخادم، فيرسل العميل أو شريحة TCP خاصة فيستجيب الخادم بشريحة خاصة ثانية، ثم يستجيب العميل بشريحة ثالثة؛ الأولى والثانية لا تحملان بيانات، أما الثالثة فقد تحمل البيانات. ولأنه تم إرسال 3 شرائح بين مضيفين، يطلق على هذا الإجراء المصافحة الثلاثية (three-way handshaking)

شكل 18-3 معضلة النوافذ الكبيرة جدا :حزمة جديدة او اعادة ارسال

وفور إنشاء الاتصال، يمكن لهما تبادل البيانات. فلو أرسل العميل، فإن العملية تمرر فيضا من البيانات مقبس (باب) العميل لتصبح في قبضة TCP قيد التشغيل لدى العميل.

يبين الشكل (19-3) أن TCP يوجه البيانات نحو مخزن الإرسال المؤقت (send buffer) وهواك المخازن التي أعدت أثناء المصافحة الثلاثية، وبين وقت وآخر ينتزع TCP قطعاً من البيانات من المخزن ليمرر شريحة إلى طبقة الشبكة. ويحدد الحد الأعلى لحجم البيانات في الشريحة بقيمة حجم الشريحة الأقصى

(maximum segment size: MSS) وتحسب بإيجاد طول أكبر إطار في طبقة الربط (link - layer frame) يستطيع المضيف المحلي المرسل إرساله ، ويطلق عليها وحدة الإرسال القصوى (maximum transmission unit:MTU) ومن ثم ضبط MSS للتأكد أن الشريحة المغلقة في مخطط بيانات ، بروتوكول الانترنت بالإضافة الى طول مقدمة TCP/IP البالغة

(40 Bytes) تناسب حجم اطار واحد لطبقة الربط والذي يبلغ (1500 Bytes) في كل من الإيثرنت وبروتوكول نقطة لنقطة (Ethernet and ppp) لاحظ أن MSS هو الحد الأعلى لبيانات طبقة التطبيقات في الشريحة وليس الحد الأقصى لحجم شريحة TCP بما في ذلك المقدمة.

يقوم TCP بمزاوجة كل قطعة من بيانات العميل بالمقدمة لتشكيل شريحة TCP . ثم تنتقل الشرائح إلى أسفل نحو طبقة الشبكة حيث تغلف كل منها على حدة ضمن مخططات بيانات بروتوكول الإنترنت في طبقة الشبكة ، وترسل هذه المخططات عبر الشبكة . وعندما يستقبل TCP شريحة في الطرف الآخر ، توضع بياناتها في مخزن الاستقبال المؤقت (receive buffer) ، وهناك يقرأ التطبيق تنفق البيانات من المخزن.

3-6-2 بنية شريحة بروتوكول التحكم بالنقل TCP Segment Structure

تتكون بنية شريحة بروتوكول التحكم بالنقل (TCP Segment Structure) من حقول المقدمة وحل البيانات ، وكما ذكرنا أعلاه ، فإن MSS تحدد الحجم الأقصى لحقل البيانات . عندما يرسل ملف كبير (مثلاً صورة في صفحة ويب) ، يقطع TCP الملف إلى قطع (chunks) حجمها MSS ، ما عدا القطعة الأخيرة التي قد تقل عن ذلك التطبيقات التفاعلية ، غالبا ، ترسل قطع بيانات أصغر من MSS ، ففي برنامج Telnet مثلا ، شريحة TCP غالبا بايت واحد فقط ، ومقدمة 20TCP بايت ، وبذلك قد يكون طول الشريحة التي يرسلها Telnet هي 21 بايت فقط . وتشمل المقدمة رقمي منفذ المصدر والوجهة اللذين يستخدمان في تجميع / فك تجميع البيانات من وإلى تطبيقات الطبقة العليا ، وحقل مجموع الاختبار (checksum) .

ضافة إلى ما ذكرنا أعلاه ، تشمل المقدمة الحقول الآتية:

*حقل رقم التسلسل (sequence number) وحقل رقم الإشعار (acknowledge number) ،وكل منهما 32 - بت ، ويستخدمان لتوفير خدمة النقل الموثوق للبيانات في المرسل والمستقبل .

*حقل نافذة الاستقبال (receive window) وطوله 16 - بت ، ويستخدم في ضبط التدفق (flow control) ، أي عدد البايتات التي يقبل بها المستقبل .

*حقل طول المقدمة (header length) وطوله 4 - بت ، ويحدد طول مقدمة TCP في كلمات 32بت ، وقد يتغير طول المقدمة تبعا لحقل الخيارات (TCP options) ، فإذا كان هذا الحقل فارغا ،فإن طول المقدمة القياسي 20 بايت .

. حقل الخيارات (options) ، وهو اختياري متغير الطول ، ويستخدم في التفاوض بين المرسل والمستقبل حول حجم الشريحة الأقصى (MSS) ، أو كعامل تحديد حجم النافذة في الشبكات عالية السرعة . كما يتم تحديد خيار خاتم الوقت (time stamping) .

*حقل الراية (flag) وطوله 6 - بت ، بت ACK يبين صلاحية حقل الإشعار ، أي أن في الشريحة إشعار بوصول شريحة أخرى بنجاح . أما بتات RST، SYN , FIN فتستخدم في إعداد الاتصال وإنهائه . بت PSH يشير أن على المستقبل تمرير البيانات فورا إلى الطبقة الأعلى وأخيرا بت URG، فيشير أن هناك بيانات في هذه الشريحة وسمت من جانب المرسل بأنها طارئة .

3-6-3 تخمين زمن الذهاب والإياب ونفاد المهلة

Round -TRIP TIME(RTT) Estimation and Timeout

يستخدم بروتوكول TCP نفاد المهلة (timeout) / إعادة الإرسال (retransmit) لمعالجة فقدان الشرائح ،وعلى الرغم أنه مفهوم بسيط ، إلا أن هناك بعض العقبات عند تطبيقه في بروتوكول حقيقي مثل TCPالسؤال الأهم يتعلق بطول المهلة ، ومن الواضح أنها أقل من زمن الذهاب والإياب Round -TRIP TIME(RTT) وهو الزمن من لحظة إرسال الشريحة إلى الإشعار بوصولها . فكيف يمكن تخمين هذه المهلة ؟ وهل يلزم مؤقت لكل شريحة بلا إشعار ؟

في البداية ، سننظر في كيفية تخمين زمن الذهاب والإياب بين المرسل والمستقبل ، سنسمي العينة(SampleRTT) ، وهي الزمن بين إرسال الشريحة (segment) و استقبال الإشعار ، وبدلا من قياس عينة لكل شريحة مرسلة ، تقاس عينة واحدة في المرة الوحدة (أي في نقطة زمنية محددة خلال زمن الذهاب والإياب تقاس عينة ما زالت بلا اشعار حاليا) ، فنحصل على قيمة جديدة للعينة مرة كل RTT تقريبا لاحظ ان TCP لا يقيس باي حال عينة لشريحة أثناء إعادة الإرسال. تتباين قيم العينة

من شريحة إلى أخرى بسبب الاحتقان في الموجهات وتغير الحمل في الأنظمة النهائية ، لذا يتم احتساب متوسط العينات المخمنة(EstimatedRTT) لتخمين RTT القياسية حسب المعادلة التالية

EstimatedRTT = (1 - @) • Estimated RTT + @ • SampleRTT

فيعتمد على الوسط المرجح لقيمة EstimatedRTT السابقة والقيمة الجديدة للعينة SampleRTT ويوصى المرجع [RFC 6298] بقيمة (a=0.125) فتصبح المعادلة

EstimatedRTT=0.875 • SampleRTT • 0 .125 + EstimatedRTT

و يطلق عليه إحصائيا الوسط الأسي المرجح المتنقل (Exponential Weighted Moving Average EWMA) ويبين الشكل 3-21 منحنى هذه المعادلة لاتصال TCP بين gaia.cs.umass.edu في الولايات المتحدة fantasia.eurecom.fr في جنوب فرنسا

ولقياس التغير في RTT ، نحسب انحراف عينات زمن الذهاب والإياب عن تلك المخمنة (DevRTT)

كما يأتي :

DevRTT=(1 - B).DevRTT +B .| SampleRTT – EstimatedRTT|

فكلما زاد التذبذب في قيم العينات يكون الانحراف أكبر ، ويوصي بقيمة (B = 0.25) .

نفاد مهلة إعادة الارسال : ضبطها وإدارتها setting and Managing the Retransmission Timeout Interval

في ضوء قياس القيم التي أشرنا إليها أعلاه (EstimatedRTT and DevRTT) ، ما قيمة فترة نفاد المهلة في TCP؟ يجب أن يكون أكبر أو يساوي EstimatedRTT . والا ستظهر حالات إعادة إرسال غير ضرورية ولكنه لا ينبغي أن يكون أكبر بكثير من EstimatedRTT والا لن يقوم بإعادة إرسال الشريحة المفقودة بسرعة ، مما يؤدي إلى تأخير كبير في نقل البيانات . لذلك يفضل تحديد مهلة تساوي EstimatedRTT إضافة إلى هامش بسيط ، ويزيد الهامش كلما زاد التذبذب في قيم SampleRTT TCP ويظهر دور الانحراف DevRTT هنا . وتراعى جميع هذه الاعتبارات في طريقة تحديد مهلة إعادة الإرسال ، كما يأتي :

Timeoutinterval = EstimatedRTT + 4. DevRTT

ويوصي المرجع [RFC 6298]إن تكون القيمة الابتدائية للمهلة Timeoutinterval 1ث . عند انتهاء المهلة ، يتم مضاعفة قيمة Timeout Interval لتجنب انتهاء مهلة قبل أوانها لشريحة لاحقة سيصدر إشعارها حالا . مع ذلك ، حالما يتم استلام شريحة وتحديث EstimatedRTT ، يعاد حساب مدة المهلة Timeout Interval باستخدام الصيغة أعلاه

3-6-4 النقل الموثوق للبيانات Reliable Data Transfer

تذكر ، عزيزي الطالب ، أن خدمة بروتوكول الإنترنت (IP service) غير موثوقة ، أي لا تضمن استلام مخططات البيانات (datagrams) ، أو ترتيبها ، أو سلامتها . فقد تفيض عن سعة المخزن المؤقت للموجه فلا تصل أبدا ، وقد تصل غير مرتبة أو تحتوي على بتات خاطئة (ينقلب 0 إلى 1 أو العكس) ، وكون شرائح طبقة النقل تنتقل عبر طبقة الشبكة من خلال مخططات بيانات (IP datagrams) ، فقد تعاني هذه الشرائح من الإشكاليات ذاتها. بالمقابل ، يتأكد بروتوكول التحكم بالنقل (TCP) ، الذي يقدم خدمات نقل موثوقة ، أن سلسلة البيانات التي تقرأها العملية من مخزنها المؤقت سليمة ، بلا ثغرات أو تكرار ، وأنها متسلسلة ؛ أي أن السلسلة (byte stream) المستلمة هي ذاتها التي أرسلت . أما آلية عمل TCP بهذا الشأن ، فتتضمن العديد من أساسيات النقل الموثوق للبيانات التي درستها في القسم 3-5

كما أسلفنا ، فإن النقل الموثوق يتطلب توفير مؤقت منفصل يرتبط بكل شريحة أرسلت ولم يصل إشعار استلامها . وهذا عظيم من الناحية النظرية ، إلا أن إدارة المؤقت قد تتطلب مزيدا من الجهد ، وللتبسيط سنستخدم مؤقتا واحدة لإعادة الإرسال ، حتى لو كان هناك أكثر من شريحة مرسلة لم يصل إشعار استلامها . سنناقش الأمر تدريجيا في خطوتين ؛ سنقدم أولا وصفة غاية في البساطة لمرسل عبر TCP يستخدم نفاد المهلة (timeouts) فقط لاستعادة الشرائح المفقودة ، ثم ننتقل إلى وصف أشمل

قليلا يضيف استخدام الإشعارات المتكررة . سنفترض هنا أن البيانات المرسلة باتجاه واحد من المضيف A إلى المضيف B ، وأن المضيف A يرسل ملفا ضخما .

هناك ثلاثة أحداث رئيسة تتعلق بنقل البيانات وإعادة الإرسال لدى المرسل عبر TCP :

*استلام البيانات من التطبيق : يستقبل TCP البيانات من التطبيق ويغلفها ضمن شريحة يمررها إلى بروتوكول الإنترنت (IP) ، لكل شريحة رقم متسلسل يحمل رقم سلسلة البيانات لأول بايت في الشريحة . وإذا لم يكن المؤقت (timer) قيد التشغيل لشريحة أخرى ، يبدأ المؤقت فور تمريرها إلىIP ، وتكون فترة الصلاحية هي نفاد المهلة (Timeoutinterval) المحسوبة من خلالEstimatedRTT و DevRTT كما ذكرنا سابقا

*نفاد مهلة المؤقت : يستجيب TCP لنفاد المهلة بإعادة إرسال الشريحة المتسببة ، ويعيد تشغيل المؤقت .

*إصدار إشعار الاستلام : عندما يصل إشعار الاستلام من المستقبل إلى المرسل ، يقارن قيمته بالرقم المتسلسل لأخر بايت بلا إشعار . من الجدير بالذكر أن TCP يستخدم الإشعارات التراكمية فالإشعار y يشير إلى استلام جميع البايتات التي سبقت البايت رقم y ، فإذا كان y أكبر من الرقم المتسلسل لآخر بايت بلا إشعار ، يكون الإشعار لواحدة أو أكثر من الشرائح السابقة التي لم يصل إشعار استلامها .

الآن ، وحتى نتعرف أكثر على آلية عمل TCP ، سنستعرض بضعا من السيناريوهات البسيطة . السيناريو الأول ، حيث أن المضيف A يرسل شريحة للمضيف B ، ولنفرض الرقم المتسلسل لهذه الشريحة 92 ، وتحمل 8 بايت . بعد إرسالها ينتظر المضيف A شريحة من المضيف B تحتوي إشعار رقمه 100 ، ولنفرض أن B أرسل الإشعار ولكنه فقد ولم يصل إلى A. في هذه الحالة تنفد المهلة ، فيقوم A بإعادة إرسال الشريحة ذاتها ، وعندما تصل إلى B يدرك من رقمها المتسلسل أنها تحمل بيانات وصلته سابقا عندها يهمل TCP هذه البيانات .

في السيناريو الثاني ، يرسل المضيف A شريحتين متتاليتين ، الأولى رقمها المتسلسل 92 وتحمل بيانات حجمها 8 بايت ، والثانية رقمها 100 وتحمل 20 بايت . ولنفرض أن كلتا الشريحتين وصلتا سليمتين إلى المضيف B ، الذي رد بإشعار استلام منفصل لكل شريحة ، الإشعار الأول رقمه 100 والثاني رقمه 120. ولنفرض أن أيا من الإشعارين لم يصل إلى A قبل نفاد المهلة ، وعند نفاد المهلة أعاد A إرسال الشريحة الأولى ورقمها 92 ، وأعاد تشغيل المؤقت . طالما أن إشعار (ACK) الشريحة الثانية سيصل قبل نفاد المهلة الجديدة ، فلن يعاد إرسالها .

في السيناريو الثالث والأخير ، افترض أن المضيف A أرسل الشريحتين ، تماما كما في المثال الثاني ، وفقد إشعار الشريحة الأولى في الشبكة ، ولكن قبل نفاد المهلة مباشرة تلقى المضيف A إشعار استلام رقمه 120 .وعليه ، فإن المضيف A يعرف أن المضيف B قد تلقى كل البيانات حتى البايت 119 ؛ لذلك لا يقوم المضيفA بإعادة إرسال أي من الشريحتين.

3-6-5 ضبط التدفق Flow Control

كما تذكر ، فإن المضيفين على طرفي اتصال TCP يعد كل منهما مخزن استقبال مؤقت (receive buffer) عندما يتلقى الاتصال بايتات سليمة وعلى الترتيب ، فإنه يضع هذه البيانات في مخزن الاستقبال للمؤقت ، لتقرأها عملية التطبيق المرتبطة بها من المخزن المؤقت ، ولكن ليس بالضرورة لحظة وصولا لبيانات . في الواقع ، قد يكون التطبيق المتلقي مشغولا ببعض المهمات الأخرى ، وربما لا يحاول قراءة البيانات حتى بعد وصولها بفترة طويلة ، فإذا كان التطبيق بطيئا نسبيا في قراءة البيانات ، فقد يفيض المخزن المؤقت بسهولة إذا أرسل المرسل بيانات كثيرة بسرعة كبيرة . يوفر TCP خدمة ضبط التدفق لمنع المرسل من إغراق المخزن المؤقت للمستقبل ، ويتضمن ذلك مواءمة المعدل الذي يرسل به المرسل مع المعدل الذي يقرا به تطبيق الاستقبال . وكما ذكر سابقا ، قد يختنق مرسلTCP بسبب الاحتقان داخل شبكة بروتوكول الإنترنت ؛ ويشار إلى ضبط هذه الحالة بضبط الاحتقان . وعلى الرغم من أن إجراءات ضبط التدفق والاحتقان متشابهة (اختناق المرسل) ، إلا أن الأسباب مختلفة كثيرا.

للأسف ، كثير من الكتب تستخدم المصطلحين بالتناوب ، ولكن يجب التفريق بينهما. دعنا نناقش الان كيف يوفر TCP خدمة ضبط التدفق ، وللتبسيط، سنفترض في هذا القسم أن TCP ينفذ بحيث يتجاهل المستقبل الشرائح التي لا تصل على الترتيب.

يوفر TCP ضبط التدفق من خلال احتفاظ المرسل بمتغير يسمى نافذة الاستقبال (receive window) ،وتستخدم لتعطي المرسل فكرة عن مساحة المخزن المؤقت المتاحة لدى المستقبل ، ولأن TCP يعمل باتجاهين (full-duplex) ، يحتفظ المرسل في كل من جانبي الاتصال بنافذة استقبال مستقلة . دعنا ، عزيزي الطالب ، نتفحص نافذة الاستقبال في سياق نقل الملفات ، ولنفترض أن المضيف A يرسل ملفا كبيرا إلى المضيف B عبر اتصال TCP ، عندها يخصص المضيف B مخزن استقبال مؤقت لهذا الاتصال ؛ ولنرمز الحجمة Rcvbuffer ، ومن وقت لآخر ، تقرأ عملية التطبيق في المضيف B من المخزن المؤقت ، ولنحدد المتغيرات الأتية:

*البايت الأخير المقروء (LastByteRead) : رقم آخر بايت في سلسلة البيانات قرأته عملية التطبيق من المخزن المؤقت في المضيف B

*البايت الأخير المستقبل (LastByteRcvd) : رقم آخر بايت في سلسلة البيانات وصل من الشبكة ووضع في المخزن المؤقت في المضيف B.

ولان TCP لا يسمح له بإغراق المخزن المؤقت المخصص يجب ان يكون لدينا :

$$Lastbytercvd-lastbyteread \leq rcvbuffer$$

تحدد نافذة الاستقبال ويرمز لها (rwnd) كمية المساحة المتوفرة في المخزن المؤقت ولأنها مساحة متغيرة مع الزمن فان قيمة (rwnd) ديناميكية كما يبين الشكل 25-3 صفحة 160 نافذة الاستقبال (rwnd) ومخزن الاستقبال المؤقت rcvbuffer

$$Rwnd= rcvbuffer-[Lastbytercvd-lastbyteread]$$

فكيف يستخدم المتغير rwnd لضبط التدفق؟ يخبر المضيف B المضيف Aعن المساحة المتوفرة في المخزن المؤقت من خلال وضع قيمة rwnd في حقل نافذة الاستقبال لكل شريحة يرسلها الى A ويحدد المضيف B مبدأيا قيمة Rwnd= rcvbuffer ولتحقيق ذلك على المضيف Bتتبع العديد من المتغيرات الخاصة بالاتصال .

من جانبه يتتبع المضيف A متغيرين آخر بايت مرسل (Lastbytesent) واخر بايت بأشعار (Lastbyteacked) والفرق بينهما هو كمية البيانات بدون اشعار استلام ارسلها المضيف A عبر الاتصال وبالمحافظة على هذه الكمية اقل من rwnd يضمن المضيف A انه لن يغرق مخزن الاستقبال المؤقت لدى المضيف B اي يتأكد A طيلة فترة الاتصال ان :

$$Lastbytesent- Lastbyteacked \leq rwnd$$

هناك اشكالية بسيطة في هذا المخطط فلنفرض ان مخزن الاستقبال المؤقت للمضيف B قد امتلأ (rwnd = 0) وبعد اعلام المضيف Aلنفرض ان Bليس لديه ما يرسله الى A فما النتيجة؟ عند افراغ المخزن لدى B لن يرسل TCPشرائح جديدة بقيم جديدة للمتغير rwnd الى A اذ انه يرسل فقط اذا كان لديه بيانات او اشعارات ليرسلها الى A وعليه لن يتم اعلام المضيف Aابدا ان بعض المساحة قد اتيحت في مخزن المضيف B فيمنع المضيف A من نقل اي بيانات اخرى ! ولمعالجة هذه المسألة تشترط مواصفات TCP ان يواصل المضيف A ارسال شرائح بقيمة بايت واحد عندما تكون نافذة الاستقبال لدى B صفرا ويرسل المستقبل اشعارات استلام هذه الشرائح في النهاية سيبدأ المخزن المؤقت بالإفراغ وستحتوي الاشعارات قيمة لا صفرية للمتغير rwnd

من الجدير بالذكر، أن بروتوكول UDP لا يوفر ضبط التدفق ، فلو أرسلت سلسلة شرائح UDP من عملية على المضيف A إلى عملية على المضيف B، سيضيف UDP الشرائح إلى مخزن مؤقت محدود الحجم يسبق المقبس المقابل. تقرأ العملية شريحة كاملة في المرة من المخزن المؤقت، فإذا لم تقرأ الشرائح بسرعة كافية، سيفيض المخزن المؤقت وتهمل الشرائح.

6-6-3 إدارة الاتصال في بروتوكول التحكم بالنقل TCP Connection management

دعنا نلقي نظرة عن قرب على كيفية إدارة اتصال TCP وتشمل إنشاء الاتصال وإنهائه ينشا اتصال TCP بناء على رغبة عملية قيد التشغيل في العميل بيده اتصال مع عملية أخرى في الخادم، فتقوم عملية تطبيق العميل باعلام TCP على العميل أنه يرغب بإنشاء اتصال بعملية في الخادم، ثم يستكمل إنشاء اتصال مع TCP على الخادم كما تبين الخطوات الآتية:

1. يرسل TCP لدى العميل أولا شريحة TCP خاصة إلى TCP لدي الخادم لا تحتوي هذه الشريدة على بيانات طبقة التطبيقات، بل تحتوي على إحدي بتات الراية في مقدمة الشريحة، فيمنح البت SYN القيمة 1 (انظر الشكل 3-26) لذا، يطلق على هذه الشريحة شريحة SYN. كما يختار العميل رقما تسلسليا مبدئيا بشكل عشوائي لتجنب الهجمات الأمنية (client_isn)، ويوضع في حقل الرقم التسلسلي لشريحة SYN المبدئية، ويتم تغليف هذه الشريحة ضمن مخطط بيانات ip المرسل إلى الخادم
2. بمجرد وصول مخطط بيانات IP الذي يحتوي على شريحة SYN إلى الخادم، يستخرج هذه الشريحة من مخطط البيانات، ويحدد المخازن المؤقتة والمتغيرات الخاصة بالاتصال، ويرسل شريحة موافقة على الاتصال إلى TCPالخاص بالعميل، لا تحتوي شريحة الموافقة على بيانات طبقة التطبيق، ولكنها تحتوي على ثلاث معلومات مهمة في مقدمة الشريحة أولا، يمنح البت SYN القيمة 1. ثانيا، يمنح حقل إشعار مقدمة الشريحة القيمة (client_isn+1). أخيرا، يختار الخادم رقما تسلسليا مبدئيا (server_isn) ويضعه في حقل الرقم التسلسلي في مقدمة شريحة TCP. وتعني شريحة الموافقة على الاتصال "تلقيت حزمة SYN لبدء اتصال برقمك التسلسلية المبدئي (client_isn)، وأوافق على إنشاء هذا الاتصال، ورقمي التسلسلي الميداني هو (server_isn)". يطلق على شريحة الموافقة على الاتصال شريحة SYNACK
3. عندما يستقبل شريحة SYNACK ، يخصص العميل أيضا المخازن المؤقتة ومتغيرات الاتصال ثم يرسل إلى الخادم شريحة أخرى كإشعار على شريحة الموافقة على الاتصال بالخادم، حيث يضع العميل القيمة (server_isn + 1) في حقل الإشعار في مقدمة شريحة TCP ثم يمنح البت SYN القيمة 0 إيذانا بإنشاء الاتصال، وتحمل هذه المرحلة من المصافحة الثلاثية البيانات من العميل إلى الخادم في الحمولة الصافية للشريحة

شكل 3-26 صفحة 162 المصافحة الثلاثية في بروتوكول TCP : تبادل الشرائح

بعد هذه الخطوات الثلاث يستطيع العميل والخادم تبادل شرائح البيانات، ولكل شريحة يعطى البت SYN القيمة 0. ولإنشاء الاتصال ترسل ثلاثة حزم بين المضيفين، لذا يطلق على هذه العملية المصافحة الثلاثية (Three-way handshake)، ويستطيع أي عملية مشاركة في الاتصال إنهاء الاتصال، وعندها يتم الإفراج عن الموارد من مخازن مؤقتة ومتغيرات. لنفرض أن العميل قرر إنهاء الاتصال، عندها تقوم عملية تطبيق العميل بإصدار أمر إغلاق الاتصال، فيقوم TCP لدي العميل بإرسال شريحة خاصة إلى عملية الخادم تشمل بت الراية في مقدمتها يسمي FIN ويعطى القيمة 1 ، كما يبين الشكل 3-27 صفحة 162 اغلاق او انهاء اتصال TCP

عندما يستلم الخادم هذه الشريحة، يرسل بدوره شريحة إشعار، ثم يرسل شريحته الخاصة بإغلاق الاتصال والتي تحتوي على البت FIN بقيمة 1. أخيرا، يقوم العميل بإصدار إشعار لشريحة الإغلاق التي استلمها من الخادم. في هذه اللحظة يفرج عن جميع الموارد الخاصة بالمضيفين.

عزيزي الطالب، لقد افترضنا في نقاشنا السابق أن كلا العميل والخادم مستعدان للاتصال، أي أن الخادم يستمع على المنفذ الذي يرسل إليه العميل شريحة SYN الخاصة به، فماذا يحدث عندما يتلقي المضيف شريحة TCP لا تتطابق أرقام منافذها أو عنوان الإنترنت (IP) المصدر مع أي من المقابس قيد التنفيذ في المضيف؟ على سبيل المثال، لنفترض أن مضيفا تلقي حزمة TCP SYN منفذها للوجهة 80 ، ولكنه لا يقبل اتصالات على المنفذ 80 (أي، خادم الويب يعمل على المنفذ 80)، عند يرسل المضيف شريحة إعادة تشغيل (Reset) خاصة إلى المصدر تحتوي على بت الراية RST ويعطى القيمة 1. وهكذا، عنها يرسل مضيف شريحة إعادة تشغيل، فإنه يخبر المصدر بالرسالة: "ليس لدي مقبس لهذه الشريحة، يرجي عدم إعادة إرسال الشريحة " وإذا استلم المضيف حزمة UDP لا يتطابق رقم منفذها مع مقبس UDP قي التنفيذ، فإنه يرسل مخطط بيانات خاصة من نوع ICMP ، كما سنوضح في الفصل الرابع

عندما يستلم الخادم هذه الشريحة، يرسل بدوره شريحة إشعار، ثم يرسل شريحته الخاصة بإغلاق الاتصال والتي تحتوي على البت FIN بقيمة 1. أخيرا، يقوم العميل بإصدار إشعار لشريحة الإغلاق التي استلمها من الخادم. في هذه اللحظة يفرج عن جميع الموارد الخاصة بالمضيفين.

عزيزي الطالب، لقد افترضنا في نقاشنا السابق أن كلا العميل والخادم مستعدان ل لاتصال، أي أن الخادم يستمع على المنفذ الذي يرسل إليه العميل شريحة SYN الخاصة به، فماذا يحدث عندما يتلقي المضيف شريحة TCP لا تتطابق أرقام منافذها أو عنوان الإنترنت (IP) المصدر مع أي من المقابس قيد التنفيذ في المضيف؟ على سبيل المثال، لنفترض أن مضيفا تلقي حزمة TCP SYNمنفذها للوجهة 80 ، ولكنه لا يقبل اتصالات على المنفذ 80 (أي، خادم الويب يعمل على المنفذ 80)، عند يرسل

المضيف شريحة إعادة تشغيل (Reset) خاصة إلى المصدر تحتوي على بت الراية RST ويعطى القيمة 1. وهكذا، عنها يرسل مضيف شريحة إعادة تشغيل، فإنه يخبر المصدر بالرسالة: "ليس لدي مقبس لهذه الشريحة، يرجي عدم إعادة إرسال الشريحة " وإذا استلم المضيف حزمة UDP لا يتطابق رقم منفذها مع مقبس UDP في التنفيذ، فإنه يرسل مخطط بيانات خاصة من نوع ICMP ، كما سنوضح في الفصل الرابع

7-3 | مبادئ ضبط الاحتقان (الازدحام)Principles of Congestion Control

عزيزي الطالب، اختبرنا في الأقسام السابقة المبادئ العامة والآليات المحددة التي يتبعها TCP لتوفير خدمة النقل الموثوق للبيانات عند فقدان الحزم. كما أشرنا أن فقدان الحزم في الممارسة العملية ينجم عادة عن فيضان مخازن الموجهات عند احتقان (ازدحام)الشبكة. وعليه، فإن إعادة إرسال الحزم، تعالج أعراض احتقان الشبكة (فقدان شريحة في طبقة النقل) دون معالجة أسبابه (محاولة العديد من مصادر البيانات الإرسال بمعدلات مرتفعة جدا). ولمعالجة أسباب الاحتقان، عند حدوثه، فإن ذلك يتطلب آليات لوقف هذه المصادر عن الإرسال (خنق المرسل).

في هذا القسم، سنغطى مشكلة ضبط الاحتقان في السياق العام، فلماذا يعتبر الاحتقان مشكلة؟ وكيف يتجلى في الأداء الذي تتلقاه التطبيقات في الطبقات العليا؟ وما الطرق المختلفة لتجنب احتقان الشبكة، أو الاستجابة له؟ ونشير هنا أننا نعتبر ضبط الاحتقان من أهم عشرة مواضيع في الشبكات. وسننهي النقاش بمثال حول خدمة معدل الإرسال المتاح (Available bit-rate: ABR)في شبكات وضع النقل غير المتزامن (Asynchronous transfer mode: ATM)

The Causes and the Costs of Congestion
اسباب الاحتقان وكلفته 3-7-1

عزيزي الطالب، سنتدارس ضبط الاحتقان من خلال ثلاثة سيناريوهات تندرج في التعقيد مع إبراز أسباب الاحتقان وكلفته من حيث المصادر غير المستغلة كلية، وضعف الأداء في الأنظمة النهائية ولن تسلط الضوء حاليا على كيفية تجنب الاحتقان أو الاستجابة له، بل إلى مسألة أبسط لفهم ما يحدث عندما يزيد المضيف من معدل الإرسال مسببة الاحتقان.

السيناريو الاول: مرسلان وموجه بمخازن مؤقتة لا محدودة (two senders and a router with infinite buffers)

سنبدأ بأبسط سيناريو ممكن: مضيفان (B,A) لكل منهما اتصال يشترك في قفزة واحدة (single hop) بين المصدر والوجهة، كما يبين الشكل (3-28) لنفرض أن التطبيق في المضيف A يرسل بيانات عبر الاتصال (مثلا ، تمرير البيانات إلى بروتوكول النقل عبر مقبس) بمعدل (λ_{in} bytes/sec) بايت في الثانية

شكل 3-28 صفحة 164 اتصالان يشاركان في فقرة (hop) واحدة ومخزن مؤقت لا محدود (السيناريو الاول)

تعتبر هذه البيانات أصيلة، حيث أن كل وحدة بيانات ترسل عبر المقبس مرة واحدة فقط، وفي هذه الحالة يكون بروتوكول النقل الأساسي بسيط، حيث تغلف البيانات وترسل ولا ينفذ تصحيح الأخطاء، أو ضبط التدفق، أو ضبط الاحتقان. ويتجاهل الكلفة الإضافية الناتجة عن إضافة معلومات المقدمة لطبقة النقل وما دونها، فإن معدل حركة المرور التي يوفرها المضيف A إلى الموجه تبعا لذلك هي λ_{in} في الثانية. ويعمل المضيف B بنفس الطريقة، وللتبسيط، نفترض أنه أيضا يرسل بنفس المعدل. تمر الحزم من المضيف A والمضيف B عبر موجه وخط اتصال خارج مشترك سعته R. للموجه مخازن مؤقتة تتيح له تخزين الحزم الداخلة عندما يتجاوز معدل وصول الحزم سعة الخط الخارج، وسنفرض ان الموجه لديه مخزن مؤقت مساحته لامحدودة.

يبين الشكل (3-29) أداء اتصال المضيف A في هذا السيناريو، فيبين الرسم البياني الأيسر الإنتاجية (throughput) لكل اتصال (عدد البايتات لكل ثانية لدى المستقبل) مقابل معدل الإرسال فعندما يكون معدل الإرسال بين 0-R/2، فإن الإنتاجية لدى المستقبل تساوي معدل إرسال المرسل، أي أن كل ما يرسله المرسل يتلقاه المستقبل بعد زمن تأخير محدود (finite delay)

شكل29-3 صفحة 165 الإنتاجية (throughput) وزمن التاخير (delay) مقابل معدل الإرسال من المضيف (السيناريو الأول)

وعندما يكون معدل الإرسال أكبر من R / 2 ، تكون الإنتاجية على أي حال R/2 فقط، ويأتي هذا الحد الأعلى نتيجة مشاركة سعة الوصلة بين الاتصالين، أي أنه لا يستطيع تسليم الحزم إلى المستقبل بسرعة ثابتة تتجاوز R/2. وبصرف النظر عن ارتفاع معدل إرسال المضيفين A،B لن تزيد الإنتاجية نهائيا عن R/2 لأي منهما في الواقع، قد يبدو أن تحقيق إنتاجية R / 2 لكل خط اتصال أمر جيد، لأنها تستغل بالكام في تسليم الحزم إلى وجهاتها.

بيد أن الرسم البياني الأيمن يبين تبعات التشغيل قريبا من سعة خط الاتصال، فعدما يقترب معدل الإرسال من R2 (من اليسار)، يصبح متوسط زمن التأخير أكبر. وعندما يتجاوز R / 2 ، يكون متوسط عدد الحزم في قائمة انتظار الموجه غير محدود، ويصبح متوسط زمن التأخير بين المصدر والوجهة لا محدود(بفرض أن الاتصالات تعمل بمعدلات الإرسال هذه لفترة غير محددة من الزمن، وأن مساحة التخزين المؤقت المتاحة لا محدودة). في حين أن التشغيل بإنتاجية إجمالية تقترب من R قد يكون مثالية من ناحية الإنتاجية، فهو غير مثالي من ناحية زمن التأخير. حتى في هذا السيناريو المثالي (للغاية)، وجدنا بالفعل كلفة واحدة لاحتقان الشبكة المعاناة من زمن تأخير كبير في الطابور كلما اقترب معدل وصول الحزم من سعة خط الاتصال.

السيناريو الثاني: مرسلان وموجه ذو مخزن مؤقت محدود (Two Senders and a Router with Finite Buffers)

عزيزي الطالب، دعنا الآن نعدل السيناريو الأول بشكل طفيف بطريقتين، انظر الشكل (30-3):

1. أولا، سنفترض أن كمية التخزين المؤقت للموجه محدودة، وبالنتيجة، سيتم إهمال الحزم عند امتلاء المخزن المؤقت.

2 .ثانياً، سنفترض أن كل الاتصالات موثوقة، فإذا أهملت حزمة تحتوي على شريحة من مستوى النقل في الموجه، فإن المرسل سيعيد إرسالها.

عزيزي الطالب، لأننا نستطيع إعادة إرسال الحزم، يجب أن نكون أكثر حذرا عند استخدام مصطلح معدل الإرسال. على وجه التحديد، دعنا مرة أخرى نعبر عن المعدل الذي يرسل به التطبيق بيانات أصلية عبر المقبس بالرمز (λ_{in}) بايت/ثانية، وعن المعدل الذي ترسل فيه طبقة النقل الشرائح (التي تحتوي على بيانات

اصلية وبيانات اعيد ارسالها) عبر الشبكة بالرمز λ'_{in} بايت/ثانية ، ويشار اليها احيانا باسم الحمولة المقدمة (offered load) للشبكة

شكل 30-3 صفحة 166 مرسلان (مضيفان) مع اعادة الارسال وموجه ذو مخازن مؤقتة محدودة (السيناريو الثاني)

يعتمد الأداء في هذا السيناريو بقوة على كيفية تنفيذ إعادة الإرسال. أولا، لنعتبر، جدلا، أن المضيف A قادر (بطريقة سحرية!) على تحديد ما إذا كان المخزن المؤقت متاحا في الموجه، وعندها فقط يرسل حزمة في هذه الحالة، لن يحدث فقدان الحزم، أي $\lambda'_{in} = \lambda_{in}$ ، والإنتاجية = λ_{in} ، كما يبين الشكل (3-31-a). بنظر الإنتاجية، هذا الأداء مثالي، إذ تم استقبال كل شيء أرسل. لاحظ، عزيزي الطالب، أن متوسط معدل إرسال المضيف لا يمكن أن يتجاوز R/2 في هذا السيناريو، أي من المفترض ألا تفقد الحزم أبدا.

انتقل إلى حالة أكثر واقعية قليلا، حيث يعيد المرسل إرسال الحزمة فقط عندما يوقن أنها فقدت. (هذا الافتراض فضفاض قليلا، ومع ذلك، فقد يحدد المضيف المرسل مهلة (timeout) كبيرة بما فيه الكفاية التاكد افتراضية أن الحزمة التي لم يصل إشعارها قد فقدت). في هذه الحالة، يبدو الأداء كما هو مبين في الشكل (3-31-b) ولتوضيح ما يحدث هنا، انظر في حالة أن الحمولة المقدمة، أي معدل إرسال البيانات الأصلية بالإضافة إلى إعادة الإرسال، (λ'_{in} =R /2). وحسب الشكل (3-31-b)، عند هذه الحمولة المقدمة يكون معدل تسليم البيانات إلى تطبيق المستقبل R / 3 . وهكذا، من أصل 0.5R من وحدات البيانات المرسلة البيانات الأصلية (بالمتوسط) R0.333 بايت/ثانية والبيانات التي أعيد إرسالها (بالمتوسط) 0.166R بايت/ثانية

شكل 31-3 صفحة 166 الاداء عندما تكون المخازن المؤقتة محدودة (السيناريو الثاني)

)

عزيزي الطالب، هنا تظهر كلفة أخرى ناتجة عن احتقان الشبكة فعلى المرسل إعادة الإرسال لتعويض الحزم (المفقودة) التي أهملت بسبب تجاوز سعة المخزن المؤقت.

وأخيراً، دعنا ننظر، عزيزي الطالب، في حالة نفاد المهلة لدى المرسل قبل أوانها، وإعادة إرسال حزمة تأخرت في قائمة الانتظار ولكن لم تفقد بعد. في هذه الحالة، قد تصل حزمة البيانات الأصلية وتلك التي أعيد إرسالها إلى المستقبل الذي يحتاج نسخة واحدة من هذه الحزمة، لذا، سيتجاهل إعادة الإرسال. في هذه الحالة، يكون دور الموجه إعادة توجيه نسخة من الحزمة الأصلية التي أعيد إرسالها على شكل هدر، كون المستقبل قد تلقي بالفعل النسخة الأصلية، إذ كان من الأجدى استغلال سعة خط الاتصال لإرسال جديدة ويبين الشكل (3-31 - c) الإنتاجية مقابل الحمولة المقدمة عند كل حزمة يفترض أن يرسلها الموجه مرتين (بالمتوسط)، ولأن كل حزمة ترسل مرتين، تقترب قيمة الإنتاجية من R / 4 عندما تقترب الحمولة المقدمة من R / 2 . عزيزي الطالب، هناك ثم كلفة أخرى لاحتقان الشبكة، تتمثل في أن إعادة الإرسال غير الضرورية قد تؤدي إلى تأخيرات كبيرة

واستنفاد عرض النطاق الترددي لخط الاتصال (link bandwidth) الذي يستخدمه الموجه في تحويل نسخ غير ضرورية من الحزمة

السيناريو الثالث: أربعة مرسلين، وموجه ذو مخزن مؤقت محدود، ومسارات متعددة القفزات, Four Senders

(Routers with Finite Buffers, and Multihop Paths

عزيزي الطالب، في السيناريو الأخير، يقوم أربعة مضيفين بإرسال الحزم، كل منها عبر موجهات متراكبة ثنائية القفز، كما هو مبين في الشكل (32-3). لنفترض أن كل مضيف يستخدم المهلة/ إعادة الإرسال لتحقيق الموثوقية وجميع المضيفين لهم نفس قيمة λ_{in} ، وأن جميع خطوط الموجه سعتها R بايت/ثانية. دعنا ننظر في الاتصال من المضيف A إلى المضيف C ، الذي يمر من خلال الموجهين R1 و R2 يشترك الاتصال A-C مع D-B في الموجه R1 ويشترك مع B-D في الموجه .R2 لقيم λ_{in} صغيرة للغاية، يندر فيضان المخزن المؤقت (كما هو الحال في السيناريو الأول والثاني)، والإنتاجية تساوي الحمولة المقدمة تقريبا لقيم أكبر بقليل تكون الإنتاجية أيضا أكبر، لأن مزيدا من البيانات الأصلية تنتقل عبر الشبكة وتسلم إلى الوجهة، ويندر الفيضان. وهكذا، للقيم الصغيرة من λ_{in} عند زيادة λ_{in} تزداد λ_{out}.

سنختبر الآن حالة حركة المرور الكبيرة للغاية، أي λ_{in} وبالتالي λ'_{in} كبيرة للغاية. دعنا نلقي نظرة في الموجه R2، معدل وصول حركة المرور A-C التي تصل إلى الموجه R2 بعد تحويلها من R1، قد يكون على الأكثر R، سعة الوصلة من R1 إلى R2 ، بغض النظر عن قيمة λ_{in}. وإذا كانت قيمة λ'_{in} كبيرة جدا لجميع الوصلات (بما في ذلك الاتصال B-D)، فإن معدل وصول حركة المرور B-D عند R2 قد يكون أكبر بكثير من حركة المرور A-C. ولأن الحركة A-C و B-D يجب أن تتنافسا عند الموجه R2 على المساحة المحدودة في مخزنه المؤقت، فإن كمية حركة المرور A-C التي تنجح في عبور R2 تصبح أصغر وأصغر كلما أصبح الحمل المقدم من B-D أكبر وأكبر. في الحد الأقصي، كلما تقترب الحمولة المقدمة من اللانهاية، يمتلئ المخزن المؤقت الفارغ للموجه R2 فورا من حزمة B-D، وتقترب الإنتاجية البوصلة A-C عند R2 إلى الصفر. أي أن إنتاجية A-C تقترب من الصفر عندما تكون حركة المرور كثيفة. وتؤدي هذه الاعتبارات إلى مقايضة الحمولة المقدمة مقابل الإنتاجية، كما يبين الشكل (33-3).

شكل 32-3 صفحة 168 اربعة مرسلين وموجه ذو مخزن مؤقت محدود ومسارات متعددة القفرات

يعزى انخفاض الإنتاجية مع زيادة الحمولة المقدمة إلى مقدار الهدر في عمل الشبكة، ففي تسهيل حركة المرور المرتفعة كما في السيناريو أعلاه، كلما أهملت حزمة في موجه القفزة الثانية فان العمل الذي ينجزه موجه القفرة الأولى في تحويل الحزمة إلى موجه القفزة الثانية يعتبر هدرا ويصبح الحال اسوء اذا تجاهل الموجه الأول ببساطة تلك الحزمة وبقي عاطلا، وقد تكون قدرة الإرسال التي يستخدمها الموجه الأول لتحويل الحزمة إلى الموجه الثاني أكثر جدوى لو استغلت لنقل حزمة مختلفة فمثلا عن اختيار حزمة للإرسال، يفضل أن يعطي الموجه الأولوية للحزم التي اجتازت بالفعل عددا من مرجيات المنبع

شكل 33-3 صفحة 168 الأداء عندما تكون المخازن المؤقتة محدودة و المسارات متعددة الفرات (السناريو الثالث)

لاحظ هنا كلفة اخرى نتجت عن الاحتقان، فعندما تهمل حزمة في المسار، فان سعة (قدرة) الإرسال، التي استخدمت في كل من وصلات المنبع لإحالة تلك الحزمة إلى النقطة التي أهملت عندها تعتبر هدرا

| 3-8 |ضبط الازدحام (الاحتقان) في بروتوكول التحكم بالنقل TCP Congestion Control

يعتبر ضبط الاحتقان من المكونات الأساسية لبروتوكول TCP، كما أشرنا في الأقسام السابقة، ويجب أن يستخدم TCP ضبط الاحتقان نهاية- نهاية (end-to-end congestion control) بدل من الاستعانة بالشبكة (network-assisted congestion control) لانه طبقة بروتوكول الانترنت لا توفر تغذية راجعة صريحة ل لأنظمة النهائية end systems) حول احتقان الشبكة. فكل مرسل يحدد معدل الإرسال في خط الاتصال الخاص به بدلالة ضبط الاحتقان، فإذا أحس المرسل بأن الاحتقان قليل في مساره نحو الوجهة، عندها يزيد معدل الإرسال، وإذا أدرك أن هناك احتقان على طول المسار، يقلل معدل الإرسال وهذه الطريقة تثير ثلاثة أسئلة؛ كيف يحدد مرسل TCP معدل الإرسال؟ وكيف يدرك المرسل أن هذا احتقان بينه وبين الوجهة؟ وما الخوارزمية التي يستخدمها لتغيير معدل الإرسال بدلالة الاحتقان؟

للإجابة على السؤال الأول، وكما أشرنا في القسم 3-6 ، يتكون كل جانب من اتصال TCP من مدرن مؤقت للاستقبال وآخر للإرسال، وعدة متغيرات(LastByteRead ، rwnd)وغيرها. وتتبع آلية ضبط الاحتقان لدى المرسل المتغير الإضافي نافذة الاحتقان، (congestion window: cwnd) والتي تضع قيدا على معدل الإرسال (حركة المرور) عبر الشبكة، وبالتحديد ألا تتجاوز البيانات بلا إشعار الحد الأدني المتغيرين Cwnd ، و rwnd ، أي:

$$LastByteSent - LastByteAcked \leq \min\{cwnd, rwnd\}$$

ولكي تركز على ضبط الاحتقان لا التدفق، دعنا نفترض أن مخزن الاستقبال المؤقت كبير يسمح بإهمال القيود على نافذة الاستقبال، وعليه فإن كمية البيانات بلا إشعار تحدد بالمتغير cwnd، وسنفترض أيضا أن لدى المرسل دائما بيانات ليرسلها، أي أن جميع الشرائح في نافذة الاحتقان قد أرسلت. وبذلك يستطيع المرسل تحديد معدل الإرسال، وللتوضيح اعتبر أن هناك خط اتصال زمن تأخير الفقدان ونقل الحزم لديه مهمل في بداية كل RTT تقريبا يسمح للمرسل بإرسال بيانات cwnd بايت، وفي نهايتها يستقبل إشعارات الاستلام أي أن معدل الإرسال تقريباً cwnd/RTT بايت /الثانية، وبضبط cwnd يتمكن المرسل من تعديل معدل الإرسال على هذا الخط

اللإجابة على السؤال الثاني، دعنا نعرف فقدان الحزم لدى مرسل TCP بالحدث (loss event)عند نفاد المهلة أو استلام 3 إشعارات متكررة من المستقبل. عندما يكون هناك احتقان كبير، يفيض مخزن واحد او أكثر(من مخازن الموجه على طول المسار، بسبب إهمال مخطط بيانات يحتوي على شريحة TCP. ويؤدي

ذلك بدوره إلى حدوث فقدان لدى المرسل، فيتخذ المرسل هذا الحدث مؤشر احتقان على مساره إلى المستقبل ولكن ماذا إذا لم يظهر هذا الحدث؟ في هذه الحالة تصل إشعارات الشرائح التي لم تصل سابقا الى المرسل فيستخدم TCP هذه الإشعارات كمؤشر أن الأمور على ما يرام، أي آن جميع الشرائح المرسلة استلمت بنجاح، فيزيد حجم نافذة الاحتقان، وبالتالي، يزيد معدل الإرسال

ننتقل إلى السؤال الثالث، فكيف يستطيع المرسل تحديد معدل الإرسال؟ إذا كان أكثر من مرسل يرسلون بسرعة فانهم يسببون احتقان الشبكة كما يبين الشكل (33-3). أما إذا كانوا يرسلون ببطء، فإنهم يقللون استغلال النطاق الترددي للشبكة، أي كان بإمكانهم الإرسال بمعدلات أعلى دون التسبب في الاحتقان. يجيب TCP على هذا السؤال بالاستعانة بالإرشادات الأساسية الآتية:

1- يشير فقدان شريحة إلى وجود احتقان، وبالتالي، على المرسل تقليل معدل الإرسال.
2- يشير وصول إشعار الاستلام لشريحة أن الشبكة تسلم الشرائح للمستقبل، وبالتالي، يستطيع المرسل زيادة معدل الإرسال.
3- التحقق من النطاق الترددي. بالنظر إلى الإشعارات ACKs التي تشير إلى خلو المسار من المصدر إلى الوجهة من الاحتقان، وحدوث فقدان الشرائح الذي يشير إلى احتقان المسار، فإن استراتيجية TCP لتعديل معدل الإرسال بزيادته استجابة لوصول الإشعارات حتى حدوث فقدان شريحة عندها، يقلل معدل الإرسال. وبالتالي فإن مرسل TCP يزيد معدل الإرسال للتحقق من المعدل الذي بدأ عنده الاحتقان، فيتراجع عن هذا المعدل، ثم يبدأ التحقق مرة أخرى لفحص ما إذا كان معدل بداية الاحتقان قد تغير.

عزيزي الطالب، في ضوء ما تقدم، يمكنك الآن الانتقال إلى خوارزمية ضبط الاحتقان-(TCP congestion control algorithm)التي وصفت لأول مرة في [Jacobson1988 ، حسب المرجع [RFC 5681]و تتكون الخوارزمية من ثلاثة مكونات رئيسة: (1) البداية البطيئة(slow start)، (2) تجنب الاحتقان (congestion avoidance)
(3) الاسترداد السريع (fast recovery) المكونان الاول والثاني الزاميان، بينما الثالث يوصي به، ولكن ليس إلزاميا، لمرسلي TCP.

البداية البطيئة (Slow Strat)

عنما يبدأ الاتصال، يتم تهيئة المتغيرcwnd بقيمة ابتدائية صغيرة (1 MSS) حسب المرجع[RFC 3390] وتزداد بقيمة (1 MSS) كلما وصل إشعار استلام لأول مرة. والقيمة الابتدائية الناتجة لمعدل الإرسال حواليMSS/RTT ، فإذا كان MSS=500 بايت، وكان RTT=200 ميلي ثانية، فإن معدل الإرسال الابتدائي حوالي 20 kbps (كيلوبت/الثانية). ولأن النطاق الترددي المتوفر لمرسل TCP قد يكون أكبر بكثير منMSS/RTT ، فإنه يرغب بإيجاد النطاق الترددي المتوفر بسرعة. في المثال المبين في الشكل (3-34)، يرسل TCP الشريحة الأولى وينتظر

إشعار الاستلام، وعندما يصل، يزيد المرسل نافذة الاحتقان بقيمة (1 MSS)، ويرسل شريحتين بحجمهما الأقصى، وعندما يصل إشعارهما يزيد المرسل : الاحتقان بقيمة (1 MSS) لكل منهما، فتصبح النافذة (4 MSS)، وهكذا.
شكل 3-34صفحة 171 البداية البطيئة لبروتوكول التحكم بالنقلTCP

العملية أعلاه، عملت على مضاعفة معدل الإرسال كل RTT ، أي يبدأ معدل الإرسال ببطء ولكنه ينمو باطراد خلال مرحلة البداية البطيئة. وينتهي هذا النمو في حالات عدة، منها:

1 .عند حدوث فقدان(loss event) بعد نفاذ المهلة، يحدد مرسل TCP قيمة cwnd= 1 وتبدأ عملية البداية البطيئة مجددا، كما أنه يحدد متغير حالة آخر يسمى العتبة (ssthresh) بقيمة (cwnd/2)،اي نصف قيمة نافذة الاحتقال عند اكتشافه
2. يرتبط انتهاء وضع البداية البطيئة مباشرة بقيمة المتغيرssthresh ، ، وهي نصف قيمة cwnd عند كشف الاحتقان مؤخرا، قد يكون من التهور الحفاظ على مضاعفة cwnd عندما يصل أو يتجاوز قيمة ssthresh وبالتالي، عندما يصبح(cwnd=ssthresh) ، تنتهي البداية البطيئة وينتقل TCP إلى وضع "تجنب الاحتقان."

3 اكتشاف وصول ثلاثة إشعارات استلام (ACKs) متكررة، عندها ينفذ TCP إعادة إرسال سريعة وينتقل إلى وضع الاسترداد السريع، ويلخص الشكل (35-3) تصرف البداية البطيئة بالتفصيل ضمن خوارزمية ضبط الاحتقان في TCP

شكل 3-35 صفحة 172 وصف آلة الحالات المنتهية (FSM) الخوارزمية TCP لضبط الاحتقان

تجنب الاحتقان(Congestion Avoidance)

عند دخول وضع تجنب الاحتقان، تكون قيمة المتغير cwnd تقريبا نصف قيمته عند مواجهة آخر احتقان و هكذا ، بدلا من مضاعفة cwnd كل RTT ، يتبنى TCP نهجا أكثر تحفظا ويزيد قيمة CWnd بمقدار .Mss واحد فقط كل RTT حسب المرجع [RFC 5681]، يطرق عدة. هناك طريقة شائعة بان يزيد المرسل cwnd بمقدار MSS بايت (cWnd/MSS) عندما يصل إشعار جديد. على سبيل المثال، إذا كان 1460 Mss بايت وكان cWnd 14،600 بايت، يتم إرسال 10 شرائح خلال فترة RTT ، وكل إشعار ACKيصل (بافتراض إشعار واحد لكل شريحة)، يزيد حجم نافذة الاحتقان بمقدار MSS 1/10 ، وبالتالي عمان قيمة نافذة الاحتقان ستزداد بمقدار MSS واحد بعد وصول إشعارات ACK عند استلام الشرائح العشر

ولكن متى تنتهي الزيادة الخطية (MSS واحد لكل RTT) في وضع تجنب الاحتقان؟ تتصرف الخوارزمية عن طريقة نفاذ المهلة، وكما هو الحال في البداية البطيئة: تحدد قيمة cwnd بمقدار MSS 1، ويتم تحديث خيمة ssthresh إلى نصف قيمة cwnd عندما حصل حدث الفقدان، الذي قد يسببه أيضا حدث تكرار الاشعار الثلاثي، وفي هذه الحالة، تواصل الشبكة تسليم الشرائح من المرسل إلى المستقبل، لذا ينبغي أن يتصرف TCP مع هذا النوع من الفقدان بشكل اقل حدة من الفقدان الناجم عن نفاد المهلة: ينصف TCP cwnd (اضافة 3 MSS كمقياس جيد لحساب الإشعارات الثلاث المتكررة المستلمة) ويسجل قيمة
ssthreshبمقدار نصف قيمة cwnd عند استلام إشعارات ACKs الثلاث المتكررة، ثم يدخل في وضع" الاسترداد السريع."

الاسترداد السريع(Fast Recovery)

في الاسترداد السريع قيمة cwnd بمقدار 1 MSS لكل إشعار ACK مكرر للشريحة المفقودة التي سببت دخول TCPفي وضع الاسترداد السريع. في نهاية المطاف، عندما يصل إشعار الشريحة المفقودة يدخل TCP في وضع تجنب الاحتقان بعد تضاؤل cwnd إذا حصل حدث نفاد المهلة، ينتقل الاسترداد السريع إلى وضع البداية البطيئة بعد تنفيذ نفس الإجراءات كما هو الحال في البداية البطيئة وتجنب الاحتقان تحدد قيمة CWnd بمقدار MSS 1 ، وتحدد قيمة ssthresh بنصف قيمة cwnd عند حصول حدث الفقدان (loss event)لاحظ أن الاسترداد السريع مكون يوصى به ولكنه غير الزامي، حسب المرجع[RFC 5681]

يبين الشكل (35-3) الوصف الكامل لخوارزميات ضبط الاحتقان في TCP؛ البداية البطيئة، وتجنب الاحتقان، والاسترداد السريع، كما يبين أين قد يحدث إرسال شرائح جديدة أو إعادة إرسال الشرائح. عزيزى الطالب، بعد أن تعرفت إلى خوارزميات ضبط الاحتقان، يمكنك التفريق بين ضبط الخطأ/إعادة الإرسال وضبط الاحتقان في TCP ، كما يمكنك تقدير الارتباط الوثيق بين هاتين الظاهرتين.

سمي الإصدار القديم TCP Tahoe وهو يقطع نافذة الاحتقان دون شرط إلى MSS 1 ويدخل مرحلة البداية البطيئة بعد فقدان حزمة إما لنفاد المهلة أو نتيجة إشعار مكرر ثلاثي .(Triple - duplicate - ACK) أما الإصدار الأحدث فيطلق عليه TCP Reno ، ويستخدم الاسترداد السريع، ويبين الشكل (3-36) تطور نافذة الاحتقان لكليهما. العتبة الابتدائية 8 MSS، في أول ثماني جولات للنقل يتصرفان بنفس الطريقة تزداد نافذة الاحتقان أسيا بسرعة أثناء البداية البطيئة لتصل إلى العتبة في الجولة 4 ، ثم تزداد خطيا حتى حدوث فقدان وإشعار مكرر ثلاثي بعد الجولة 8 فوراً لتصبح 12MSS وتعطى قيمة ssthresh حينها MSS • 6 = cwnd • 0.5 في حالة TCP Renoتعطي نافذة الاحتقان القيمة MSS •6 = cwnd ثم تزداد بشكل خطي، أما في حالة TCP Tahoe تعطي نافذة الاحتقان القيمة MSS 1 وتكبر بشكل أسي حتى تصل قيمة ssthresh، وعندها تزداد بشكل خطي.

وصف دقيق ل لإنتاجية في بروتوكول التحكم بالنقل (Macroscopic Description Of TCP Throughput)

ما متوسط الإنتاجية الاتصال TCP طويل الأجل؟ بإهمال مراحل البداية البطيئة بعد حدث نفاد المهلة، كونها معبرة جدا، وخلال فترة ذهاب وإياب معينة، يكون المعدل الذي يرسل فيه TCP البيانات كدالة لنافذة الاحتقان وقيمة RTT الحالية. إذا كان حجم النافذة W بايت، وزمن الذهاب والإياب الحالي RTT ثانية يكون معدل الإرسال w/RT T تقريبا. ثم يتحقق TCP من النطاق الترد دي الإضافي بزيادة w بمقدار Mss 1 كل RTT إلى أن يحصل حدث الفقدان، عندها سنعبر عن حجم النافذة بالرمز W. وبافتراض أن RT T وW ثابتان تقريبا خلال مدة الاتصال، يتراوح معدل الإرسال من W/2•RTT إلى W/RT T. وهذا يقودنا إلى نموذج مبسط جدا لسلوك TCP في الحالة الثابتة. تهمل الشبكة حزمة إذا زاد المعدل إلى W/RT T، ثم يهبط المعدل إلى النصف ويزداد بمقدار MSS/RTT كل RT T حتى يصل إلى W/RT T مرة أخرى، وتتكرر العملية. ولأن الإنتاجية تزداد خطية بين قيمين متطرفتين، يصبح متوسط الإنتاجية.0.75 W/RTT

بروتوكول التحكم بالنقل عبر مسارات ذات نطاق ترد دي عالي Over High-Bandwidth TCP paths

ما زالت الحاجة إلى تطوير بروتوكول التحكم بالنقل مستمرة لتوفير الاتصالات عالية السرعة اللازمة وأن ، التطبيقات الشبكة والحوسبة السحابية. على سبيل المثال، اعتبر أن حجم شريحة اتصال TCP 1500 بايت م RTT = 100 ، ولنفترض أننا نريد إرسال البيانات عبر هذا الاتصال بمعدل 10 Gbps. حسب المرجع [RFC 3649]، وباستخدام الصيغة اعلاه لتحقيق إنتاجية مقدارها 10 Gbps ، فإن متوسط حجم نافذة الاحتقان 83333 شريحة، وهي كثيرة قد تفقد واحدة منها. فكم من الشرائح تسمح خوارزمية ضبط الاحتقان بفقدانها مع تحقيق المعدل المطلوب (10 Gbps)؟ يمكن كتابة الإنتاجية كدالة لمعدل الفقدان)L(، وزمن الذهاب والإياب (RTT)، والحد الأقصى لحجم الشريحة (MSS) بالصيغة الآتية

متوسط الإنتاجية : =L√ VL 1.22.MSS/RTT

باستخدام هذه الصيغة، ولتحقيق الإنتاجية المطلوبة، نجد أن الخوارزمية قد تسمح بفقدان 2 × 10-10 فقط (أي فقدان شريحة واحدة كل 0 0 0 0,0 0 0,0 0 5,0 شريحة)، وهو معدل منخفض جدة.

3-8-1 العدل والمساواة Fairness

نعتبر أن لدينا عدد K من خطوط اتصال TCP لكل منها مسار مختلف، ولكنها جميعا تمر عبر وصلة تشكل عنق الزجاجة(bottleneck) معدل إرسالها R bps (أي أن لكل خط اتصال، جميع الوصلات على مسار خط الاتصال لا تعاني من الاحتقان ولديها قدرة إرسال وفيرة مقارنة بقدرة وصلة عنق الزجاجة). ولنفرض أن كل خط اتصال ينقل ملف كبيرا دون حركة مرور من نوع UDP عبر وصلة عنق الزجاجة يمكن تحقيق العدل في آلية ضبط الاحتقان إذا كان متوسط معدل الإرسال لكل خط اتصال R/K تقريبا، أي أن لكل خط حصة متساوية من عرض النطاق الترددي للوصلة

فهل خوارزمية الزيادة المضافة، النقصان المضاعف additive-increase, multiplicative- decrease: AIMD عادلة؟ إذا علمت أن خطوط الاتصال المختلفة قد تبدأ في أوقات مختلفة، وعليه، لها أحجام نوافذ مختلفة في نقطة زمنية محددة؟ لنعتبر أن لدينا حالة مبسطة تتكون من خطي اتصال TCPيشتركان في وصلة واحدة ذات معدل إرسال R، كما يوضح الشكل (37-3)

شكل 3-37 خط اتصال TCP يشتركان في وصلة عنق زجاجة (Bottleneck) واحدة اعتبر أن خطي الاتصال لهما نفس قيمة MSS ونفس قيمة RTT (أي إذا كان لهما نفس حجم نافذة الاحتقان فان لهما نفس الإنتاجية)، ولديهما كمية كبيرة من البيانات لإرسالها، وليس هناك خطوط اتصال TCP أو مخططات بيوت UDP تجتاز هذه الوصلة المشتركة لنتجاهل أيضا مرحلة البداية البطيئة ولنفرض آن خطوط الاتصال تعمل في وضع (collision avoidance: CA) lتجنب التصادم (AIMD) طول الوقت. بين الرسم البياني في الشكل (3-38) إنتاجية خطي الاتصال، ولتقسيم عرض النطاق الترددي بينهما بالتساوي يجب أن تقع الإنتاجية على النسيم بزاوية 45 درجة (أي حصة متساوية من عرض النطاق في الحالة المثالية يكون مجموع الإنتاجية ليما يساوي R و الهدف هو الوصول إلى إنتاجية تقترب من تقاطع خط الحصة المتساوية من عرض النطاق الترددي وخط استغلال النطاق الكامل

لنفترض أن أحجام نافذة TCP في نقطة زمنية معينة، بحلق خط الاتصال او 2 الإنتاجية المشار إليها في النقطة في الشكل)3-38) ولأن كمية عرض النطاق الذي يشترك فيه الخطان أقل من R ، فلن يحدث أو فقدان، وسيزيدان ناقلتيهما بمقدار IMSS لكل RTT تبعا لخوارزمية تجنب الاحتقان. وهكذا فان الإنتاجية المشتركة تقع على خط 45 درجه بدءا من النقطة A في نهاية المطاف، سيكون عرض النطق الذي يستهلكه الخطان أكبر من R ، ويحث فقدان الحزم لنفترض أن خطي الاتصال او 2 يعانين من فقدان الحرم عند تحقيق الإنتاجية المشار إليها في النقطة B، عندها يخفضان حجم نافذتيهما بمعامل 2 فيجاء قان الإنتاجية عند النقطة C)منتصف المسافة على متجه بينا في B وينتهي في نقطة الأصل ولأن عرض النطاق المشترك المستخدم أقل من R عند النقطة ، يزيد الحصان إنتاجيتهما مرة أخرى على خط 45 درجة بدها عن) في النهاية، سيتكرر فكان الحزم، مثلا، عند النقطة ، فيخفضان حجم نافذتيهما بمعامل 2 وهكذا .

عزيز الطالب، عليك ان تقع بار عرض النطاق الذي يحقه الخطان يتأرجح في نهاية المطاف على خط الحصة المتساوية، وآنهما سيتقاربان نحو هذا التصرف على الرغم من الافتراضات المالية في هذا السيناريو، فيه ولك شعورا لديك لما بقسم TCP عرض النطاق بالتساوي بين خطوط الاتصال من الناحية العملية المثالية، وبالتالي قد تحصل تطبيقات العميل الخام على حصص غير متساوية من عرض النطاق، وخاصة، عندما تشترك اتصالات متعددة في عنق الزجاجة، فإن الجلسات ذات RTT الأصغر نستحوذ على عرض النطاق الترددي المتاح بسرعة أكبر، وبالتالي سوف تتمتع بإنتاجية أعلى من غيرها

شكل 3-38 ، إنتاجية خطى الاتصال الأول و الثاني (Through put of Connections 1 and 2)

العدالة في UDP و اتصالات TCP المتوازية

عادة، لا تستخدم تطبيقات الوسائط المتعددة TCP لأنها لا تريد أن يقل معدل الإرسال نتيجة لضبط الاحتقان، لذا تلجأ إلى UDP التي تمكن من نقل الصوت والفيديو بمعدل ثابت، مع التساهل في أمر الحزم المفقودة ولمعالجة الأمر، يلجا TCP إلى الاتصالات المتوازية، إذ تستطيع التطبيقات فتح اتصالات متعدد على التوازي بين مضيفين، وهذا ما تقوم به متصفحات الإنترنت. مثلا، إذا كان معدل الإرسال لوصلة R ، وتدعم و تطبيقات عميل خادم، كل منها يستخدم اتصال TCP واحد، فإذا طلب تطبيق جديد اتصال TCP واحد، سيحصل كل تطبيق على معدل إرسال متساو تقريبا R / 10 . وإذا كان التطبيق الجديد يستخدم 11 اتصال TCP معا على التوازي، سيخصص له بشكل غير منصف أكثر من R/2 أي R.(11/20.)

مسرد المصطلحات

إعادة الإرسال (retransmit)

الوسط الأسي المرجح المتنقل (EWMA :Exponential Weighted Moving Average)

انحراف عينات زمن الذهاب والإياب عن تلك المخمنة (DevRTT) .

بروتكول نقطة لنقطة (Ethernet and PPP)

حجم الشريحة الأقصى ((MSS :maximum segment size)

رقم الاشعار (acknowledge number)

رقم التسلسل (sequence number)

رقم منفذ المصدر (source port number)

رقم منفذ الوجهة (destination port number)

زمن الذهاب والاياب (RTT :Round-Trip Time)

شريحة بروتوكول التحكم بالنقل(TCP Segment Structure)

فك التجميع (demultiplexing)

قطع البيانات (data chunks)

متوسط العينات المخمنة (Estimated RTT)

مجموع الاختبار (checksum)

معدل الإرسال المتاح (ABR: Available - bit rate) .

نفاد المهلة (timeout)

وحدة الارسال القصوى (MTU maximum transmission unit)

وضع النقل غير المتزامن (ATM:Asynchronous transfer mode)