طبقة الشبكـة

The Network Layer

محتويات الفصل:

- مقدمة
- شبكات الدائرة الافتراضية وشبكات وحدات البيانات
- ماذا بداخل الموجّه؟
- بروتوكول الإنترنت (IP): التمرير والعنونة في الإنترنت
- خوارزميات التوجيه
- التوجيه في شبكة الإنترنت
- توجيه البث الإذاعي (العام) والتوجيه المتعدد (الجماعي)
- الخلاصة

عرفنا في الفصل السابق أن طبقة النقل توفر أشكالاً مختلفة للاتصال من عملية إلى عملية بالاعتماد على خدمة طبقة الشبكة للاتصال من مضيف إلى مضيف. وتعلمنا أيضاً أن طبقة النقل تؤدي هذا الدور دون أية معرفة عن كيفية تحقيق طبقة الشبكة في واقع الأمر لهذه الخدمة. لذا ربما تتساءل الآن: ما الذي تحت قلنسوة خدمة الاتصال من مضيف إلى مضيف، وما الذي يجعلها تحدث؟ وكيف تتم؟

في هذا الفصل سوف نتعلم كيف تحقق طبقة الشبكة بالضبط خدمة الاتصال من مضيف إلى مضيف. وسوف نرى أنه على خلاف طبقة النقل يوجد جزء من طبقة الشبكة في كل مضيف وموجّه في الشبكة. ولهذا السبب فإن بروتوكولات طبقة الشبكة من بين البروتوكولات الأكثر تحدّياً في رصّة البروتوكولات (ولذا فقد حظيت باهتمام كبير!).

وطبقة الشبكة أيضاً هي إحدى الطبقات الأكثر تعقيداً في رصّة البروتوكولات، ولذا فسيكون لدينا الكثير من الموضوعات للتغطية هنا. سنبدأ دراستنا بنظرة عامة عن طبقة الشبكة والخدمات التي يمكن أن توفرها. ثم سنعاود التعرض مرةً أخرى للطريقتين الرئيستين لهيكلة طبقة الشبكة لتوصيل الرزم: نموذج إرسال وحدات البيانات (datagram model) ونموذج الدائرة الافتراضية (virtual circuit model)، واللتين سبق أن تناولناهما لأول مرة في الفصل الأول. وسنرى الدور الأساسي الذي تلعبه "العنونة" (addressing) في توصيل الرزمة إلى مضيف الوجهة.

وسنجري في هذا الفصل تمييزاً مهماً بين وظيفة التمرير (forwarding) ووظيفة التوجيه (routing) لطبقة الشبكة. حيث يتضمّن "التمرير" نقل الرزمة من وصلة داخلة إلى وصلة خارجة ضمن "موجّه" واحد؛ بينما يشمل التوجيه كل موجّهات الشبكة والتي تحدد تفاعلاتها الجماعية ـ عن طريق بروتوكولات التوجيه ـ المسارات التي تأخذها الرزم أثناء رحلتها من المصدر إلى الوجهة. وسوف يساعدك

تذكر هذا التمييز (الفرق) كلما تقدمت خلال هذا الفصل على وضع العديد من الموضوعات التي سنتناولها في سياق ملائم.

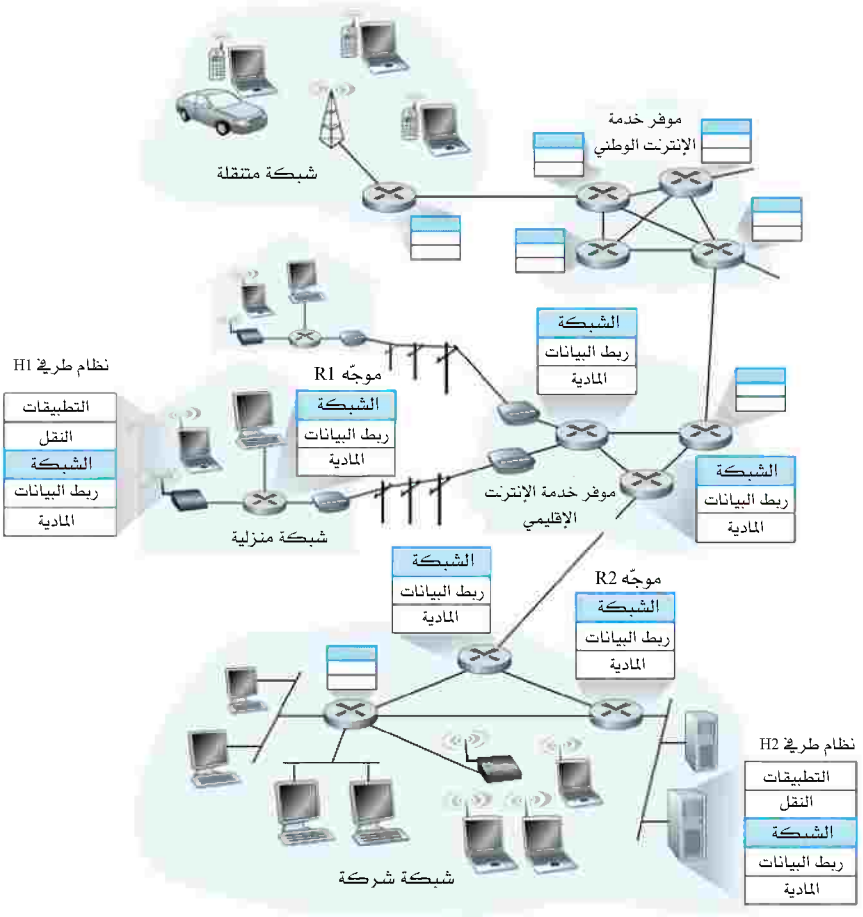
ولكي نُعمّق فهمنا لتمرير الرزم سننظر "داخل" موجّه لنرى بنيته المعمارية وتنظيم مكوناته المادية. ثم ننظر إلى تمرير الرزم في الإنترنت سوية مع بروتوكول الإنترنت الشهير IP. وسنفحص العنونة في طبقة الشبكة وصيغة رزمة بيانات الإصدار الرابع لبروتوكول الإنترنت (IPv4). وسوف ندرس أيضاً ترجمة عناوين الشبكة (NAT)، وتجزئة رزم البيانات (fragmentation)، وبروتوكول رسائل التحكم في الإنترنت (ICMP)، وبروتوكول IPv6.

بعد ذلك سنحوّل انتباهنا إلى وظيفة التوجيه في طبقة الشبكة، حيث سنرى أن مهمة خوارزمية التوجيه هي تحديد مسارات (طرق) جيدة من مصادر البيانات إلى وجهاتها. وسندرس أولاً نظرية خوارزميات التوجيه مع التركيز على النوعين الأكثر انتشاراً من الخوارزميات: خوارزمية "حالة الوصلة" (link state) وخوارزمية "متجه المسافة" (distance vector). ولأن خوارزميات التوجيه تزداد تعقيداً إلى حدٍّ كبير كلما زاد عدد الموجّهات في الشبكة، فإن أساليب التوجيه الهرمي (hierarchical routing) ستكون أيضاً محل اهتمامنا. وسنرى كيف توضع النظرية موضع التطبيق عندما نغطي بروتوكولات التوجيه داخل النظم المستقلة ذاتياً (intra-autonomous systems) في الإنترنت (مثل RIP وOSPF وIS-IS) وكذلك بروتوكول BGP للتوجيه بين النظم المستقلة ذاتياً (inter-autonomous systems) بها. وسننهي هذا الفصل بمناقشة توجيه البث الإذاعي (broadcast) والإرسال المتعدّد (multicast).

وباختصار يتكون هذا الفصل من ثلاثة أقسام رئيسة: حيث يغطي القسم الأول (الجزءان 4-1 و4-2) وظائف وخدمات طبقة الشبكة، ويغطي القسم الثاني (الجزءان 4-3 و4-4) التمرير، وأخيراً يغطي القسم الثالث (من الجزء 4-5 إلى الجزء 4-7) التوجيه.

1-4 مقدمة

يوضح الشكل 4-1 شبكة بسيطة ذات مضيفين H1 وH2، وعدّة موجّهات على المسار بين H1 وH2. افترض أن H1 يرسل معلومات إلى H2، ولننظر إلى دور طبقة الشبكة في كلا المضيفين وفي الموجّهات المتوسطة بينهما. تأخذ طبقة



الشكل 4-1 طبقة الشبكة.

الشبكة في H1 "قطع البيانات" (segments) من طبقة النقل في H1 ، وتغلف كل قطعة في "رزمة بيانات" (packet)، ثم بعد ذلك ترسل تلك الرزم إلى الموجّه المجاور R1. في مضيف الاستقبال H2، تستلم طبقة الشبكة رزم البيانات من الموجّه المجاور R2، وتستخرج "قطع بيانات" طبقة النقل، ثم تسلم تلك القطع إلى طبقة النقل في H2. إن الدور الأساسي للموجّهات هو إرسال رزم البيانات من الوصلات الداخلة الى الوصلات الخارجة. لاحظ أن رصّة البروتوكولات على الموجّهات في الشكل 4-1 مقطوعة (أى بدون طبقات عليا فوق طبقة الشبكة)، وذلك لأن الموجّهات لا تستخدم بروتوكولات طبقتي النقل والتطبيق كالتي درسناها في الفصلين الثانى والثالث (إلا لأغراض التحكم).

4-1-1 التمرير والتوجيه

هكذا قد يبدو – بشكلٍ خادع – أن دور طبقة الشبكة بسيط، فهي تنقل الرزم من مضيف الإرسال إلى مضيف الاستقبال فقط. يمكننا من البداية التعرف على وظيفتين مهمتين لطبقة الشبكة لتتمكن من أداء هذا الدور:

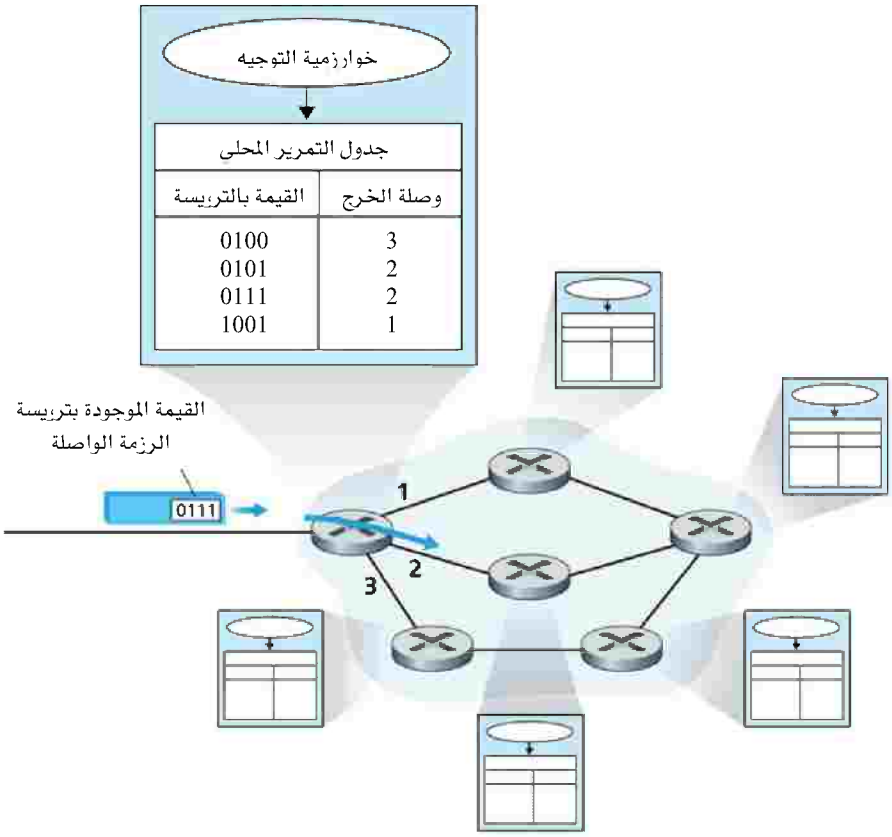
- التمرير: عندما تصل رزمة إلى وصلة مدخل (input link) للموجّه يجب عليه أن ينقل تلك الرزمة إلى وصلة مخرج (output link) مناسبة. على سبيل المثال عند وصول رزمة من المضيف H1 إلى الموجّه R1 يجب عليه أن يرسلها إلى الموجّه التالي على المسار إلى H2. في الجزء 4-3 سننظر داخل موجّه لنرى كيف تُرسل رزمة في الواقع من وصلة مدخل إلى وصلة مخرج على موجّه.
- التوجيه: يجب أن تقرّر طبقة الشبكة المسار الذي تتبعه الرزم وهي تتدفّق من المُرسِل إلى المستقبل. ويطلق على الخوارزميات التي تحسب هذه المسارات "خوارزميات التوجيه". على سبيل المثال ستحدد خوارزمية التوجيه الطريق الذي تتدفق خلاله الرزم من H1 إلى H2.

غالباً ما يُستخدم المصطلحان "تمرير" و"توجيه" بمعنى واحد من قِبَل بعض المؤلفين أثناء مناقشة طبقة الشبكة. لكننا في هذا الكتاب سوف نستخدمهما بدقة أكثر. يشير "التمرير" إلى عمل الموجّه المحلي لنقل رزمة من واجهة وصلة مدخل

(input link interface) إلى واجهة وصلة مخرج (output link interface) مناسبة. بينما يشير "التوجيه" إلى عملية تتم في كافة أنحاء الشبكة لتحديد المسارات التي تأخذها الرزم من المصدر إلى وجهتها النهائية. وبالتناظر تأمَّل الرحلة من بينسلفانيا إلى فلوريدا التي قام بها المسافر في الجزء 1-3-2. أثناء تلك الرحلة يمر المسافر خلال العديد من المفارق في الطريق حتى يصل إلى فلوريدا. يمكن أن نعتبر "التمرير" كعملية عبور مفرق واحد: تدخل السيارة المفرق من طريق ما ثم تقرّر الطريق الذي يجب أن تأخذه لتغادر المفرق. ويمكن أن نعتبر "التوجيه" كعملية تخطيط الرحلة من بينسلفانيا إلى فلوريدا: فقبل أن يبدأ الرحلة يراجع السائق الخريطة ويختار طريقاً واحداً من بين عدة طرق محتملة. يتكون كل طريق من سلسلة من القطع المتصلة عند المفارق. في هذا الفصل سنفحص أولاً نماذج خدمة طبقة الشبكة، ثم سنركّز على مواضيع طبقة الشبكة المتعلقة بالتمرير، وبعد ذلك نحوّل انتباهنا إلى التوجيه.

يوجد في كل موجّه "جدول تمرير" (forwarding table). عندما تصل رزمة إلى الموجّه يقوم بفحص قيمة حقل في ترويسة الرزمة ويستعملها للبحث في جدول التمرير لديه لتحديد أيٍّ من واجهات الوصلات يجب إرسال الرزمة إليها. ويمكن على حسب بروتوكول طبقة الشبكة المستخدم أن تمثل هذه القيمة في ترويسة الرزمة عنوان وجهة الرزمة أو إشارة تحدد التوصيلة التي تنتمي لها الرزمة. يوضح الشكل 4-2 مثالاً لهذه العملية. ففي هذا الشكل تصل رزمة تحتوي على القيمة 0111 في حقل الترويسة إلى الموجّه. يبحث الموجّه في جدول التمرير لديه، ويقرّر أن واجهة وصلة المخرج لهذه الرزمة هي الواجهة 2، ومن ثمّ يرسل الموجّه تلك الرزمة داخلياً إلى الواجهة 2. في الجزء 4-3 سننظر داخل موجّه ونفحص وظيفة التمرير بتفصيل أكثر.

قد تتساءل الآن كيف تُعَدّ جداول التمرير في الموجّهات؟ إن هذه قضية حاسمة، فهي توضح التفاعل الهام بين التوجيه والتمرير. كما هو مبين في الشكل 4-2 تقرّر خوارزمية التوجيه القيم التي توضع في جداول التمرير بالموجّهات. قد تكون خوارزمية التوجيه مركزية (أي توجد خوارزمية تُنفّذ على موقع مركزي ثم تُحمِّل معلومات التوجيه إلى كل موجّه) أو غير مركزية (أي يوجد جزء من



الشكل 4-2 خوارزميات التوجيه تحدد القيم الموجودة بجداول التمرير.

خوارزمية التوجيه الموزّعة يُنفّذ في كل موجّه). في كلا الحالتين يتلقى الموجّه رسائل بروتوكول التوجيه التي تستعمل لإعداد جدول التمرير لديه. ويمكن أن نوضح الأهداف الأكثر تمييزاً واختلافاً لوظائف التمرير والتوجيه بالنظر في حالة افتراضية (وغير واقعية ولكن ممكنة تقنياً) لشبكة يتم فيها إعداد كل جداول التمرير مباشرة بواسطة "مشغلي الشبكة" الموجودين فعلياً عند الموجّهات. في هذه الحالة لا نحتاج إلى بروتوكولات توجيه! يحتاج مشغلو الشبكة بالطبع للتفاعل مع بعضهم البعض للتأكد من أن جداول التمرير قد أعدت بطريقة تضمن وصول الرزم إلى وجهاتها النهائية. من المحتمل أيضاً أن يكون مثل هذا الإعداد البشري أكثر

عرضة للأخطاء وأبطأ بكثير للاستجابة للتغييرات في طبوغرافية الشبكة من استخدام بروتوكول للتوجيه. ومن حسن الحظ إن كل الشبكات فيها كلا الوظيفتين: التمرير والتوجيه!

وبينما نحن بصدد مصطلحات الشبكات، يجدر بنا التنويه إلى مصطلحين آخرين يستعملان في أغلب الأحيان بشكلٍ متبادل، ولكننا سوف نستعملهما بعناية أكثر. سنخصص المصطلح "محوّل رزم" (packet switch) ليعنى أداة عامة لتحويل الرزم حيث تنقل الرزم من واجهة وصلة المدخل الى واجهة وصلة المخرج طبقاً للقيم الموجودة بحقل ما في ترويسة كل رزمة. تبنى بعض محوّلات الرزم – وتعرف بمحوّلات طبقة ربط البيانات (والتي سنتناولها في الفصل الخامس) – قرار التمرير على القيمة الموجودة في حقل من حقول الترويسة لطبقة ربط البيانات. في حين تبني محوّلات الرزم الأخرى – وتسمى الموجّهات – قرار التمرير على القيمة الموجودة في حقل من حقول ترويسة طبقة الشبكة. (ولتقدير أهمية هذا التمييز قد تحتاج إلى مراجعة الجزء 1-5-2 حيث ناقشنا رزم بيانات طبقة الشبكة وإطارات (frames) طبقة ربط البيانات والعلاقة بينهما). ولأن تركيزنا في هذا الفصل على طبقة الشبكة سنستعمل المصطلح "موجّه" بدلاً من "محوّل الرزم". وحتى عندما نتحدّث عن محوّلات الرزم في شبكات الدائرة الافتراضية (والتي سيتم مناقشتها قريباً) سوف نستعمل المصطلح موجّه.

إعداد التوصيلة (Connection Setup)

ذكرنا للتو أن طبقة الشبكة لها وظيفتان مهمتان: التمرير والتوجيه. ولكننا سنرى قريباً أن في بعض شبكات الحاسب توجد في الحقيقة وظيفة ثالثة مهمة لطبقة الشبكة؛ ألا وهي إعداد التوصيلة. تذكّر من دراستنا لبروتوكول TCP أن المصافحة ثلاثية الاتجاه (three-way handshake) مطلوبة قبل تدفق البيانات من المُرسِل إلى المستقبل. وهذا يسمح للمُرسِل والمُستقبِل بإعداد المعلومات المطلوبة عن الحالة، كالرقم التسلسلي والحجم الابتدائي لنافذة التحكم في التدفق (flow control window). وبأسلوب مماثل تتطلب بعض البنى المعمارية لطبقة الشبكة

(على سبيل المثال شبكة نمط النقل اللاتزامني ATM وشبكة تحويل الإطارات frame relay)، على خلاف شبكة الإنترنت، أن تتصافح الموجّهات على طول المسار المختار من المصدر إلى الوجهة مع بعضها البعض لكي تقوم بإعداد معلومات الحالة قبل أن تبدأ رزم البيانات في التدفق خلال المسار. تسمى هذه العملية في طبقة الشبكة إعداد التوصيلة (وسوف نتناولها في الجزء 4-2).

4-1-2 نماذج الخدمة للشبكة

قبل التنقيب في طبقة الشبكة دعنا نلقِ نظرة أشمل على أنواع الخدمات المختلفة التي قد تقدمها طبقة الشبكة. عندما ترسل طبقة النقل في مضيف الإرسال رزمة إلى الشبكة (أى تمررها الى طبقة الشبكة في مضيف الإرسال) هل يمكن أن تعتمد طبقة النقل على طبقة الشبكة لتسليم الرزمة إلى وجهتها النهائية؟ وعند إرسال رزم متعدّدة هل تصل إلى طبقة النقل في مضيف الاستقبال بنفس ترتيبها عند الإرسال؟ وهل مقدار الوقت بين إرسال رزمتين متتاليتين هو نفسه تماماً مقدار الوقت بين استقبالهما؟ وهل توفر الشبكة أي تغذية مرتجعة (feedback) حول الازدحام (congestion) في الشبكة؟ وما هو التمثيل المجرّد (أي الصفات) للقناة التي تربط بين طبقة النقل في مضيف الإرسال ومضيف الاستقبال؟ تتحدد الإجابة على هذه الأسئلة وأسئلة أخرى بنموذج الخدمة الذي توفره طبقة الشبكة. يُعرِّف نموذج خدمة الشبكة خصائص نقل الرزم من طرف إلى طرف بين حافة وأخرى للشبكة (أي بين الأنظمة الطرفية للإرسال والاستقبال).

دعنا الآن نراجع بعض الخدمات المحتملة التي يمكن أن توفرها طبقة الشبكة. عندما ترسل طبقة النقل رزمة إلى طبقة الشبكة في مضيف الإرسال فإن الخدمات التي يمكن أن توفرها طبقة الشبكة تشمل:

- ضمان التوصيل: تضمن هذه الخدمة وصول الرزمة في النهاية إلى وجهتها.
- ضمان التوصيل مع تأخير محدود: هذه الخدمة لا تضمن فقط توصيل الرزمة ولكن توصيلها في زمن تأخير محدد من مضيف إلى مضيف (مثلاً في خلال 100 ميلى ثانية).

كما يمكن أن توفر طبقة الشبكة علاوةً على ذلك الخدمات التالية لتدفق الرزم بين مصدر معين ووجهة معينة:

- توصيل الرزم بنفس الترتيب: تضمن هذه الخدمة وصول الرزم إلى وجهتها بنفس الترتيب الذي أرسلت به.
- ضمان الحد الأدنى للحيز الترددي: تحاكي هذه الخدمة في طبقة الشبكة سلوك وصلة إرسال ذات معدل بيانات محدد (على سبيل المثال ميجابت واحد في الثانية) بين مضيفات الإرسال والاستقبال (بالرغم من أن المسار الفعلي من طرف إلى طرف قد يتكون من عدّة وصلات مادية). طالما يرسل المضيف البتات (كجزء من الرزم) بمعدل أقل من المعدل المحدّد فإن الرزم لا تفقد، وتصل كل رزمة في غضون تأخير "من مضيف إلى مضيف" محدد مسبقاً (مثلاً خلال 40 ميلي ثانية).
- ضمان الحد الأقصى للتفاوت الزمني للتأخير (delay jitter): تضمن هذه الخدمة أن يكون مقدار الوقت بين رزمتين متتاليتين عند المُرسِل يساوي مقدار الوقت بينهما عند الوجهة النهائية (أو ألا تتجاوز التغيرات في الفترة الزمنية بينهما قيمة محدّدة).
- خدمات الأمن: باستعمال مفتاح جلسة سري معروف فقط للمصدر والوجهة يمكن لطبقة الشبكة في مضيف المصدر أن تشفّر (encrypt) بيانات كل رزمة تُرسلها إلى مضيف الوجهة. وستكون طبقة الشبكة في مضيّف الوجهة مسؤولة عن حل الشفرة (decrypt) واسترجاع الشكل الأصلي للبيانات. بمثل هذا الخدمة ستُضمن السريّة (الخصوصية) لكل قطع بيانات طبقة النقل (TCP وUDP) بين مضيفي المصدر والوجهة. وبالإضافة إلى السريّة يمكن أن توفر طبقة الشبكة خدمات أخرى كسلامة البيانات (data integrity) والتوثيق (authentication) للتحقق من المصدر.

هذه فقط قائمة جزئية من الخدمات التي يمكن أن توفرها طبقة الشبكة؛ فهناك العديد من الاختلافات الممكنة التي لا تحصى.

توفر طبقة الشبكة في الإنترنت خدمة واحدة تعرف بخدمة "أفضل جهد" (best-effort). يتضح من الجدول 4-1 أن خدمة "أفضل جهد" هي تعبير تلطيفي لـ "لا خدمة على الاطلاق". فمع هذه الخدمة لا ضمان للإبقاء على الوقت بين الرزم، ولا ضمان لتوصيل الرزم المرسلة بنفس الترتيب، ولا ضمان لتوصيلها نهائياً. بهذا التعريف فإن الشبكة التي لم توصّل أي رزمة إلى الوجهة توافق تعريف خدمة أفضل جهد للتوصيل. ومع ذلك فكما سنناقش بعد قليل هناك أسباب معقولة وراء وجود هذا النموذج لخدمة الحد الأدنى لطبقة الشبكة. وسوف نغطّي نماذج خدمة إضافية – ما زالت في مرحلة التطوير – للإنترنت في الفصل السابع.

البنية المعمارية للشبكة	نموذج الخدمة	الحيز الترددي	عدم الفقد	الترتيب	التوقيت	الإشارة إلى الازدحام
الإنترنت	أفضل جهد	غير مضمون	غير مضمون	أي ترتيب	غير مضمون	غير متوفر
ATM	معدل ثابت للبتات (CBR)	المعدل الثابت مضمون	مضمون	مضمون	مضمون	لا يحدث ازدحام
ATM	معدل البتات المتاح (ABR)	المعدل الأدنى مضمون	غير مضمون	مضمون	غير مضمون	متوفر

الجدول 4-1 نماذج الخدمة في الإنترنت وشبكة ATM.

هناك أيضاً بنيات معمارية أخرى لشبكات تُعرِّف وتُحقِّق نماذج خدمة تتجاوز خدمة أفضل جهد في الإنترنت، لكنها خارج نطاق هذا الكتاب. على سبيل المثال توفر بنية شبكة ATM [ATM Forum 2007; Black 1995] عدة نماذج للخدمة وهذا يعني أنه يمكن عمل اتصالات بأنواع مختلفة للخدمة ضمن نفس الشبكة. إن مناقشة كيفية توفير شبكة ATM لمثل هذه الخدمات تقع خارج نطاق هذا الكتاب، فهدفنا هنا فقط هو التنويه عن وجود بدائل لنموذج "خدمة أفضل جهد" المستخدم في الإنترنت. اثنان من نماذج خدمة ATM الأكثر أهميّة هما: خدمة معدل البتات الثابت (Constant Bit Rate (CBR)) وخدمة معدل البتات المتوفر (Available Bit Rate (ABR)):

- خدمة معدل البتات الثابت (CBR) لشبكة ATM: كان هذا أول نموذج خدمة قياسي لشبكة ATM، وهو يعكس اهتماماً مبكّراً من شركات الهاتف بشبكة ATM ومدى ملاءمة خدمة CBR لنقل بيانات الصوت والفيديو ذات المعدل الثابت في الوقت الحقيقي. إن هدف خدمة CBR بسيط من حيث المفهوم: ألا وهو توفير تدفق من الرزم (المعروفة بالخلايا (cells) في مصطلحات ATM) خلال أنبوب افتراضي له نفس الخواص تماماً كما لو كان وصلة إرسال مخصصة ذات حيز ترددي ثابت بين المضيفين المُرسِل والمستقبل. ومع خدمة CBR يُحمَل تدفق من خلايا ATM عبر الشبكة بطريقة بحيث يضمن أن يكون تأخير الخلايا من طرف إلى طرف (end-to-end delay) والتفاوت الزمني للتأخير (delay jitter) ومعدل فقد الخلايا أقل من قيم محدّدة. هذه القيم يُتّفق عليها بين المضيف المُرسِل وشبكة ATM عند بداية إرساء اتصال بمعدل إرسال ثابت (CBR).
- خدمة معدل البتات المتوفر (ABR) لشبكة ATM: في حين تقدم الإنترنت ما يسمى بخدمة أفضل جهد، فإن خدمة ABR لشبكة ATM يمكن تمييزها على أنها تعديل بعض الشيء لخدمة أفضل جهد. وكما هو الحال مع نموذج خدمة الإنترنت قد تُفقد الخلايا مع خدمة ABR. لكن على خلاف الإنترنت فإنه لا يمكن أن يختلف ترتيب الخلايا (بالرغم من أن البعض قد يفقد)، والمعدل الأدنى لإرسال الخلايا ((Minimum Cell Rate (MCR) مضمون لاتصال يستخدم خدمة ABR. وإذا كانت الموارد المتاحة في الشبكة في وقت ما كافية يمكن للمُرسِل أيضاً إرسال الخلايا بنجاح بنسبة أعلى من MCR. إضافة إلى ذلك – كما رأينا في الجزء 3-6 – يمكن أن توفر خدمة ABR في شبكة ATM تغذية مرتجعة إلى المُرسِل على شكل بت إخطار الازدحام (congestion notification bit) أو معدل إرسال صريح (explicit transmission rate) لضبط معدل الإرسال بين MCR والمعدل الأقصى المسموح به.

4-2 شبكات الدائرة الافتراضية وشبكات وحدات البيانات

تذكّر من الفصل الثالث أن طبقة النقل يمكن أن تقدم للتطبيقات خدمة لاتوصيلية (connectionless) أو خدمة توصيلية (connection-oriented). على سبيل المثال تزوّد طبقة النقل في الإنترنت كل تطبيق بإمكانية الاختيار بين خدمتين: UDP (خدمة لاتوصيلية) أو TCP (خدمة توصيلية). وبطريقة مماثلة يمكن أن تقدّم طبقة الشبكة أيضاً خدمة لاتوصيلية أو خدمة توصيلية، وهي توازي خدمات طبقة النقل من عدّة أوجه. على سبيل المثال تبدأ الخدمة التوصيلية في طبقة الشبكة بالمصافحة (handshaking) بين مضيفي المصدر والوجهة، بينما لا توجد أية تمهيدات للمصافحة في خدمة طبقة الشبكة اللاتوصيلية.

بالرغم من وجود بعض أوجه التشابه بين خدمات طبقة الشبكة اللاتوصيلية والتوصيلية مع خدمات طبقة النقل إلا أن هناك اختلافات جوهرية:

- تكون الخدمات في طبقة الشبكة من مضيف إلى مضيف وتوفرها طبقة الشبكة لطبقة النقل. أما في طبقة النقل فتكون الخدمات من عملية إلى عملية وتوفرها طبقة النقل لطبقة التطبيقات.
- في كل البنى الرئيسة لشبكات الحاسب حتى الآن (الإنترنت، وشبكة ATM، وشبكة تحويل الإطارات، وغيرها) توفر طبقة الشبكة إمّا خدمة لاتوصيلية من مضيف إلى مضيف أو خدمة توصيلية من مضيف إلى مضيف؛ لكن ليسا معاً. تسمى شبكات الحاسب التي تقدّم خدمة توصيلية فقط في طبقة الشبكة شبكات الدائرة الافتراضية ((Virtual Circuit (VC)؛ في حين تسمى شبكات الحاسب التي تقدّم خدمة لا توصيلية فقط في طبقة الشبكة شبكات وحدات البيانات (datagram networks).
- يختلف تحقيق الخدمة التوصيلية في طبقة النقل بشكلٍ أساسي عن تحقيقها في طبقة الشبكة. سبق أن رأينا في الفصل السابق أن خدمة طبقة النقل التوصيلية مطبّقة في حافة الشبكة في الأنظمة الطرفية، لكن كما سنرى بعد قليل توجد الخدمة التوصيلية في طبقة الشبكة في الموجهّات الموجودة في قلب الشبكة بالإضافة إلى وجودها في الأنظمة الطرفية.

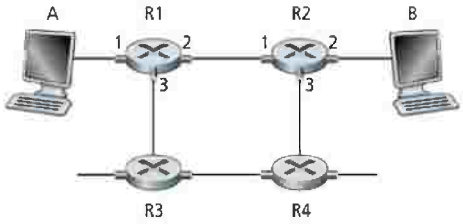
إن شبكات الدائرة الافتراضية وشبكات وحدات البيانات نوعان أساسيان لشبكات الحاسب، وهما يستعملان معلومات مختلفة جداً في اتخاذ قرارات التمرير. دعنا الآن نلقِ نظرة أقرب في كيفية تحقيقهما.

1-2-4 شبكة الدائرة الافتراضية

لقد عرفنا أن الإنترنت هي إحدى شبكات وحدات البيانات (datagram networks). ولكن هناك العديد من بنى الشبكات المعمارية البديلة – بما في ذلك شبكة ATM وشبكة تحويل الإطارات – تستخدم الدائرة الافتراضية، ولذا فهي تستعمل التوصيلات في طبقة الشبكة، والتي يطلق عليها "دوائر افتراضية" (VCs). دعنا الآن نرى كيفية تحقيق خدمة الدائرة الافتراضية VC في شبكات الحاسب.

تتكون الدائرة الافتراضية من: (1) مسار (أي سلسلة من الوصلات والموجّهات) بين مضيفي المصدر والوجهة، (2) أرقام الدائرة الافتراضية وتشمل رقماً واحداً لكل وصلة على طول المسار، (3) مدخلات في جدول التمرير في كل موجّه على طول المسار. تحمل كل رزمة تنتمي لدائرة افتراضية رقماً للدائرة الافتراضية (VC number) في ترويستها. ولأن الدائرة الافتراضية قد يكون لها رقم مختلف على كل وصلة فعند مرور الرزمة على المسار يجب أن يستبدل كل موجّه بيني رقم الدائرة الافتراضية برقم جديد يحصل عليه من جدول التمرير.

لتوضيح هذا المفهوم انظر إلى الشبكة الموضحة في الشكل 4-3. تمثل الأرقام الموجودة بجانب الوصلات للموجّه R1 أرقام واجهات الوصلات للموجّه. افترض الآن أن المضيف A يطلب من الشبكة إعداد دائرة افتراضية VC بينه وبين المضيف B. وافترض أيضاً أن الشبكة تختار المسار A-R1-R2-B وتخصص الأرقام 12، 22، 32 للوصلات الثلاث في هذا المسار لهذه الدائرة الافتراضية. في هذه الحالة عندما تغادر رزمة المضيف A تكون القيمة في حقل رقم الدائرة الافتراضية في ترويسة الرزمة 12، وعندما تغادر R1 تصبح القيمة 22، وعندما تغادر R2 تصبح القيمة 32.



الشكل 4-3 مثال بسيط لشبكة الدائرة الافتراضية.

كيف يُقرّر الموجّه رقم الدائرة الافتراضية الجديد لرزمة تعبر خلاله؟ في شبكة الدائرة الافتراضية يتضمّن كل جدول تمرير للموجّه ترجمة لأرقام الدوائر الافتراضية. على سبيل المثال جدول التمرير في الموجّه R1 قد يبدو مثلاً كما يلي:

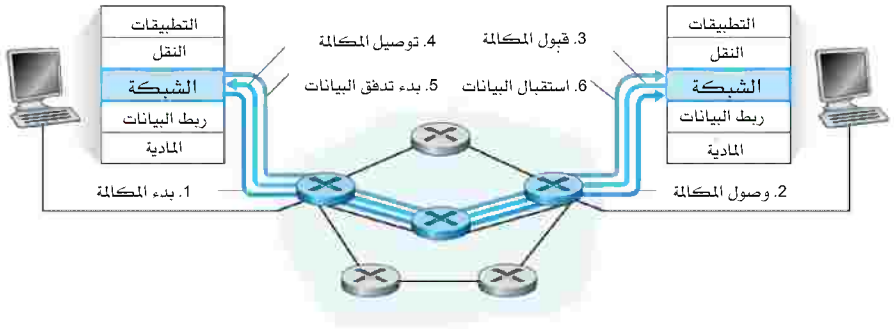
واجهة المدخل	رقم الدائرة الافتراضية للمدخل	واجهة المخرج	رقم الدائرة الافتراضية للمخرج
1	12	2	22
2	63	1	18
3	7	2	17
1	97	3	87
...	...	...	...

عندما تؤسّس VC جديدة عبر موجّه يضاف مُدخل جديد إلى جدول التمرير لديه. وبنفس الطريقة عند إنهاء VC تُزَال المُدخلات المتعلقة بها من كل جداول التمرير على طول مسارها.

قد تتساءل لماذا لا تحتفظ الرزمة بنفس رقم VC على كل وصلة من الوصلات على طول المسار. يرجع ذلك إلى سببين. الأول هو أن تغيير الرقم من وصلة إلى وصلة يقلل طول حقل VC في ترويسة الرزمة. والثاني – وهو الأهم – أن عملية إعداد VC تكون أبسط بكثير عند السماح لرقم VC بالتغيّر لكل وصلة على طول مسار VC. وبالتحديد باستعمال أرقام VC متعدّدة يمكن أن تختار كل وصلة على المسار رقم VC بشكلٍ مستقل عن أرقام VC التي يتم اختيارها على الوصلات الأخرى على طول المسار. أما إذا تطلبنا أن يكون رقم VC ثابتاً لكل الوصلات على طول

المسار فإن الموجهات يجب أن تتبادل وتعالج عدداً كبيراً من الرسائل للموافقة على رقم مشترك (مثلاً رقم يكون غير مستخدم من قِبَل أي دائرة افتراضية أخرى موجودة حالياً في تلك الموجّهات) لكي يُستعمَل لهذه التوصيلة الجديدة.

في شبكة VC يجب أن تحتفظ موجّهات الشبكة بمعلومات حالة عن التوصيلات الموجودة حالياً. بالتحديد في كل مرة يتم تأسيس توصيلة جديدة عبر موجّه يجب أن يضاف مُدخل جديد عن التوصيلة إلى جدول التمرير، وفي كل مرة يتم إنهاء توصيلة يجب أن يحذف المُدخل المتعلق بها من الجدول. لاحظ أنه حتى في حالة عدم وجود ترجمة لأرقام الدوائر الافتراضية ما زال من الضروري الاحتفاظ بمعلومات حالة عن التوصيلات تقرن أرقام VC بأرقام واجهات المخرج. تُعتبر قضية احتفاظ الموجّه أو عدم احتفاظه بمعلومات حالة لكل توصيلة موجودة حالياً من القضايا الهامة والتي سنعود إليها مراراً وتكراراً في هذا الكتاب.



الشكل 4-4 إعداد دائرة افتراضية.

هناك ثلاث مراحل مميّزة للدائرة الافتراضية:

- إعداد VC: أثناء مرحلة إعداد VC تتصل طبقة النقل للمُرسِل بطبقة الشبكة، وتحدد عنوان المستقبل، وتنتظر حتى تقوم الشبكة بإعداد VC. تحدد طبقة الشبكة المسار بين المُرسِل والمستقبل، أي سلسلة الوصلات والموجّهات التي تمر خلالها كل رزم الدائرة الافتراضية VC. كما تحدد طبقة الشبكة أيضاً رقم VC لكل وصلة على طول المسار. وفي النهاية

تضيف طبقة الشبكة مُدخلاً في جدول التمرير في كل موجّه على طول المسار. أثناء إعداد VC قد تحجز طبقة الشبكة أيضاً الموارد اللازمة (على سبيل المثال الحيز الترددي) على طول مسار الدائرة الافتراضية.

- نقل البيانات: كما هو موضح في الشكل 4-4 بمجرد إعداد دائرة افتراضية يمكن أن تبدأ الرزم بالتدفق خلال تلك الدائرة الافتراضية.
- إنهاء (فض) الدائرة الافتراضية: تبدأ هذه الخطوة عندما يخبر المُرسِل (أو المستقبل) طبقة الشبكة عن رغبته في إنهاء الدائرة الافتراضية. بعد ذلك تخبر طبقة الشبكة عادةً النظام الطرفي على الجانب الآخر للشبكة لإنهاء الاتصال وتُعدَّل جداول التمرير في كل الموجّهات على المسار للإشارة إلى أن الدائرة الافتراضية لم تعد قائمة.

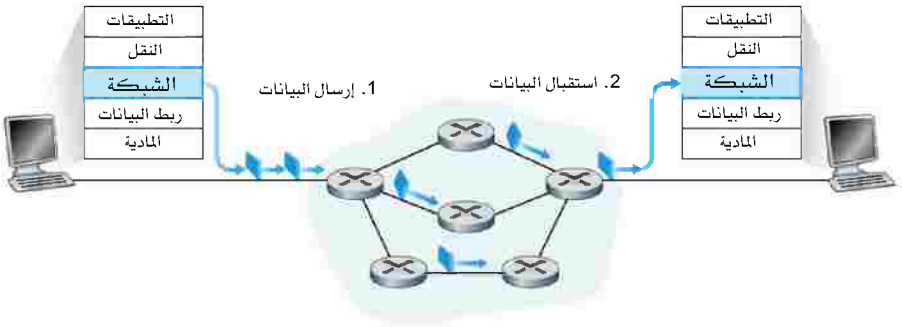
هناك اختلاف دقيق ولكنه مهم بين إعداد VC في طبقة الشبكة وإعداد التوصيلة في طبقة النقل (على سبيل المثال المصافحة الثلاثية لبروتوكول TCP الذي درسناه في الفصل الثالث). يقتصر إعداد التوصيلة في طبقة النقل على النظامين الطرفيين فقط، فهما وحدهما يحددان البارامترات المطلوبة لتوصيلة طبقة النقل (على سبيل المثال الرقم التسلسلي الأولي وحجم نافذة التحكم في التدفق). ورغم أن النظامين الطرفيين يكونان على دراية بتوصيلة طبقة النقل، إلا أن الموجّهات على طول المسار بين النظامين الطرفيين تكون غافلة عنها تماماً. في المقابل في شبكة الدوائر الافتراضية تشترك كل الموجّهات على طول المسار بين النظامين الطرفيين في إعداد الدائرة الافتراضية، ويكون كل موجّه على دراية تامة بكل الدوائر الافتراضية التي تعبره.

وتُعرَف الرسائل التي ترسلها الأنظمة الطرفية إلى الشبكة لبدء أو إنهاء VC، والرسائل التي تعبر بين الموجّهات لبدء VC (أي لتَعديل حالة الاتصال في جداول الموجّه) باسم "رسائل التحكم" (رسائل التأشير) (signaling messages)، وغالباً ما تسمى البروتوكولات المستخدمة لتبادل تلك الرسائل بروتوكولات التحكم (بروتوكولات التأشير). يوضح الشكل 4-4 إعداد VC بشكلٍ تصويري. سوف لا نغطي بروتوكولات التأشير للدوائر الافتراضية في هذا الكتاب. راجع [Black

[1997 لمناقشة عامة عن التأشير في الشبكات التوصيلية، وراجع معيار الاتحاد الدولي للمواصلات السلكية واللاسلكية [ITU-T Q.2931 1994] لمواصفات بروتوكول التأشير Q.2931 المستخدم في شبكات ATM.

4-2-2 شبكات وحدات البيانات (Datagram Networks)

في كل مرة يريد نظام طرفي في شبكة وحدات البيانات إرسال رزمة يختم الرزمة بعنوانِ النظام الطرفي للوجهة ثم يدفعها إلى الشبكة. يتم ذلك بدون أي إعداد للدوائر الافتراضية كما هو موضح في الشكل 4-5. ولا تحتفظ الموجّهات في شبكة وحدات البيانات بأية معلومات حالة حول الدوائر الافتراضية (لأنها لا توجد أصلاً!).



الشكل 4-5 شبكة وحدات البيانات.

وبينما تنتقل رزمة من المصدر إلى الوجهة فإنها تمر خلال سلسلة من الموجّهات. تَستعمل كلٌّ من تلك الموجّهات عنوان الوجهة في ترويسة الرزمة لتمريرها. وبشكلٍ مُحدّد، كل موجّه له "جدول تمرير" (forwarding table) يترجم عناوين الوجهة النهائية إلى واجهات الوصلات عليه. وعندما تصل رزمة إلى الموجّه، يستعمل الموجّه عنوان الوجهة النهائية لها للبحث عن واجهة وصلة المخرج الملائمة في جدول التمرير لديه، ثم يرسلها الموجّه عمداً إلى تلك الواجهة.

لفهم عملية البحث في الجدول (lookup) بشكلٍ أفضل، دعنا ننظر إلى مثالٍ محدد. افترض أن عناوين كل الوجهات تتكون من 32 بتاً (وهو نفس طول عنوان الوجهة في رزمة بيانات IP). تقتضي الطريقة المباشرة لتكوين جدول التمرير وجود مُدخل واحد في الجدول لكل عنوان محتمل للوجهة. ولأن هناك أكثر من 4 بلايين عنوان محتمل فهذا الخيار غير ممكن عملياً لأنه يتطلب جدول تمرير ضخم للغاية.

الآن دعنا نفترض بعد ذلك أن موجّهنا له أربع وصلات مرقمة من 0 إلى 3، وأن الرزم سترسل إلى واجهات الوصلات كالتالي:

واجهة الوصلة	مدى عناوين الوجهة
0	11001000 00010111 00010000 00000000 حتى 11001000 00010111 00010111 11111111
1	11001000 00010111 00010000 00000000 حتى 11001000 00010111 00011000 11111111
2	11001000 00010111 00011001 00000000 حتى 11001000 00010111 00011111 11111111
3	ما عدا ذلك

ومن الواضح في هذا المثال أنه ليس من الضروري وجود 4 بلايين مُدخل في جدول التمرير للموجّه. يمكن أن نستخدم على سبيل المثال جدول التمرير التالي بأربعة مُدخلات فقط:

واجهة الوصلة	تطابق البادئة
0	11001000 00010111 00010
1	11001000 00010111 00011000
2	11001000 00010111 00011
3	ما عدا ذلك

بهذا الأسلوب لجدول التمرير يطابق الموجّه بادئة (prefix) في عنوان الوجهة للرزمة بالمُدخلات في الجدول. إذا حدث تطابق بين عنوان الوجهة للرزمة وأحد مُدخلات الجدول يرسل الموجّه الرزمة إلى الوصلة المقترنة بذلك التطابق. على سبيل المثال افترض أن عنوان وجهة الرزمة هو 10100001 00010110 00010111 11001000. لأن بادئة العنوان المكونة من 21 بتاً توافق أول مُدخل في الجدول سوف يرسل الموجّه الرزمة لواجهة الوصلة رقم 0. أما إذا لم يوجد تطابق مع أيٍّ من المُدخلات الأولى الثلاثة فإن الموجّه سيُرسِل الرزمة إلى الواجهة رقم 3. وبالرغم من أن هذه الطريقة تبدو بسيطة للغاية إلا أنه يوجد هنا معنى دقيق ومهم. ربما لاحظت أنه من المحتمل أن يطابق عنوان الوجهة أكثر من مُدخل واحد في الجدول. على سبيل المثال الـ 24 بتاً الأولى من العنوان 10101010 00011000 00010111 11001000 تطابق المُدخل الثاني في الجدول، والـ 21 بتاً الأولى تطابق المُدخل الثالث في الجدول. عند وجود تطابقات متعدّدة يستعمل الموجّه قاعدة تطابق البادئة الأطول (longest prefix matching rule)، وهذا يعني تحديد المُدخل الذي يحقق أطول تطابق في الجدول ويرسل الرزمة إلى واجهة الوصلة المقترنة به.

بالطبع لكي تكون هذه القاعدة فعّالة يجب أن تكون كل واجهة وصلة مخرج مسؤولة عن تمرير كتل كبيرة من عناوين متجاورة للوجهات. سنرى في الجزء 4-4 أن عناوين الإنترنت عادة ما تخصّص بطريقة هرمية لكي تكون خاصية "التجاور" هذه سائدة في جداول التمرير لأكثر الموجّهات. وعلى الرغم من هذا هناك بعض القلق في المجتمع البحثي للإنترنت حول وجود ثقوب (عناوين غير مستخدمة) أكثر في فضاء العناوين يسبّب صغر الكتل المتجاورة أكثر، وبالتالي تصبح جداول التمرير أكبر (راجع [Meng 2005]، و[RFC 3221]، والمناقشة "المبادئ في الواقع العملي" في الجزء 4-4).

رغم أن الموجّهات في شبكات وحدات البيانات لا تحتفظ بأية معلومات عن حالة التوصيلة إلا أنها تحتفظ بمعلومات عن حالة التمرير في جداول التمرير. لكن المعدّل الزمني الذي تتغيّر فيه معلومات الحالة هذه بطيء نسبياً. في الواقع يتم تعديل جداول التمرير في شبكة وحدات البيانات بواسطة خوارزميات التوجيه، وعادة ما

يتم ذلك على فترات تتراوح من دقيقة إلى خمس دقائق أو نحوها. أما في شبكة الدائرة الافتراضية فيتم تعديل الجدول في الموجّه عند بدء توصيلة جديدة خلاله أو انتهاء توصيلة موجودة حالياً خلاله. وهذا يمكن أن يحدث بسهولة في مقياس زمني بالميكروثانية (جزء من مليون من الثانية) في موجّهات المستوى الأول لشبكة العمود الفقري للإنترنت (backbone tier-1 router).

ولأنه يمكن أن تُعدَّل جداول التمرير في شبكات وحدات البيانات في أي وقت فإن سلسلة من الرزم المرسلة من نظام طرفي إلى آخر قد تتبع مسارات مختلفة خلال الشبكة وقد تصل بترتيب مختلف. وقد قدَّم [Paxson 1997] و[Jaiswal 2003] دراسات لقياس إعادة ترتيب الرزم وظواهر أخرى في الإنترنت العامة.

3-2-4 نشأة شبكات الدوائر الافتراضية وشبكات وحدات البيانات

يعكس تطور شبكات وحدات البيانات وشبكات VC منشأها. ففكرة الدائرة الافتراضية كمبدأ تنظيم مركزي لها جذورها في عالم اتصالات الهاتف الذي يستعمل الدوائر الحقيقية. فبإعداد التوصيلة، والاحتفاظ بمعلومات عن حالة كل توصيلة في الموجّهات الموجودة في الشبكة، تصبح شبكة VC جدلياً أكثر تعقيداً من شبكة وحدات البيانات؛ وهذا يتوافق أيضاً مع جذورها المتمثلة في شبكات الاتصالات الهاتفية. ويمكنك الاطلاع على [Molinero-Fernandez 2002] لمقارنة هامة بين مدى تعقيد شبكات تحويل الدوائر (circuit-switched networks) وشبكات تحويل رزم البيانات (packet-switched networks). فقد كان التعقيد في شبكة الهواتف بالضرورة ضمن الشبكة، لأنها كانت توصّل بين أجهزة أنظمة طرفية غير ذكية كالهواتف الدوّارة (rotary telephones) – ولأولئك الشباب الذين لا يعرفون الهاتف الدوّار هو هاتف تناظري (analog telephone) بدون أزرار؛ فقط يوجد قرص دوَّار (dial).

ومن ناحية أخرى نشأت الإنترنت (كشبكة وحدات بيانات) لسد الحاجة لتوصيل الحاسبات مع بعضها. ومع أجهزة أنظمة طرفية أكثر تطوراً اختار مصمّمو الإنترنت أن يجعلوا نموذج خدمة طبقة الشبكة بسيطاً قدر الإمكان. كما رأينا في

الفصلين الثانى والثالث تم تطبيق وظائف إضافية (على سبيل المثال توصيل الرزم بنفس الترتيب، والنقل الموثوق للبيانات ، والتحكم في الازدحام، وترجمة أسماء النطاقات DNS) في طبقة أعلى في الأنظمة الطرفية. وقد كان لهذا النموذج عدة نتائج مثيرة:

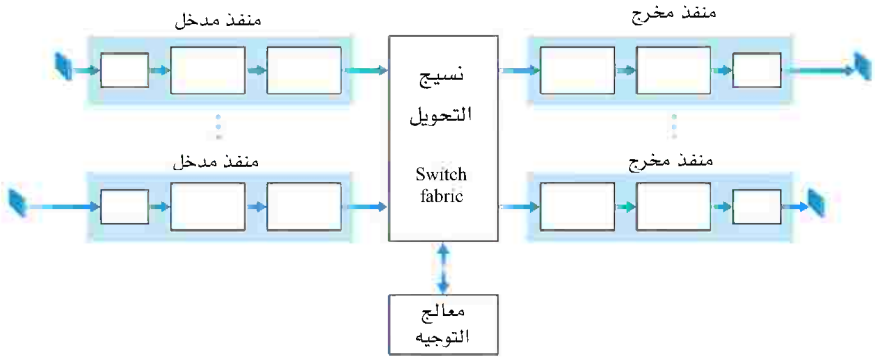
- يُسهِّل نموذج خدمة طبقة شبكة الإنترنت الناتج والذي يوفر أقل ضمانات للخدمة (أو لا ضمانات في الواقع!) (ومن ثم يفرض الحد الأدنى من المتطلبات على طبقة الشبكة) ربط الشبكات التي تستعمل تقنيات مختلفة جداً لطبقة ربط البيانات (على سبيل المثال الأقمار الصناعية، والإيثرنت، والألياف الضوئية، وموجات الراديو) ولها خصائص معدلات إرسال ونسب فقد مختلفة جداً. سوف ندرس توصيل شبكات IP بالتفصيل في الجزء 4-4.
- كما رأينا في الفصل الثانى يتم تحقيق التطبيقات كالبريد الإلكتروني والويب وحتى خدمة مركزية لطبقة الشبكة مثل DNS في المضيفات (الخادمات) على حافة الشبكة. وقد أدت تلك القدرة على إضافة خدمة جديدة عن طريق ربط المضيف بالشبكة وتعريف بروتوكول جديد لطبقة الشبكة (مثل HTTP) إلى انتشار استعمال التطبيقات الجديدة كالويب على الإنترنت في فترة زمنية قصيرة جداً.

كما سنرى في الفصل السابع يسود جدلٌ حادٌّ في مجتمع الإنترنت حول كيفية تطوّر البنية المعمارية لطبقة الشبكة في الإنترنت لكي تدعم الخدمات الفورية (real time) كتطبيقات الصوت والصورة. وتوجد مقارنة هامة بين البنية المعمارية لشبكة ATM المعتمدة على الدوائر الافتراضية ومقترح للبنية المعمارية للجيل القادم للإنترنت في [Crowcroft 1995].

4-3 ماذا بداخل الموجّه؟

الآن وبعد أن رأينا مخططاً عاماً لوظائف وخدمات طبقة الشبكة دعنا نحوّل انتباهنا إلى وظيفة "التمرير" لطبقة الشبكة (أي النقل الفعلي للرزم داخل الموجّه من وصلات المدخل إلى وصلات المخرج المناسبة). ولقد ألقينا نظرة سريعة حول بضع

قضايا لوظيفة التمرير في الجزء 4-2، وبالتحديد العنونة وتطابق أطول بادئة. سنتناول في هذا الجزء البنية المعمارية لموجّه معيّن لنقل الرزم من وصلات المدخل إلى وصلات المخرج. وقد تعمدنا الاختصار لأن تغطية تصميم موجّه بتعمّق يحتاج إلى منهج كامل. وبالتالي سنبذل جهداً خاصاً في هذا الجزء لتزويد القارئ ببعض المراجع التي تغطّي هذا الموضوع بتعمق أكثر. ونُذكّر هنا بأن المصطلحين "تمرير" (forwarding) و"تحويل" (switching) يستعملان في أغلب الأحيان بالتبادل من قِبَل الباحثين والعاملين في حقل شبكات الحاسب، وسوف نستعملهما في هذا الكتاب الدراسي.



الشكل 4-6 البينة المعمارية للموجّه.

يبين الشكل 4-6 مخططاً عالي المستوى لبنية معمارية عامة للموجّه؛ ومنه يمكن تمييز أربعة مكوّنات رئيسة للموجّه:

- منافذ المدخل: تؤدّي هذه المنافذ عدّة وظائف. فهي تقوم بوظائف الطبقة المادية (الصندوق الموجود في أقصى اليسار لمنافذ المدخل، وأقصى اليمين لمنافذ المخرج في الشكل 4-6) عند نهاية وصلة مادية قادمة للموجّه، وتؤدّي وظائف طبقة ربط البيانات (ممثّلة بالصناديق الموجودة في المنتصف لمنافذ المدخل والمخرج) والمطلوبة للتعامل مع وظائف طبقة ربط البيانات في الجانب البعيد للوصلة القادمة، كما تؤدّي أيضاً وظائف البحث في الجدول والتمرير

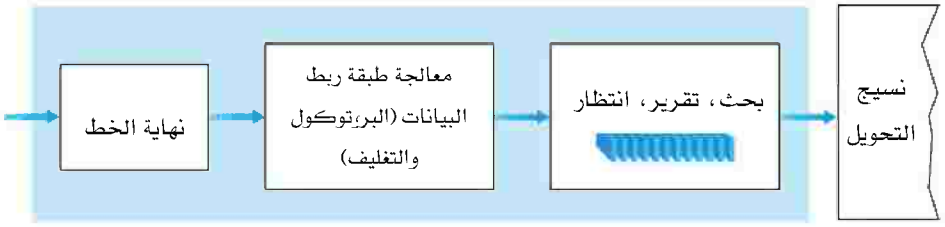
(الصندوق الموجود في أقصى اليمين لمنافذ المدخل وأقصى اليسار لمنافذ المخرج) لكي تخرج كل رزمة مرسلة إلى نسيج التحوّيل للموجّه من منفذ المخرج المناسب لها. وترسل رزم التحكم (على سبيل المثال الرزم التي تحمل معلومات بروتوكول التوجيه) من منفذ المدخل إلى معالج التوجيه (routing processor). وعملياً تتجمع عدة منافذ في أغلب الأحيان على "بطاقة خط" (line card) واحدة داخل الموجّه.

- نسيج التحويل: يوصّل نسيج التحويل منافذ المدخل للموجّه بمنافذ المخرج. يوجد نسيج التحويل بالكامل داخل الموجّه (شبكة داخل الموجّه!).
- منافذ المخرج: يخزّن منفذ المخرج الرزم التي أرسلت إليه خلال نسيج التحويل، ثم بعد ذلك يرسلها على وصلة المخرج. وهكذا يؤدّي منفذ المخرج الوظائف العكسية لطبقة ربط البيانات والطبقة المادية لمنفذ المدخل. وعندما تكون الوصلة مزدوجة الاتجاه (أي تحمل بيانات في كلا الاتجاهين) فإن منفذ المخرج إلى الوصلة عادة ما يتزاوج مع منفذ المدخل لتلك الوصلة على نفس بطاقة الخط.
- معالج التوجيه: ينفّذ معالج التوجيه بروتوكولات التوجيه (على سبيل المثال البروتوكولات التي سندرسها في الجزء 4-6)، ويحتفظ بمعلومات التوجيه وجداول التمرير، ويؤدّي وظائف إدارة الشبكة (انظر الفصل التاسع) داخل الموجّه.

في الأجزاء التالية سوف ننظر بتفصيل أكثر إلى منافذ المدخل، ونسيج التحويل، ومنافذ المخرج. راجع [Chuang 2005; Keslassy 2003; Chao 2001; Turner 1988; Giacopelli 1990; McKeown 1997a; Partridge 1998] لمناقشة لبعض البنى المعمارية المحددة للموجّهات. قدّم [McKeown 1997b] نظرة عامة سهلة القراءة للبنى الحديثة للموجّهات مستخدماً موجّه Cisco 1200 كمثال. وللدقة تفترض المناقشة التالية أن شبكة الحاسب شبكة رزم، وأن قرارات التمرير مستندة على عنوان وجهة الرزمة (بدلاً من رقم VC في شبكة الدائرة الافتراضية). ومع ذلك فالمفاهيم والأساليب مماثلة لشبكة الدائرة الافتراضية.

1-3-4 منافذ المدخل

يبين الشكل 4-7 رؤية أكثر تفصيلاً لوظائف منفذ المدخل. كما ذكرنا آنفاً فإن وظائف منفذ المدخل في توفير النهاية للخط ومعالجة وصلة البيانات هي تحقيق للطبقة المادية وطبقة ربط البيانات الخاصة بذلك المنفذ للموجّه. وتُعتبر وحدة البحث في الجدول والتمرير الموجودة في منفذ المدخل أساسية لوظيفة التمرير في الموجّه. في العديد من الموجّهات، يتم هنا تحديد منفذ المخرج الذي سترسل إليه الرزمة الواصلة وذلك عن طريق نسيج التحويل. ويتم اختيار منفذ المخرج باستعمال المعلومات الموجودة في جدول التمرير. ورغم أن جدول التمرير يُحسب بمعالج التوجيه، تُخزن نسخة ظلّ (shadow copy) عادةً من جدول التمرير في كل منفذ مدخل وتُحدَّث حسب الحاجة بواسطة معالج التوجيه. وباستخدام النسخ المحلية من جدول التمرير يمكن أن يتم اتخاذ قرار التمرير محلياً في كل منفذ مدخل بدون استدعاء معالج التوجيه المركزي. مثل هذه المعالجة اللامركزية تتجنب وجود اختناق (عنق زجاجة) عند نقطة واحدة داخل الموجّه.



الشكل 4-7 المعالجة بمنفذ المدخل.

تاريخ حالة (Case History)

هيمنة أنظمة سيسكو على قلب الشبكة

في أكتوبر/تشرين الأول عام 2006 (عند إعداد الكتاب الأصلي) بلغ عدد الموظفين بشركة سيسكو أكثر من 30 ألف شخصٍ، وبلغ رأسمالها في السوق حوالي 140 بليون دولاراً. تهيمن أنظمة سيسكو حالياً على سوق موجهات الإنترنت، وفي السنوات الأخيرة تحركت إلى سوق هاتف الإنترنت حيث تتنافس نداً لند مع شركات أجهزة الهاتف مثل Lucent، Alcatel، Norte، Siemens. فكيف نشأت هذه الغوريلا كشركة شبكات؟ كانت البداية في عام 1984 في غرفة المعيشة بشقّة في وادي السيليكون (Silicon Valley).

كان 'لن بوزاك' (Len Bosak) وزوجته 'ساندي لرنر' (Sandy Lerner) يعملان في جامعة ستانفورد عندما تكونت لديهم فكرة بناء وبيع موجّهات شبكة الإنترنت للمؤسسات الأكاديمية والبحثية. جاءت ساندي لرنر بالاسم سيسكو (كاختصار لـ 'سان فرانسيسكو')، كما صمّمت شعار الشركة. وكان المقر الرئيس للشركة هو غرفة المعيشة في منزلهم، وتم تمويل المشروع عن طريق بطاقات الائتمان ووظائف العمل الجزئي الاستشارية. في نهاية عام 1986 بلغت عائدات سيسكو 250 ألف دولار في الشّهر، وفي نهاية عام 1987 نجحت سيسكو في جذب رأسمال استثماري بلغ مليوني دولار من Sequoia في مقابل ثلث الشركة. وواصلت سيسكو نموها على مدى السنوات القليلة التالية وزادت حصة الشركة في السوق أكثر فأكثر.

في نفس الوقت توترت علاقات بوزاك ولرنر مع إدارة سيسكو. أشهرت سيسكو للجمهور في عام 1990، وفي نفس العام ترك بوزاك ولرنر الشركة. وعلى مرّ السنين توسّعت سيسكو أكثر لتشمل منتجات وخدمات أخرى غير الموجّهات كأجهزة الأمن واللاسلكي، ومنتجات وخدمات نقل الصوت عبر الإنترنت.

ومع ذلك واجهت سيسكو منافسة دولية متزايدة مع شركات أخرى مثل شركة Huawei وهي شركة صينية سريعة النمو. بلغ عدد موظفي شركة Huawei أكثر من 38 ألف موظفٍ حول العالم واستحوذت الشركة ـ حسبما أُعلن مؤخراً ـ على أكثر من 7٪ من أسواق الموجّهات ومحولات الإيثرنت. ومن الشركات المنافسة لسيسكو أيضاً في مجال الموجّهات ومحولات الإيثرنت شركة ألكاتيل (Alcatel) ولوسينت (Lucent) وجونيبر (Juniper).

في الموجّهات ذات قدرة المعالجة المحدودة في منفذ المدخل قد يرسل منفذ المدخل الرزمة ببساطة إلى معالج التوجيه المركزي، والذي يقوم بدوره بالبحث في جدول التمرير وإرسال الرزمة إلى منفذ المخرج المناسب. هذه الطريقة تُتبع عندما تعمل محطة عمل فرعية (workstation) أو خادم كموجّه. في تلك الحالة يكون معالج التوجيه في الحقيقة هو وحدة المعالجة المركزية CPU لمحطة العمل الفرعية، ويكون منفذ المدخل في الحقيقة هو بطاقة مواءمة الشبكة (network interface card)؛ على سبيل المثال بطاقة الإيثرنت.

بوجود جدول التمرير تكون عملية البحث بسيطة من حيث المفهوم، فقط نبحث خلال جدول التمرير عن تطابق أطول بادئة، كما وصفنا في الجزء 4-2-2. ومع ذلك فليست الحياة عملياً بهذه البساطة. ربما يكون عامل الصعوبة الأول والأكثر أهمية هو أن موجّهات شبكة العمود الفقري يجب أن تعمل بسرعة عالية مؤدّيةً الملايين من عمليات البحث كل ثانية. في الحقيقة من المرغوب فيه أن تتم معالجة منفذ المدخل بسرعة الخط، أي تؤدِّي عملية البحث في زمن أقل من الوقت اللازم لاستلام رزمة في منفذ المدخل. في هذه الحالة يمكن أن تكتمل معالجة الرزمة المستلمة قبل أن تكتمل عملية الاستلام التالية. ولفهم متطلبات الأداء لعملية البحث، تصور وصلة من النوع OC-48 بسرعة 2.5 جيجابت/ثانية ورزم بحجم 256 بايت، فهذا يعني ضمناً أن سرعة البحث تبلغ تقريباً مليون عملية في الثانية.

ومع الحاجة للتشغيل بالسرعات العالية المتاحة للوصلة الآن يتضح أن البحث التتابعي (sequential search) خلال جدول تمرير كبير يصبح أمراً مستحيلاً. وهناك طريقة أخرى أكثر معقولية هي أن تُخزن مُدخلات جدول التمرير في هيكل بيانات على شكل "شجرة". حيث يمكن أن تفكر في كل مستوى في الشجرة بمثابة بتٍ في عنوان الوجهة. وللبحث عن عنوان نبدأ ببساطة من عقدة "الجذر" للشجرة ونتحرك خلال الشجرة. فإذا كان البت الأول في العنوان "0" نتجه للشجرة الفرعية اليسرى، وإذا كان "1" نتجه للشجرة الفرعية اليمنى. ثمّ نتتبع الشجرة الفرعية المناسبة باستعمال البتات الأخرى في العنوان. بهذه الطريقة يمكن أن نبحث في جدول التمرير عن عنوان في عدد من الخطوات يساوي N (وهي تمثل عدد البتات في

العنوان). لاحظ أن هذه الطريقة أساساً هي بحث ثنائي (binary search) خلال فضاء عناوين حجمه يساوي 2^N. ويمكنك الاطلاع على تعديل لطريقة البحث الثنائي في [Srinivasan 1999] وعلى دراسة مسحية عامة لخوارزميات تصنيف الرزم في [Gupta 2001].

لكن حتى مع $N = 32$ (على سبيل المثال عنوان IP يتكون من 32 بتاً) لا تكون سرعة البحث بأسلوب "البحث الثنائي" سريعة بما فيه الكفاية لمتطلبات توجيه شبكة العمود الفقري اليوم. على سبيل المثال افترض أن كل خطوة تحتاج إلى وصول للذاكرة (memory access)، فإن أقل من مليون بحث في الثانية يمكن أن يتم إذا كانت سرعة الوصول للذاكرة 40 نانو ثانية (40ns). وعليه تم اقتراح عدّة أساليب لزيادة سرعة البحث. على سبيل المثال تسمح ذاكرة CAM – وهي ذاكرة معنونة بمحتوياتها (content addressable memory) – باستخراج محتوى مُدخل جدول التمرير المناظر لعنوان IP المقدّم للذاكرة في وقت ثابت بالضرورة. وتُعتبر سلسلة سيسكو 8500 مثالاً للموجهات التي تحتوي على ذاكرة CAM حيث توجد ذاكرة من هذا النوع بسعة 64 كيلوبايت (64KB) لكل منفذ مدخل.

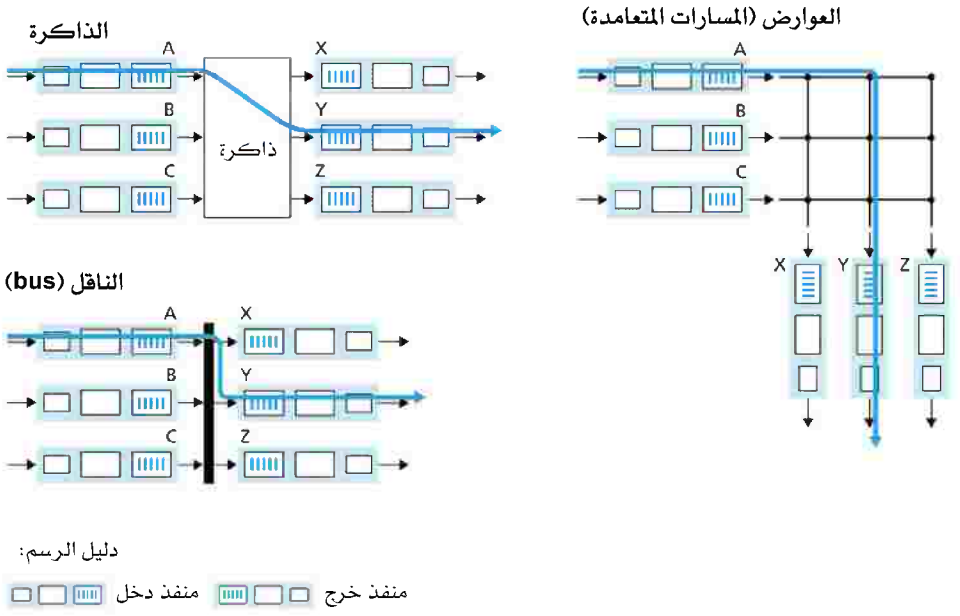
طريقة أخرى لتسريع البحث هي الاحتفاظ بمُدخلات جدول التمرير التي استُخدمت مؤخراً في ذاكرة وسيطة (cache) [Feldmeier 1988]. لكن تبقى مشكلة الحجم المحتمل للذاكرة الوسيطة. اقتُرح أيضاً تراكيب (هياكل) بيانات سريعة تسمح بالبحث عن مُدخلات جدول التمرير في عدد خطوات يساوي $\log(N)$ [Waldvogel 1997] وأساليب أخرى تقوم بضغط جداول التمرير بطرق مبتكرة [Brodnik 1997]. كما تم مناقشة طريقة "مادية" (hardware approach) محسّنة للبحث تفيد في الحالة الشائعة عندما يكون العنوان المراد البحث عنه يتضمن 24 بتاً أو أقل [Gupta 1998]. للحصول على دراسة مسحية وتصنيف لأنواع الخوارزميات السريعة للبحث داخل جداول التمرير اطّلع على [Ruiz Sanchez 2001].

وبمجرد تحديد منفذ المخرج للرزمة عن طريق البحث يمكن أن تُرسَل الرزمة خلال نسيج التحويل. لكن قد تُمنع الرزمة بشكلٍ مؤقت من دخول نسيج التحويل

(بسبب انشغال النسيج حالياً لنقل رزم من منافذ المدخل الأخرى). لذا يجب أن تنتظر الرزمة المستوقفة في منفذ المدخل وبعد ذلك تُجدول (scheduled) لعبور نسيج التحويل في وقتٍ لاحق. سنلقي نظرة أكثر تفحصاً على الإيقاف (blocking) والانتظار في الصف (الطابور) (queueing) وجدولة الرزم (في منافذ المدخل ومنافذ المخرج) داخل موجّه في الجزء 4-3-4.

4-3-2 نسيج التحويل

يوجد نسيج التحويل في صميم قلب الموجّه. فمن خلال نسيج التحويل تنتقل الرزم في الحقيقة (أي تُمرر) من منفذ مدخل إلى منفذ مخرج. يمكن أن ينجز التحويل بعدة طرق كما هو مبين في الشكل 4-8:



الشكل 4-8 ثلاثة طرق للتحويل.

- التحويل عن طريق الذاكرة: كانت الموجّهات الأولى البسيطة حاسبات تقليدية في أغلب الأحيان، وكان التحويل يتم بين منافذ المدخل ومنافذ المخرج تحت السيطرة المباشرة لوحدة المعالجة المركزية (معالج التوجيه). وكانت منافذ المدخل والمخرج تمثل أجهزة إدخال وإخراج تقليدية في نظام التشغيل التقليدي. عند وصول رزمة إلى منفذ مدخل يبعث "إشارة مقاطعة" (interrupt) إلى معالج التوجيه. بعد ذلك تُنسَخ الرزمة من منفذ المدخل إلى ذاكرة المعالج، حيث يقوم معالج التوجيه باستخلاص عنوان الوجهة من ترويسة الرزمة، والبحث عن منفذ المخرج المناسب في جدول التمرير، ثم نسخ الرزمة إلى الذاكرة المؤقتة (المرحلية) (buffer) لمنفذ المخرج. لاحظ أنه إذا كان الحيز الترددي للذاكرة بحيث يمكن الكتابة فيها أو القراءة منها بسرعة B رزمة في الثانية، فإن الطاقة الإنتاجية الكلية للتمرير (المعدل الكلي لنقل الرزم من منافذ المدخل إلى منافذ المخرج) يجب أن تكون أقل من $B/2$.

  يتم التحويل أيضاً في العديد من الموجّهات الحديثة عن طريق الذاكرة. ومع ذلك هناك اختلاف أساسي عن الموجّهات المبكرة، وهو أن البحث عن عنوان الوجهة وتخزين الرزمة في موقع الذاكرة المناسب يتم بواسطة المعالجات على بطاقات الخط للإدخال. وتشبه الموجّهات التي تحول الرزم عن طريق الذاكرة إلى حدٍ كبير المعالجات المتعددة ذات الذاكرة المشتركة (shared-memory multiprocessors)، حيث تقوم المعالجات على بطاقة الخط بتحويل الرزم إلى ذاكرة منفذ المخرج المناسب. وكمثال تقوم سلسلة محوّلات Cisco Catalyst 8500 [Cisco 8500 2007] بتحويل الرزم عن طريق الذاكرة المشتركة. ويمكنك الاطلاع على نموذج تجريدي لدراسة خواص التحويل المبني على الذاكرة ومقارنته بالأشكال الأخرى من التحويل في [Iyer 2002].
- التحويل عن طريق "ناقل" (bus): في هذه الطريقة تنقل منافذ المدخل الرزمة مباشرةً إلى منفذ المخرج على ناقل مشترك بدون تدخُّل من معالج التوجيه

(لاحظ أنه عند التحويل عن طريق الذاكرة يجب أيضاً أن تعبر الرزمة ناقل النظام إلي الذاكرة أو منها). وبالرغم من أن معالج التوجيه لم يشترك في النقل على الناقل إلا أنه يمكن أن تنتقل رزمة واحدة فقط في كل مرة على الناقل (لأن الناقل مشترك). عندما تصل رزمة إلى منفذ مدخل وتجد الناقل مشغولاً بنقل رزمة أخرى فسوف تُمنع من عبور نسيج التحويل وعليها الانتظار في طابور منفذ المدخل. ولأن كل رزمة يجب أن تعبر الناقل الوحيد فإن سعة التحويل (switching bandwidth) للموجّه محدودة بسرعة الناقل.

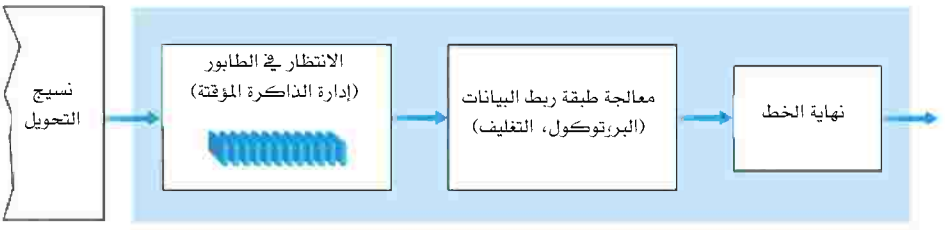
ومع توفر سعات للناقل في تقنيات اليوم تزيد على جيجابت في الثانية، فإن التحويل عن طريق الناقل يعتبر كافياً في أغلب الأحيان لموجّهات شبكات الوصول وشبكات المؤسسات (كالشبكات المحلية وشبكات الشركات). ويُستخدم التحويل عن طريق الناقل في عدد من منتجات الموجّهات الحالية مثل Cisco 5600 [Cisco Switches 2007] والتي تحوّل الرزم على ناقل لوحات الربط الخلفية (backplane bus) بسرعات تزيد على 32 جيجابت/ثانية.

- التحويل عن طريق شبكة ربط بينية (interconnection network): أحد الطرق للتغلب على قيود الحيز الترددي لناقل مشترك وحيد هو استخدام شبكة ربط بيني أكثر تطوّراً كتلك التي استعملت في الماضي لربط المعالجات في البنية المعمارية للحاسب متعدد المعالجات. يمثل محوّل العوارض (محوّل بمسارات متعامدة) (crossbar switch) شبكة ربط بيني تتكون من ناقلات عددها $2n$ لتوصل n منفذ مدخل إلى n منفذ مخرج كما هو مبين في الشكل 4-8. تنتقل الرزمة التي تصل إلى منفذ مدخل على الناقل الأفقي المتصل بمنفذ المدخل حتى يتقاطع بالناقل العمودي المؤدّي إلى منفذ المخرج المطلوب. إذا كان الناقل العمودي المؤدّي إلى منفذ المخرج حراً ("غير مشغول") تنتقل الرزمة إلى منفذ المخرج. أما إذا كان الناقل العمودي مستخدماً لنقل رزمة من منفذ مدخل آخر إلى نفس منفذ المخرج فإن تلك الرزمة تُمنَع ويجب عليها أن تنتظر في طابور منفذ المدخل.

اقتُرحت أيضاً أنسجة تحويل الدلتا والأوميجا (Delta and Omega switching fabrics) كشبكة ربط بين منافذ المدخل والمخرج. انظر [Tobagi 1990] لدراسة مسحية للبنى المعمارية للمحوّلات. وكمثال، تُستعمل محوّلات عائلة سيسكو 12000 [Cisco 12000 2007] شبكة ربط بسرعات تصل إلى 60 جيجابت/ثانية خلال نسيج التحويل. أحد الاتجاهات في تصميم شبكة الربط [Keshav 1998] هو تجزئة رزمة IP ذات الطول المتغير إلى خلايا ثابتة الطول، ثم تُعلَّم وتُنقَل تلك الخلايا خلال شبكة الربط. يعاد تجميع الخلايا إلى الرزمة الأصلية في منفذ المخرج. يمكن أن يبسّط استخدام الخلايا ثابتة الطول والتعليم الداخلي عملية تحويل الرزم خلال شبكة الربط إلى حدٍّ كبير ويزيد من سرعتها.

3-3-4 منافذ المخرج

تتضمن المعالجة – التي تتم في منافذ المخرج والموضحة في الشكل 4-9 – أخذ الرزم التي خُزنت في ذاكرة منفذ المخرج وإرسالها على الوصلة الخارجة. تمثل معالجة بروتوكول وصلة البيانات وتوفير نهاية للخط وظائف طبقة ربط البيانات والطبقة المادية لجهة الإرسال التي تتفاعل مع منفذ المدخل على الطرف الآخر للوصلة الخارجة، كما نوقش في الجزء 4-3-1. إن وظائف إدارة الانتظار في الصف وإدارة الذاكرة المؤقتة (buffer) مطلوبة عندما يُسلِّم نسيج التحويل الرزم إلى منفذ المخرج بمعدل يتجاوز معدل الإرسال على وصلة المخرج، وسوف نغطي الانتظار في صفوف منافذ المخرج فيما بعد.



الشكل 4-9 المعالجة في منفذ المخرج.

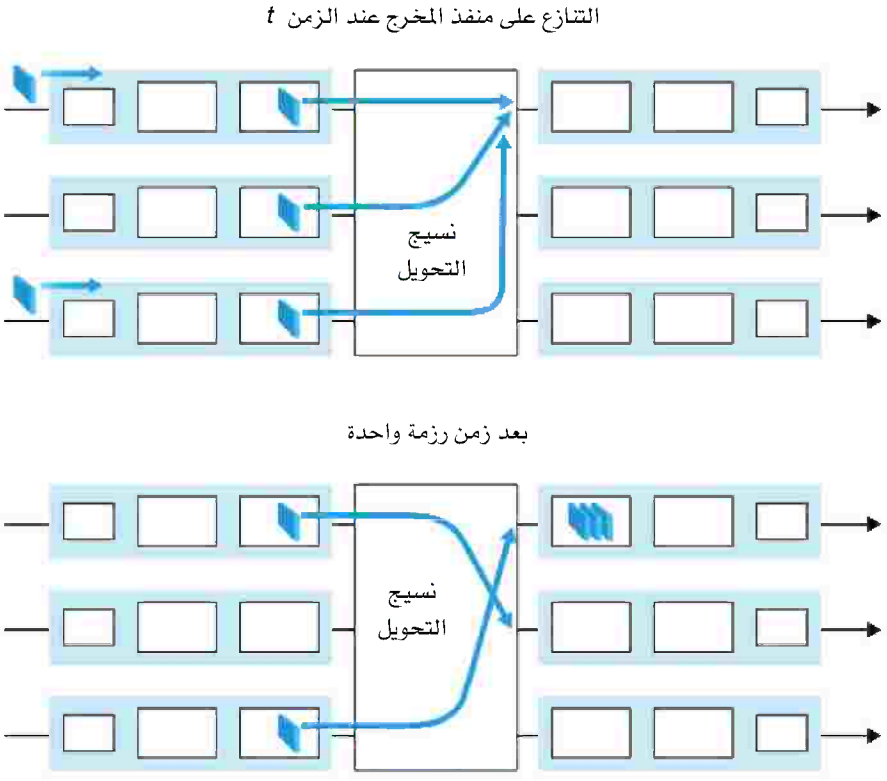
4-3-4 أين يحدث الانتظار في الطابور؟

إذا نظرنا إلى وظائف منافذ المدخل والمخرج والترتيبات المبينة في الشكل 4-8 يتضح أن طوابير الرزم يمكن أن تُشكَّل في كلٍّ من منافذ المدخل ومنافذ المخرج. من المهم دراسة هذه الطوابير بشيء من التفصيل لأنه مع زيادة حجمها ستُستنزَف ذاكرة الموجّه المؤقتة في النهاية، وسوف يؤدي ذلك إلى فقد الرزم. تذكّر أننا ذكرنا في مناقشاتنا السابقة أن الرزم تُفقَد داخل الشبكة أو تُسقَط عند الموجّه. وها نحن نرى كيف تُسقَط مثل تلك الرزم وتُفقَد هنا في هذه الطوابير داخل الموجّه. يعتمد الموقع الفعلي لفقد الرزم (في طوابير منفذ المدخل أو طوابير منفذ المخرج) – كما سنناقش فيما بعد – على حِمْل مرور البيانات (traffic load) والسرعة النسبية لنسيج التحويل وسرعة الخط.

افترض أن سرعة الخط لكلٍّ من منافذ المدخل والمخرج متماثلة، وأن هناك عدد n منفذ مدخل وعدد n منفذ مخرج. ولنُعرّف سرعة نسيج التحويل على أنها المعدل الذي يمكن به لنسيج التحويل أن يحرّك الرزم من منافذ المدخل إلى منافذ المخرج. إذا كانت سرعة نسيج التحويل تعادل على الأقل n مرة سرعة خط المدخل فعندئذ لا يمكن أن يحدث أي انتظار في الطوابير في منافذ المدخل. وذلك لأنه حتى في أسوأ الأحوال عندما تستلم كل منافذ المدخل رزماً فلا يزال بوسع المحوّل نقل عدد n رزمة من منفذ المدخل إلى منفذ المخرج في نفس الوقت الذي يأخذه كل منفذ من منافذ المدخل (بشكلٍ آني) لاستلام رزمة واحدة.

لكن ماذا يمكن أن يحدث في منافذ المخرج؟ دعنا نفترض بأنه ما زالت سرعة نسيج التحويل تعادل على الأقل n مرة سرعة الخط. في أسوأ الأحوال تكون كل الرزم التي تصل إلى منافذ المدخل (وعددها n) متجهة إلى نفس منفذ المخرج. في هذه الحالة في خلال الوقت اللازم لاستلام (أو إرسال) رزمة واحدة ستصل n رزمة إلى منفذ المخرج هذا. ولأنه يمكن أن يرسل منفذ المخرج رزمة واحدة فقط في وحدة الزمن (وقت إرسال الرزمة) فعندئذ سيكون على الرزم المُستلَمة (وعددها n) أن تصطف (تنتظر) للإرسال على الوصلة الخارجة. وسيكون من المحتمل وصول رزم

أكثر عددها n رزمة في الوقت اللازم لإرسال رزمة واحدة من تلك التي سبق وضعها في الطابور. وهكذا في النهاية يمكن أن يزيد عدد الرزم المنتظرة بدرجة كافية لاستنزاف الذاكرة المؤقتة في منفذ المخرج وعند ذلك يبدأ إسقاط (فقد) الرزم.



الشكل 4-10 الانتظار في منافذ المخرج.

يوضح الشكل 4-10 الانتظار في طوابير منافذ المخرج. عند زمن t تصل رزمة إلى كل منفذ من منافذ المدخل، وكلٌّ منها متجه إلى منفذ المخرج الموجود في أعلى الشكل. افترض أن سرعة الخط متماثلة وأن المحوّل يعمل بسرعة تعادل ثلاثة أضعاف سرعة الخط. بعد وحدة زمن (أي الوقت اللازم لاستلام أو إرسال رزمة) تكون الرزم الثلاثة الأصلية قد نُقلت إلى منفذ المخرج واصطفت منتظرة الإرسال.

في وحدة الزمن التالية سترسل إحدى هذه الرزم الثلاثة على الوصلة الخارجة. في مثالنا ستصل رزمتان جديدتان إلى الجانب القادم للمحوّل؛ إحداهما متجهة إلى نفس منفذ المخرج الموجود في أعلى الشكل.

بافتراض أن الذاكرة المؤقتة لازمة لامتصاص التقلبات في حِمْل مرور البيانات، فالسؤال الطبيعي الآن هو "ما الحجم المطلوب لتلك الذاكرة؟". لعدة سنوات كانت القاعدة التقريبية الشائعة (المبنية على التجربة العملية وليس المعرفة العلمية) [RFC 3439] لاختيار حجم الذاكرة المؤقتة B هي أن يكون مساوياً لحاصل ضرب متوسط زمن الرحلة ذهاباً وإياباً (RTT مثلا 250 ميلي ثانية) وسعة الوصلة C. وهذه النتيجة مستندة على تحليل ديناميكا الطوابير لعدد صغير نسبياً من مسارات تدفق TCP [Villamizar 1994]. فمثلاً إذا كانت سعة الوصلة 10 جيجابت في الثانية وقيمة RTT تساوي 250 ميلي ثانية فإن حجم الذاكرة المؤقتة التي نحتاجها $B=RTT\times C=2.5\text{Gb}$. وتقترح الجهود النظرية والتجريبية الحديثة حساب حجم الذاكرة عندما يكون هناك عدد كبير من مسارات تدفق TCP يمر بالوصلة من المعادلة $B = RTT \times C/\sqrt{N}$ حيث تمثل N عدد تلك المسارات [Appenzeller 2004]. ومع وجود عدد كبير من مسارات التدفق يمر خلال وصلات شبكة عمود فقري كبيرة (انظر على سبيل المثال [Fraleigh 2003]) يمكن أن تكون قيمة N كبيرة جداً، وبالتالي يقل حجم الذاكرة المطلوبة بشكلٍ ملحوظ للغاية. قدَّم [Appenzeller 2004] و [Wischik 2005] مناقشات سهلة القراءة لمشكلة اختيار حجم الذاكرة المؤقتة من منطلقات نظرية وتطبيقية وتشغيلية.

نتيجة للانتظار في طابور منفذ المخرج يجب أن يختار مُجدول الرزم (packet scheduler) في منفذ المخرج رزمة واحدة من بين تلك الرزم المنتظرة للإرسال. قد يتم هذا الاختيار بقاعدة بسيطة مثل قاعدة "الأول وصولاً ... الأول خدمةً" (FCFS) والتي تعطي أفضلية الخدمة للأول وصولاً، أو نظام جدولة أكثر تطوّراً مثل قاعدة طوابير الانتظار العادلة ذات الأوزان (WFQ)، والتي يتم فيها تقاسم الوصلة الخارجة بإنصاف بين التوصيلات المختلفة من طرف لطرف والتي لها رزم منتظرة للإرسال. وتلعب جدولة الرزم دوراً هاماً لتوفير ضمانات جودة للخدمة (quality of service).

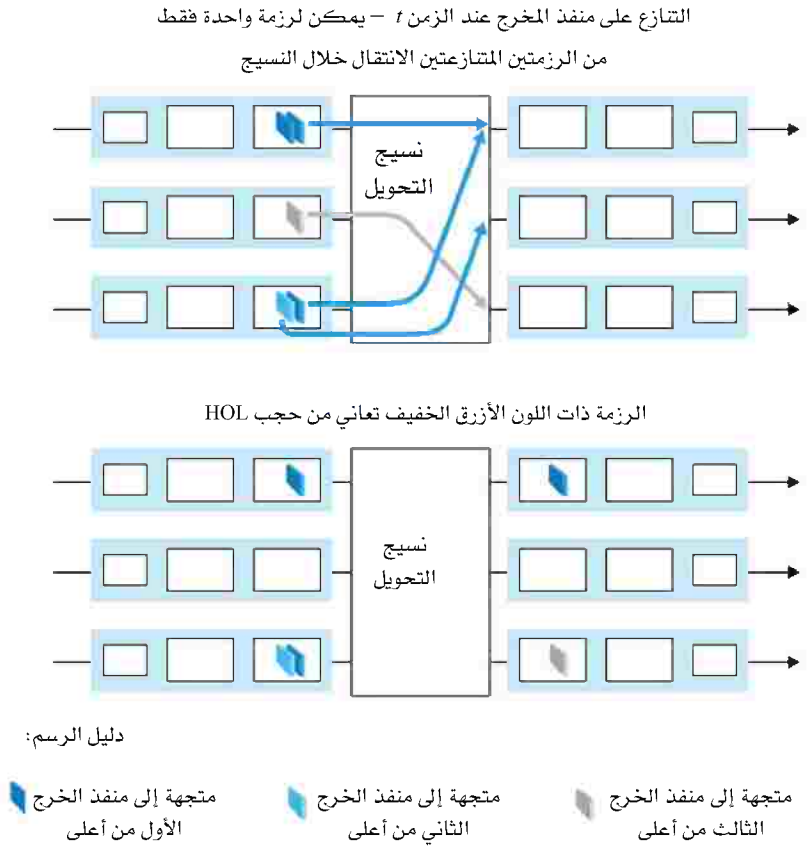
سوف نغطي جدولة الرزم بتوسع أكثر في الفصل السابع، وهناك مناقشة حول قواعد جدولة الرزم لمنافذ المخرج في [Cisco Queue 2007].

بنفس الطريقة إذا لم تكن هناك ذاكرة مؤقتة تكفي لتخزين الرزمة القادمة يجب أن يُتخَذ قرار إما بإسقاط الرزمة المستلمة (وهي سياسة تعرف بالإسقاط الذيلي) أو لإزالة واحدة أو أكثر من الرزم المنتظرة في الطابور لإخلاء مكان للرزمة الواصلة حديثاً. وقد يكون من المفيد في بعض الحالات إسقاط رزمة (أو التأشير على ترويستها) قبل امتلاء الذاكرة المؤقتة لكي تعطي إشارة ازدحام إلى المُرسِل. تم اقتراح وتحليل عدة سياسات لإسقاط الرزم والتأشير عليها، وأصبحت تعرف مجتمعةً بخوارزميات إدارة الطابور النشطة (Active Queue Management (AQM)) [Labrador 1999; Hollot 2002]. يطلق على إحدى هذه الخوارزميات المدروسة والمطبّقة على نحو واسع خوارزمية "الكشف المبكّر العشوائي" ((Random-Early Detection (RED)، وبهذه الطريقة يمكن الاحتفاظ بمتوسط موزون (weighted average) لطول طابور المخرج. إذا كان طول الطابور المتوسط أقل من عتبة الحد الأدنى (minimum threshold) min_{th} فعندما تصل رزمة سوف يسمح لها بالانتظار في الطابور. وبالمقابل إذا كان الطابور ممتلئاً بالكامل أو أن طول الطابور المتوسط أعلى من عتبة الحد الأقصى max_{th} فعندما تصل رزمة سوف يُؤشَّر عليها أو تُسقَط. وأخيراً إذا وصلت الرزمة وكان طول الطابور المتوسط في المدى $[min_{th}, max_{th}]$ سوف يُؤشَّر عليها أو تُسقَط باحتمالية معينة والتي عادة ما تكون دالة في (أي تعتمد على) طول الطابور المتوسط وmin_{th} وmax_{th}. اقتُرحت عدة طرق للتأشير والإسقاط الاحتمالي، وتم نمذجة وتحليل ومحاكاة وتطبيق نسخ مختلفة من طريقة RED. قدَّم [Christiansen 2001] و[Floyd 2007] نظرة عامة حول هذا الموضوع مع ذكر مراجع للقراءة الإضافية.

إذا كان نسيج المحوّل ليس سريعاً بما فيه الكفاية (مقارنةً بسرعة خط المدخل) لتحويل كل الرزم الواصلة خلال النسيج بدون تأخير فإن الانتظار يمكن أن يحدث أيضاً في منافذ المدخل؛ لأن الرزم يجب أن تلتحق بطوابير منفذ المدخل لانتظار دورها قبل أن تنتقل خلال نسيج التحويل إلى منفذ المخرج. ولتوضيح نتيجة

مهمة لهذا الانتظار افترض وجود نسيج تحويل بمسارات متعامدة (crossbar switching fabric) وأن (1) كل سرعات الوصلات متماثلة، (2) زمن تحويل رزمة واحدة من أي منفذ مدخل إلى منفذ مخرج معين هو نفسه الزمن الذي تأخذه رزمة لاستلامها على منفذ مدخل، (3) نقل الرزم من طابور منفذ المدخل إلى طابور المخرج المطلوب يتم بأسلوب FCFS. يمكن أن تحوّل رزم متعدّدة بالتوازي (في نفس الوقت) طالما أن منافذ المخرج مختلفة. لكن إذا كانت رزمتان في مقدمة طابورَي دخل تتّجهان إلى نفس طابور المخرج فإن إحديهما ستوقف ويجب أن تنتظر في طابور المدخل (لأن نسيج التحويل يمكن أن يحوّل رزمة واحدة فقط إلى منفذ مخرج معين في وقتٍ ما).

يبين الشكل 4-11 مثالاً فيه رزمتان (مظلّلتان على نحو داكن) في مقدمة طابورَي دخل ومتجهتان إلى نفس منفذ المخرج الموجود أعلى الشكل. افترض أن نسيج المحوّل يختار تحويل الرزمة من مقدمة الطابور الموجود في أعلى اليسار. في هذه الحالة يجب أن تنتظر الرزمة المظلّلة على نحو داكن في الطابور الموجود أسفل اليسار. وليس هذا فحسب بل يجب أيضاً أن تنتظر الرزمة المظلّلة قليلاً والموجودة وراء تلك الرزمة في الطابور أسفل اليسار بالرغم من عدم وجود تنازع على منفذ المخرج في وسط اليمين (والذي يمثل وجهة الرزمة المظلّلة قليلاً). تعرف هذه الظاهرة بـ "حجب مقدمة الطابور (HOL blocking)" في المحوّلات ذات الطوابير عند منافذ المدخل (أي يجب أن تنتظر الرزمة الموجودة في طابور منفذ مدخل بالرغم من أن منفذ المخرج لها قد يكون حراً وذلك نظراً لوجود رزمة أخرى منتظرة في مقدمة الطابور). بيّن [Karol 1987] أنه – تحت بعض الفرضيات – تؤدي ظاهرة حجب HOL إلى زيادة طول طابور المدخل بطريقة غير محدودة (أي يحدث فقدٌ ملحوظٌ في الرزم) بمجرد أن يصبح معدل وصول الرزم على وصلات المدخل 58٪ فقط من سعتها. وتوجد مناقشة لعدد من الحلول لمشكلة "حجب HOL" في [McKeown 1997b].



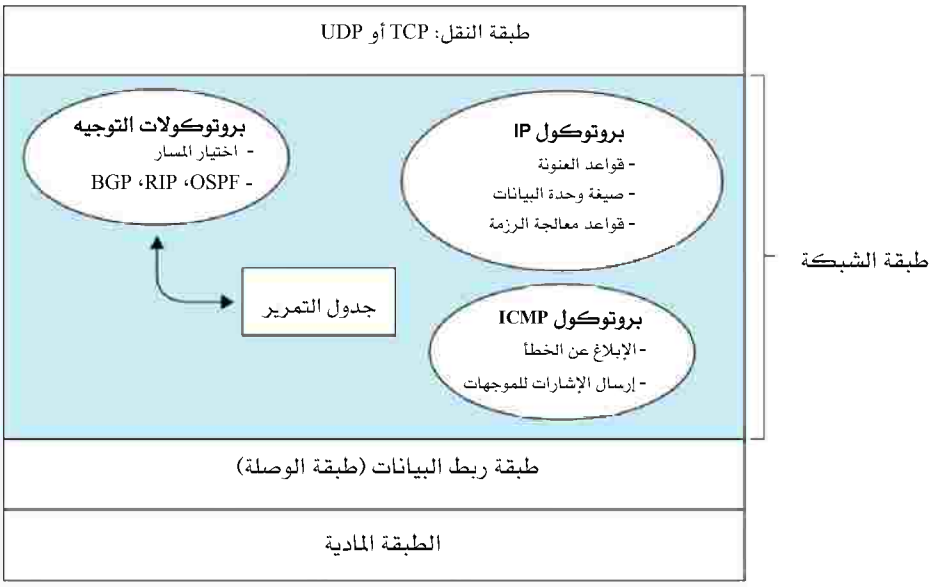
الشكل 4-11 حجب مقدمة الطابور.

4-4 بروتوكول الإنترنت (IP): التمرير والعنونة في الإنترنت

لقد كانت مناقشتنا للعنونة والتمرير في طبقة الشبكة حتى الآن عامة وغير مرتبطة بشبكة محددة. في هذا الجزء سوف نحوّل انتباهنا لكيفية إنجاز العنونة والتمرير في شبكة الإنترنت بصفة خاصةً. وسنرى بأن العنونة والتمرير في الإنترنت تعد مكوّنات هامة في بروتوكول الإنترنت IP. توجد نسختان من بروتوكول الإنترنت IP قيد الاستعمال اليوم. سوف نفحص أولاً نسخة بروتوكول الإنترنت 4 الواسعة الانتشار والتي عادةً ما يشار إليها بـ IPv4 [RFC 791]. وسوف نفحص نسخة

بروتوكول الإنترنت 6 في نهاية هذا الجزء [RFC 2460; RFC 3513] (والتي اقتُرِحَت لتحل محل IPv4).

لكن قبل أن نبدأ حملتنا لاستكشاف بروتوكول الإنترنت دعنا نراجع مكوّنات طبقة شبكة الإنترنت. كما هو موضح في الشكل 4-12 تتكون طبقة شبكة الإنترنت من ثلاثة مكوّنات رئيسة. الأول بروتوكول الإنترنت (موضوع هذا الجزء)، والثاني التوجيه لتحديد المسار الذي تتبعه رزمة البيانات من مصدرها إلى وجهتها. ذكرنا في وقت سابق أن بروتوكولات التوجيه تحسب جداول التمرير التي تستعمل لإرسال الرزم خلال الشبكة. وسوف ندرس بروتوكولات التوجيه في الإنترنت في الجزء 4-6. أما المكوّن النهائي لطبقة الشبكة فهو وسيلة للإبلاغ عن الأخطاء في رزم البيانات والرد على الطلبات لمعلومات معينة من طبقة الشبكة. وسوف نغطي بروتوكول الإبلاغ عن الأخطاء والمعلومات في طبقة شبكة الإنترنت والمشار إليه في الجزء 4-4-3 ببروتوكول رسائل التحكم في الإنترنت (Internet Control Message Protocol (ICMP)).



الشكل 4-12 طبقة الشبكة في الإنترنت.

1-4-4 صيغة وحدة البيانات

تذكّر أن رزمة طبقة الشبكة تُدعى وحدة بيانات. سنبدأ دراستنا لبروتوكول الإنترنت بنظرة عامة لتراكيب (دراسة القواعد النحوية) ودلالات (دراسة المعاني) وحدة بيانات بروتوكول IPv4. قد تعتقد أنه لا شيء يمكن أن يكون أكثر جفافاً من دراسة النحو ودراسة معاني البتات في الحقول المختلفة للرزمة. على الرغم من ذلك تلعب وحدة البيانات دوراً هاماً في الإنترنت (فكل دارس ومحترف للشبكات يحتاج لأن يراها ويستوعبها ويتقنها). يوضح الشكل 4-13 صيغة وحدة بيانات IPv4. إن الحقول الرئيسة في وحدة بيانات IPv4 كالتالي:

الشكل 4-13 صيغة وحدة بيانات بروتوكول IPv4.

- رقم النسخة: تحدّد الـ 4 بتات الأولى رقم نسخة بروتوكول الإنترنت لوحدة البيانات تلك. بالنظر إلى رقم النسخة يستطيع الموجّه أن يحدّد كيف يترجم بقيّة حقول وحدة البيانات. تستعمل النسخ المختلفة لبروتوكول الإنترنت صيغ

وحدات بيانات مختلفة. يوضح الشكل 4-13 صيغة وحدة البيانات للنسخة الحالية لبروتوكول الإنترنت IPv4. سوف نناقش صيغة وحدة البيانات للنسخة الجديدة لبروتوكول الإنترنت IPv6 في نهاية هذا الجزء.

- طول الترويسة: لأن وحدة بيانات IPv4 يمكن أن تحتوي على عدد متغير من الخيارات (التي تُتضمّن في ترويسة وحدة البيانات) فهذه البتات الأربعة مطلوبة لتقرير أين تبدأ البيانات فعلاً في وحدة البيانات. معظم وحدات بيانات بروتوكول الإنترنت لا تحتوي على خيارات في الترويسة، ولذا فإن ترويسة وحدة بيانات بروتوكول الإنترنت العادية تتكون من 20 بايتاً.
- نوع (نمط) الخدمة: تشتمل ترويسة IPv4 على حقل نوع الخدمة (TOS) للسماح بتمييز أنواع مختلفة من وحدات البيانات عن بعضها البعض (مثلاً رزم بيانات تتطلّب بصفة خاصة تأخيراً منخفضاً أو طاقة إنتاجية عالية أو موثوقية نقل). على سبيل المثال قد يكون من المفيد تمييز وحدات البيانات الفورية (كتلك المستخدمة من قِبَل تطبيق هاتف الإنترنت) عن غيرها (على سبيل المثال FTP). يعتبر المستوى المعين للخدمة الذي يمكن توفيره قضية سياسة تُحدَّد من قِبَل مدير الموجّه. سوف نستكشف موضوع الخدمة التفاضلية (differentiated service) بالتفصيل في الفصل السابع.
- طول وحدة البيانات: وهو يمثل الطول الكلي لوحدة بيانات بروتوكول الإنترنت (الترويسة والبيانات) مقاسة بالبايتات. ولأن هذا الحقل طوله 16 بتاً، فإن الحجم الأقصى النظري لوحدة بيانات بروتوكول الإنترنت هو 65535 بايتاً. لكن من النادر أن تكون وحدات البيانات أكبر من 1500 بايت.
- المُعرّف (مُميّز الرزمة) والأعلام والعنوان النسبي للتجزئة: تُستخدم هذه الحقول الثلاثة مع ما يسمّى بـ "التجزئة" (fragmentation)، وهو موضوع سوف ندرسه بتعمّق بعد قليل. وبشكلٍ مثيرٍ للانتباه لا تسمح النسخة الجديدة لبروتوكول الإنترنت IPv6 بالتجزئة في الموجّهات.
- فترة العُمر ((Time-To-Live (TTL): يستعمل هذا الحقل لضمان أن وحدة البيانات لا تظل تدور إلى الأبد خلال الشبكة (على سبيل المثال بسبب وجود حلقة توجيه طويلة الأمد (long-lived routing loop). تخفض قيمة هذا الحقل

بمقدار واحد في كل مرة تُعالج وحدة البيانات بموجّه. ويجب أن تُسقَط وحدة البيانات إذا أصبحت قيمة الحقل تساوى صفراً.

- البروتوكول: يُستعمَل هذا الحقل فقط عندما تصل وحدة البيانات إلى الوجهة حيث تشير قيمته إلى البروتوكول المحدد في طبقة النقل الذي يجب أن يعبر إليه هذا الجزء من وحدة البيانات. على سبيل المثال القيمة 6 تدل على أن هذا الجزء من وحدة البيانات يسلم إلى TCP، بينما القيمة 17 تدل على أنه يسلم إلى UDP. وللاطلاع على قائمة بكل القيم المحتملة راجع [RFC 1700; RFC 3232]. لاحظ أن رقم البروتوكول في وحدة بيانات IP له دور مماثل لدور حقل رقم المنفذ في قطعة بيانات طبقة النقل (segment). ويعتبر رقم البروتوكول الصمغ الذي يربط طبقة الشبكة وطبقة النقل سويةً، في حين يعتبر رقم المنفذ الصمغ الذي يربط طبقة النقل مع طبقة التطبيقات سويةً. سنرى في الفصل الخامس أن إطار طبقة ربط البيانات له أيضاً حقل خاص يربط طبقة ربط البيانات بطبقة الشبكة.
- المجموع التدقيقي للترويسة (header checksum): يساعد هذا الحقل الموجّه في اكتشاف حدوث خطأ في وحدة البيانات المستلمة. يُحسَب المجموع التدقيقي بمعاملة كل بايتين في الترويسة كعدد، ثم تجمع الأعداد الناتجة بطريقة حساب مكمل الواحد (1's complement arithmetic). وكما ناقشنا في الجزء 3-3 يُعرَف مكمل الواحد لهذا المجموع باسم المجموع التدقيقي للإنترنت ويخزَّن في حقل المجموع التدقيقي للرزمة. ويحسب الموجّه المجموع التدقيقي للترويسة لكل وحدة بيانات مستلمة، فإذا كانت القيمة المحسوبة لا تساوى القيمة المتضمنة في وحدة البيانات فإن هذا يدل على حدوث خطأ بها. وعادةً ما تُسقِط الموجّهات وحدات البيانات التي يُكتشَف وجود خطأ فيها. لاحظ أنه يجب أن تُحسَب قيمة المجموع التدقيقي عند كل موجّه ويعاد تخزينها في وحدة البيانات لأن بعض الحقول قد تتغيّر مثل حقل TTL ومن المحتمل حقل الخيارات أيضاً. يمكنك الاطلاع على مناقشة هامة لخوارزميات سريعة لحساب المجموع التدقيقي للإنترنت في [RFC 1071]. وعادة ما يُطرَح سؤال عند تلك النقطة "لماذا يقوم نموذج TCP/IP بفحص

الأخطاء في كلٍّ من طبقة النقل وطبقة الشبكة؟". هناك عدّة أسباب لهذا التكرار. أولاً: لاحظ أنه في طبقة الشبكة يتم حساب المجموع التدقيقي للترويسة فقط، بينما في طبقة النقل يتم حساب المجموع التدقيقي لقطعة البيانات بكاملها. ثانياً: بروتوكول IP وبروتوكول TCP أو UDP لا ينتميان بالضرورة لنفس رصّة البروتوكولات. فمن حيث المبدأ يمكن أن يعمل بروتوكول TCP على بروتوكول مختلف عن IP (مثلاً ATM) وكذلك بروتوكول IP يمكن أن يحمل بيانات غير متجهة إلى أيٍّ من TCP أو UDP.

- عنوان IP للمصدر وعنوان IP للوجهة: عندما يُنشئ المصدر وحدة بيانات فإنه يضع عنوانه في حقل عنوان IP للمصدر ويضع عنوان الوجهة في حقل عنوان IP للوجهة. ويحصل مضيف المصدر على عنوان الوجهة غالباً عن طريق بحث DNS كما نوقش في الفصل الثاني. سوف نناقش عنونة بروتوكول الإنترنت بالتفصيل في الجزء 4-4-2.
- الخيارات: تسمح حقول الخيارات لترويسة وحدة بيانات IP بالتمدد. غير أن خيارات الترويسة نادراً ما تستعمل، لذا كان القرار بعدم تضمين البيانات الموجودة في الحقول الاختيارية بصفة ثابتة في كل وحدة بيانات وذلك لتقليل العبء الإضافي (overhead). ومع ذلك فمجرّد وجود تلك الحقول يعقّد الأمور، فتغيير طول ترويسة وحدة البيانات يعوق إمكانية تحديد مكان بداية حقل البيانات مسبقاً. أيضاً قد تتطلّب بعض وحدات البيانات معالجة الخيارات في حين لا تحتاج بعضها الآخر لذلك، وبالتالي يمكن أن يتفاوت مقدار الوقت اللازم لمعالجة وحدة البيانات في الموجّه تفاوتاً كبيراً. هذه الاعتبارات مهمة جداً لمعالجة IP في الموجّهات والمضيفات ذات الأداء العالي. لهذه الأسباب وأسباب أخرى لم تُستخدَم خيارات IP في ترويسة IPv6 كما سنناقش في الجزء 4-4-4.
- البيانات (الحمل الآجر): نأتي أخيراً إلى الحقل الأخير والأكثر أهمية فهو المبرر الأساسي لرزمة البيانات! يحتوي هذا الحقل في أغلب الأحيان على قطعة بيانات طبقة النقل (من بروتوكول TCP أو UDP) المطلوب تسليمها إلى

وجهتها. ومع ذلك يمكن أن يحمل هذا الحقل أنواعاً أخرى من البيانات كرسائل ICMP (ستناقش في الجزء 4-4-3).

لاحظ أن وحدة بيانات IP بها 20 بايتاً للترويسة (بافتراض عدم وجود خيارات). إذا كانت وحدة البيانات تحمل قطعة TCP فذلك يعني أن كل وحدة بيانات غير مجزأة تحمل ما مجموعه 40 بايتاً للترويسة (20 بايتاً لترويسة IP و20 بايتاً لترويسة TCP) بالإضافة إلى رسالة طبقة التطبيقات.

تجزئة وحدة بيانات IP

سنرى في الفصل الخامس أنه ليست كل بروتوكولات طبقة ربط البيانات يمكن أن تحمل رزم طبقة شبكة بنفس الحجم. يمكن أن تحمل بعض البروتوكولات رزم بيانات كبيرة، بينما يمكن أن تحمل بروتوكولات أخرى رزماً صغيرة فقط. على سبيل المثال يمكن أن تحمل إطارات إيثرنت في حدود 1500 بايت من بايتات البيانات، بينما لا يمكن أن تحمل إطارات بعض وصلات شبكة المنطقة الواسعة (Wide-Area Network (WAN)) أكثر من 576 بايتاً. الكمية القصوى للبيانات التي يمكن أن يحملها إطار طبقة ربط البيانات تدعى وحدة الإرسال القصوى (MTU). ولأن كل وحدة بيانات IP تكون مُغلّفة ضمن إطار طبقة ربط البيانات لنقلها من موجّه لآخر تمثل الكمية MTU ببروتوكول طبقة ربط البيانات حدّاً أقصى لطول وحدة بيانات IP. لا يُشكِّل وجود مثل هذا الحد الأقصى على حجم وحدة بيانات IP مشكلةً كبيرةً. وإنما تأتي المشكلة من استخدام بروتوكولات مختلفة في طبقة ربط البيانات على الوصلات المختلفة على طول المسار بين المصدر والوجهة وكلٌّ منها تستخدم قيماً مختلفة لـ MTU.

ولفهم قضية التمرير بطريقة أفضل تخيل بأنك موجّهٌ يربط عدّة وصلات لكلٍّ منها بروتوكول مختلف لطبقة ربط البيانات ولها حجم أقصى مختلف لوحدة النقل (MTU). افترض أنك عندما تستلم وحدات بيانات IP من وصلة واحدة تقوم بفحص جدول التمرير لديك لتقرير وصلة المخرج. افترض أن وصلة المخرج هذه لها MTU أصغر من طول وحدة بيانات IP. كيف لك أن تضغط وحدة بيانات IP تلك

الكبيرة جداً في حقل الحمل الآجر لإطار طبقة ربط البيانات؟ يتمثل الحل في تجزئة وحدة بيانات IP الأصلية إلى وحدتين أو أكثر تكون أصغر حجماً ، ثم تغليف كلٍّ منهما في إطار منفصل يُرسَل على وصلة المخرج. يُطلَق على كل وحدة من وحدات البيانات الأصغر تلك جزءاً (fragment).

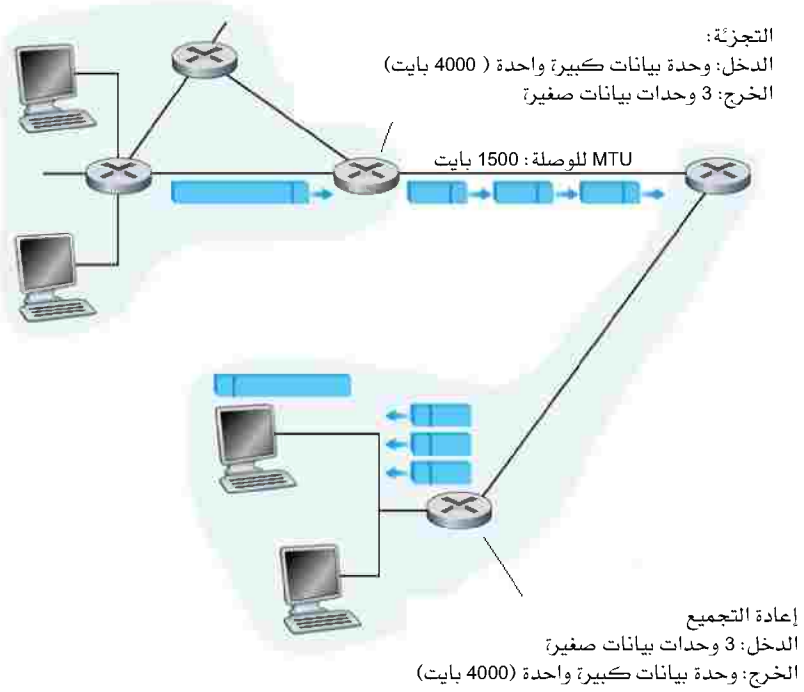
يجب إعادة تجميع الأجزاء قبل تسليمها إلى طبقة النقل في الوجهة. وفي واقع الأمر يتوقع كلٌّ من TCP وUDP استلام قطعاً كاملة غير مجزأة من طبقة الشبكة. لقد أحس مصممو بروتوكول IPv4 أن إعادة تجميع الوحدات الجزئية في الموجّهات سيؤدي إلى تعقيد ملحوظ في البروتوكول مما يقلل كفاءته. تخيل نفسك مكان الموجّه، هل تريد القيام بإعادة تجميع الوحدات الجزئية بجانب كل شيء آخر يفترض أن تقوم به؟ بالتمسّك بمبدأ إبقاء الشبكة الرئيسة بسيطة، قرّر مصممو IPv4 إيكال مهمة إعادة تجميع الوحدات الجزئية إلى الأنظمة الطرفية بدلاً من موجّهات الشبكة.

عندما يستلم مضيف الوجهة سلسلة من وحدات البيانات من نفس المصدر يحتاج لتحديد ما إذا كانت أيٌّ من تلك الوحدات هي أجزاء من وحدة أصلية أكبر. إذا كانت بعض وحدات البيانات الواصلة أجزاء، فعليه أيضاً أن يقرّر متى استلم الجزء الأخير وكيف يجب أن توضع الأجزاء التي استلمها سويةً لتشكيل وحدة البيانات الأصلية. وللسماح لمضيف الوجهة بأداء هذه المهمّة، وضع مصممو بروتوكول الإنترنت (النسخة 4) حقول المعرّف (identifier) والعلم (flag) والعنوان النسبي للتجزئة (fragmentation offset) في ترويسة وحدة بيانات بروتوكول الإنترنت. عند تكوين وحدة بيانات يختم مضيف الإرسال وحدة البيانات بعدد تعريفي (المعرّف) بالإضافة إلى عناوين المصدر والوجهة. ويزيد مضيف الإرسال العدد التعريفي لكل وحدة بيانات يرسلها بعد ذلك. وعندما يحتاج موجّه لتجزئة وحدة بيانات تختم كل الوحدات الجزئية الناتجة بنفس عنوان المصدر وعنوان الوجهة والعدد التعريفي لوحدة البيانات الأصلية. بعد أن تستلم الوجهة سلسلة من وحدات البيانات من نفس مضيف الإرسال، تفحص العدد التعريفي لكل وحدة بيانات لتقرير أيٍّ منها يمثل في الحقيقة جزءاً من وحدة أكبر. ولأن بروتوكول الإنترنت

يوفر خدمة غير موثوق فيها لنقل البيانات، فيمكن ألا يصل جزء أو أكثر من تلك الأجزاء إلى الوجهة. لهذا السبب، وحتى يتمكّن مضيف الوجهة من التأكد من استلام الجزء الأخير للوحدة الأصلية ، توضع قيمة حقل العلم 0 بهذا الجزء بينما تكون قيمة هذا الحقل في كل الأجزاء الأخرى 1. أيضاً لكي يتمكّن مضيف الوجهة من تقرير ما إذا كان جزء قد فقد (وأيضاً من إعادة تجميع الأجزاء في ترتيبها الصحيح)، يُستعمَل حقل العنوان النسبي ليحدّد أين يقع هذا الجزء ضمن وحدة البيانات الأصلية.

يوضح الشكل 4-14 مثالاً لذلك حيث تصل وحدة بيانات مكونة من 4000 بايت (20 بايتاً للترويسة بالإضافة إلى 3980 بايتاً حمل آجر) إلى موجّه، وعليه أن يرسلها إلى وصلة ذات حجم أقصى لوحدة البيانات MTU يساوي 1500 بايت. يشير هذا ضمناً إلى أن بايتات البيانات الـ 3980 في الوحدة الأصلية يجب أن تقسم إلى ثلاثة أجزاء منفصلة (كلٌّ منها سيمثل أيضاً وحدة بيانات IP). افترض أن الوحدة الأصلية مختومة بعدد تعريفي قيمته 777. يوضح الجدول 4-2 خصائص الأجزاء الثلاثة. تعكس القيم في الجدول 4-2 المطلب بأن كمية بيانات الحمل الآجر الأصلية في كل جزء فيما عدا الجزء الأخير يجب أن تكون مضاعفات لـ 8 بايتات، وأن تحدد قيم حقل العنوان النسبي بوحدات مكونة من 8 بايتات.

عند الوجهة تعبر بيانات الحمل الآجر لوحدة البيانات فقط إلى طبقة النقل بعد أن تكون طبقة الشبكة قد أعادت بناء الوحدة الأصلية بالكامل. إذا لم يصل جزء أو أكثر إلى الوجهة فسوف تسقط وحدة البيانات التي ينقصها ذلك الجزء بالكامل ولا تمرر إلى طبقة النقل. لكن – كما عرفنا في الفصل السابق – إذا استُخدم بروتوكول TCP في طبقة النقل فإنه سيعوّض هذا الفقد بجعل المصدر يعيد إرسال البيانات المفقودة من جديد.



الشكل 4-14 تجزئة وإعادة تجميع وحدة بيانات بروتوكول IP.

جزء الرزمة	عدد البايتات	الرقم التعريفي	قيمة العنوان النسبي	قيمة بت العَلَم
الأول	1480 بايت من بيانات رزمة IP	777	0 (أي يجب وضع البيانات في البداية عند البايت 0)	1 (أي أنه ليس الجزء الأخير في الرزمة)
الثاني	1480 بايت من البيانات	777	135 (أي يجب وضع البيانات عند البايت 1480، حيث أن 8 × 135 = 1480)	1 (أي أنه ليس الجزء الأخير في الرزمة)
الثالث	1020 بايت من البيانات (وهي البايتات المتبقية)	777	370 (أي يجب وضع البيانات عند البايت 2960، حيث أن 8 × 370 = 2960)	0 (أي أنه الجزء الأخير في الرزمة)

الجدول 4-2 الوحدات الجزئية الناتجة.

لقد عرفنا للتوّ أن التجزئة في بروتوكول الإنترنت تلعب دوراً مهماً في توصيل العديد من التقنيات المتباينة لطبقة ربط البيانات. لكن التجزئة لها ثمنها أيضاً. فهي أولاً تُعقّد الأنظمة الطرفية والموجّهات، حيث يتعين أن تصمم بحيث يمكنها القيام بالتجزئة وإعادة تجميع وحدات البيانات. ثانياً يمكن أن تُستخدَم التجزئة لشن هجمات قاتلة لحجب الخدمة (DoS)، حيث يرسل المهاجم سلسلة من الوحدات الجزئية الغريبة وغير المتوقّعة. وكمثال تقليدي لذلك ما يعرف بهجوم Jolt2، حيث يرسل المهاجم فيضاً من الوحدات الجزئية الصغيرة ـ التي ليس لأيٍّ منها القيمة "صفر" في حقل العنوان النسبي ـ إلى مضيف الهدف. يمكن أن ينهار مضيف الهدف وهو يحاول إعادة بناء وحدات بيانات من تلك الوحدات الجزئية التالفة. وفي نوع آخر من تلك الحيل يتم إرسال وحدات جزئية متداخلة (أي لها قيم عناوين نسبية لا تسمح بإعادة وضع الوحدات الجزئية بشكلٍ صحيح). يمكن أن تنهار أنظمة التشغيل الضعيفة ـ أي التي لا تعرف ماذا تفعل مع تلك الوحدات الجزئية المتداخلة [Skoudis 2006]. وسنرى في نهاية هذا الجزء أن النسخة الجديدة من بروتوكول الإنترنت IPv6 تخلصت بالجملة من التجزئة، وذلك لتحسين معالجة وحدات البيانات وجعله أقل عرضة للهجمات.

يوجد على موقع الويب لهذا الكتاب (http://www.awl.com/kurose-ross) برنامج جافا صغير لتوليد وحدات جزئية. من خلال هذا البرنامج يحدد المستخدم حجم وحدة البيانات الواصلة وقيمة MTU وعدد تعريفي لتلك الرزمة، فيولّد البرنامج الوحدات الجزئية آلياً.

2-4-4 العنونة في بروتوكول IPv4

سنحوّل انتباهنا الآن إلى عناوين IPv4. بالرغم من أنك قد تعتقد بأن العنونة يجب أن تكون موضوعاً بسيطاً إلا أننا نأمل أن تقتنع مع نهاية هذا الفصل بأن هذا الموضوع ليس فقط مثيراً ودقيقاً بل وأنه يحظى بقدرٍ كبير من الأهمية في الإنترنت. من المعالجة الممتازة لموضوع عنونة IPv4 المقال [Com Addressing 20073] والفصل الأول في كتاب [Stewart 1999].

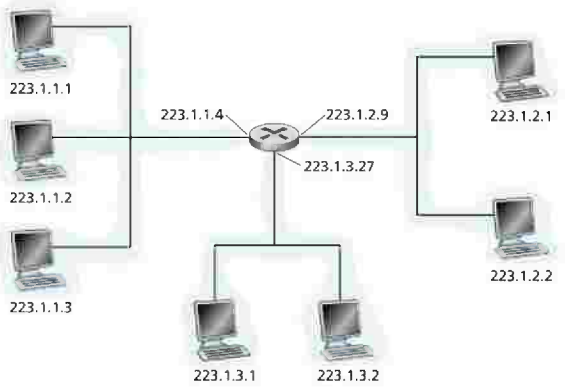
لكن قبل مناقشة عنونة IP سنحتاج لقول بضع كلمات حول كيفية توصيل المضيفات والموجّهات بالشبكة. يُوصّل المضيف بالشبكة عادةً بوصلة واحدة والتي عن طريقها يرسل بروتوكول IP على المضيف وحدات البيانات للشبكة. يطلق على نقطة تلاقي المضيف والوصلة المادية "واجهة" (interface). لننظر الآن إلى موجّه وواجهاته. نظراً لأن وظيفة الموجّه هي استقبال وحدة بيانات على وصلةٍ ما وإعادة إرسالها على وصلة أخرى، فإن الموجّه يوصّل بالشبكة بالضرورة عن طريق وصلتين أو أكثر. يطلق أيضاً على نقطة تلاقي الموجّه وأي وصلة مادية "واجهة"، وبالتالى يكون للموجّه واجهات متعددة (واحدة لكل وصلة). لأن كل مضيف وموجّه قادرٌ على إرسال واستلام وحدات بيانات IP ، يتطلب برتوكول IP أن يكون لكل واجهة لمضيف أو موجّه عنوان IP خاص بها (أي أنه من الناحية الفنية يرتبط عنوان IP بواجهة ما وليس بالمضيف أو بالموجّه الذي يحتوي على تلك الواجهة).

يتكون كل عنوان IP من 32 بتاً (أي أربعة بايتات)، ولهذا يكون العدد الإجمالي لعناوين IP المحتملة = 2^{32}. بتقريب 2^{10} إلى 10^3 فمن السّهل ملاحظة أن هناك حوالي 4 بلايين عنوان IP محتمل. وفي العادة تكتب تلك العناوين في صيغة عشرية منقوطة، أي يكتب كل بايت من العنوان في شكلٍ عشري ويفصل بنقطة عن البايتات الأخرى في العنوان. على سبيل المثال يتكون العنوان 193.32.216.9 من أربعة بايتات: تمثل القيمة 193 (بنظام العد العشري) البايت الأول، والقيمة 32 البايت الثاني، وهكذا. والصيغة المكافئة لهذا العنوان بنظام العد الثنائي هي

11000001 00100000 11011000 00001001

يجب أن يكون لكل واجهة على كل مضيف وموجّه في شبكة الإنترنت العالمية عنوان IP فريدٌ عالمياً (أي غير مكرر). يُستثنى من ذلك الواجهات وراء أنظمة NAT كما سنناقش في نهاية هذا الجزء. لكن لا يمكن اختيار تلك العناوين بطريقة عشوائية. فجزء من عنوان بروتوكول الإنترنت للواجهة يحدد الشبكة الفرعية التي تتصل بها تلك الواجهة.

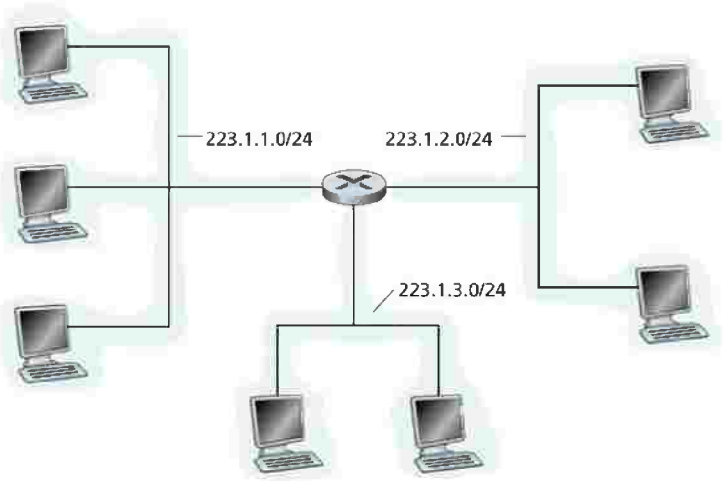
يبين الشكل 4-15 مثالاً لعنونة IP للواجهات. حيث يظهر موجّه واحد (بثلاث واجهات) يربط بين سبعة مضيفات. لنلقِ نظرة فاحصة على عناوين بروتوكول الإنترنت المخصّصة لواجهات المضيفات والموجّهات حيث توجد عدّة أمور يجب ملاحظتها. إن عنوان IP لكلٍّ من واجهات المضيفات الثلاثة في الجزء الأعلى يساراً في الشكل 4-15 وواجهة الموجّه التي يرتبط بها كلٌّ منهم له الصيغة 223.1.1.xxx.



الشكل 4-15 عناوين الواجهات والشبكات الفرعية.

أي أن لكلٍّ منها نفس الـ 24 بتاً من ناحية اليسار. ترتبط الوصلات الأربعة أيضاً ببعضها البعض من قِبَل شبكة لا تحتوي على موجّهات. يمكن أن تكون هذه الشبكة على سبيل المثال شبكة إيثرنت محلية حيث توصل الواجهات بمجمّع إيثرنت (hub) أو محول إيثرنت (switch) (انظر الفصل الخامس). في مصطلحات بروتوكول الإنترنت تعد الشبكة التي تصل بين واجهات المضيفات الثلاثة وأحد واجهات الموجّه شبكة فرعية [RFC 950] (يطلق أيضاً على تلك الشبكة شبكة IP أو ببساطة شبكة). تخصّص عنونة IP العنوان 223.1.1.0/24 لهذه الشبكة الفرعية، وأحياناً يطلق على الصيغة 24/ قناع الشبكة الفرعية (subnet mask)، وهي تشير إلى أن الـ 24 بتاً من يسار العنوان تمثل عنوان الشبكة الفرعية. تتكون الشبكة الفرعية 223.1.1.0/24 من ثلاث واجهات للمضيفات (223.1.1.1، 223.1.1.2، 223.1.1.3) وواجهة موجّه واحدة (223.1.1.4). ويجب أن يكون لأي مضيفات أخرى

توصَّل بالشبكة الفرعية 223.1.1.0/24 عنوان بالصيغة 223.1.1.xxx. توجد شبكتان فرعيتان إضافيتان في الشكل 4-15: شبكة 223.1.2.0/24 وشبكة 224.1.3.0/24 يوضح الشكل 4-16 شبكات IP الفرعية الثلاثة الموجودة في الشكل 4-15.



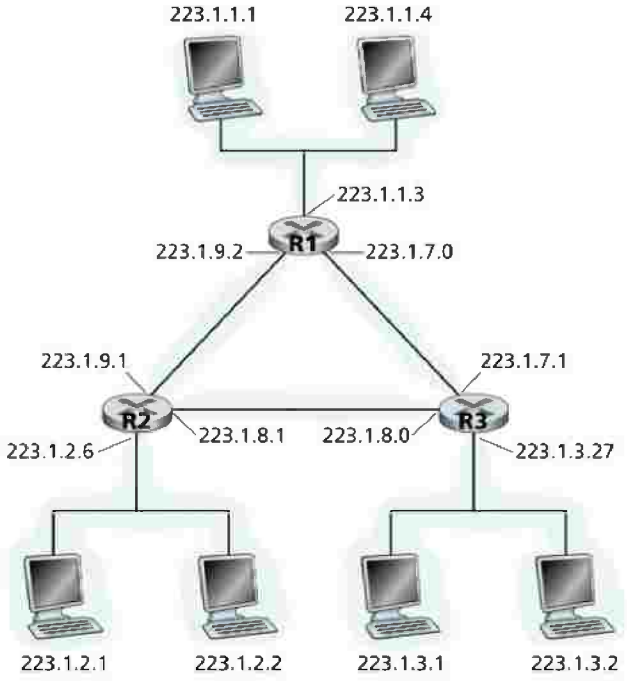
الشكل 4-16 عناوين الشبكات الفرعية.

لا يقتصر تعريف بروتوكول IP لشبكة فرعية على قطع الإيثرنت (Ethernet segments) التي توصل عدة مضيفات إلى واجهة موجّه. ولتوضيح ذلك انظر الشكل 4-17 حيث يوجد ثلاثة موجّهات متصلة مع بعضها البعض بوصلات من نوع "نقطة إلى نقطة". كل موجّه له ثلاث واجهات: واحدة لكل وصلة "نقطة إلى نقطة" وواحدة لوصلة الإذاعة التي توصل الموجّه مباشرة مع زوج من المضيفات. ما هي إذن الشبكات الفرعية الموجودة هنا؟ توجد ثلاث شبكات فرعية بالعناوين 223.1.1.0/24 و223.1.2.0/24 و223.1.3.0/24، وهي مشابهةٌ لتلك الموجودة في الشكل 4-15. كما أن هناك ثلاث شبكات فرعية إضافية في هذا المثال: شبكة فرعية 223.1.9.0/24 للواجهات التي توصّل الموجّه R1 مع R2 وشبكة فرعية أخرى 223.1.8.0/24 للواجهات التي توصل الموجّه R3 مع R2 وشبكة فرعية ثالثة 223.1.7.0/24 للواجهات التي توصل الموجّه R3 مع R1. في شبكة عامة مكونة من

مضيفات وموجّهات، يمكن استخدام الوصفة التالية لتعريف الشبكات الفرعية التي تتضمنها تلك الشبكة:

لتحديد الشبكات الفرعية نفصل كل واجهة من كل المضيفات والموجّهات، وبالتالي تصبح الشبكة عدداً من الجزر المعزولة، حيث تمثل الواجهات النهايات للنقاط الطرفية لتلك الجزر. ويطلق على كل جزيرة من تلك الجزر المعزولة شبكة فرعية.

إذا طبّقنا هذه القاعدة على الشبكة في الشكل 4-17 فسنحصل على ست جزر تمثل شبكات فرعية.



الشكل 4-17 ثلاثة موجّهات تربط بين ست شبكات فرعية.

من المناقشة السابقة يتضح أن شبكة منظمة (كشركة أو مؤسسة أكاديمية) مكونة من عددٍ من قطع الإيثرنت ووصلات "نقطة إلى نقطة" سيكون فيها شبكات فرعية متعدّدة، وسيكون للأجهزة على كل شبكة فرعية نفس عنوان الشبكة الفرعية. يمكن من حيث المبدأ أن تأخذ الشبكات الفرعية المختلفة عناوين شبكة فرعية مختلفة جداً. لكن عملياً ستشترك عناوين الشبكات الفرعية في أغلب الأحيان في أمور كثيرة. لفهم السبب دعنا نلفت الانتباه لكيفية معالجة العنونة في شبكة الإنترنت العالمية.

تعرف استراتيجية تخصيص عناوين الإنترنت بأسلوب التوجيه اللانوعي بين النطاقات (Classless InterDomain Routing (CIDR)) [RFC 4632]، وهي تعميم لفكرة عنونة الشبكة الفرعية. كما هو الحال مع عنونة الشبكة الفرعية يقسّم العنوان المكون من 32 بتاً إلى جزأين ويكتب أيضاً في الصيغة العشرية المنقوّطة *a.b.c.d/x* حيث تشير *x* إلى عدد البتات في الجزء الأول من العنوان.

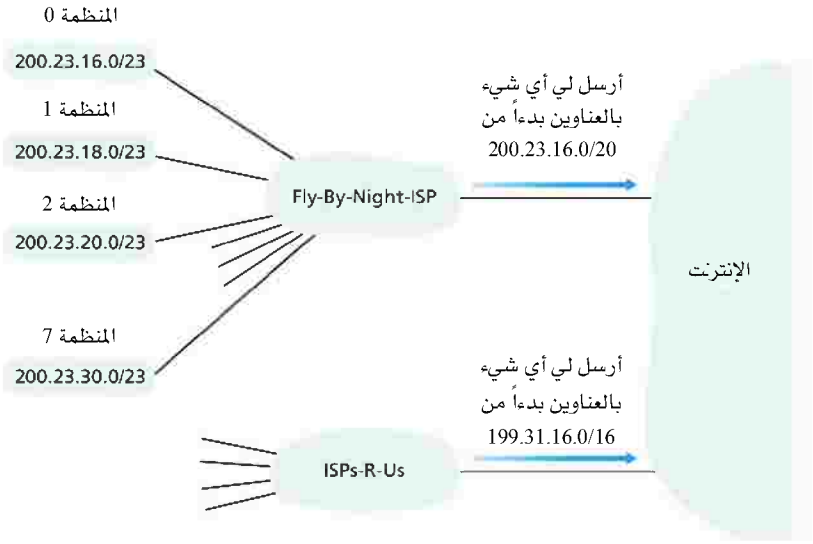
تمثل البتّات الأعلى رتبة، وعددها *x* بت، من عنوان ما بالصيغة *a.b.c.d/x* عنوان الشبكة التي ينتمي لها هذا العنوان، وفي أغلب الأحيان يشار إليها باسم البادئة (أو بادئة الشبكة) للعنوان. وعادة ما يخصّص لمنظمة ما عددٌ من العناوين المتجاورة – أي التي لها نفس البادئة (ويطلق عليها كتلة العناوين (address block)). في تلك الحالة ستشترك عناوين IP للأجهزة الموجودة ضمن شبكة المنظمة في البادئة. سنرى عندما نغطّي بروتوكول التوجيه BGP في الجزء 4-6 أن الموجّهات خارج شبكة المنظمة تفحص فقط بتات البادئة *x* هذه عند اختيار مسار وحدات البيانات. وهذا يُخفض حجم جداول التوجيه إلى حدٍّ كبير في تلك الموجّهات، لأنه سيكفي مُدخل واحد فقط بالصيغة *a.b.c.d/x* بالجداول لإرسال وحدات البيانات لأي وجهة ضمن المنظمة.

المبادئ في الواقع العملي (Principles in Practice)

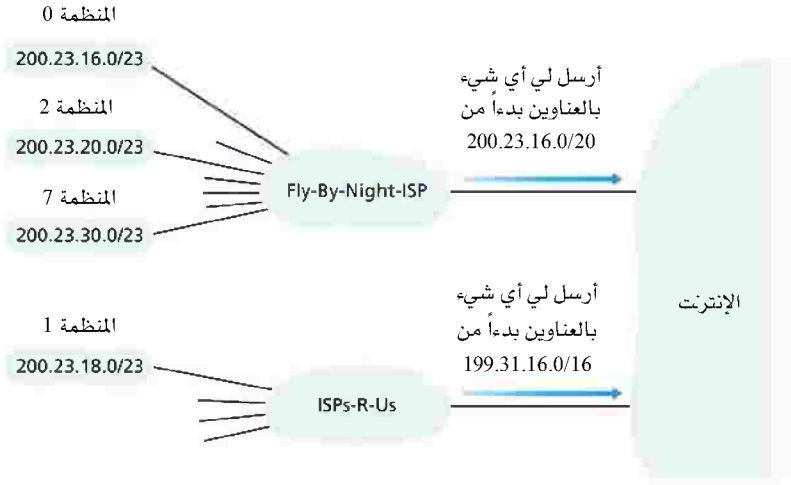
هذا المثال لموفر لخدمة الإنترنت يوصِّل ثماني منظمات إلى الإنترنت ويوضح بشكلٍ رائع كيف خُصِّصت عناوين CIDR بعناية لتسهيل التوجيه. افترض ـ كما هو موضح في الشكل 4-18 أن موفر خدمة الإنترنت (والذي سنطلق عليه Fly-By-Night-ISP) يعلن للعالم الخارجي بأنه يجب أن ترسل إليه أي رزم الـ 20 بتاً الأولى في عناوينها تطابق 200.23.16.0/20. لا يلزم أن يعرف بقيّة العالم أن ضمن كتلة العناوين 200.23.16.0/20 توجد في الحقيقة ثماني منظمات أخرى لكلٍّ منها شبكاتها الفرعية الخاصة بها. غالباً ما يُشار إلى القدرة على استعمال بادئة واحدة بتجميع العناوين (address aggregation) (أيضاً تسمى تجميع أو دمج المسارات).

يعمل هذا الأسلوب بطريقة جيدة للغاية عندما تُخصّص العناوين على شكل كتل لموفري خدمة الإنترنت ومنهم إلى المنظمات الزبائن. لكن ماذا يحدث عندما لا تُخصص العناوين بهذا الأسلوب الهرمي (hierarchical)؟

ماذا كان سيحدث – على سبيل المثال – إذا امتلك موفر خدمة الإنترنت Fly-By-Night-ISP موفر خدمة إنترنت آخر يدعى ISPs-R-Us ثم جعل المنظمة 1 تتصل بالإنترنت من خلال ذلك التابع ISPs-R-Us؟ كما هو موضح في الشكل 4-18 يمتلك موفر خدمة الإنترنت التابع ISPs-R-Us كتلة العناوين؛ لكن ولسوء الحظ عناوين المنظمة 1 خارج هذه الكتلة. ما الذي يجب فعله هنا؟ بالتأكيد يمكن أن تعيد المنظمة 1 ترقيم كل موجّهاتها ومضيفاتها لتكون عناوينها ضمن كتلة موفر خدمة الإنترنت ISPs-R-Us. إلا أن هذا الحل مكلّف، خصوصاً أن المنظمة 1 قد يعاد توصيلها في المستقبل عن طريق موفر خدمة إنترنت تابع آخر. إن الحل الذي غالباً ما يُستخدَم هو جعل المنظمة 1 تحتفظ بالعناوين في الكتلة 200.23.18.0/23. في هذه الحالة – كما هو مبين في الشكل 4-19 – يواصل موفر خدمة الإنترنت Fly-By-Night-ISP إعلان كتلة العناوين 200.23.16.0/20 ويواصل موفر خدمة الإنترنت ISPs-R-Us إعلان 199.31.16.0/16. ومن ناحية أخرى يعلن موفر خدمة الإنترنت ISPs-R-Us كتلة العناوين للمنظمة 1 أي 200.23.18.0/23. عندما ترى الموجّهات الأخرى في الإنترنت كتلة العناوين 200.23.16.0/20 (من Fly-By-Night-ISP) وكتلة العناوين 200.23.18.0/23 (من ISPs-R-Us) وتريد توجيه رزم لوجهة تقع في نطاق الكتلة 200.23.18.0/23، ستستعمل تطابق البادئة الأطول (longest prefix matching) (راجع الجزء 4-2-2) وتوجه نحو موفر خدمة الإنترنت ISPs-R-Us لأنه في هذه الحالة يمثل البادئة الأطول (أي الأكثر تحديداً) التي تطابق عنوان الوجهة.



الشكل 4-18 العنونة الهرمية وتجميع المسارات.



الشكل 4-19 ISPs-R-Us له مسار أكثر تحديداً للمنظمة 1.

يمكن اعتبار بقية البتات في العنوان على أنها تُميّز بين الأجهزة ضمن المنظمة التي لها نفس بادئة الشبكة. هذه البتات هي التي ستُفحص عند توجيه الرزم داخل المنظمة. وقد يكون (أو لا يكون) لهذه البتات ذات الرتبة الأدنى تركيب لتفريع شبكي إضافي كالذي ناقشناه من قبل. على سبيل المثال افترض أن البتات الـ 21 الأولى من العنوان a.b.c.d/21 تحدّد بادئة شبكة المنظمة وهي ثابتة في عناوين كل الأجهزة في تلك المنظمة. أما البتات الباقية الإحدى عشرة الأخرى فهي لتمييز المضيفات في المنظمة. قد يكون التركيب الداخلي لشبكة المنظمة بحيث تستخدم تلك البتات الإحدى عشرة في أقصى اليمين لعناوين الشبكات الفرعية ضمن المنظمة كما ذكرنا سابقاً. على سبيل المثال قد يشير العنوان a.b.c.d/24 إلى شبكة فرعية معينة ضمن المنظمة.

قبل استخدام أسلوب CIDR للعنونة كان جزء العنوان الذي يدل على الشبكة مقيداً بواحد من الأطوال 8 أو 16 أو 24 بتاً، وهو ما عرف بالعنونة النوعية (classful) وتعرف الشبكات التي تنتمي لكل نوع من هذه العناوين بالفئة A أو B أو C على الترتيب. لكن المطلب أن يكون طول الجزء الدال على الشبكة لعنوان محصوراً في تلك القيم (أي 1 أو 2 أو 3 بايتات) سبّب مشكلةً لدعم العدد المتزايد بسرعة من المنظمات التي تمتلك شبكات فرعية صغيرة ومتوسطة الحجم. كما أن تخصيص عناوين من الفئة C (/24) يدعم فقط $2^8 - 2 = 254$ مضيف كحد أقصى (حيث إن اثنين من تلك العناوين محجوزان للاستعمال الخاص) والذي قد يكون صغيراً جداً بالنسبة للعديد من المنظمات. في حين أن أقصى عدد تدعمه الفئة B (/16) من المضيفات يساوي 65634 مضيفاً والذي قد يعتبر كبيراً جداً لتلك المنظمات. بالتالي إذا استخدمنا هذا الأسلوب للعنونة فإن منظمة لديها فقط 2000 مضيفٍ ستحتاج إلى عنوان من الفئة B؛ الأمر الذي يؤدي لاستنزاف سريع لفضاء عناوين الفئة B واستخدام سيّئ للعناوين المخصّصة. على سبيل المثال ستستخدم المنظمة السابقة التي لديها 2000 مضيفٍ فقط 2000 عنوانٍ من عناوين الفئة B التي خصصت لها تاركة بذلك أكثر من 63000 عنوانٍ لا يمكن استخدامها من قِبَل منظمات أخرى.

سنكون مقصرين إذا لم نذكر نوعاً آخر من عناوين بروتوكول الإنترنت وهو عنوان الإذاعة 255.255.255.255. فعندما يرسل مضيف وحدة بيانات لهذا العنوان كعنوان الوجهة تُسلّم الرسالة إلى كل المضيفات على نفس الشبكة الفرعية، ويمكن أن ترسل الموجّهات الرسالة إلى الشبكات الفرعية المجاورة أيضاً (غير أن هذا الاختيار لا يُستخدَم عادة).

بعد أن درسنا عنونة بروتوكول الإنترنت بالتفصيل نحتاج لمعرفة كيفية حصول المضيفات والشبكات الفرعية على عناوينها في البداية. دعنا نبدأ بالنظر إلى كيفية حصول منظمة ما على كتلة عناوين لأجهزتها، ثم إلى كيفية حصول جهاز (كمضيف مثلاً) على عنوان من بين كتلة عناوين المنظمة.

الحصول على كتلة العناوين

لكي تحصل منظمة على كتلة عناوين للاستعمال ضمن شبكتها قد يتّصل المشرف على الشبكة أولاً بموفر خدمة الإنترنت لتخصيص عناوين من كتلة أكبر من العناوين التي تم تخصيصها لموفر الخدمة من قبل. على سبيل المثال افترض أن موفر خدمة إنترنت قد حصل على الكتلة 200.23.16.0/20. يقوم موفر خدمة الإنترنت بدوره بتقسيم تلك الكتلة إلى ثماني كتل متجاورة بأحجام متساوية ويعطي لكل منظمة من المنظمات الثماني التي يدعمها كتلة منها كما هو مبين أدناه (وللتوضيح قمنا بوضع خط تحت جزء الشبكة الفرعية لهذه العناوين):

كتلة موفّر خدمة الإنترنت	200.23.16.0/20	**11001000 00010111 0001**0000 00000000
المنظمة 0	200.23.16.0/23	**11001000 00010111 0001000**0 00000000
المنظمة 1	200.23.18.0/23	**11001000 00010111 0001001**0 00000000
المنظمة 2	200.23.20.0/23	**11001000 00010111 0001010**0 00000000
.....	.....	.....
المنظمة 7	200.23.30.0/23	**11001000 00010111 0001111**0 00000000

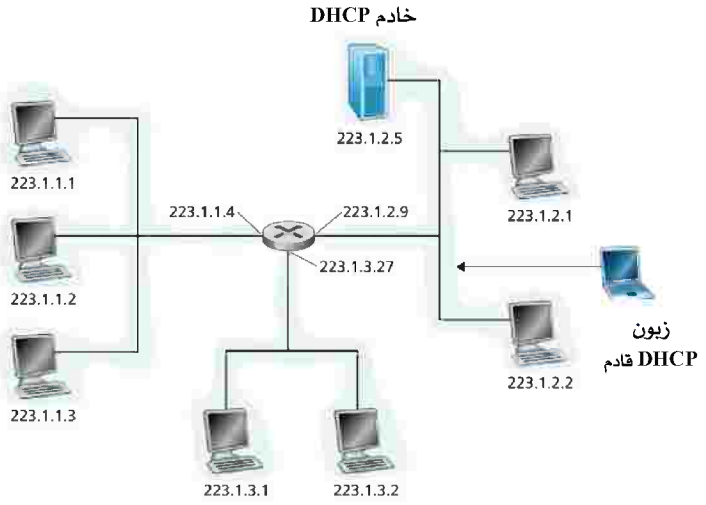
ليست هذه هي الطريقة الوحيدة للحصول على العناوين ولكنها إحدى الطرق. واضح أنه يجب أيضاً أن تكون هناك طريقة لموفر خدمة الإنترنت نفسه للحصول على كتلة عناوين، فهل هناك سلطة عالمية لها مسؤولية نهائية لإدارة فضاء عناوين الإنترنت وتخصيص كتل منها لموفري خدمة الإنترنت والمنظمات الأخرى؟ في الحقيقة نعم هناك سلطة! فعناوين الإنترنت مدارة تحت سلطة شركة الإنترنت للأسماء والأعداد المخصّصة (ICANN) [ICANN 2007] بناءً على الإرشادات الموجودة بـ RFC 2050. ودور هذه المنظمة اللاربحية ليس فقط تخصيص عناوين بروتوكول الإنترنت ولكن أيضاً إدارة خادمات أسماء النطاقات الجذرية (DNS root servers). كذلك تعمل على تخصيص أسماء النطاقات وحل النزاعات المتعلقة بها. تخصّص ICANN عناوين مكاتب تسجيل الإنترنت الإقليمية (مثل: ARIN، RIPE، APNIC، LACNIC) والتي تُشكِّل سويةً المنظمة المساندة للعناوين لـ (ICANN) [ASO-ICANN 2007] وتعالج تخصيص وإدارة العناوين ضمن مناطقها.

الحصول على عنوان مضيف: بروتوكول تهيئة المضيف الديناميكي (DHCP)

بعد أن تحصل منظمة على كتلة عناوين يمكنها تخصّيص عناوين لواجهات الموجّهات والمضيفات لديها. عادة ما يقوم المسؤول عن الشبكة بتهيئة عناوين IP للموجّهات يدوياً (غالباً ما يتم ذلك عن بُعد باستخدام أداة إدارة الشبكة). وبالمثل يمكن أيضاً تهيئة عناوين المضيفات بطريقة يدوية إلا أنه في أغلب الأحيان تستخدم هذه العملية بروتوكول DHCP لتهيئة المضيفات ديناميكياً [RFC 2131]. يسمح بروتوكول DHCP لمضيف بالحصول على عنوان IP آلياً. يمكن أن يهيئ المشرف على الشبكة بروتوكول DHCP بحيث يعطي دائماً نفس العنوان لمضيف ما في كل مرة يتصل بالشبكة، أو قد يخصّص العنوان مؤقتاً للمضيف وبالتالي سيكون العنوان مختلفاً في كل مرة. بالإضافة إلى مهمة تخصيص عناوين المضيفات يسمح بروتوكول DHCP أيضاً للمضيف بالحصول على معلومات إضافية مثل قناع شبكته الفرعية وعنوان الموجّه الأول (والذي يطلق عليه في أغلب الأحيان البوابة الاعتيادية (default gateway) وعنوان خادم أسماء النطاقات المحلي.

بسبب قدرة بروتوكول DHCP على أتمتة خصائص الشبكة فيما يتعلق بتوصيل مضيف بها، فإنه غالبا ما يطلق عليه بروتوكول "وصّل وشغّل". هذه القدرة تجعله جذّاباً جداً لمشرف الشبكة الذي بدونه كان سيؤدي تلك المهمة يدوياً! كما يتمتّع بروتوكول DHCP بالاستعمال الواسع الانتشار في شبكات الإتصال بالإنترنت السكني وفي الشبكات المحلية اللاسلكية، حيث تتصل المضيفات بالشبكة وتغادرها كثيراً. تصوّر على سبيل المثال الطالب الذي يحمل حاسباً نقالاً من غرفة مسكنه إلى المكتبة إلى قاعة الدروس. من المحتمل أنه في كل موقع سيوصل بشبكة فرعية جديدة ولذلك سيحتاج عنوان IP جديد في كل موقع. يناسب DHCP بطريقة مثالية تلك الحالة، حيث العديد من مستخدمي الشبكة يجيؤون ويغادرون، والتي تكون فيها العناوين مطلوبة لفترة محدودة فقط. بالمثل يفيد DHCP بنفس الطريقة في شبكات الوصول السكني لموفر خدمة إنترنت. تصور مثلاً موفر خدمة إنترنت سكني لديه 2000 عميلٍ لكن لا يتصل أكثر من 400 عميلٍ منهم بالإنترنت في نفس الوقت. في هذه الحالة بدلاً من الحاجة إلى كتلة من 2048 عنوانٍ يمكن لموفر الخدمة استخدام خادم DHCP الذي يخصّص العناوين ديناميكياً، وفي هذه الحالة سيحتاج فقط كتلة من 512 عنواناً (على سبيل المثال كتلة عناوين بالصيغة a.b.c.d/23). وسيقوم خادم DHCP بتحديث قائمة العناوين المتوفرة لديه بينما تلتحق المضيفات بالشبكة أو تغادرها. في كل مرة ينضمّ مضيف للشبكة يخصّص خادم DHCP عنواناً اعتباطياً من القائمة الحالية للعناوين المتوفرة، وفي كل مرة يغادر مضيّف الشبكة يرجع عنوانه إلى قائمة العناوين المتوفرة.

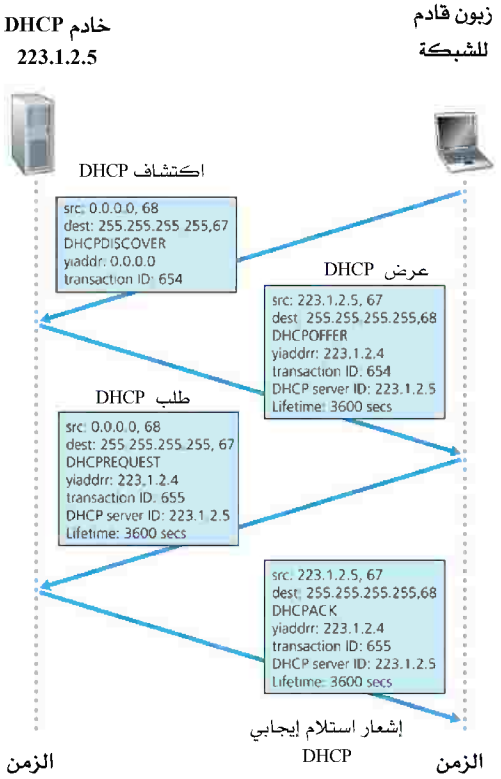
يستخدم بروتوكول DHCP بنية خادم/زبون. عادة ما يحتاج مضيف قادم للشبكة للحصول على معلومات تهيئة بما في ذلك عنوان IP له. في الحالة الأبسط يوجد في كل شبكة (بطريقة العنونة الموضحة في الشكل 4-17) خادم DHCP. أما إذا لم يوجد خادم DHCP فسنحتاج إلى وجود وكيل ترحيل (relay agent) يعرّف عنوان خادم DHCP. يوضح الشكل 4-20 خادم DHCP متصل بالشبكة الفرعية 223.1.2/24، ويعمل الموجّه وكيل ترحيل للزبائن الجديدة التي تصل للشبكات الفرعية 223.1.1/24 و223.1.3/24. سنفترض في مناقشتنا التالية أن خادم DHCP متوفر على الشبكة الفرعية.



الشكل 4-20 سيناريو التفاعل بين خادم وزبون DHCP.

لمضيف جديد قادم يعتبر بروتوكول DHCP عملية مكونة من أربع خطوات؛ كما يبين الشكل 4-21 لإعدادات الشبكة المعروضة في الشكل 4-20. في هذا الشكل (كما في "عنوان الإنترنت" الخاص بك) يشير الرمز yiaddr إلى العنوان المخصّص للمضيف الجديد القادم. والخطوات الأربعة هي:

- اكتشاف خادم DHCP: إن المهمّة الأولى للمضيف القادم حديثاً للشبكة هي أن يجد خادم DHCP الذي سيتفاعل معه. ويتم ذلك باستعمال رسالة اكتشاف DHCP والتي يرسلها المضيف ضمن رزمة UDP إلى المنفذ رقم 67. تغلّف رزمة UDP في رزمة بيانات IP. لكن إلى من يجب أن ترسل هذه الرزمة؟ إن المضيف لا يعرف على الإطلاق حتى عنوان IP للشبكة التي يتصل بها وبالأحرى لا يعرف عنوان خادم DHCP لهذه الشبكة. ولذا ينشئ زبون DHCP وحدة بيانات IP تحتوي على رسالته لاكتشاف DHCP سويّة مع عنوان IP الإذاعي للوجهة 255.255.255.255 وعنوان IP "لهذا المضيّف" (أي 0.0.0.0) للمصدر. يمرر زبون DHCP وحدة بيانات IP إلى طبقة ربط البيانات والتي تقوم بدورها بإذاعة الإطار الناتج إلى كل العقد المتصلة بالشبكة الفرعية (سنغطّي تفاصيل إذاعة طبقة ربط البيانات في الجزء 5-4).



الشكل 4-21 التفاعل بين خادم وزبون DHCP.

- عروض خادمات DHCP : يرد خادم DHCP الذي استلم رسالة اكتشاف DHCP على الزبون برسالة عرض DHCP تذاع إلى كل العقد على الشبكة الفرعية (مستعملاً العنوان 255.255.255.255 مرة أخرى للوجهة). (قد تحتاج لأن تفكر في سبب ضرورة إذاعة الرد من الخادم!). ونظراً لاحتمال وجود عدّة خادمات DHCP على الشبكة الفرعية قد يجد الزبون نفسه في وضع يحسد عليه حيث يستطيع الاختيار من بين عدّة عروض. تحتوي كل رسالة عرض من الخادم على الرقم التعريفي لرسالة الاكتشاف التي تلقاها، وعنوان بروتوكول الإنترنت المقترح للزبون، وقناع الشبكة، ومدة إيجار عنوان IP (أي المدة التي سيكون العنوان فيها صحيحاً – أي محجوزاً

لاستخدام المضيف ولا يمكن تخصيصه لمضيف آخر). من الشائع أن يضع الخادم مدة الإيجار عدّة ساعات أو أيام [Droms 1999].

- طلب DHCP: بعد أن يختار المضيف الواصل حديثاً للشبكة واحداً من عروض DHCP المقدمة له سيردّ عليها برسالة طلب DHCP ويضع بها نفس قيم بارامترات التهيئة الموجودة في العرض المختار.
- إشعار استلام DHCP: يردّ الخادم على رسالة طلب DHCP برسالة إشعار استلام DHCP مؤكّداً قيم البارامترات المطلوبة.

بمجرد استلام الزبون إشعار استلام DHCP يكون التفاعل بين الزبون والخادم قد اكتمل، ويمكن أن يستعمل الزبون عنوان IP المخصص له من خادم DHCP حتي تنتهي مدة الإيجار. ولأن الزبون قد يرغب في استعمال عنوانه بعد انتهاء مدة الإيجار يوفر DHCP أيضاً آلية تسمح للزبون بتجديد إيجار عنوان IP.

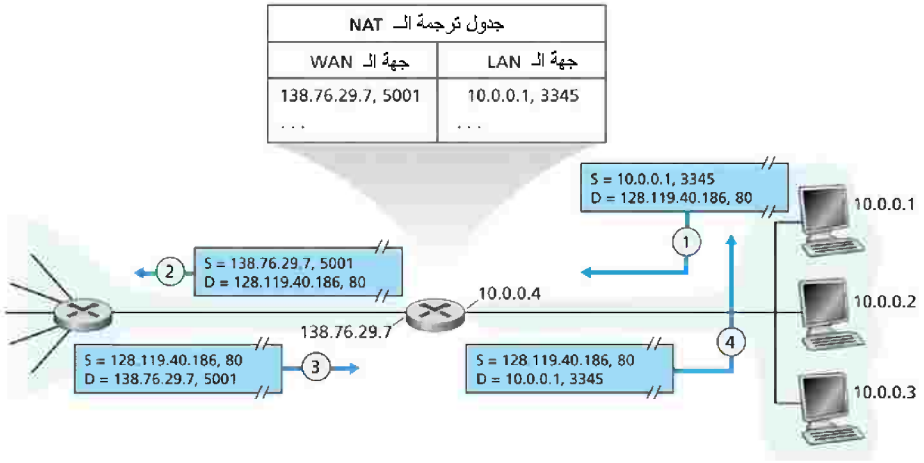
تتضح الفائدة الجليلة لخاصية "وصل وشغل" في بروتوكول DHCP إذا ما أخذنا في الاعتبار أن البديل هو تهيئة عنوان IP للمضيّف يدوياً. مثلاً تصور طالباً ينتقل من قاعة الدروس إلى المكتبة إلى غرفة المسكن مع حاسب نقال وفي كل موقع يتصل بشبكة فرعية جديدة فإنه يحتاج إلى عنوان IP جديد في كل موقع. من المستحيل تصور أن مدير النظام يجب أن يعيد تهيئة الحاسبات النقالة في كل موقع، كما أن الكثير من الطلاب (ما عدا أولئك الذين يأخذون مادة شبكات الحاسب!) ليس لديهم خبرة لتهيئة حاسباتهم النقالة يدوياً. ومع ذلك يعاني بروتوكول DHCP من بعض أوجه القصور من منظور قابلية الحركة. فمثلاً لا يمكن الاحتفاظ بتوصيلة TCP لعقدة تتحرّك بين شبكات فرعية لأنها تحصل على عنوان IP جديد من DHCP في كل مرة توصَّل بشبكة فرعية جديدة. في الفصل السادس سنفحص بروتوكول IP النقّال كامتداد حديث للبنية التحتية لبروتوكول IP يسمح لعقدة متنقلة باستعمال عنوان دائم وحيد بينما تتحرّك بين الشبكات الفرعية. يمكنك الاطلاع على التفاصيل الإضافية حول بروتوكول DHCP في [Droms 1999] و[dhc 2007]. يوجد برنامج مصدر مفتوح (open source code)

لتحقيق بروتوكول DHCP من اتحاد نظم الإنترنت (Internet Systems Consortium) [ISC 2007].

ترجمة عناوين الشبكة (NAT)

بعد أن ناقشنا عناوين الإنترنت وصيغة وحدة بيانات IPv4 ندرك الآن جيداً أن كل جهاز يعمل ببروتوكول IP يحتاج إلى عنوان IP. ومع انتشار شبكات سوهو SOHO (المكتب الصغير والمكتب المنزلي (small office home office) فإن ذلك يشير ضمناً إلى أنه حينما تريد سوهو تركيب شبكة اتصالات محلية لتوصيل عدد من الأجهزة فمن الضّروري أن يخصّص موفّر خدمة الإنترنت مدى من العناوين لتغطية كل أجهزة تلك الشبكة. لو اتسعت تلك الشبكة الفرعية أكثر (على سبيل المثال إذا اشترى الأطفال في المنزل بالإضافة إلى حاسباتهم الخاصة أجهزة PDA وهواتف IP ولعبة Game Boys للشبكة) فإنها ستحتاج إلى تخصيص مجموعة أكبر من العناوين. لكن ماذا لو أن موفر خدمة الإنترنت قد خصّص الأجزاء المتاخمة لعناوين شبكة سوهو الحالية لشبكة أخرى؟ وماذا يحتاج صاحب المنزل لمعرفة كيف يدير عناوين IP في المقام الأول؟ لحسن الحظ هناك طريقة بسيطة لتخصيص العناوين والتي لاقت استعمالاً واسعاً جداً في مثل هذه السيناريوهات يطلق عليها ترجمة عناوين الشبكة (NAT) [RFC 2663; RFC 3022].

يوضح الشكل 4-22 كيفية عمل موجّه مزوّد بـ NAT. توجد واجهة لموجّه الـ NAT القابع في المنزل كجزء من شبكة المنزل على الجانب الأيمن للشكل 4-22. تتم العنونة داخل شبكة المنزل بالضبط كما رأينا من قبل؛ فالواجهات الأربعة في شبكة المنزل لها نفس عنوان الشبكة الفرعية 10.0.0/24. يمثل فضاء العناوين 10.0.0.0/8 أحد ثلاثة أجزاء لعناوين IP المحجوزة في [RFC 1918] للشبكات الخاصّة أو لمنطقة بعناوين خاصّة كشبكة المنزل في الشكل 4-22. والمقصود بـ"منطقة بعناوين خاصّة" هنا شبكة يكون لعناوينها معنى فقط لدى الأجهزة الموجودة ضمن تلك الشبكة. كي نعي أهمية ذلك، تذكّر أن هناك مئات الآلاف



الشكل 4-22 ترجمة عنوان الشبكة.

من الشبكات المنزلية التي يستعمل الكثير منها نفس فضاء العناوين 10.0.0.0/24. يمكن أن ترسل الأجهزة الموجودة بشبكة منزلية لبعضها البعض باستعمال العنونة 10.0.0.0/24، ولكن واضح أنه لا يمكن أن تستعمل الرزم المرسلة خارج نطاق الشبكة المنزلية إلى شبكة الإنترنت العالمية الأكبر هذه العناوين (لا كمصدر ولا كوجهة) نظراً لأن هناك مئات الآلاف من الشبكات تستخدم نفس تلك الكتلة من العناوين. أي أن العناوين 10.0.0.0/24 يمكن أن يكون لها معنى فقط في نطاق الشبكة المنزلية المحددة. لكن إذا كانت العناوين الخاصّة لها معنى فقط ضمن الشبكة المحددة فكيف تعالج العنونة عند إرسال أو استلام رزم من الإنترنت العالمية (حيث يكون من الضروري استعمال عناوين فريدة) ؟ يكمن الجواب في فهم نظام ترجمة عناوين الشبكة NAT.

لا يبدو الموجّه القادر على ترجمة عناوين الشبكة NAT للعالم الخارجي كموجّه، وإنما يتصرّف كجهاز واحد له عنوان IP وحيد. في الشكل 4-22، كل البيانات التي تغادر موجّه الشبكة الأم إلى الإنترنت الأكبر تستخدم العنوان 138.76.29.7 كعنوان المصدر، كما أن البيانات القادمة من الإنترنت إلى الموجّه لها العنوان 138.76.29.7 كعنوان الوجهة. بشكلٍ أساسي يحجب موجّه الـ NAT

تفاصيل الشبكة المنزلية عن العالم الخارجي. (قد تتساءل ومن أين تحصل حاسبات الشبكة المنزلية على عناوينها؟ ومن أين يحصل الموجّه على عنوانه الوحيد؟. في أغلب الأحيان يكون الجواب هو نفسه "عن طريق DHCP"! يحصل الموجّه على عنوانه من خادم DHCP لمزوّد خدمة الإنترنت، ويشغل الموجّه خادم DHCP لتزويد العناوين إلى الحاسبات ضمن فضاء عناوين الشبكة المنزلية التي تقع في نطاق تحكم موجّه DHCP).

إذا كانت كل وحدات البيانات التي تصل إلى موجّه NAT من الشبكة الواسعة النطاق (WAN) لها نفس عنوان IP للوجهة (بالتحديد عنوان واجهة موجّه NAT التي تتصل بالشبكة الواسعة النطاق) فكيف يعرف الموجّه المضيف الداخلي الذي يجب أن يرسل له وحدة البيانات؟ تكمن الحيلة هنا في استعمال جدول ترجمة NAT في الموجّه يتضمن في مدخلاته أرقام المنافذ بالإضافة إلى عناوين IP.

بالنظر إلى المثال الموضح في الشكل 4-22، وبافتراض أن مستخدماً يعمل على الشبكة المنزلية من خلال المضيّف 10.0.0.1 يطلب صفحة ويب من خادم ويب (منفذ 80) بعنوان 128.119.40.186. يخصّص المضيّف 10.0.0.1 رقم منفذ (اعتباطي) للمصدر وليكن 3345، ويرسل وحدة البيانات إلى شبكة الاتصالات المحلية. يستلم موجّه NAT وحدة البيانات، ويولّد رقم منفذ جديد وليكن 5001 لمصدر وحدة البيانات تلك، ثم يستبدل عنوان IP للمصدر بعنوان IP لواجهته المتصلة بالشبكة واسعة النطاق أي 138.76.29.7، ويستبدل رقم منفذ المصدر الأصلي 3345 برقم منفذ المصدر الجديد 5001. عندما يولّد موجّه NAT رقماً جديداً لمنفذ المصدر يمكن أن يختار أي رقم غير موجود حالياً في جدول ترجمة NAT. (لاحظ أن حقل رقم المنفذ مكون من 16 بتاً ولذا يمكن أن يدعم بروتوكول NAT أكثر من 60 ألف توصيلة في نفس الوقت مع عنوان واحد لواجهة الموجّه المتصلة بالشبكة واسعة النطاق!). يضيف NAT الموجود في الموجّه أيضا مُدخلاً إلى جدول ترجمة NAT. لا يدرك خادم الويب أن رزمة البيانات الواصلة والتي تحتوي على طلب HTTP قد عولجت بموجّه NAT، ويردّ بإرسال رزمة بيانات تحتوي عنوان IP لموجّه NAT لعنوان الوجهة ورقم منفذ الوجهة 5001. عندما تصل وحدة البيانات هذه إلى موجّه NAT

يقوم الموجّه بالبحث في جدول ترجمة الـ NAT مستخدماً عنوان IP للوجهة ورقم منفذ الوجهة للحصول على عنوان IP المناسب (10.0.0.1) ورقم منفذ الوجهة (3345) للمتصفّح في الشبكة المنزلية. عندئذ يعيد الموجّه كتابة عنوان الوجهة لرزمة البيانات ورقم منفذ الوجهة، ويرسل رزمة البيانات إلى الشبكة المنزلية.

حظيت NAT بانتشار واسع في السنوات الأخيرة. لكننا يجب أن نذكر بأن العديد من المثاليين في محيط فريق عمل هندسة الإنترنت (IETF) يعترضون على NAT بصوت عالٍ. يعترضون أولاً لأنه من المفروض أن تُستعمل أرقام المنافذ لعنونة العمليات وليس لعنونة المضيفات (هذا الانتهاك يمكن أن يسبّب في الحقيقة مشاكل للخادمات التي تعمل على الشبكة المنزلية لأنه كما رأينا في الفصل الثاني تنتظر عمليات الخادم الطلبات القادمة لأرقام منافذ معروفة ومحددة). ويعترضون ثانياً لأنه من المفروض أن الموجّهات تعالج الرزم حتى الطبقة 3 فقط. والسبب الثالث لاعتراضهم أن بروتوكول NAT ينتهك ما يسمّى بقضية من طرف إلى طرف؛ أي أن المضيفات يجب أن تتكلم مباشرة مع بعضها البعض بدون تدخّل عقد لتعديل عناوين IP وأرقام المنافذ. ويعترضون رابعاً وأخيراً لأنه يجب استخدام IPv6 (راجع الجزء 4-4-4) للتغلب على مشكلة النقص في عناوين IP، بدلاً من تلك الحلول الترقيعية المؤقّتة للمشكلة كحلول NAT. لكن سواء شئنا أم أبينا أصبح الـ NAT مكوّناً مهماً للإنترنت.

من المشكلات الرئيسة الأخرى التي تواجه NAT تداخلها مع تطبيقات النظائر، كمشاركة النظائر للملفات وتطبيقات النظائر لنقل الصوت عبر الإنترنت. تذكّر من الفصل الثاني أنه في تطبيقات النظائر يستطيع أي نظير مشارك A أن يبدأ توصيلة TCP مع أي نظير مشارك آخر B. تكمن المشكلة في أنه لو كان النظير B وراء الـ NAT فإنه لا يستطيع العمل كخادم وبالتالي لا يستطيع أن يقبل توصيلات TCP. كما سنرى في مسائل الواجب المنزلي يمكن التخلص من مشكلة NAT هذه إذا لم يكن النظير A وراء الـ NAT. في هذه الحالة يمكن أن يتصل النظير A أولاً بالنظير B عن طريق نظير آخر C ليس وراء الـ NAT ويجري معه النظير B حالياً اتّصال TCP. يمكن أن يسأل النظير A النظير B عن طريق النظير C

أن يبدأ اتّصال TCP خلفي مباشرةً مع النظير A. بمجرد إنشاء الاتصال المباشر بين النظيرين A وB يمكن أن يتبادلا الرسائل أو الملفات. تسمّى هذه العملية "الاتّصال الخلفي" وتستخدم في الواقع من قبل الكثير من تطبيقات النظائر لتجاوز NAT. إذا كان كلٌّ من النظيرين A وB وراء الـ NAT الخاص به تكون هذه الحالة أصعب نوعاً ما لكن يمكن أن تعالج باستعمال تطبيقات الترحيل (relays) كما رأينا مع مرحّلات سكاي (Skye) في الفصل الثاني.

بروتوكول UPnP

يوفر بروتوكول UPnP (Universal Plug and Play) على نحو متزايد عبوراً للـ NAT وذلك بالسماح للمضيف باكتشاف وتهيئة الـ NAT القريب [UPnP Forum 2007]. يتطلّب UPnP أن يكون كلٌّ من المضيف والـ NAT متوافقين مع UPnP. وباستخدام UPnP يمكن أن يطلب تطبيقٌ ما يجري تشغيله على المضيف من الـ NAT الترجمة بين (عنوان IP ورقم المنفذ الخاصين به) و(عنوان IP العام ورقم المنفذ العام) عند توجيه طلب إلى رقم منفذٍ عامٍ ما. إذا قَبِل الـ NAT الطلب فيمكن للعُقد من الخارج أن تبدأ توصيلات TCP مع (عنوان IP العام ورقم المنفذ العام). وعلاوةً على ذلك يُمكّن UPnP التطبيق من معرفة قيمة (عنوان IP العام ورقم المنفذ العام) وبالتالي يمكن أن يعلنه التطبيق إلى العالم الخارجي.

كمثال افترض أن مضيفك وراء الـ NAT ويستعمل UPnP وله عنوان خاص 10.0.0.1 ويشغل تطبيق BitTorrent على منفذ 3345. وافترض أيضاً أن عنوان IP العام للـ NAT هو 138.76.29.7. من الطبيعي أن تطبيق BitTorrent لديك يحتاج أن يكون قادراً على قبول توصيلات من المضيفات الآخرى لكي يمكنه تبادل البيانات معهم. ولذلك يطلب تطبيق BitTorrent في مضيفك من الـ NAT تكوين "فتحة" لترجمة (10.0.0.1، 3345) إلى (138.76.29.7، 5001) (يختار التطبيق رقم المنفذ العام؛ في هذا المثال 5001). يمكن أن يعلن تطبيق BitTorrent في مضيفك أيضاً إلى مقتفيه بأنه موجود في العنوان (138.76.29.7، 5001). بهذا الأسلوب يمكن أن يتصل مضيف خارجي يشغل BitTorrent بالمقتفي ويعرف أن تطبيق BitTorrent

لديك موجود من خلال العنوان (138.76.29.7، 5001). وبالتالي يمكن أن يرسل المضيف الخارجي TCP SYN إلى العنوان (138.76.29.7، 5001). عندما يستلم NAT رزمة SYN سيغيّر عنوان IP ورقم منفذ الوجهة في الرزمة إلى (10.0.0.1، 3345) ثم يرسلها عبر الـ NAT.

الخلاصة هي أن UPnP يسمح للمضيفات الخارجية ببدء الاتصال مع مضيفات وراء الـ NAT باستعمال TCP أو UDP. لقد كان الـ NAT ولفترة طويلة عدواً لتطبيقات P2P؛ وقد وفر UPnP حلاً فعّالاً ومتيناً لاجتياز الـ NAT والذي ربما كان المنقذ لتلك التطبيقات. لقد كانت مناقشتنا هنا للـ NAT وUPnP مختصرة بالضرورة، وللمزيد من التفصيل راجع [Huston and UPnP 2004; Cisco NAT 2007].

3-4-4 بروتوكول رسائل التحكم في الإنترنت (ICMP)

تذكّر أن طبقة شبكة الإنترنت لها ثلاثة مكوّنات رئيسة: بروتوكول IP (نوقش في الجزء السابق)، وبروتوكولات التوجيه (وتضم RIP وOSPF وBGP) (وسوف نغطيها في الجزء 4-6)، وبروتوكول ICMP (وهو موضوع هذا الجزء).

تم وصف بروتوكول ICMP في [RFC 792]، وتستخدمه المضيفات والموجّهات لتبادل معلومات طبقة الشبكة فيما بينها. إن أكثر استعمالات بروتوكول ICMP هو للإبلاغ عن الخطأ. على سبيل المثال ربما صادفت رسالة خطأ مثل "لا يمكن الوصول لشبكة الوجهة" عند تشغيلك Telnet أو FTP أو HTTP. هذه الرسالة أصلها بروتوكول ICMP. في وقت ما قد لا يستطيع موجّه IP إيجاد مسار يصل إلى المضيف المحدّد عند تشغيلك تطبيق Telnet أو FTP أو HTTP. عندئذٍ ينشئ الموجّه رسالة ICMP من النوع 3 تتضمن وصفاً للخطأ الذي حدث ويرسلها إلى مضيفك.

غالباً ما يُعتبر بروتوكول ICMP جزءاً من بروتوكول IP، ولكنه من الناحية المعمارية يقع فوق IP مباشرة حيث إن رسائل ICMP يتم حملها داخل وحدات بيانات IP. بمعنى أن رسائل ICMP تمثل الحمل الآجر ضمن وحدات بيانات IP تماماً كما

تُحمل قطع بيانات TCP أو UDP في وحدات بيانات IP. وبنفس الطريقة عندما يستلم مضيف وحدة بيانات IP تحمل رسالة ICMP فإنه يقوم بانتزاع محتويات الرسالة تماماً كما يفعل مع قطع بيانات TCP أو UDP.

تحتوي رسالة ICMP على حقل يحدد النوع والكود لها، كما تتضمن ترويسة وحدة بيانات IP التي تسببت في توليد رسالة ICMP تلك في المقام الأول وأول ثماني بايتات منها (كي يتمكن المُرسِل من تحديد وحدة البيانات التي سببت الخطأ). يبين الشكل 4-23 بعض أنواع رسائل ICMP. لاحظ أن رسائل ICMP لا يقتصر استخدمها على الإبلاغ عن حالات الأخطاء.

الوصف	الكود	النوع
رد الصدى	0	0
لا يمكن الوصول لشبكة الوجهة	0	3
لا يمكن الوصول لمضيف الوجهة	1	3
لا يمكن الوصول لبروتوكول الوجهة	2	3
لا يمكن الوصول لمنفذ الوجهة	3	3
شبكة الوجهة غير معروفة	6	3
مضيف الوجهة غير معروف	7	3
خنق المصدر (التحكم في الازدحام)	0	4
طلب صدى	0	8
إعلان من موجّه	0	9
اكتشاف موجّه	0	10
انتهاء فترة TTL	0	11
ترويسة وحدة البيانات غير صحيحة	0	12

الشكل 4-23 أنواع رسائل بروتوكول ICMP.

يرسل برنامج البينج (ping) الشهير رسالة ICMP من النوع "8" بالكود "0" إلى المضيف المحدّد. يرد مضيف الوجهة الذي يرى رسالة "طلب الصدى" (echo request) برسالة ICMP "رد الصدى" (echo reply) من النوع "0" بكود "0". تدعم معظم نظم TCP/IP تحقيق خادم البينج كجزء مباشر من نظام التشغيل (أي أن الخادم ليس عملية (process)). يحتوي الفصل الأول من كتاب [Stevens 1990] على النص الأصلي لبرنامج زبون البينج. لاحظ أنه من الضروري أن يكون بوسع برنامج الزبون أن يطلب من نظام التشغيل توليد رسالة ICMP من النوع "8" بكود "0".

من رسائل ICMP الأخرى الشائقة رسالة خنق المصدر (source quench)، رغم أنها نادراً ما تستخدم حالياً. كان الغرض الأساسي من هذه الرسالة السماح لموجّه يعاني من الازدحام بإرسال تلك الرسالة إلى مضيف لإجباره على تخفيض معدل إرساله. لقد رأينا في الفصل الثالث أن بروتوكول TCP لديه آلية للتحكم في الازدحام تعمل في طبقة النقل بدون استعمال رسائل طبقة الشبكة لخنق المصدر.

قدّمنا في الفصل الأول برنامج متتبع المسار (Traceroute) والذي يسمح لنا بتتبّع المسار من مضيف معين إلى أي مضيف آخر في العالم. ومن الجدير بالذكر أن هذا البرنامج أيضاً يستخدم رسائل ICMP. لتقرير أسماء وعناوين الموجّهات بين المصدر والوجهة يقوم برنامج تتبع المسار بإرسال سلسلة من وحدات بيانات IP العادية إلى الوجهة. تحمل كلٌّ من هذه الوحدات قطعة UDP برقم منفذ UDP غير محتمل الوجود. ويكون زمن TTL في أول هذه الوحدات له القيمة 1، وفي الثانية له القيمة 2، وفي الثالثة له القيمة 3، وهكذا. يبدأ المصدر أيضاً موقّتات لكل وحدة من وحدات البيانات. عندما تصل وحدة البيانات n إلى الموجّه n يلاحظ الموجّه n أن مدة TTL لوحدة البيانات قد انتهت. وفقاً لقواعد بروتوكول IP يتخلص الموجّه من وحدة البيانات ويرسل رسالة ICMP تحذيرية إلى المصدر (من النوع 11 بكود 0) تتضمّن اسم الموجّه وعنوان IP له. عندما تصل تلك الرسالة إلى المصدر يحصل على زمن رحلة الذهاب والإياب من الموقّت واسم وعنوان IP للموجّه n من رسالة ICMP.

نبذة عن الأمن (Focus on Security)

تفتيش وحدات البيانات: برامج الحماية وأنظمة اكتشاف الاختراق

لنفترض أنك تضطلع بمهمّة إدارة شبكة في البيت أو القسم أو الجامعة أو الشركة. من السهل على المهاجمين الذين يعرفون حيز العناوين لتلك الشبكة إرسال وحدات بيانات IP إلى أيٍّ من تلك العناوين. يمكن أن تقوم وحدات البيانات تلك بأي نوع من الأشياء المخادعة مثل رسم مخطط لشبكتك عن طريق ما يسمى بمسح البينج (ping sweeps) ومسح المنافذ (port scans)، وتخريب المضيفات الضعيفة برزم مشوّهة، وإغراق الخادمات بفيضان من رسائل ICMP، وإصابة المضيفات بتضمين برمجيات خبيثة (malware) في الرزم.

بصفتك المشرف على الشبكة ما الذي يمكنك فعله إزاء كل أولئك الأشرار القادرين على إرسال رزم خبيثة إلى شبكتك؟ هناك آليتان شائعتان لصد هجمات الرزم الخبيثة: برامج الحماية (firewalls) وأنظمة اكتشاف الاختراق (Intrusion Detection Systems (IDSs)).

قد تحاول أولاً تركيب برامج الحماية بين شبكتك والإنترنت (معظم موجّهات الوصول (access routers) اليوم مزودة ببرامج الحماية). تقوم برامج الحماية بتفتيش ترويسات وحدات وقطع البيانات، وتمنع وحدات البيانات المريبة من الدخول إلى الشبكة الداخلية. على سبيل المثال قد تُعدّ برامج الحماية لمنع كل رزم رسائل ICMP لطلبات الصدى، وبذلك تمنع المهاجمين من عمل مسح بينج لحيز العناوين لشبكتك. يُمكن أيضاً أن تمنع برامج الحماية الرزم بناءً على عناوين بروتوكول الإنترنت للمصدر والوجهة وأرقام المنافذ. كما يُمكن أن تُعدّ برامج الحماية لتعقب توصيلات TCP والسماح فقط بدخول وحدات البيانات التي تنتمي إلى التوصيلات المسموح لها.

مكن توفير حماية إضافية باستخدام نظام IDS، والذي عادةً ما يركّب على تخوم (حدود) الشبكة ليقوم بتفتيش "أعمق" للرزم وذلك بفحص ليس فقط حقول الترويسة ولكن أيضاً بيانات الحمل الآجر في وحدة البيانات (بما في ذلك بيانات طبقة البرامج). يحتوي نظام IDS على قاعدة بيانات لتوقيعات الرزم (packet signature) المعروفة بأنها تشكل جزءاً من هجمات. يتم تحديث قاعدة البيانات تلك آلياً كلما اكتشفت هجمات جديدة. بينما تعبر الرزم نظام IDS يحاول النظام مطابقة حقول الترويسة وحمولات تلك الرزم بالتوقيعات الموجودة في قاعدة البيانات. إذا عثر على تطابق يقوم IDS بإصدار إنذار. وهناك أيضاً أنظمة لمنع الاختراق (IPS) وهي تشبه أنظمة IDS إلا أنها تمنع تلك الرزم من دخول الشبكة بالإضافة إلى إصدار الإنذارات. سنستعرض في الفصل الثامن تفاصيل أكثر حول برامج الحماية ونظم IDS.

هل بإمكان برامج الحماية ونظم IDS حماية شبكتك حمايةً كاملةً من كل الهجمات؟ من الواضح أن الجواب لا، لأن المهاجمين يقومون بشكلٍ مستمر بهجمات جديدة ليست لها أية توقيعات متوفرة في قاعدة البيانات. لكن برامج الحماية ونظم IDS التقليدية المعتمدة على التوقيعات تفيد بلا شك في حماية شبكتك من الهجمات المعروفة.

لكن كيف يعرف برنامج متتبع المسار متى يجب التوقّف عن إرسال قطع UDP؟ تذكّر أن البرنامج يقوم بزيادة مدة TTL لكل وحدة بيانات يرسلها، وبالتالي ستقطع في النهاية إحدى وحدات البيانات الطريق بطوله إلى مضيف الوجهة. ولأن وحدة البيانات تلك تحتوي على قطعة UDP برقم منفذ غير محتمل الوجود فإن مضيف الوجهة سيرسل رسالة ICMP (من النوع 3 بكود 3) إلى المصدر للإبلاغ عن أن منفذ UDP في وحدة البيانات بعيد المنال. عندما يستلم مضيف المصدر رسالة ICMP تلك، يعرف بأنه ليس بحاجة إلى أن يرسل وحدات بيانات اختبار إضافية (في الحقيقة يرسل برنامج متتبع المسار القياسي مجموعات من ثلاث وحدات بيانات لها نفس مدة TTL، وهكذا يتكون مخرج متتبع المسار من ثلاث قيم لكل TTL).

بهذه الطريقة يُحدِّد مضيف المصدر عدد وهويّات الموجّهات التي تقع بينه وبين مضيف الوجهة وكذلك مدة رحلة الذهاب والإياب بينهما. لاحظ أن برنامج زبون متتبع المسارات يجب أن يكون قادراً على الإيعاز إلى نظام التشغيل بتوليد وحدات بيانات UDP بقيم TTL معينة، ويجب أيضاً أن يكون قادراً على تلقي إخطارات من نظام التشغيل عندما تصل رسائل ICMP. الآن وبعد أن عرفت كيف يعمل برنامج متتبع المسارات، قد تريد معاودة تشغيله والتمرن عليه أكثر.

4-4-4 بروتوكول IPv6

في أوائل التسعينيات بدأ فريق عمل هندسة الإنترنت (IETF) محاولة لتطوير بروتوكول يخلف بروتوكول IPv4. كان أحد الحوافز الأساسية لهذا الجهد هو إدراك أن فضاء العناوين المكونة من 32 بتاً يشرف على النفاذ بمعدل سريع، مع توصيل شبكات فرعية وعُقد بروتوكول IP جديدة بالإنترنت (وتخصيص عناوين IP فريدة) بمعدّلات عالية للغاية. لتلبية هذه الحاجة لفضاء كبير لعناوين IP تم تطوير بروتوكول جديد للإنترنت IPv6. وانتهز مصممو IPv6 الفرصة أيضاً لتحسين وتوسيع الإمكانيات الأخرى لبروتوكول IPv4 استناداً على الخبرة التشغيلية المتراكمة مع ذلك البروتوكول.

كان الموضوع الذي دار عليه نقاش كبير هو متى سيتم نفاذ عناوين IPv4 بالكامل (وبالتالي لا يُمكن توصيل شبكات فرعية جديدة بالإنترنت)؟ كانت تخمينات زعيمي مجموعة توقعات عمر العناوين في فريق عمل هندسة الإنترنت هو أنها ستنفذ في عامي 2008 و2018، على التوالي [Solensky 1996]. في عام 1996 أعلن مكتب التسجيل الأمريكي لأعداد الإنترنت (ARIN) أن كل عناوين الفئة A لبروتوكول IPv4 قد خصّصت، وأن 62 بالمائة من عناوين الفئة B قد خصّصت، وأن 37 بالمائة من عناوين الفئة C قد خصّصت [ARIN 1996]. ويمكنك الاطلاع على تقرير حديث عن تخصيص عناوين IPv4 في [Hain 2005]. رغم أن هذه التقديرات والإحصاءات تشير إلى أنه لا يزال هناك متسع من الوقت حتى تُستنزف عناوين IPv4، إلا أنه تم أيضاً إدراك أن وقتاً طويلاً سيكون مطلوباً لانتشار تقنية جديدة على مثل هذا النطاق الواسع؛ ولذا بدأ العمل في بروتوكول الإنترنت القادمة IPng [Bradner 1996; RFC 1752]. كانت نتيجة هذا الجهد وضع مواصفات نسخة بروتوكول الإنترنت 6 (IPv6) [RFC 2460]. غالباً ما يطرح هنا السؤال التالي: "وماذا حدث لبروتوكول IPv5"؟ كان التصوّر الأولي بأن بروتوكول ST-2 سيصبح IPv5، لكن ST-2 أسقط لاحقاً لصالح بروتوكول RSVP والذي سنناقشه في الفصل السابع.

من مصادر المعلومات الممتازة حول IPv6 صفحة الويب الخاصة بالجيل القادم من بروتوكول الإنترنت IPng [Hinden 2007] وكتاب هويتيما حول هذا الموضوع [Huitema 1998].

صيغة وحدة بيانات بروتوكول IPv6

يبين الشكل 4-24 صيغة وحدة بيانات IPv6. ومن أهم التغييرات في بروتوكول IPv6 والتي تتضح من صيغة وحدة البيانات:

- التوسع في العناوين: زاد بروتوكول IPv6 حجم عنوان بروتوكول الإنترنت من 32 بتاً إلى 128 بتاً. وهذا يضمن بأن العالم لن يستنفذ عناوين بروتوكول الإنترنت (فيمكن الآن لكل حبة رمل على كوكب الأرض أن يكون لها

عنوان IP). بالإضافة إلى عناوين الإرسال الفردي (unicast) وعناوين الإرسال الجماعي (multicast)، يوفر IPv6 نوعاً جديداً من العناوين يسمّى عنوان (anycast) والذي يسمح بتوصيل وحدة البيانات إلى أي مضيف ضمن مجموعة من المضيفات. (يمكن استخدام هذه الميزة على سبيل المثال لإرسال تعليمة "GET" في بروتوكول HTTP إلى الخادم الأقرب في عدد من مواقع الويب البديلة التي تحتوي على وثيقةٍ ما).

الشكل 4-24 صيغة وحدة بيانات IPv6.

- انسيابية الترويسة المكونة من 40 بايتاً: كما سنتناول لاحقاً، تم إسقاط عدد من حقول IPv4 أو جعلها اختيارية. من شأن استخدام الترويسة الجديدة بطول ثابت قدره 40 بايتاً تسريع معالجتها. كما يسمح استخدام نظام تكويد جديد للخيارات بمرونة أكثر في معالجة تلك الخيارات.
- وسم التدفق والأولوية: يستخدم IPv6 تعريفاً مبهماً بعض الشيء للتدفق، حيث ذكر في RFC 1752 وRFC 2460 أن هذا يسمح بـ "وسم الرزم التي تنتمي إلى تدفقات معينة والتي يطلب المُرسِل معالجة خاصة لها كنوعية غير معتادة من الخدمة أو خدمة فورية". على سبيل المثال قد يُعامَل إرسال الفيديو والتسجيل الصوتي كتدفق، بينما لا تُعد التطبيقات الأكثر تقليدية – كإرسال الملفات والبريد الإلكتروني – تدفقاً. من الممكن معاملة حركة

مرور البيانات من قِبَل مستخدم له أولوية عالية كتدفق (كما في حالة شخص ما يدفع ثمناً أعلى لخدمة أفضل لحركة مرور بياناته). ومع ذلك فمن الواضح أن مصممي IPv6 يتوقعون ضرورة توفير القدرة على التمييز بين أنواع التدفق المختلفة، حتى وإن كان المعنى الدقيق لـ "تدفق" لم يحدد بعد. يوجد أيضاً في ترويسة IPv6 حقلٌ مكون من 8 بتات لتحديد نوع حركة مرور البيانات. يمكن استخدام هذا الحقل – مثل حقل TOS في IPv4 – لإعطاء أولوية لبعض وحدات البيانات ضمن تدفق معين، أو لإعطاء أولوية لوحدات البيانات من بعض التطبيقات (مثل ICMP) على وحدات البيانات من التطبيقات الأخرى (مثل شبكات الأخبار).

كما لاحظنا سابقاً تكشف المقارنة بين الشكل 4-24 والشكل 4-13 بساطة وانسيابية أكثر لتركيب وحدة بيانات IPv6. تعرّف الحقول التالية في بروتوكول IPv6:

- رقم النسخة (الإصدار): يُميّز هذا الحقل المؤلف من 4 بتات رقم نسخة بروتوكول الإنترنت. وليس مستغرباً أن يحتوي هذا الحقل على القيمة 6 لبروتوكول IPv6. لاحظ أن وضع القيمة 4 في هذا الحقل لا يُكوِّن وحدة بيانات IPv4 صحيحة. (لو حدث ذلك لكانت الحياة أسهل بكثير!، راجع المناقشة الخاصة بالانتقال من IPv4 إلى IPv6 فيما بعد).
- نوع حركة مرور البيانات: يماثل هذا الحقل المكون من 8 بتات – من حيث المبدأ – حقل TOS الموجود في IPv4.
- وسمة التدفق (flow label): كما ناقشنا سابقاً يُستعمَل هذا الحقل المكون من 20 بتاً لتمييز تدفق من وحدات البيانات.
- طول الحمل الآجر: تعامل هذه القيمة المكونة من 16 بتاً كعدد صحيح بدون إشارة، وذلك لتحديد عدد البايتات في وحدة بيانات IPv6 التي تلي الترويسة ثابتة الطول والمكونة من 40 بايتاً.

- الترويسة التالية (next header): يميّز هذا الحقل البروتوكول الذي ستسلم إليه محتويات وحدة البيانات تلك (مثلاً TCP أو UDP). يستخدم هذا الحقل قيماً مماثلة لتلك المستخدمة في IPv4.
- الحد الأعلى لعدد القفزات (hop limit): تخفض محتويات هذا الحقل بمقدار واحد عند كل موجّه يقوم بإرسال وحدة البيانات تلك. فإذا أصبحت قيمته صفراً، سيهمل الموجّه وحدة البيانات ولا يرسلها.
- عناوين المصدر والوجهة: يوجد وصف للصيغ المختلفة لعناوين IPv6 والمكونة من 128 بتاً في RFC 4291.
- البيانات: يمثل هذا الجزء "الحمل الآجر" لوحدة بيانات IPv6. عندما تصل وحدة بيانات إلى وجهتها يتم استخلاص هذا الجزء من وحدة البيانات ونقله إلى البروتوكول المحدد في حقل "الترويسة التالية".

حددت المناقشة السابقة الغرض من الحقول المتضمنة في وحدة بيانات IPv6. بمقارنة صيغة وحدة بيانات IPv6 في الشكل 4-24 مع صيغة وحدة بيانات IPv4 التي رأيناها في الشكل 4-13 سنلاحظ أن عدّة حقول في وحدة بيانات IPv4 لم تعد موجودة في وحدة بيانات IPv6:

- التجزئة وإعادة التجميع لوحدات البيانات: لا يسمح IPv6 بتجزئة وإعادة تجميع وحدات البيانات في الموجّهات المتوسّطة، وإنما يمكن أن تؤدَّى هذه العمليات فقط بواسطة المصدر والوجهة. إذا استلم موجّه وحدة بيانات IPv6 كبيرة جداً لكي ترسل على الوصلة الخارجة، فإن الموجّه ببساطة يسقط وحدة البيانات ويرسل رسالة خطأ ICMP "رزمة كبيرة جداً" إلى المُرسِل. يمكن حينئذ أن يعيد المُرسِل إرسال البيانات مستخدماً حجماً أصغر لوحدة بيانات IP. إن التجزئة وإعادة التجميع عملية مضيِّعة للوقت، لذا ستؤدي إزالة هذه الوظيفة من الموجّهات ووضعها مباشرة في الأنظمة الطرفية إلى تسريع بروتوكول الإنترنت إلى حدٍّ كبير.
- المجموع التدقيقي للترويسة: نظراً لأن بروتوكولات طبقة النقل (مثل TCP وUDP) وطبقة ربط البيانات (مثل الإيثرنت) في رصّة بروتوكولات الإنترنت

تقوم بتدقيق البيانات، شعر مصممو بروتوكول الإنترنت الجديد بأن هذه الوظيفة في طبقة الشبكة زائدة ويمكن إزالتها. مرة أخرى كانت المعالجة السريعة لرزم بروتوكول الإنترنت محط اهتمام المصممين. تذكّر من مناقشتنا لـ IPv4 في الجزء 4-4-1 أنه يلزم حساب المجموع التدقيقي للترويسة بعد كل قفزة نظراً لوجود حقل TTL والذي تتغيّر قيمته مع كل قفزة. كما هو الحال مع التجزئة وإعادة التجميع كانت هذه العملية مكلّفة في بروتوكول IPv4.

- الخيارات: لم يعد حقل الخيارات جزءاً من ترويسة IP المعيارية. ومع ذلك فإنه لم يُلغ تماماً وإنما أصبح حقل الخيارات أحد الترويسات التالية المحتملة والمشار إليها من ترويسة IPv6. أي مثلما يمكن أن تكون ترويسة TCP أو UDP الترويسة التالية ضمن حزمة بروتوكول الإنترنت كذلك يمكن أن يكون حقل الخيارات. يؤدّي إزالة حقل الخيارات إلى جعل ترويسة بروتوكول الإنترنت ثابتة الطول ومؤلفة من 40 بايتاً.

تذكّر من مناقشتنا في الجزء 4-4-3 أن بروتوكول ICMP يُستخدم من قِبَل عُقد بروتوكول الإنترنت للإبلاغ عن حالات الخطأ وتزويد النظام الطرفي بمعلومات محدودة (على سبيل المثال رسالة رد الصدى لرسالة البينج). لقد تم تعريف نسخة جديدة من بروتوكول ICMP لبروتوكول IPv6 في RFC 4443. بالإضافة إلى إعادة تنظيم تعريفات الأنواع والأكواد الموجودة، أضاف ICMPv6 أيضاً أنواعاً وأكواداً جديدة تطلّبتها وظائف IPv6 الجديدة. يشمل ذلك "رزمة كبيرة جداً"، و"خيارات IPv6 غير معروفة". كما يتضمّن ICMPv6 وظائف بروتوكول IGMP والتي سندرسها في الجزء 4-7. في السابق كان IGMP – والذي يستعمل لإدارة انضمام مضيف ومغادرته لمجموعات الإرسال الجماعي – بروتوكولاً منفصلاً عن ICMP في IPv4.

الانتقال من IPv4 إلى IPv6

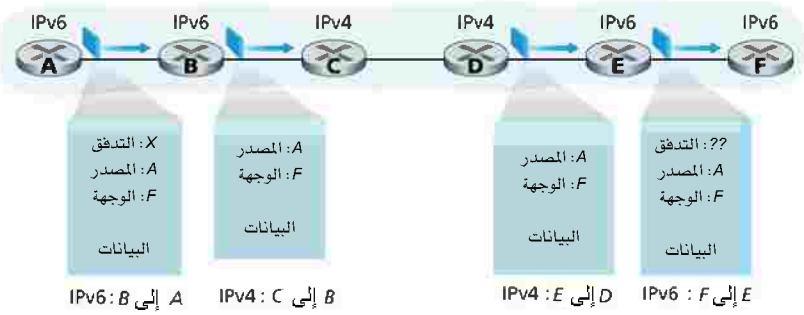
دعنا الآن بعد أن رأينا تفاصيل تقنية IPv6 النظر في مسألة عملية جداً: كيف ستتحول الإنترنت العامّة والتي تعمل طبقاً لبروتوكول IPv4 إلى بروتوكول IPv6؟ تكمن المشكلة في أنه بالرغم من توافق أنظمة IP الجديدة للعمل مع بروتوكول IPv4 (أي يمكنها إرسال وتوجيه واستقبال وحدات بيانات IPv4) إلا أن أنظمة IPv4 المستخدمة حالياً لا يمكنها معالجة وحدات بيانات IPv6. هناك عدّة خيارات ممكنة.

أحد تلك الخيارات هو الإعلان عن موعد محدد يتم فيه توقف تام للإنترنت وترقية كل أجهزتها من IPv4 إلى IPv6. ولقد كان آخر تحول هام في تقنية الإنترنت ما حدث منذ ما يقرب من 25 سنة تقريباً للإنتقال من بروتوكول NCP إلى بروتوكول TCP لخدمة النقل الموثوق فيه للبيانات. وحتى في ذلك الوقت [RFC 801] عندما كانت الإنترنت صغيرة جداً وتدار بواسطة عدد صغير من البرامج المساعدة (wizards) كان تحديد يومٍ بعينه لحدوث تحول في التقنية غير ممكن. هذا الأمر مستحيل بدرجة أكبر اليوم لأنه يتضمّن مئات الملايين من الأجهزة والملايين من مشرفي الشبكات ومستخدميها. يقدم RFC 4213 وصفاً لطريقتين (يمكن اتباعهما منفردتين أو معاً) للإحلال التدريجي لمضيفات وموجّهات IPv6 ضمن عالم IPv4 (بالطبع مع الهدف بعيد المدى لتحويل كل عقد IPv4 في النهاية إلى IPv6).

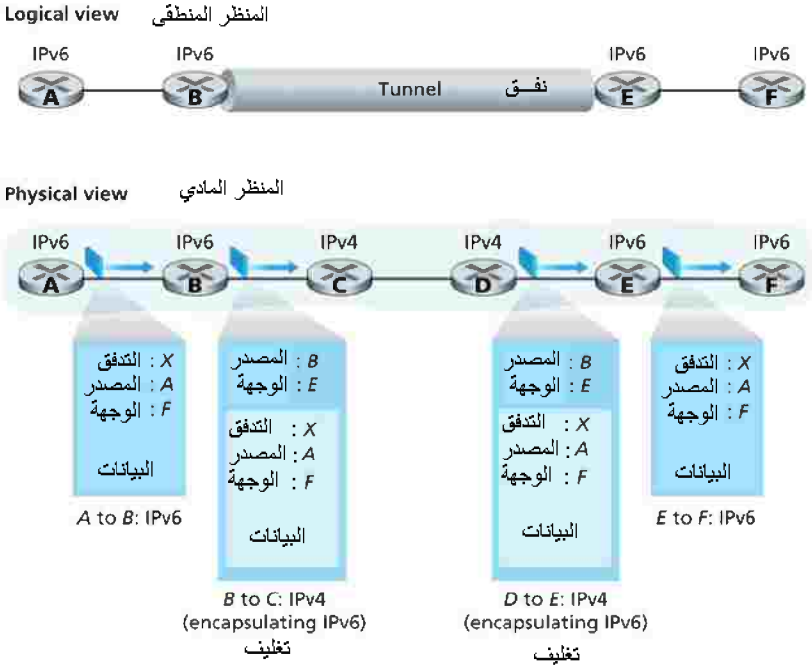
ربما تكون الطريقة الأبسط هي تقديم عقد IP برصّة بروتوكولات مزدوجة، حيث تتضمن عقد IPv6 تحقيقاً لبروتوكول IPv4 أيضاً. تدعى مثل هذه العقدة عقدة IPv6/IPv4 في RFC 4213، ولها القدرة على إرسال واستلام وحدات بيانات لكلٍّ من IPv4 وIPv6. عند توصيل عقد IPv6/IPv4 مع عقد IPv4 تستخدم وحدات بيانات IPv4، وعند توصيلها مع عقد IPv6 تستخدم وحدات بيانات IPv6. يجب أن يكون لعقد IPv6/IPv4 عناوين IPv6 وIPv4. وعلاوةً على ذلك يجب أن تكون قادرة على تحديد ما إذا كانت عقدة أخرى قادرة على التعامل بكلٍّ من IPv6 وIPv4 أو بـ IPv4 فقط. هذه المشكلة يمكن أن تحل عن طريق DNS (راجع الفصل الثاني)،

والذي يمكن أن يُرجع عنوان IPv6 إذا كانت العقدة المعطى اسمها قادرة على التعامل ببروتوكول IPv6 أو يُرجِع عنوان IPv4 فيما عدا ذلك. بالطبع يُرجع DNS عنوان IPv4 فقط إذا كانت العقدة التي تصدر طلب DNS قادرة فقط على استعمال IPv4.

في طريقة رصّة البروتوكولات المزدوجة، إذا كان المُرسِل أو المُستقبِل قادراً على التعامل ببروتوكول IPv4 فقط فإنه يجب استخدام وحدات بيانات IPv4. ونتيجة لذلك قد يصل الأمر في النهاية إلى أن عقد IPv6 ترسل وحدات بيانات IPv4 إلى بعضها البعض. هذه الحالة موضحة في الشكل 4-25. افترض أن العقدة A قادرة على التعامل ببروتوكول IPv6 وتريد إرسال وحدة بيانات إلى العقدة F والتي تعمل أيضاً ببروتوكول IPv6. يمكن أن تتبادل العقد A وB وحدات بيانات IPv6. ومع ذلك يجب أن تكوّن العقدة B وحدة بيانات IPv4 للإرسال إلى C. بالتأكيد يمكن أن ينسخ حقل البيانات من وحدة بيانات IPv6 إلى حقل بيانات وحدة بيانات IPv4 مع عمل تحويل للعناوين. ومع ذلك فعند التحويل من IPv6 إلى IPv4 سيكون هناك حقول معينة في وحدة بيانات IPv6 (مثل حقل معرّف التدفق) ليس لها نظير في IPv4 وبالتالي ستفقد المعلومات الموجودة في تلك الحقول. وهكذا بالرغم من أن E وF يمكن أن يتبادلا وحدات بيانات IPv6 إلا أن وحدات بيانات IPv4 الواصلة لـ E من D لا تحتوي على كل الحقول التي كانت في وحدة بيانات IPv6 الأصلية التي أرسلت من A.



الشكل 4-25 أسلوب رصة البروتوكولات المزدوجة.



الشكل 4-26 استخدام الأنفاق.

هناك طريقة بديلة لطريقة رصّة البروتوكولات المزدوجة تم مناقشتها أيضاً في RFC 4213، وتعرف باستخدام الأنفاق (tunnels). يمكن أن يحل استخدام الأنفاق المشكلة المذكورة سابقاً وذلك بالسماح لـ E على سبيل المثال باستلام وحدة بيانات IPv6 مرسلة من قِبَل A. الفكرة الأساسية وراء استخدام الأنفاق هي كالتالي: افترض أن عُقدتي IPv6 (مثلاً B وE في الشكل 4-25) تريدان استخدام وحدات بيانات IPv6 لكن الموجهات الموصلة بينهما تستخدم IPv4. سنشير لمجموعة موجّهات IPv4 الموصلة بين اثنين من موجّهات IPv6 كـ "نفق" كما هو موضح في الشكل 4-26. باستخدام الأنفاق تأخذ عقدة IPv6 على جانب الإرسال للنفق (على سبيل المثال B) وحدة بيانات IPv6 كاملة وتضعها في حقل البيانات (الحمل الآجر) لوحدة بيانات IPv4، ثم تعنون وحدة بيانات IPv4 هذه إلى عقدة IPv6 على جانب الاستلام للنفق (على سبيل المثال E)، وترسل إلى العقدة الأولى في النفق (على سبيل المثال C). تقوم موجّهات IPv4 الفاصلة على طول النفق بتوجيه وحدة بيانات IPv4 هذه بينها

تماماً مثلما تفعل مع أي وحدة بيانات أخرى دون أن تدرك أن وحدة بيانات IPv4 تلك تحتوي على وحدة بيانات IPv6. في النهاية تستلم عقدة IPv6 على جانب الاستقبال للنفق وحدة بيانات IPv4 (حيث إنها تمثل وجهة وحدة بيانات IPv4)، وتدرك أن وحدة بيانات IPv4 تحتوي على وحدة بيانات IPv6، فتنتزع وحدة بيانات IPv6 وتقوم بتوجيه وحدة بيانات IPv6 تماماً كما تفعل عندما تستلم وحدة بيانات IPv6 من أحد جيرانها الذين يتعاملون ببروتوكول IPv6 مباشرة.

ننهي هذا الجزء بملاحظة أنه بينما كان الشروع في تبني IPv6 بطيئاً في بداية الأمر [Lawton 2001] إلا أنه قد أخذ في التسارع مؤخراً. وقد طلب المكتب الأمريكي للإدارة والميزانية (OMB) التحول إلى IPv6 بحلول شهر يونيو/حزيران 2008. يعطي انتشار الأجهزة كهواتف الإنترنت والأجهزة النقّالة الأخرى دفعاً إضافياً لانتشار أوسع لـ IPv6. ولقد حدد برنامج شراكة جيل أوروبا الثالث IPv6 كأسلوب معياري للعنونة للوسائط المتعددة النقالة [3GPP 2007]. حتى مع عدم انتشار IPv6 على نحو واسع في السنوات العشر الأولى من حياته القصيرة إلا أن الدعوة قوية الآن لاعتماده على المدى البعيد. إن نظام أرقام الهواتف المستعمل اليوم قد أخذ عدّة عقود للتطبيق، ولكنه مطبّق الآن لما يقرب من نصف قرن وبدون ما ينم عن الحاجة لتغييره. بنفس الطريقة قد يستغرق IPv6 بعض الوقت للسيطرة، ولكنه أيضاً قد يبقى لمدة طويلة فيما بعد. يقول براين كاربينتر ـ الرئيس السابق لمجلس بنية الإنترنت [IAB 2007] ومؤلف عدّة RFCs متعلّقة بـ IPv6 – "نظرت دائماً إلى هذا الأمر على أنه عملية مدتها 15 سنة تبدأ من عام 1995" [Lawton 2001]. وعلى حسب كلام كاربينتر نكون قد اقتربنا من ثلاثة أرباع المدة!

من الدروس الهامة التي يمكن أن نتعلّمها من تجربة IPv6 أنه من الصّعب جداً تغيير بروتوكولات طبقة الشبكة. منذ أوائل التسعينيات أُعلِنت العديد من البروتوكولات الجديدة لطبقة الشبكة على أنها ستحدث انقلاباً كبيراً في الإنترنت لكن أغلب هذه البروتوكولات لاقت قبولاً محدوداً حتى الآن. من بين هذه البروتوكولات IPv6، وبروتوكولات الإرسال المتعدد (الجزء 4-7)، وبروتوكولات حجز الموارد (الفصل السابع). في الحقيقة يشبه تغيير بروتوكولات طبقة الشبكة

استبدال أساسات البيت (فمن الصعب إجراء ذلك بدون هدم للبيت بالكامل أو على الأقل نقل سكّان البيت بشكلٍ مؤقت). على الجانب الآخر شهدت الإنترنت استخدامات سريعة لبروتوكولات جديدة في طبقة التطبيقات. من الأمثلة التقليدية لذلك بالطبع الويب والرسائل الفورية ومشاركة النظائر للملفات. تتضمّن الأمثلة الأخرى التسجيل الصوتي وعرض الفيديو والألعاب الموزعة. يشبه تقديم بروتوكولات جديدة في طبقة التطبيقات إضافة طبقة جديدة من الطلاء إلى البيت والتي من السّهل نسبياً عملها (كما أنك لو اخترت لوناً جذاباً قد يقلدك آخرون في الحي). الخلاصة أنه يمكن أن نرى في المستقبل تغييرات في طبقة شبكة الإنترنت؛ لكن من المحتمل أن هذه التغييرات ستحدث على فترة زمنية أبطأ بكثير من التغييرات التي تحدث في طبقة التطبيقات.

5-4-4 رحلة قصيرة مع أمن بروتوكول IP

غطى الجزء 4-4-3 بعض تفاصيل بروتوكول IPv4 بما في ذلك الخدمات التي يوفرها وكيفية تحقيقها. وخلال قراءتك لذلك الجزء ربما لاحظت أنه لم يرد ذِكر أي خدمات للأمن (security). في الحقيقة صُمّم IPv4 في عصر (السبعينيات) عندما كان استعمال الإنترنت محصوراً بين باحثي الشبكات الموثوق فيهم والمؤتمنين فيما بينهم. كانت عملية بناء شبكة حاسب تتضمن العديد من تقنيات طبقة ربط البيانات تمثل تحدياً كافياً دون الحاجة للقلق حول هواجس الأمن.

لكن اليوم أصبح الأمن من القضايا الرئيسة، وانتقل باحثو الإنترنت لتصميم بروتوكولات جديدة لطبقة الشبكة توفر تشكيلة من خدمات الأمن. أحد هذه البروتوكولات الشائعة هو IPSec، والذي انتشر أيضاً على نحو واسع في الشبكات الخاصة الافتراضية (Virtual Private Networks (VPNs)). وبالرغم من أن تغطية تفاصيل IPSec وعمليات التشفير التي يعتمد عليها ستأتي في الفصل الثامن إلا أننا سنعطي هنا مقدمة مختصرة عن خدمات IPSec.

لقد تم تصميم IPSec ليكون متوافقاً مع IPv4 وIPv6. وبالتحديد لكي نجني فوائد IPSec لسنا بحاجة إلى أن نستبدل رصّة البروتوكولات في كل الموجّهات

والمضيفات في الإنترنت. على سبيل المثال عند استعمال نمط النقل (transport mode) (أحد نمطين يوفرهما IPSec)، إذا أراد مضيفان الاتصال بشكلٍ آمن فمن الضروري أن يكون IPSec متوفراً فقط في هذين المضيفين. أما كل الموجّهات والمضيفات الأخرى فيمكن أن تستمر في استعمال IPv4 العادي.

وللدقة سنركّز هنا على نمط النقل لبروتوكول IPSec. في هذا النمط يؤسّس مضيفان جلسة IPSec أولاً فيما بينهما (ومن ثم فإن IPSec بروتوكول توصيلي). وأثناء تلك الجلسة تتمتع كل قطع بيانات TCP وUDP المرسلة بين المضيفين بخدمات الأمن المتوفرة من قِبَل IPSec. على جانب الإرسال تمرّر طبقة النقل القطعة إلى IPSec. يقوم IPSec بتشفير القطعة وإلحاق حقول الأمن الإضافية بها، ثم تغلّيف الحمل الآجر الناتج في وحدة بيانات IP عادية. (وفي الحقيقة فإن الأمر أكثر تعقيداً بعض الشيء من هذا كما سنرى في الفصل الثامن). بعد ذلك يرسل مضيف المصدر وحدة البيانات إلى الإنترنت لنقلها إلى مضيف الوجهة. وهناك يقوم IPSec بفك التشفير واسترجاع القطعة الأصلية وتمريرها إلى طبقة النقل.

تشمل الخدمات التي يوفرها IPSec ما يلي:

- الاتفاق على آليات التشفير: تسمح للمضيفين المتصلين بالاتفاق على خوارزميات ومفاتيح التشفير.
- تشفير الحمل الآجر لوحدة بيانات IP: عندما يستلم مضيف الإرسال قطعة من طبقة النقل يقوم IPSec بتشفير الحمل الآجر، ولا يمكن أن يُفك التشفير إلا من قِبَل IPSec في مضيف الاستقبال.
- سلامة البيانات (data integrity): يسمح بروتوكول IPSec لمضيف الاستقبال بالتحقق من عدم تعديل حقول الترويسة والحمل الآجر المشفر في وحدة البيانات أثناء رحلتها من المصدر إلى الوجهة.
- توثيق المصدر (origin authentication): عندما يستلم مضيف وحدة بيانات IPSec من مصدر موثوق به (لديه مفتاح تشفير موثوق فيه كما سنرى في الفصل الثامن) يطمئن المضيف بأن عنوان بروتوكول الإنترنت للمصدر في وحدة البيانات هو المصدر الفعلي لوحدة البيانات.

عندما يؤسّس مضيفان جلسة IPSec بينهما يتم تشفير وتوثيق جميع قطع بيانات TCP وUDP المرسلة بينهما. يزوّد IPSec تغطية عامّة لتأمين جميع الاتصالات بين المضيفين لكل تطبيقات الشبكة.

يمكن أن تستخدم شركةٌ ما بروتوكول IPSec للاتصال بشكلٍ آمن بالإنترنت العامّة غير الآمنة. ولتوضيح ذلك دعنا ننظر إلى مثال بسيط هنا. تصوّر شركة لديها عدد كبير من مندوبي المبيعات الجوّالين ومع كل واحد منهم حاسب نقال من الشركة. افترض أن مندوبي المبيعات يحتاجون للرجوع إلى معلومات حسّاسة جداً عن الشركة (كمعلومات عن المنتجات والأسعار) والمخزنة على خادم في مقر الشركة. افترض كذلك أن مندوبي المبيعات يحتاجون أيضاً لإرسال وثائق حسّاسة إلى بعضهم البعض. كيف يتم ذلك من خلال IPSec؟ والجواب أننا نركّب IPSec في الخادم وفي جميع حاسبات مندوبي المبيعات النقالة. وبذلك حينما يحتاج مندوب مبيعات للاتصال بالخادم أو مع مندوب آخر يتم إنشاء جلسة اتصال آمنة بينهما.

5-4 خوارزميات التوجيه (Routing Algorithms)

تعرضنا حتى الآن في هذا الفصل في الغالب لوظيفة التمرير في طبقة الشبكة. وعرفنا أنه عندما تصل رزمة إلى موجّه فإنه يبحث في جدول التمرير لديه ليحدد الوصلة التي ستوجّه إليها الرزمة. وعرفنا أيضاً أن خوارزميات التوجيه والتي تعمل في موجّهات الشبكة تتبادل وتحسب المعلومات التي تُستخدَم لتهيئة جداول التمرير تلك. وقد سبق توضيح التفاعل بين خوارزميات التوجيه وجداول التمرير في الشكل 2-4. بعد أن تعرفنا على بعض تفاصيل التمرير نحوّل انتباهنا الآن إلى الموضوع الأساسي الآخر في هذا الفصل والذي يمثل الوظيفة الهامة الأخرى لطبقة الشبكة، ألا وهو التوجيه. وسواء كانت طبقة الشبكة توفر خدمة وحدات البيانات (datagrams) (في هذه الحالة قد تأخذ الرزم مسارات مختلفة بين زوج ما من المُرسِل والمستقبل) أو خدمة الدائرة الافتراضية (VC) (في هذه الحالة ستتبع كل الرزم نفس المسار من المصدر إلى الوجهة)، فإنه يجب عليها أن تحدد المسارات التي

ستأخذها الرزم من المُرسِلين إلى المستقبلين. سنرى أن وظيفة التوجيه هي المسؤولة عن تحديد مسارات جيدة من المُرسِلين إلى المستقبلين خلال شبكة الموجّهات.

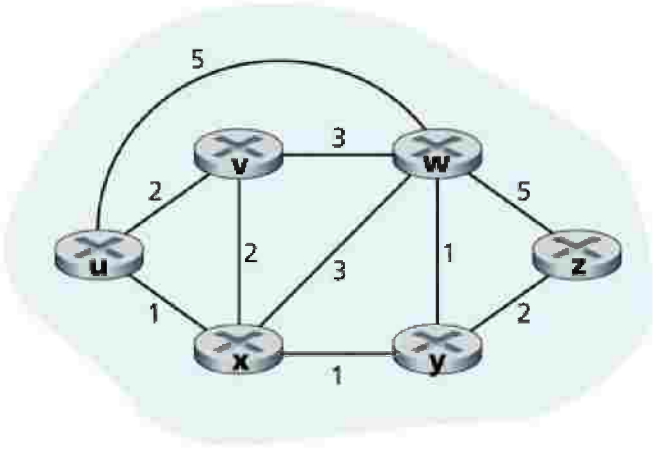
عادةً ما يوصّل كل مضيف مباشرة بموجّه واحد يمثل الموجّه الاعتيادي (default router) للمضيف (أيضاً يسمى موجّه القفزة الأولى للمضيف). وحينما يرسل مضيفٌ ما رزمة تنتقل الرزمة إلى الموجّه الاعتيادي له. نشير إلى الموجّه الاعتيادي لمضيف المصدر بموجّه المصدر والموجّه الاعتيادي لمضيف الوجهة بموجّه الوجهة. وبالتالي تصبح مشكلة توجيه رزمة من مضيف المصدر إلى مضيف الوجهة تماماً هي مشكلة توجيه الرزمة من موجّه المصدر إلى موجّه الوجهة، وهو ما سيمثل بؤرة التركيز لهذا الجزء.

إن الغرض من خوارزمية التوجيه بسيط: هو إيجاد مسار جيد من المصدر إلى الوجهة خلال مجموعة من الموجّهات المتصلة فيما بينها بوصلات. ويعرّف المسار الجيد بأنه أحد المسارات بين المصدر والوجهة والذي له أدنى كلفة. سنرى مع ذلك أنه من الناحية العملية تلعب بعض الاعتبارات الأخرى في الشبكات الحقيقية مثل قضايا سياسة الشبكة (مثلاً القاعدة "الموجّه x ينتمي للمنظمة y، ولذا يجب ألا ترسل إليه أيّة رزم مصدرها شبكة المنظمة z") دوراً أيضاً في تعقيد الخوارزميات البسيطة والرائعة التي تعتمد عليها شبكات اليوم.

يُستخدم رسم بياني (graph) لصياغة مشاكل التوجيه. تذكّر أن الرسم البياني $G = (N, E)$ هو مجموعة من العقد (النقط) N ومجموعة من الحافات (الخطوط) E حيث تمثّل كل حافة بزوج من العقد الموجودة في N. في سياق توجيه طبقة الشبكة تمثّل العقد في الرسم البياني الموجّهات (أي النقط التي يتم عندها اتخاذ قرارات التوجيه) وتمثل الحافات الوصلات المادية بين هذه الموجّهات. يوضح الشكل 4-27 مثالاً لهذا الرسم البياني المجرد لشبكة حاسب. يمكنك الاطلاع على بعض الرسوم البيانية التي تمثّل خرائط شبكات حقيقية في [Dodge 2007; Cheswick 2000]؛ وللاطلاع على مناقشة حول مدى تمثيل النماذج المختلفة المعتمدة

على الرسم البياني لشبكة الإنترنت انظر [Zegura 1997; Faloutsos 1999; Li 2004].

كما يوضّح الشكل 4-27، تقترن بكل حافة أيضاً قيمة تمثّل كلفتها، والتي قد تعكس الطول الطبيعي للوصلة المناظرة (على سبيل المثال قد يكون لوصلة عبر المحيطات كلفة أعلى من وصلة أرضية قصيرة المدى)، أو سرعة الوصلة، أو الكلفة النقدية للوصلة. لغرض التوجيه سنعتبر ببساطة أن لكل حافة كلفة دون الاكتراث بماهية تلك الكلفة. سنشير إلى كلفة أي حافة (x, y) في المجموعة E بالرمز $c(x, y)$ وهي تمثل كلفة الحافة بين العقدتين x وy. وإذا كانت الحافة (x, y) لا تنتمي إلى E تكون $c(x, y) = \infty$. أيضاً سنعتبر فقط الرسوم البيانية غير المتجهة (undirected graphs) (أي الرسوم البيانية ذات الحافات غير المتجهة)، وبالتالي تكون الحافة (x, y) هي نفسها تماماً الحافة (y, x)؛ وكذلك $c(x, y) = c(y, x)$. أيضاً يقال: إن العقدة y جار للعقدة x إذا كانت الحافة (x, y) تنتمي إلى E.



الشكل 4-27 نموذج رسم بياني تجريدي لشبكة حاسب.

بهذه الكلفة للحافات المختلفة في الرسم البياني التجريدي للشبكة يكون الهدف الطبيعي لخوارزمية التوجيه هو ايجاد المسارات الأقل كلفة بين المصادر والوجهات. ولجعل هذه المشكلة أكثر دقة تذكر أن المسار في الرسم البياني $G = (N, E)$ هو سلسلة من العقد $(x_1، x_2، ...، x_p)$ بحيث كل زوج من الأزواج $\{(x_1، x_2)، (x_2، x_3)، ...، (x_{p-1}، x_p)\}$ يمثل حافة ضمن E. تساوي كلفة المسار $(x_1، x_2، ...، x_p)$ مجموع كلف جميع الحافات على طول المسار أي

$$c(x_1, x_2) + c(x_2, x_3) + \ldots + c(x_{p-1}, x_p)$$

عادةً ما توجد عدة مسارات بين كل عقدتين x وy في الرسم البياني ولكلٍّ منها كلفة، ولواحد أو أكثر منها الكلفة الأدنى. يمكن الآن صياغة مسألة الكلفة الأدنى كالآتي: أوجد مساراً له أقل كلفة بين المصدر والوجهة. على سبيل المثال في الشكل 4-27 المسار أقل كلفة بين عقدة المصدر u وعقدة الوجهة w هو $(u، x، y، w)$ وله كلفة مقدارها 3. لاحظ أنه إذا كانت كل الحافات في الرسم البياني لها نفس الكلفة فسيكون المسار الأدنى كلفةً هو نفسه أقصر مسار (أي المسار الذي يتكون من أقل عدد من الوصلات بين المصدر والوجهة).

كتمرين بسيط حاول إيجاد مسار أقل كلفة من العقدة u إلى z في الشكل 4-27 وتأمل لبرهة كيف حسبت ذلك المسار. إذا كنت مثل الكثير من الناس ستحدد المسار المطلوب من u إلى z بفحص الشكل 4-27، مقتفياً بضعة مسارات من u إلى z وبطريقة ما ستقنع نفسك أن المسار الذي اخترته هو الأقل كلفةً بين جميع المسارات المحتملة. هل فحصت كل المسارات المحتملة بين u وz (وعددها 17 مساراً في هذا المثال)؟ من المحتمل لا!. تُعتبر هذه العملية مثالاً لخوارزمية توجيه مركزية (centralized routing algorithm)، حيث يتم تشغيلها في موقع واحد (دماغك في هذا المثال) بمعلومات كاملة عن الشبكة. وبشكلٍ عام من الطرق التي يمكن بها تصنيف خوارزميات التوجيه هو كونها إما عالمية أو لامركزية:

- خوارزمية توجيه عالمية: تحسب المسار الأدنى كلفةً بين المصدر والوجهة بناءً على معرفة كاملة بجميع أجزاء الشبكة. أي أنها تستخدم الوصلات بين كل العقد وكلفة كل وصلة كمُدخلات. وهذا يتطلب في الحقيقة حصول

الخوارزمية على تلك المعلومات بطريقةٍ ما قبل إجراء حساب أفضل مسار. يمكن إجراء الحساب نفسه في موقع واحد (مركز خوارزمية التوجيه العالمية) أو في مواقع متعدّدة مكررة (replicated). ولكن الميزة الهامة لهذا النوع من الخوارزميات هي أنها تمتلك معلومات كاملة حول كل الوصلات وكلفتها. من الناحية العملية غالباً ما يشار إلى هذه الخوارزميات باسم خوارزميات حالة الوصلة ((Link State (LS) حيث يتعين أن تعرف الخوارزمية كلفة كل وصلة في الشبكة. سندرس خوارزميات LS في الجزء 4-5-1.

- خوارزمية توجيه لامركزية: في هذه الحالة يُنفّذ حساب المسار أقل كلفة بأسلوب موزّع تكراري. لا تمتلك أي عقدة المعلومات الكاملة عن كلفة كل الوصلات الموجودة بالشبكة، وإنما تبدأ كل عقدة بمعرفة كلفة الوصلات المتصلة بها مباشرة. ثم من خلال عملية تكرارية من حساب وتبادل المعلومات مع العقد المجاورة (أي العقد التي في النهايات الأخرى للوصلات المتصلة مباشرة بها)، وبشكلٍ تدريجي تحسب العقدة مسار أقل كلفة إلى وجهة ما أو إلى مجموعة من الوجهات. تدعى خوارزمية التوجيه اللامركزية التي سندرسها في الجزء 4-5-2 بخوارزمية متجه المسافة (DV)، لأن كل عقدة تحتفظ بمتجه لتقديرات الكلفة (أي المسافة) إلى كل العقد الأخرى في الشبكة.

كما يُمكن تصنيف خوارزميات التوجيه أيضاً حسب كونها ثابتة أو ديناميكية. في خوارزميات التوجيه الثابتة تتغيّر المسارات ببطء شديد بمرور الوقت، وغالباً كنتيجة للتدخّل البشري (كأن يعدّل شخصٌ ما يدوياً جدول التمرير بالموجّه). تُغيّر خوارزميات التوجيه الديناميكية المسارات مع تغيّر أحمال أو طبوغرافية الشبكة. يمكن أن تنفذ الخوارزمية الديناميكية إمّا بشكلٍ دوري أو كرد فعل مباشر للتغيرات في طبوغرافية الشبكة أو كلفة الوصلات. رغم أن الخوارزميات الديناميكية تمتاز باستجابتها السريعة للتغيرات في الشبكة، إلا أنها أيضاً تُعتبر أكثر عُرضةً للمشاكل مثل حلقات التوجيه (routing loops) (أي المسارات المغلقة) وتذبذب المسارات (route oscillation).

يعتمد أسلوب ثالث لتصنيف خوارزميات التوجيه على كونها تتأثر بأحمال الشبكة أو لا تتأثر. في الخوارزمية التي تتأثر بالأحمال، تتغيّر كلفة الوصلة ديناميكياً لتعكس المستوى الحالي للازدحام في تلك الوصلة. إذا تم تحديد كلفة عالية لوصلة مزدحمة حالياً، فإن خوارزمية التوجيه ستميل إلى اختيار مسارات بعيدة عن تلك الوصلة المزدحمة. وقد كانت خوارزميات التوجيه الأولى في شبكة Arpanet متأثرة بالأحمال [McQuillan 1980]، إلا أنها صادفت عدداً من الصعوبات [Huitema 1998]. خوارزميات التوجيه في الإنترنت اليوم (مثل RIP، OSPF، BGP) لا تتأثر بالأحمال لأن كلفة الوصلات لا تعكس بشكلٍ واضح مستوى الازدحام الحالي (أو في الماضي القريب).

1-5-4 خوارزمية التوجيه من نوع حالة الوصلة (LS)

تذكّر أنه في خوارزمية حالة الوصلة تكون طبوغرافية الشبكة وكلفة كل الوصلات معروفة (أي متوفرة كمدخلات إلى الخوارزمية). عملياً يتم ذلك بجعل كل عقدة تذيع رزماً عن حالة الوصلة إلى كل العقد الأخرى في الشبكة، حيث تحتوي كل رزمة على هويّات وكلف الوصلات الملحقة بها. عملياً (على سبيل المثال في بروتوكول التوجيه في الإنترنت OSPF والذي سنناقشه في الجزء 4-6-1) يتم ذلك في أغلب الأحيان بواسطة خوارزمية إذاعة حالة الوصلة [Perlrnan 1999]. سنغطّي خوارزميات الإذاعة في الجزء 4-7. ونتيجة لذلك ستحصل كل العقد على معلومات مماثلة عن الشبكة بكاملها. وعندئذ يمكن أن تُشغل كل عقدة خوارزمية LS عليها لحساب مجموعة المسارات ذات الكلفة الأدنى إلى العقد الأخرى.

سنقدم فيما يلي خوارزمية توجيه من نوع حالة الوصلة تُعرف بخوارزمية Dijkstra على اسم مخترعها. وهناك خوارزمية أخرى وثيقة الصلة بها يطلق عليها خوارزمية Prim؛ راجع [Cormen 2001] لمناقشة عامّة عن خوارزميات الرسم البياني. تحسب خوارزمية Dijkstra المسارات ذات الكلفة الأدنى من عقدة معينة (عقدة المصدر، وسنشير لها بـ u) إلى كل العقد الأخرى في الشبكة. خوارزمية Dijkstra تكرارية ولها خاصية أنه بعد التكرار k مرة تكون المسارات ذات الكلفة الأدنى

معروفة إلى عدد k من عقد الوجهات، وأنه من بين المسارات ذات الكلفة الأقل لجميع عقد الوجهات تكون تلك المسارات الـ k هي المسارات الـ k الأدنى كلفة.

دعنا نعرّف الرموز التالية:

- $D(v)$: كلفة المسار الأدنى كلفة من عقدة المصدر إلى الوجهة v خلال هذا التكرار للخوارزمية.
- $p(v)$: العقدة السابقة (جار v) على طول المسار الأدنى كلفة من المصدر إلى v.
- N': مجموعة جزئية من العقد؛ حيث تكون v ضمن N' إذا كان المسار الأدنى كلفة من المصدر إلى v معروفاً بشكلٍ حاسم.

تتكون خوارزمية التوجيه العالمية من خطوة تهيئة يتبعها حلقة تكرارية. عدد مرات تكرار الحلقة يساوي عدد العقد في الشبكة. عند الانتهاء تكون الخوارزمية قد حسبت أقصر مسارات من عقدة المصدر u إلى كل عقدة في الشبكة.

خوارزمية حالة الوصلة من عقدة المصدر *u*:

```
1  Initialization:
2     N' = {u}
3     for all nodes v
4        if v is a neighbor of u
5           then D(v) = c(u, v)
6        else  D(v) = ∞
7
8  Loop
9     find w not in N' such that D(w) is a minimum
10    add w to N'
11    update D(v) for each neighbor v of w and not in N':
12          D(v) = min{D(v), D(w) + c(w, v)}
13    /* new cost to v is either old cost to v or known
14          least path cost to w plus cost from w to v */
15 until N' = N
```

وكمثال دعنا نحسب المسارات الأدنى كلفة من u إلى جميع الوجهات المحتملة ضمن الشبكة المبينة في الشكل 4-27. يبين الجدول 4-3 ملخصاً بحسابات الخوارزمية بحيث يعطي كل سطر في الجدول قيم متغيرات الخوارزمية في نهاية كل تكرار. دعنا ننظر أولاً للخطوات القليلة الأولى بالتفصيل:

step	*N'*	$D(v), p(v)$	$D(w), p(w)$	$D(x), p(x)$	$D(y), p(y)$	$D(z), p(z)$
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	$uxyv$		3,y			4,y
4	$uxyvw$					4,y
5	$uxyvwz$					

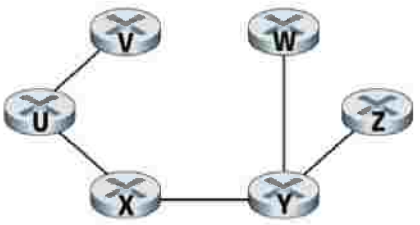
الجدول 4-3 تنفيذ خوارزمية حالة الوصلة على الشبكة الموجودة في الشكل 4-27.

- في خطوة التهيئة تحدد المسارات الأدنى كلفةً والمعروفة حالياً من عقدة المصدر إلى جيرانها الملحقين مباشرة v، x، w بالقيم 2، 1، 5 على التوالي. لاحظ بشكلٍ خاص أن الكلفة إلى w تأخذ القيمة 5 لأن هذه الكلفة هي الكلفة المباشرة (قفزة واحدة) للوصلة من u إلى w (رغم أننا سنرى قريباً أنه يوجد في الحقيقة مسار بكلفة أقل). أما الكلف إلى y وz فتأخذ القيمة لانهاية لعدم وجود وصلة مباشرة بين كلٍّ منهما والمصدر.
- في التكرار الأول نبحث بين العقد التي لم تضاف بعد إلى المجموعة N' عن العقدة التي لها أدنى كلفة من المصدر في نهاية التكرار السابق. تلك العقدة هي x وبكلفة تساوي 1، وهكذا تضاف x إلى المجموعة N'. يُنفّذ السطر 12 من خوارزمية LS حينئذ لتحديث قيمة $D(v)$ لكل العقد v فنحصل على النتائج المبينة في السطر الثاني (الخطوة 1) في الجدول 4-3. لا تتغيّر كلفة المسار إلى v. أما كلفة المسار إلى w (والتي كانت 5 في نهاية التهيئة) عبر العقدة x فلها الآن كلفة تساوي 4. لذا نختار هذا المسار الأدنى كلفة ونعدل

العقدة السابقة لـ w على طول المسار الأقصر من u لتصبح x. بنفس الطريقة تكون الكلفة إلى y (عبر x) تساوي 2، ويعدّل الجدول وفقاً لذلك.

- في التكرار الثاني نجد أن العقد v وy لها أدنى كلفة (تساوي 2)، لذا سنختار أحدهما بشكلٍ اعتباطي وليكن y ونضيفها إلى المجموعة N' والتي ستتضمن الآن u، x، y. ثم نعدل الكلفة إلى العقد الأخرى غير الموجودة في N' (أي العقد v، w، z) عن طريق السطر 12 في خوارزمية LS فنحصل على الناتج الموضح في السطر الثالث في الجدول 4-3.
- وهكذا .....

عندما تنتهي خوارزمية LS نكون قد حصلنا لكل عقدة على سلفها (predecessor) على طول مسار أدنى كلفة من عقدة المصدر إلى تلك العقدة، ولكل سلف لدينا سلفه أيضاً وهكذا يمكننا أن نبني كامل المسار من المصدر إلى كل الوجهات. ومن ثم يمكن أن نبني جدول التمرير في كل عقدة (مثلاً العقدة u) وذلك بتخزين لكل وجهة عقدة القفزة التالية على مسار أدنى كلفة من u إلى الوجهة. يبين الشكل 4-28 مسارات أدنى كلفة وجدول التمرير في العقدة u للشبكة الموجودة في الشكل 4-27.



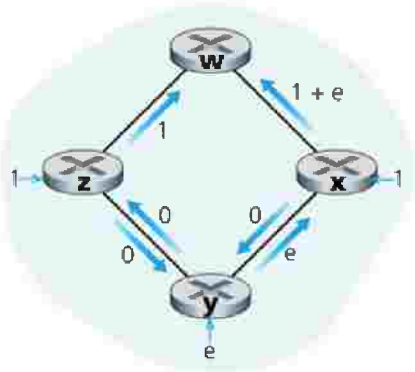
واجهة	وصلة
v	(u, v)
w	(u, x)
x	(u, x)
y	(u, x)
z	(u, x)

الشكل 4-28 المسارات الأدنى كلفة وجدول التمرير على العقدة u.

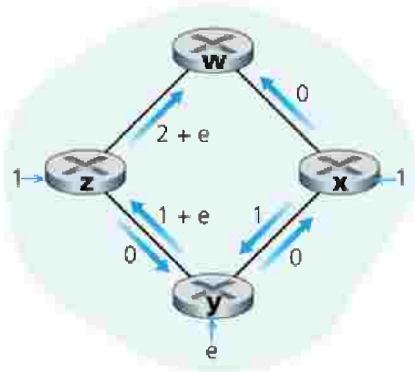
ما حجم التعقيد الحسابي (computational complexity) لهذه الخوارزمية؟ أي ما كمية الحسابات التي يجب إنجازها في أسوأ الأحوال لإيجاد المسارات الأدنى كلفة من المصدر إلى وجهات عددها n عقدة؟ في التكرار الأول نحتاج للبحث خلال كل العقد n لتقرير العقدة w غير الموجودة في N' ولها أدنى كلفة. في التكرار الثاني نحتاج للبحث في n -1 عقدة لتقرير الكلفة الدنيا، وفي التكرار الثالث n - 2 عقدة، وهكذا. وبالتالي يكون العدد الكلي للعقد التي نحتاج البحث خلالها في كل التكرارات يساوي $n(n + 1)/2$، وهكذا نقول بأن التطبيق السابق لخوارزمية LS له تعقيد في أسوأ الأحوال $O(n^2)$. وهناك تطبيق أكثر تطوّراً لهذه الخوارزمية باستعمال هياكل بيانات تُعرف بالكومة (heap)، حيث يمكن إيجاد الحد الأدنى في السطر 9 في زمن لوغاريتمي بدلاً من زمن خطي مما يقلل التعقيد.

قبل أن نغادر خوارزمية LS، دعنا نناقش بعض المشاكل التي يمكن أن تظهر. يبين الشكل 4-29 طبوغرافية شبكة بسيطة وفيها كلفة الوصلة تساوي الحمل الفعلي للوصلة، ومن ثم يعكس على سبيل المثال التأخير الذي سيواجَه. في هذا المثال كُلَف الوصلات ليست متماثلة؛ أي أن $c(u, v)$ تساوي $c(v, u)$ فقط إذا كان الحمل الفعلي في كلا الاتجاهين على الوصلة (u, v) متساوياً. في هذا المثال افترض أن العقدة z تُرسل وحدة بيانات إلى w، والعقدة x أيضا تُنشئ وحدة بيانات إلى w، أما العقدة y فترسل كمية بيانات مقدرها e إلى w أيضاً. يبين الشكل 4-29 (a) التوجيه الأولي حيث تناظر كلفة الوصلة كمية البيانات التي تنقل خلالها.

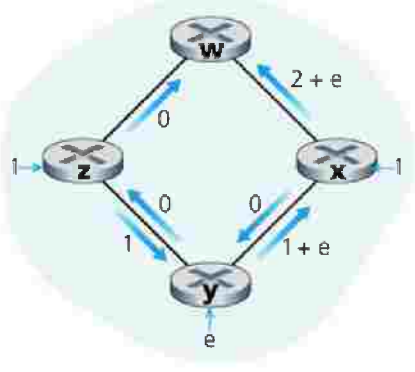
عند تشغيل خوارزمية LS في المرة التالية تقرر العقدة y (بناءً على كلفة الوصلة في الشكل 4-29 (a)) أن المسار في اتجاه عقارب الساعة إلى w له كلفة 1، بينما المسار في عكس عقارب الساعة (والذي تستخدمه حالياً) له كلفة تساوي e+1. لذلك تُعدِّل الآن مسارها إلى w باتجاه عقارب الساعة. وبنفس الطريقة تقرّر x أن المسار الجديد أقل كلفة إلى w في اتجاه عقارب الساعة أيضاً، وبالتالي نحصل على الكُلَف في الشكل 4-29 (b). عند تشغيل خوارزمية LS في المرة التالية تكتشف العقد x، y، z وجود مسار بكلفة صفر إلى w في اتجاه عكس عقارب الساعة، وبالتالي تُغيِّر مساراتها في اتجاه عكس عقارب الساعة. في المرة التالية لتشغيل خوارزمية LS توجه x، y، z مساراتها باتجاه عقارب الساعة، وهكذا.



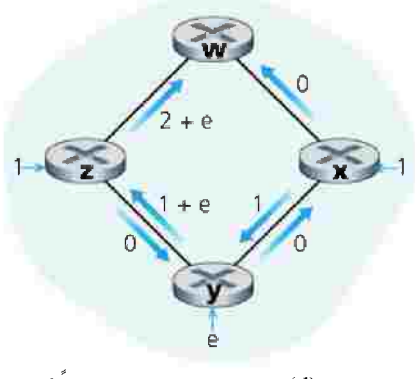
(a) التوجيه الأولي



(b) تكتشف x، y مساراً أفضل إلى w في اتجاه عقارب الساعة



(c) تكتشف x، y، z مساراً أفضل إلى w في اتجاه عكس عقارب الساعة



(d) تكتشف x، y، z مساراً أفضل إلى w في اتجاه عقارب الساعة

الشكل 4-29 تذبذبات في التوجيه الحساس للازدحام.

ماذا يُمكن فعله لمنع مثل تلك التذبذبات (والتي يمكن أن تحدث في أي خوارزمية تستخدم معياراً يعتمد على الازدحام أو التأخير على الوصلة كخوارزمية LS)؟ يتطلب أحد الحلول ألا تعتمد كلف الوصلات على كمية المرور التي تحملها الوصلة فعلاً، وهو حل غير مقبول لأنه قد يكون من أهداف التوجيه تجنب مشاكل الازدحام على الوصلات (كالتأخير العالي مثلاً). يكمن حل آخر في

تجنب تشغيل خوارزمية LS في كل الموجّهات في نفس الوقت. هذا الحل يبدو أكثر معقولية، حيث إننا نأمل أنه رغم استخدام الموجّهات لخوارزمية LS بنفس الدورية (periodicity) فإن لحظة التنفيذ لن تكون نفسها في كل عقدة. وبشكلٍ مثير للانتباه وجد الباحثون أنه يمكن أن يحدث تزامن ذاتي فيما بين موجّهات الإنترنت [Floyd Synchronization 1994]. بمعنى أنه رغم أن الموجّهات تنفّذ في البداية الخوارزمية بنفس الفترة الدورية وفي لحظات مختلفة لكن يمكن أن يؤول الأمر في النهاية إلى أن تنفذ الموجّهات الخوارزمية بصورة متزامنة (ومن ثم تستمر على هذا الحال). أحد طرق تفادي مثل هذا التزامن الذاتي هو أن يختار كل موجّه وقت الإعلان عن حالة الوصلة بطريقة عشوائية.

بعد أن تناولنا خوارزمية LS دعنا نناقش خوارزمية التوجيه الرئيسة الأخرى (أي خوارزمية توجيه متجه المسافة).

2-5-4 خوارزمية توجيه متجه المسافة (DV)

بينما تستخدم خوارزمية LS معلومات عالمية، تُعتبر خوارزمية توجيه متجه المسافة (DV) موزّعة وتكرارية ولاتزامنية. فهي موزّعة حيث إن كل عقدة تستلم بعض المعلومات من واحد أو أكثر من جيرانها الموصّلين بها مباشرة، وتقوم بإجراء الحسابات، ثم بعد ذلك توزّع النتائج إلى جيرانها. وهي تكرارية حيث إن هذه العملية تستمر حتى ينتهي تبادل المزيد من المعلومات بين الجيران. (وبشكلٍ هام هذه الخوارزمية ذاتية الانتهاء أيضاً حيث لا توجد إشارة لإنهاء الحسابات وإنما فقط تنتهي). وأخيراً هي خوارزمية لاتزامنية حيث إنها لا تتطلّب أن تعمل كل العقد بإيقاع موحد مع بعضها البعض. سنرى بعد قليل كيف أن خوارزمية لاتزامنية وتكرارية وذاتية الانتهاء وموزّعة تكون أكثر تشويقاً ومتعة من خوارزمية مركزية!

قبل أن نقدّم خوارزمية DV من المفيد مناقشة علاقة مهمة توجد بين كُلَف مسارات أدنى كلفة. لنرمز لكلفة مسار أدنى كلفة من العقدة x إلى العقدة y بـ

$d_x(y)$. عندئذ ترتبط الكلف بمعادلة بلمن - فورد (Bellman-Ford) الشهيرة كما يلي:

$$d_x(y) = \min_v\{c(x, v) + d_v(y)\} \quad (4\text{-}1)$$

حيث إن $\min_v$ في المعادلة تطبق على كل جيران x. إن معادلة بلمن - فورد بدهية نوعاً ما. في الحقيقة بعد الانتقال من x إلى v نأخذ عندئذ المسار الأدنى كلفة من v إلى y، وبالتالي تكون كلفة المسار $c(x,v) + d_v(y)$. ولأننا يجب أن نبدأ بالانتقال إلى أحد الجيران v، فإن أدنى كلفة من x إلى y تكون أصغر قيمة لـ $c(x,v) + d_v(y)$ على كل الجيران.

لكن لأولئك الذين قد يساورهم الشك في صحة المعادلة، دعنا نختبرها لعقدة المصدر u وعقدة الوجهة z في الشكل 4-27. لعقدة المصدر u ثلاثة جيران: v، x، w. بتتبع المسارات المختلفة في الرسم البياني من السّهل رؤية أن $d_v(z)=5$، $d_x(z)=3$، $d_w(z)=3$. بتعويض هذه القيم في المعادلة 4-1 مع الكلف $c(u,v)=2$، $c(u,x)=1$، $c(u,w)=5$ نحصل على $d_u(z) = \min\{2+5, 5+3, 1+3\} = 4$. من الواضح أن هذا صحيح ويمثل تماماً ما نحصل عليه من خوارزمية Dijskstra لنفس الشبكة. هذا التحقّق السريع يجب أن يساعد في التخفيف من أي تشكك عندك.

إن معادلة بلمن - فورد ليست مجرد فضول ثقافي، وإنما لها في الحقيقة أهمية عملية هامّة. وبشكلٍ محدد يعطي الحل لمعادلة بلمن - فورد المدخلات في جدول التمرير للعقدة x. ولرؤية هذا نرمز بـ v^* لأي عقدة مجاورة لها الحدّ الأدنى في المعادلة 4-1. عندئذ إذا أرادت العقدة x إرسال رزمة إلى العقدة y على المسار الأدنى كلفة، يجب أن ترسل الرزمة أولاً إلى العقدة v^*. وهكذا يحتوي جدول التمرير للعقدة x العقدة v^* كموجّه القفزة التالية للوجهة النهائية y. وكمساهمة عملية مهمة أخرى لمعادلة بلمن - فورد فإنها تقترح شكل الاتصال من جار لجار والذي يحدث في خوارزمية DV.

تتلخص الفكرة الأساسية في التّالي: تبدأ كل عقدة x بتقدير $D_x(y)$ لكلفة مسار أدنى كلفة منها إلى العقدة y، وذلك لكل العقد في N. ولنرمز لمتجه المسافة

للعقدة x بـ $\boldsymbol{D}_x = [D_x(y): y \in N]$. تحتفظ كل عقدة x في خوارزمية DV بمعلومات التوجيه التالية:

- الكلفة $c(x,v)$ من x لكل جار v موصّل بها مباشرة.
- متجه المسافة للعقدة x أي $\boldsymbol{D}_x = [D_x(y): y \in N]$ والذي يحتوي على مسافات تقديرية لكل وجهة y في N.
- متجه المسافة لكل عقدة من جيرانها أي $\boldsymbol{D}_v = [D_v(y): y \in N]$ لكل جار من جيران العقدة x.

من وقت لآخر في الخوارزمية اللاتزامنية الموزّعة ترسل كل عقدة نسخة من متجه المسافة الخاص بها إلى كل جيرانها. عندما تستلم العقدة x متجه مسافة جديد من أيٍّ من جيرانها وليكن v تقوم بتخزين متجه المسافة للعقدة v ثم بعد ذلك تستخدم معادلة بلمن - فورد لتحديث متجه المسافة الخاص بها كالتّالي:

$$D_x(y) = \min_v\{c(x, v)+D_v(y)\}$$

لجميع العقد y في N. إذا تغيّر متجه المسافة للعقدة x كنتيجة لخطوة التحديث هذه، عندئذ سترسل العقدة x متجه المسافة الجديد الخاص بها إلى كل جيرانها، والذين يمكن تباعاً أن يُعدِّلوا متجهات المسافة الخاصة بهم. وطالما أن كل العقد تواصل تبادل متجهات المسافة بشكلٍ غير متزامن، فإن تقديرات المسافة $D_x(y)$ ستؤول للقيمة الحقيقية لمسار أدنى كلفة من x لـ y أي $d_x(y)$ [Bertsekas 1991].

في خوارزمية DV تقوم العقدة x بتحديث متجه المسافة التقديري عندما ترى تغيّراً في كلفة إحدى وصلاتها المرتبطة بها مباشرة أو عندما تستلم متجه مسافة جديد من أحد جيرانها. لكن لتحديث جدول التمرير الخاص لوجهة معينة y تحتاج العقدة x حقاً لمعرفة ليس فقط مسافة المسار الأقصر إلى y وإنما العقدة المجاورة $v^*(y)$ التي تمثل أول قفزة على طول المسار الأقصر إلى y. وربما تكون قد توقّعتَ أن موجّه القفزة التالية $v^*(y)$ هو العقدة المجاورة التي تحقق أصغر قيمة في السطر 14 لخوارزمية DV. إذا كان هناك عدة جيران v لهم نفس القيمة الصغرى فيمكن استخدام أيٍّ منهم. هكذا في السطور 13-14 تقوم العقدة x أيضاً بتحديد $v^*(y)$ وتحديث جدول التمرير لكل وجهة y.

خوارزمية متجه المسافة (DV)

في كل عقدة x:

```
1  Initialization:
2    for all destinations y in N:
3      Dx(y)=c(x, y)     /* if y is not a neighbor then c(x, y) = ∞ */
4    for each neighbor w
5      Dw(y)= ∞ for all destinations y in N
6    for each neighbor w
7      send distance vector Dx = [Dx(y): y in N] to w
8
9  Loop
10   wait (until I see a link cost change to some neighbor w or
11        until I receive a distance vector from some neighbor w)
12
13   for each y in N:
14     Dx(y) = minv{c(x, v) + Dv(y)}
15
16   if Dx(y) changes for any destination y
17     send distance vector Dx = [Dx(y): y in N] to all neighbors
18
19 forever
```

تذكّر أن خوارزمية LS عالمية بمعنى أنها تتطلّب من كل عقدة الحصول أولاً على خريطة كاملة للشبكة قبل تشغيل خوارزمية Dijkstra؛ في حين خوارزمية DV لامركزية ولا تستخدم مثل هذه المعلومات العالمية. في الحقيقة المعلومات الوحيدة التي لدى كل عقدة هي كلفة كل الوصلات إلى جيرانها الملحقين بها مباشرةً والمعلومات التي تستقبلها من هؤلاء الجيران. تنتظر كل عقدة رسالة تحديث من أي جار (السطور 10-11)، وتحسب متجه المسافة الجديد عندما تستقبل رسالة التحديث تلك (السطر 14)، ثم توزّع متجه المسافة الجديد إلى جيرانها (السطور 16-17). تستخدم الخوارزميات التي تشبه DV عملياً في العديد من بروتوكولات

التوجيه، بما في ذلك بروتوكولات الإنترنت RIP وBGP، وبروتوكول ISO IDRP، وبروتوكول Novell IPX، وبروتوكول شبكة Arpanet الأصلي.

يوضح الشكل 4-30 طريقة عمل خوارزمية DV لشبكة بسيطة مكونة من ثلاث عقد والموضحة في أعلى الشكل. إن طريقة عمل الخوارزمية مصوّرة بطريقة متزامنة، حيث تستلم كل العقد متجهات المسافة بشكلٍ آني من جيرانها، وتحسب متجهات المسافة الجديدة الخاصة بها، ومن ثَمَّ تُخبر جيرانها إذا تغيّرت متجهات المسافة. يجب أن تقنع نفسك بعد هذا المثال أنه يمكن إجراء الخوارزمية بشكلٍ صحيح في أسلوب لاتزامني أيضاً، بمعنى إجراء الحسابات وإرسال واستقبال رسائل التحديث في أي وقت كان.

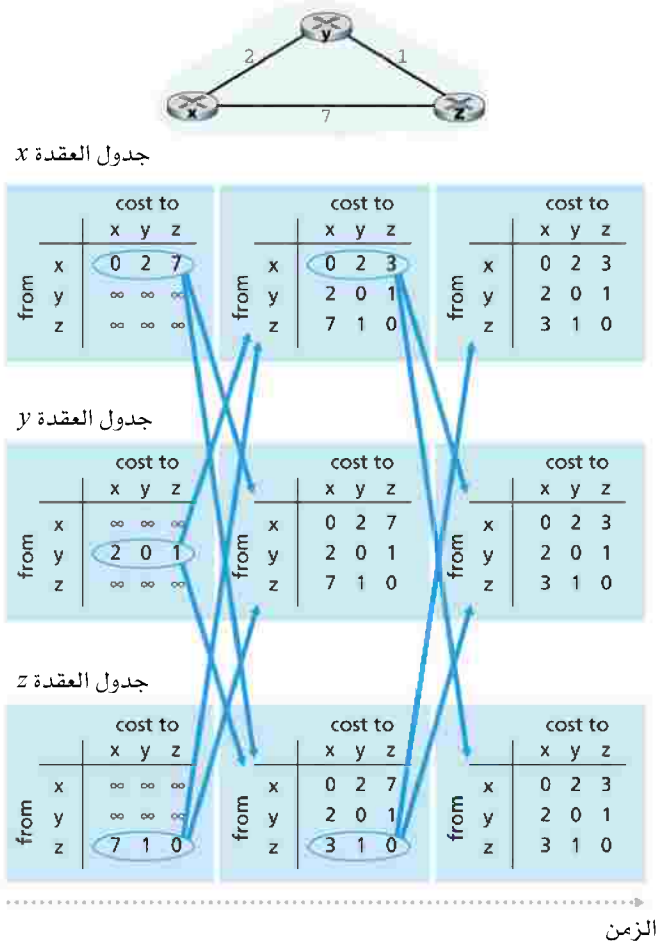
يعرض العمود في أقصى اليسار للشكل ثلاثة جداول توجيه أوّليَّة لكلٍّ من العقد الثلاث. على سبيل المثال الجدول في الزاوية اليسرى من أعلى هو جدول التوجيه الأوَّلي للعقدة x. بداخل جدول توجيه معين يمثل كل صف متجه مسافة. وبالتحديد يتضمّن جدول التوجيه لكل عقدة متجه المسافة الخاص بها وذلك الخاص بكلٍّ من جيرانها. وهكذا يكون الصف الأول في جدول التوجيه الأوَّلي للعقدة x كالآتي:

$$\boldsymbol{D}_x=[D_x(x), D_x(y), D_x(z)]= [0, 2, 7]$$

أما الصفان الثاني والثالث في هذا الجدول فيمثلان متجهي المسافة المستلمين مؤخراً من العقد y وz على التوالي. ولأنه عند التهيئة لم تستلم العقدة x أي شيء من العقد y وz فإن المدخلات في الصفين الثاني والثالث تأخذ القيمة ما لانهاية.

بعد التهيئة ترسل كل عقدة متجه المسافة الخاص بها إلى كلٍّ من جيرانها. هذا موضح في الشكل 4-30 بالأسهم من جداول العمود الأول إلى جداول العمود الثاني. فمثلاً ترسل العقدة x متجه المسافة $\boldsymbol{D}_x=[0, 2, 7]$ إلى كلٍّ من y وz. بعد استلام رسالة التحديث تلك تقوم كل عقدة بإعادة حساب متجه المسافة الخاص بها. على سبيل المثال تحسب العقدة x:

$D_x(x) = 0$
$D_x(y)=\min\{c(x,y)+D_y(y), c(x,z)+D_z(y)\}=\min\{2+0, 7+1\}=2$
$D_x(z)=\min\{c(x,y)+D_y(z), c(x,z)+D_z(z)\}=\min\{2+1, 7+0\}=3$



الشكل 4-30 خوارزمية متجه المسافة (DV).

لذا يعرض العمود الثاني لكل عقدة متجه المسافة الجديد للعقدة سويةً مع متجهات المسافة التي استلمتها للتّوّ من جيرانها. لاحظ أن تقديرات العقدة x – على سبيل المثال – لأدنى كلفة إلى العقدة z أي $D_x(z)$ قد تغيّرت من 7 إلى 3. لاحظ أيضاً أن العقدة y كأحد جيران العقدة x تحقق لها أدنى قيمة في السطر 14 من

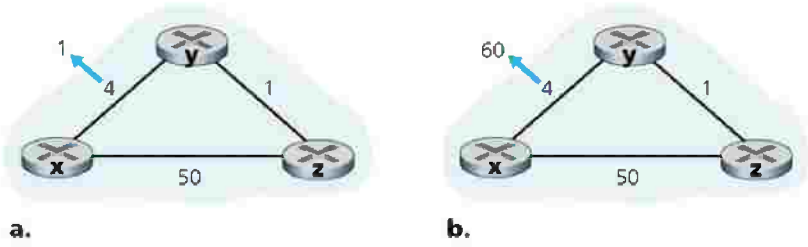
خوارزمية DV؛ وبهذا تصبح y عقدة أول قفزة للعقدة x في هذه المرحلة $v^*(y)=y$, $v^*(z)=y$.

بعد انتهاء العقد من حساب متجهات المسافة الخاصة بها تقوم بإرسالها مجدداً إلى جيرانها (في حالة حدوث تغيير)، كما هو موضح في الشكل 4-30 بالأسهم من جداول العمود الثاني إلى جداول العمود الثالث. لاحظ أن العقد x وz فقط هي الوحيدة التي ترسل رسائل تحديث؛ أما متجه المسافة للعقدة y فلم يتغيّر ولذا لا ترسل العقدة y رسالة تحديث. بعد استلام العقد لرسائل التحديث تقوم بإعادة حساب متجهات المسافة وتحديث جداول التوجيه كما هو موضح في العمود الثالث.

تتكرر عملية استلام متجهات المسافة من الجيران، وإعادة حساب مدخلات جدول التوجيه، وإعلام الجيران بالتغيّرات في كلف مسارات أدنى كلفة إلى الوجهة إلى أن يتوقف إرسال رسائل تحديث جديدة. عند هذه النقطة ونظراً لعدم استقبال رسائل تحديث لا تتم أية تعديلات أخرى على جداول التوجيه وتدخل الخوارزمية حالة خمود (أي تنفذ كل العقد الانتظار في السطور 10-11 من خوارزمية DV). تبقى الخوارزمية في حالة الخمود حتى تتغيّر كلفة وصلة كما سنناقش فيما يلي.

خوارزمية متجه المسافة: تغيّر كلفة وصلة أوعطلها

عندما تكتشف عقدة تستخدم خوارزمية DV تغيّراً في كلفة وصلة بينها وبين أحد جيرانها (السطور 10-11)، تقوم بتحديث متجه المسافة الخاص بها (السطور 13-14) وإذا حدث تغيّر في كلفة مسار أدنى كلفة فإنها تخبر جيرانها (السطور 16-17) بمتجه المسافة الجديد الخاص بها. يوضح الشكل 4-31 (a) سيناريو تغيّرت فيه كلفة الوصلة بين y وx من 4 إلى 1. سنركّز هنا فقط على مدخلات جداول العقد y وz إلى الوجهة x. تتسبّب خوارزمية DV في سلسلة الأحداث التالية:



الشكل 4-31 تغيّر كلفة الوصلة.

- في الوقت t_0 تكتشف y تغيّر كلفة الوصلة (من 4 إلى 1) فتقوم بتحديث متجه المسافة الخاص بها، وتخبر جيرانها بهذا التغيير.
- في الوقت t_1 تستلم z رسالة التحديث من y فتقوم بتحديث جدولها. تحسب z المسار الجديد الأدنى كلفة إلى x (تقل الكلفة من 5 إلى 2) وترسل متجه المسافة الجديد إلى جيرانها.
- في الوقت t_2 تتسلم y رسالة تحديث من z وتعدّل متجه المسافة الخاص بها، غير أن أقل كلفة لا تتغيّر ولذا لا ترسل أي رسالة إلى z. عندها تدخل الخوارزمية حالة خمود.

وهكذا تحتاج خوارزمية DV فقط لتكرارين حتى تصل لحالة الخمود. تنتشر الأخبار الجيدة حول النقص في الكلفة بين x وy بسرعة خلال الشبكة.

دعنا نرى الآن ما يمكن أن يحدث عندما تزيد كلفة وصلة. افترض بأن كلفة الوصلة بين x وy زادت من 4 إلى 60 كما هو موضح في الشكل 4-31 (b).

1. قبل أن تتغيّر كلفة الوصلة $D_y(x) = 4$، $D_y(z) = 1$، $D_z(y) = 1$، $D_z(x) = 5$. في الوقت t_0 تكتشف y تغيّر كلفة الوصلة (تغيّرت الكلفة من 4 إلى 60). تحسب y المسار الجديد الأدنى كلفة إلى x لتصبح

$$D_y(x)=\min\{c(y,x)+D_x(x),\ c(y,z)+D_z(x)\}=\min\{60+0,\ 1+5\}=6$$

بالطبع مع نظرتنا الكلية للشبكة يمكن أن نرى أن هذه الكلفة الجديدة عن طريق z خاطئة. لكن المعلومات الوحيدة المتوفرة لدى العقدة y هي أن الكلفة المباشرة إلى x تساوي 60 وأن z قد أخبرت y أخيراً بأنها يمكنها

الوصول إلى x بكلفة 5. لذا لكي تصل y إلى x ستوجّه الآن عبر z وتتوقّع أن تكون z قادرة على الوصول إلى x بكلفة 5.

عند الوقت t_1 تتكون لدينا حلقة توجيه مفرغة (مسار مغلق): فلكي تصل y إلى x توجّه إلى z، ولكن z توجه إلى x عبر y. إن حلقة التوجيه تشبه ثقباً مظلماً (black hole)، حيث تبقى الرزمة الموجهة إلى x ترتد من y إلى z والعكس ابتداءً من زمن t_1 إلى الأبد (أو إلى أن تتغيّر جداول التمرير).

2. لأن العقدة y قد حسبت حداً أدنى جديداً للكلفة إلى x، فإنها تخبر z بمتجه المسافة الجديد عند الزمن t_1.
3. في وقتٍ ما بعد t_1 تتسلم z متجه المسافة الجديد الخاص بـ y، والذي يشير بأن أقل كلفة من y إلى x تساوي 6. عندها تعلم z بأنه يمكن أن تصل إلى y بكلفة مقدرها 1 ومن ثَمَّ تحسب أقل كلفة جديدة إلى x من $D_z(x)=\min\{50+0, 1+6\}=7$. ونظراً لزيادة أدنى كلفة من z إلى x فإنها تخبر y بمتجه المسافة الجديد عند t_2.
4. بطريقة مماثلة بعد استلام متجه المسافة الجديد الخاص بـ z تحدد y أن $D_y(x)=8$ وترسل إلى z متجه المسافة الخاص بها. ومن ثَمَّ تقرّر z بأن $D_z(x)=9$ وترسل إلى y متجه المسافة الخاص بها، وهكذا.

إلى متى ستستمر تلك العملية؟ يجب أن تقنع نفسك أن الحلقة تُكرَّر 44 مرة (تبادل رسائل بين y وz) حتى تحسب z في النهاية كلفة مسارها عن طريق y ليصبح أكبر من 50. في هذه النقطة ستقرر z (أخيراً!) أن مسار أدنى كلفة إلى x هو عن طريق الوصلة المباشرة إلى x. عندئذ ستوجه y إلى x عن طريق z. النتيجة هي أن الأخبار السيئة حول الزيادة في كلفة الوصلة تنتشر في الحقيقة ببطء! ماذا كان سيحدث إذا كانت كلفة الوصلة $c(y, x)$ قد تغيّرت من 4 إلى 10000 وكانت الكلفة $c(z, x) = 9999$؟ بسبب مثل هذه السيناريوهات يطلق على هذه المشكلة أحياناً اسم مشكلة العد لما لانهاية (count-to-infinity problem).

خوارزمية متجه المسافة: إضافة اتجاه عكسي مسمّم (Poisoned Reverse)

يمكن تجنب سيناريو حلقة التوجيه المفرغة الذي تم وصفه للتو باستخدام تقنية تعرف بالاتجاه العكسي المسمّم. إن الفكرة بسيطة؛ إذا وجهت z من خلال y للوصول إلى الوجهة x فإن z تُعلِم y أن المسافة منها إلى x مالانهاية، أي أن $D_z(x) = \infty$ رغم معرفتها حالياً أنه في الحقيقة $D_z(x)=5$. تستمر z في إطلاق هذه الكذبة البيضاء الصغيرة إلى y طالما توجّه z إلى x عن طريق y. ولأن y تعتقد أن z ليس لديها مسار إلى x فلن تحاول y التوجيه إلى x عن طريق z طالما أن z تواصل التوجيه إلى x عن طريق y (وتستمر في كذبها حول ذلك!).

دعنا الآن نرى كيف أن تسميم الاتجاه العكسي يحلّ مشكلة حلقة التوجيه المفرغة التي واجهناها من قبل في الشكل 4-31 (b). كنتيجة لتسميم الاتجاه العكسي يشير جدول المسافة الخاص بـ y لأن $D_z(x) = \infty$. عندما تتغيّر كلفة الوصلة (x, y) من 4 إلى 60 في الوقت t_0 تقوم y بتحديث جدولها وتستمر في التوجيه مباشرة إلى x رغم أن الكلفة الآن أعلى وتساوي 60، وتخبر z الكلفة الجديدة إلى x، (أي أن $D_y(x)=60$). بعد استلام z لرسالة التحديث في الوقت t_1 تحوّل طريقها فوراً إلى x ليكون عن طريق الوصلة (z, x) وبكلفة تساوي 50. ولأن هذا يمثل مساراً جديداً إلى x ولأن المسار لم يعد يمرّ من خلال y، سوف تخبر z الآن y بأن $D_z(x)=50$ في الوقت t_2. بعد استلام رسالة التحديث من z تقوم y بتحديث جدول المسافة ليتضمن $D_y(x)=51$. أيضاً لأن z الآن على مسار أقلّ كلفة من y إلى x تقوم y بتسميم المسار العكسي من z إلى x بإعلام z في الوقت t_3 أن $D_y(x)=\infty$ (رغم أن y في الحقيقة تعرف أن $D_y(x)=51$).

هل تسميم الاتجاه العكسي يحل أيضاً مشكلة العد لما لانهاية؟ الجواب لا. يجب أن تقنع نفسك بأن الحلقات التي تتضمّن ثلاثة عقد أو أكثر لن يتم اكتشافها بأسلوب تسميم الاتجاه العكسي.

مقارنة بين خوارزميات التوجيه LS وDV

تتبع خوارزميات DV وLS طرقاً تتكامل فيما بينها لحساب التوجيه. في خوارزمية DV تخاطب كل عقدة جيرانها المرتبطين بها مباشرة فقط، لكنها تزودهم بتقديرات أدنى كلفة للمسارات بينها وبين كل العقد (التي تعرفها) في الشبكة. في خوارزمية LS تخاطب كل عقدة جميع العقد الأخرى (عن طريق الإرسال الإذاعي)، لكنها تخبرهم بكلف الوصلات المرتبطة مباشرة بها فقط. دعنا ننهي دراستنا لخوارزميات LS وDV بمقارنة سريعة لبعض خواصهما. تذكّر أن N هي مجموعة العقد (الموجّهات) وE هي مجموعة الحافات (الوصلات).

- تعقيد الرسالة (message complexity): لقد رأينا أن خوارزمية LS تحتاج لمعرفة كل عقدة بكلفة كل وصلة في الشبكة، مما يتطلب أن تكون الرسائل المرسلة $O(|N|.|E|)$. أيضاً عندما تتغيّر كلفة وصلة يجب أن ترسل الكلفة الجديدة للوصلة إلى كل العقد. أما خوارزمية DV فتتطلب تبادل الرسائل بين الجيران المرتبطين مباشرة فقط في كل تكرار. كما رأينا أن الوقت الذي تستغرقه الخوارزمية للتقارب يمكن أن يعتمد على العديد من العوامل. وعند تغيير كلفة الوصلة فإن خوارزمية DV ستبث نتائج الوصلة التي تغيرت كلفتها فقط إذا أدت كلفة الوصلة الجديدة إلى مسار جديد أقل كلفةً لأحد العقد المرتبطة بتلك الوصلة.
- سرعة التقارب (speed of convergence): لقد رأينا أن تحقيق خوارزمية LS من الدرجة $O(|N|^2)$ ويتطلب رسائل $O(|N|.|E|)$. يمكن أن تتقارب خوارزمية DV ببطء ويمكن أن تعاني من حلقات التوجيه المفرغة أثناء تقارب الخوارزمية. تعاني DV أيضاً من مشكلة العد لما لانهاية.
- المتانة (robustness): ماذا يمكن أن يحدث لو تعطل أحد الموجّهات، أو أساء التصرف، أو تم اختراقه؟ في حالة LS يمكن أن يذيع الموجّه كلفة خاطئة لأحد وصلاته المرتبطة به مباشرة (لكن يقتصر الخطأ على هذا التصرف). يمكن أيضاً أن تخرّب أحد العقد أو تسقط أياً من الرزم التي تستلمها كجزء من إرسال إذاعة ضمن خوارزمية LS. لكن عقدة LS تحسب جداول

التمرير الخاصة بها فقط، وتقوم العقد الأخرى بحسابات مماثلة لنفسها. هذا يعني فصل حسابات المسارات بعض الشيء عند استخدام LS، مما يزيد درجة متانة الخوارزمية. عند استخدام DV يمكن أن تعلن عقدة مسارات أدنى كلفةً خاطئة لأي وجهة من الوجهات أو لها جميعاً. مثال ذلك ما حدث عام 1997 حيث أدى عطب بموجّه في شبكة موفر خدمة إنترنت صغير إلى تزويد موجّهات شبكة العمود الفقري القومية بمعلومات توجيه خاطئة. نتج عن ذلك إغراق موجّهات أخرى للموجّه المتعطل بحركة مرور البيانات وبقيت أجزاء كبيرة من الإنترنت مقطوعة لعدّة ساعات [Neumann 1997]. وبتعميم أكثر نلاحظ أنه في كل تكرار يُنقل ناتج حساب عقدة في DV إلى جيرانها وبعد ذلك بشكلٍ غير مباشر إلى جيران جيرانها في التكرار التالي. وهذا يعني أنه يمكن أن ينتشر حساب عقدة خاطئ خلال الشبكة بكاملها عند استخدام DV.

في النهاية يتضح أن أياً من خوارزميات LS وDV لا يُعتبر فائزاً على الآخر، فكلاهما مستعمل في الإنترنت في حقيقة الأمر.

خوارزميات التوجيه الأخرى

لا تُعتبر خوارزميات LS وDV التي درسناها فقط واسعة الانتشار عملياً، ولكنهما بالضرورة بمثابة خوارزميات التوجيه الوحيدة المستعملة واقعياً اليوم في الإنترنت. ومع ذلك فقد تم اقتراح خوارزميات توجيه أخرى عديدة من قبل الباحثين خلال السنوات الـ 30 الماضية، تتراوح من البسيط جداً إلى المتطوّر والمعقّد جداً. يعتمد أحد الأنواع العامة لخوارزميات التوجيه على النظر إلى مرور الرزم كتدفقات (flows) بين المصادر والوجهات في الشبكة. من هذا المنطلق يمكن صياغة مشكلة التوجيه بطريقة رياضية مثل مشكلة تحقيق حل أمثل ذات محدودات (constrained optimization problem) والمعروفة بمشكلة تدفق الشبكة (network flow problem) [Bertsekas 1991]. نذكر هنا نوعاً آخر من خوارزميات التوجيه مشتقّ من عالم الإرسال الهاتفي يعرف بخوارزميات التوجيه بتحويل الدوائر، ولهذا النوع أهمية في

شبكات البيانات بتحويل الرزم في الحالات التي نحتاج فيها لحجز موارد الوصلة (مثل الذاكرة المؤقتة والحيز الترددي للوصلة) لكل توصيلة تمر خلال الوصلة. بينما قد تبدو صياغة مشكلة التوجيه بهذا الأسلوب مختلفة تماماً عن صياغة توجيه أدنى كلفة التي رأيناها في هذا الفصل، إلا أن هناك عدد من التشابهات على الأقل من حيث خوارزمية إيجاد المسار (خوارزمية التوجيه). اطّلع على [Ash 1998; Ross 1995; Girard 1990] لمناقشة أكثر تفصيلاً لأبحاث في هذا الموضوع.

3-5-4 التوجيه الهرمي (Hierarchical Routing)

في دراستنا لخوارزميات LS وDV نظرنا للشبكة ببساطة كمجموعة من الموجّهات المرتبطة ببعضها البعض. لم نُفرِّق بين موجّه وآخر حيث إنها جميعاً تُنفِّذ نفس خوارزمية التوجيه لحساب مسارات التوجيه خلال الشبكة بكاملها. عملياً هذا النموذج ونظرته للموجّهات كمجموعة متجانسة يُنفّذ كلٌّ منها نفس خوارزمية التوجيه يُعد تبسيطاً للأمور إلى حد كبير، وذلك لسببين مهمين:

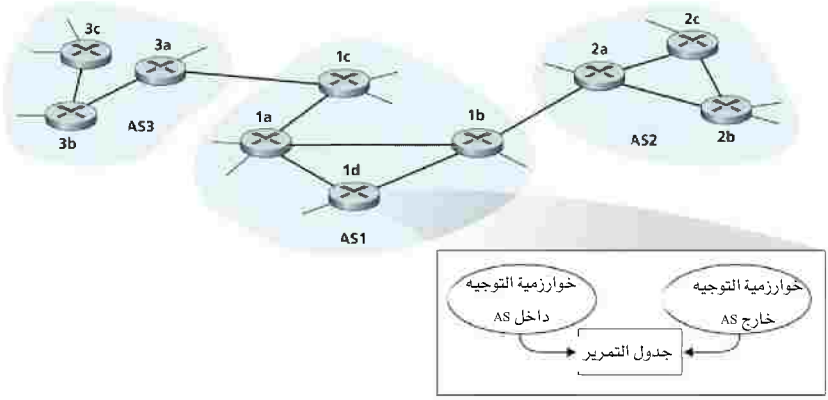
- حجم الشبكة: مع زيادة عدد الموجّهات يصبح العبء الإضافي في حساب وتخزين وتبادل معلومات التوجيه (على سبيل المثال رسائل التحديث في LS وتغييرات المسار الأدنى كلفة) عائقاً. تشمل الإنترنت العامّة اليوم مئات الملايين من المضيفات، وواضح أن تخزين معلومات التوجيه في كل تلك المضيفات يتطلب كميات هائلة من الذاكرة. وكذلك فإن العبء الإضافي لإذاعة التحديثات في LS بين كل الموجّهات في الإنترنت العامّة لن يترك أي حيز ترددي لإرسال رزم البيانات! وبالتأكيد فإن خوارزمية متجه المسافة التي تتكرّر بين مثل هذا العدد الكبير من الموجّهات لن تتقارب أبداً. واضح أنه يجب عمل شيء لتخفيض تعقيد حساب المسار في الشبكات الكبيرة كالإنترنت العامّة.
- ذاتية الحكم الإداري: يميل الباحثون لإهمال بعض القضايا كرغبة شركة ما في تشغيل موجّهاتها كما يحلو لها (على سبيل المثال تشغيل خوارزمية التوجيه التي تختارها) أو في إخفاء السمات التنظيمية للشبكة الداخلية عن خارج الشبكة، ولكن تلك في واقع الأمر اعتبارات مهمة. فمن الناحية

المثالية يجب أن تكون المنظمة قادرة على تشغيل وإدارة شبكتها كما تحب، وفي الوقت ذاته تبقى قادرة على توصيل شبكتها بالشبكات الأخرى الخارجية.

يمكن أن تُحل كلتا هاتين المشكلتين بتنظيم الموجّهات في أنظمة مستقلة ذاتياً ((Autonomous Systems (ASs)، يتكون كلٌّ منها من مجموعة من الموجّهات التي عادةً ما تكون تحت نفس الرقابة الإدارية (مثلاً تُشغَّل من قِبَل نفس موفِّر خدمة الإنترنت أو تنتمي لشبكة شركة بعينها). تُنفِّذ جميع الموجّهات داخل النظام المستقل ذاتياً نفس خوارزمية التوجيه (كخوارزمية LS أو DV) وتتوافر لديها المعلومات عن بعضها البعض، بالضبط كما في حالة النموذج المثالي الذي تناولناه في الجزء السابق. يطلق على خوارزمية التوجيه التي تعمل ضمن نظام مستقل ذاتياً بروتوكول توجيه داخل نظام مستقل ذاتياً (intra-AS routing). بالطبع سيكون من الضروري توصيل تلك الأنظمة المستقلة ذاتياً إلى بعضها البعض، وبالتالي سيضاف موجّه أو أكثر تناط به المسؤولية الإضافية لتوجيه الرزم إلى الوجهات خارج النظام المستقل ذاتياً (inter-AS routing)؛ تسمى تلك الموجّهات موجّهات البوّابة (gateway routers).

يبين الشكل 4-32 مثالاً بسيطاً لثلاثة أنظمة مستقلة ذاتياً: AS1، AS2، وAS3. تمثّل الخطوط الثقيلة في هذا الشكل الوصلات المباشرة بين أزواج الموجّهات. أما الخطوط الأخف بين الموجّهات فتمثّل الشبكات الفرعية التي تتصل مباشرة بتلك الموجّهات. يتكون AS1 من أربعة موجّهات (1a، 1b، 1c) تدير عملية التوجيه داخل AS1. ولذا فكلٌّ من هذه الموجّهات الأربعة تعرف كيف ترسل الرزم على طول المسار الأمثل (optimal route) إلى أي وجهة ضمن AS1. وبنفس الطريقة لكلٍّ من الأنظمة المستقلة ذاتياً AS2 وAS3 ثلاثة موجّهات. لاحظ أن بروتوكولات التوجيه داخل الأنظمة AS1 وAS2 وAS3 لا يشترط أن تكون هي نفسها. لاحظ أيضاً أن الموجّهات 1b و1c و2a و3a تعمل كموجّهات بوّابة.

نأمل أن يكون واضحاً لديك الآن كيف تقوم الموجّهات في AS بتحديد مسارات التوجيه بين أزواج المصدر والوجهة الموجودة داخل AS. لكن ما زالت هناك قطعة كبيرة مفقودة من لغز التوجيه من طرف إلى طرف. كيف لموجّه يعمل داخل AS أن يعرف كيف يوجّه رزماً إلى وجهة خارج AS؟ من السّهل الإجابة على هذا السؤال إذا كان نظام AS لديه موجّه بوّابة واحد يوصله إلى AS آخر فقط. في هذه الحالة تحدد خوارزمية التوجيه داخل AS مسار أدنى كلفة من كل موجّه داخلي إلى موجّه البوّابة، والذي يعرف بدوره كيف يوجه الرزمة. بمجرد استلام موجّه البوّابة الرزمة فإنه يرسلها على الوصلة الواحدة التي تقود إلى خارج AS. يتحمل نظام AS على الجانب الآخر من الوصلة مسؤولية توجيه الرزمة إلى الوجهة النهائية. وكمثال افترض أن الموجّه 2b في الشكل 4-32 يستلم رزمة وجهتها خارج AS2. سيرسل الموجّه 2b بعد ذلك الرزمة لأيٍّ من الموجّهات 2a أو 2c حسبما هو مبين بجدول التمرير للموجّه 2b، والذي أُعد من قِبَل بروتوكول التوجيه داخل AS2. ستصل الرزمة في النهاية إلى موجّه البوّابة 2a، والذي سيرسلها إلى 1b. بمجرد مغادرة الرزمة 2a تكون مهمة AS2 قد انتهت مع تلك الرزمة.



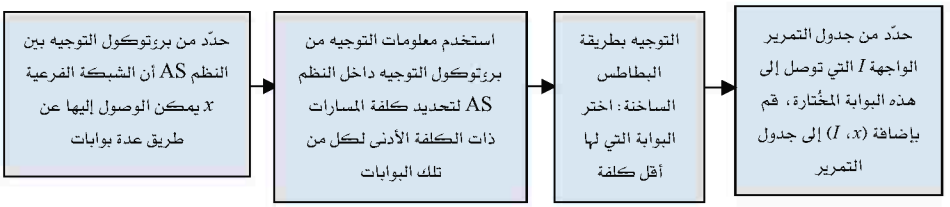
الشكل 4-32 مثال لعدة أنظمة مستقلة ذاتياً متصلة ببعضها.

ولذا فالمشكلة سهلة عندما يكون لـ AS المصدر وصلة واحدة فقط تقود خارجه. لكن ماذا لو أن AS المصدر كانت له وصلتان أو أكثر (خلال موجّهي بوّابة أو أكثر) تقود خارجه؟ عندئذ تصبح مشكلة معرفة أين تُرسل الرزمة أكثر تحدّياً. على سبيل المثال خذ في الاعتبار موجّهاً في AS1 كان قد تلقى رزمة وجهتها خارج AS1. واضح أنه يجب على الموجّه أن يرسل الرزمة إلى إحدى موجهات البوّابة 1b أو 1c. لكن لأيٍّ منهما؟ لحلّ هذه المشكلة يحتاج AS1 إلى (1) معرفة أي الوجهات يمكن الوصول إليها عن طريق AS2 وأيٍّ منها يمكن الوصول إليه عن طريق AS3، و(2) نشر معلومات الوصول هذه إلى كل الموجهات داخل AS1 حتى يمكن لكل موجّه أن يعد جدول تمرير لمعالجة الوجهات الخارجية. يتم أداء هاتين المهمّتين (الحصول على معلومات الوصول من نظم AS المجاورة ونشر تلك المعلومات لكل الموجّهات الداخلية) بواسطة بروتوكول التوجيه بين الأنظمة المستقلة ذاتياً. ولأن بروتوكول التوجيه بين ASs يتضمّن الاتصال بين نظامي AS، فمن الضروري أن يستخدم النظامان نفس بروتوكول التوجيه البيني. في الحقيقة تستخدم كل ASs في الإنترنت نفس بروتوكول التوجيه بين الأنظمة المستقلة ذاتياً، والمعروف ببروتوكول BGP4، والذي سنناقشه في الجزء التالي. كما هو موضح في الشكل 4-32 يستلم كل موجّه معلومات من بروتوكول التوجيه داخل AS ومن بروتوكول التوجيه خارج AS، ويستخدم المعلومات من كليهما لإعداد جدول التمرير.

وكمثال افترض شبكة فرعية x (مُعرّفة بعنوان CIDR)، وافترض بأن AS1 علِم من بروتوكول التوجيه بين ASs أنه يمكن الوصول لتلك الشبكة الفرعية x عن طريق AS3 ولكن لا يمكن الوصول لها عن طريق AS2. عندئذ يذيع AS1 هذه المعلومات إلى كل موجّهاته. عندما يعلم الموجّه 1d بأن الشبكة الفرعية x يمكن الوصول لها من AS3 ومن ثمَّ من موجّه البوّابة 1c، عندئذ يقرّر من المعلومات المتوفرة من قِبَل بروتوكول التوجيه بين ASs واجهة الموجّه التي على مسار أدنى كلفة من الموجّه 1d إلى موجّه البوّابة 1c. افترض أن هذه الواجهة هي I. يمكن أن يضيف الموجّه 1d المدخل (x, I) إلى جدول التمرير لديه. (هذا المثال وأمثلة أخرى قدمت في هذا الجزء تبين الفكرة العامّة لكنها تبسيط لما يحدث فعلاً في الإنترنت. في الجزء

التالي سنزوّدك بوصف أكثر تفصيلاً ولو أنه أكثر تعقيداً عندما نناقش بروتوكول BGP).

وتعقيباً على المثال السابق افترض الآن أن AS2 وAS3 متصلان بـ ASs أخرى غير موضحة في الشكل. افترض أيضاً أن AS1 علِم من بروتوكول التوجيه بين ASs أن الشبكة الفرعية x يمكن الوصول لها من AS2 (عن طريق موجّه البوّابة 1b) ومن AS3 (عن طريق موجّه البوّابة 1c). عندئذ سيذيع AS1 هذه المعلومات إلى كل الموجّهات بداخله بما في ذلك 1d. ولكي ينشئ الموجّه 1d جدول التمرير لديه عليه أن يحدد مَنْ من الموجهين 1b أو 1c يجب أن يوجّه الرزم المتجهة إلى x. أحد الطرق التي غالباً ما تُستخدَم عملياً تتبع أسلوب توجيه البطاطس الساخنة (hot-potato routing). في هذه الطريقة يتخلص AS من الرزمة (البطاطس الساخنة) بأسرع ما يمكن (بدقة أكثر بأرخص ما يمكن). يتم ذلك بجعل الموجّه يرسل الرزمة إلى موجّه البوّابة الذي يعتبر المسار إليه له أدنى كلفة ويتوفر لديه مسار إلى الوجهة المطلوبة. في سياق المثال الحالي يستخدم الموجّه 1d توجيه البطاطس الساخنة والذي يستعمل معلومات من بروتوكول التوجيه داخل AS لتحديد كلفة المسارات إلى 1b و1c، وحينئذ سيختار المسار الذي له أدنى كلفة. بمجرد اختيار هذا المسار يضيف الموجّه 1d مدخلاً جديداً للشبكة الفرعية x في جدول التمرير لديه. يلخّص الشكل 4-33 ما يفعله الموجّه 1d لإضافة المدخل الجديد لـ x إلى جدول التمرير.



الشكل 4-33 خطوات إضافة وجهة خارج AS في جدول تمرير الموجّه.

عندما يعلم AS عن وجهة من نظام AS مجاور يمكن أن يعلن AS معلومات التوجيه تلك لبعض ASs أخرى مجاورة له. على سبيل المثال افترض أن AS1 يعلم من AS2 أن الشبكة الفرعية x يمكن الوصول لها عن طريق AS2. يمكن لـ AS1 أن يخبر AS3 أنه يمكن الوصول لـ x عن طريق AS1. بهذه الطريقة إذا كانت AS3 تحتاج لتوجيه رزمة إلى x فإن AS3 سيرسل الرزمة إلى AS1 والذي بدوره يرسلها إلى AS2. كما سنرى في مناقشتنا لبروتوكول BGP أن لـ AS مرونة في تحديد أيٍّ من الوجّهات سيعلن عنها لنظم ASs المجاورة. هذا قرار تمليه سياسة التشغيل المتبعة، ويعتمد عادةً على القضايا الاقتصادية أكثر من اعتماده على القضايا التقنية.

تذكّر من الجزء 1-5 أن الإنترنت تتكون من مزوّدي خدمة الإنترنت الموزعين في تركيب هرمي. إذاً ما العلاقة بين موفري خدمة الإنترنت وASs؟ قد تعتقد بأن الموجّهات التابعة لموفر خدمة إنترنت والوصلات التي تربط بينها تُعتبر بمثابة نظام AS واحد. بالرغم من أن الأمر يكون كذلك في أغلب الأحيان، إلا أن العديد من موفري خدمة الإنترنت يقسمون شبكتهم إلى عدة ASs. على سبيل المثال بعض موفري خدمة الإنترنت من الطبقة الأولى (tier-1) يصممون شبكتهم بكاملها كنظام AS واحد، بينما يقسم البعض الآخر شبكته إلى عشرات من أنظمة ASs المتصلة ببعضها.

الخلاصة أنه يتم حل مشاكل حجم الشبكة والسلطة الإدارية بتعريف الأنظمة المستقلة ذاتياً. تستخدم كل الموجّهات الموجودة داخل AS نفس بروتوكول التوجيه داخله، في حين تستخدم كل أنظمة ASs المختلفة نفس بروتوكول التوجيه فيما بينها. تُحل مشكلة حجم الشبكة بهذه الطريقة لأن بروتوكول التوجيه داخل AS يحتاج فقط لمعرفة الموجّهات داخل AS. وتحل مشكلة السلطة الإدارية لأن أي منظمة يمكن أن تستخدم أي بروتوكول توجيه تختاره داخل نظام AS الخاص بها؛ لكن كل زوج من ASs المتصلة ببعضهما يحتاج لتشغيل نفس بروتوكول التوجيه بين ASs لتبادل معلومات الوصول (reachability).

في الجزء التالي سنتناول ثلاثة من البروتوكولات المستخدمة في الإنترنت اليوم: اثنين للتوجيه داخل AS (RIP وOSPF) وبروتوكولاً للتوجيه بين ASs (BGP). بدراسات الحالة تلك ستكتمل دراستنا للتوجيه الهرمي بشكلٍ جيد.

6-4 التوجيه في الإنترنت

بعد أن درسنا عناوين الإنترنت وبروتوكول IP، نحوّل انتباهنا الآن إلى بروتوكولات التوجيه في الإنترنت (وهي المسؤولة عن تحديد المسارات التي تأخذها وحدات البيانات من المصدر إلى الوجهة). سنرى أن بروتوكولات التوجيه في الإنترنت تجسّد العديد من المبادئ التي تعلّمناها في وقت سابق في هذا الفصل. إن طرق حالة الوصلة ومتجه المسافة التي درسناها في الأجزاء 4-5-1 و4-5-2 وفكرة الأنظمة المستقلة ذاتياً في الجزء 4-5-3 تمثل جميعاً محاور مركزية للتوجيه في الإنترنت اليوم.

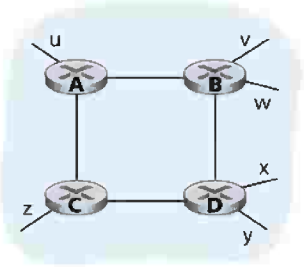
تذكّر من الجزء 4-5-3 أن النظام المستقل ذاتياً (AS) هو مجموعة من الموجّهات تحت نفس السلطة الإدارية والتقنية، وتستخدم جميعها نفس بروتوكول التوجيه فيما بينها. وبدوره يحتوي كل AS عادة على عدة شبكات فرعية (راجع الاستعمال الدقيق لتعبير شبكة فرعية مع العنونة في الجزء 4-4-2).

4-6-1 التوجيه داخل نظام AS في الإنترنت: بروتوكول RIP

يُستخدم بروتوكول التوجيه داخل AS لتحديد المسارات داخل نظام مستقل ذاتيا (AS). تعرف بروتوكولات التوجيه داخل AS أيضاً ببروتوكولات البوّابة الداخلية (interior gateway protocols). من الناحية التاريخية اُستخدِم اثنان من بروتوكولات التوجيه على نطاق واسع للتوجيه ضمن نظام مستقل ذاتياً في الإنترنت: بروتوكول معلومات التوجيه RIP (Routing Information Protocol) وبروتوكول المسار الأقصر أولاً المفتوح OSPF (Open Shortest Path First). وهناك بروتوكول توجيه آخر وثيق الصلة بـ OSPF يعرف بـ IS-IS [RFC 1142; Perlman 1999]. سنناقش أولاً RIP وبعد ذلك OSPF.

يُعتبر RIP أحد البروتوكولات الأولى للتوجيه داخل AS في الإنترنت وما زال واسع الانتشار اليوم. ترجع أصوله واسمه إلى البنية المعمارية لأنظمة شبكة Xerox (XNS)، ويعزى استخدامه على نطاق واسع الانتشار بدرجة كبيرة إلى إدراجه ضمن نسخة BSD لنظام التشغيل يونيكس UNIX بدعم لبروتوكولات TCP/IP والتي ظهرت عام 1982. تم تعريف النسخة 1 من بروتوكول RIP في [RFC 1058]، والنسخة 2 بتوافق خلفي في [RFC 2453].

يُعد RIP بروتوكول متجه المسافة ويعمل بأسلوب قريب جداً من الحالة المثالية لبروتوكول DV الذي فحصناه في الجزء 4-5-2. تُستخدم نسخة RIP المعرَّفة في RFC 1058 عدد القفزات كمعيار للكلفة؛ أي أن كل وصلة لها كلفة تساوي 1. في خوارزمية DV في الجزء 4-5-2 وللتبسيط كنا قد عرّفنا الكلفة بين أزواج الموجّهات. في RIP (وأيضاً في OSPF) تعرّف الكلفة في الحقيقة من موجّه المصدر إلى شبكة الوجهة الفرعية. يستخدم RIP المصطلح قفزة (hop) للإشارة إلى عدد الشبكات الفرعية التي يتم عبورها على طول المسار الأقصر من موجّه المصدر إلى شبكة الوجهة الفرعية، بما في ذلك شبكة الوجهة الفرعية. يوضح الشكل 4-34 نظام AS بست شبكات فرعية طرفية. يبين الجدول الموجود بالشكل عدد القفزات من المصدر A إلى كلٍّ من الشبكات الفرعية الطرفية.

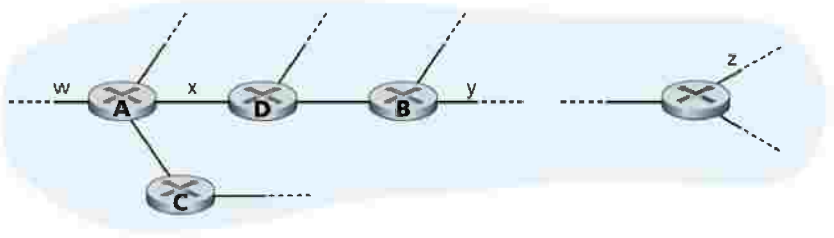


الواجهة	عدد القفزات
u	1
v	2
w	2
x	3
y	3
z	2

الشكل 4-34 عدد القفزات من موجّه المصدر A إلى الشبكات الفرعية المختلفة.

تم تحديد الكلفة القصوى لمسارٍ ما بالعدد 15، وبالتالي يمكن استخدام RIP فقط للأنظمة المستقلة ذاتياً التي لا يزيد طول قطرها عن 15 قفزة. تذكّر أن الموجّهات المتجاورة في بروتوكولات DV تتبادل متجهات المسافة مع بعضها البعض. يمثل متجه المسافة لأيٍّ من تلك الموجّهات التقدير الحالي لأطوال المسارات الأقصر بين ذلك الموجّه والشبكات الفرعية في AS. يتم تبادل رسائل تحديث التوجيه في RIP بين الجيران كل 30 ثانية تقريباً عن طريق رسالة ردّ RIP. تحتوي رسالة الردّ التي يرسلها موجّه أو مضيف قائمةً لا يزيد طولها عن 25 شبكة وجهة فرعية ضمن AS، بالإضافة إلى المسافة من المُرسِل إلى كلٍّ من تلك الشبكات الفرعية. تعرف رسائل الرد أيضاً بإعلانات RIP (RIP advertisements).

لنلقِ نظرةً على مثال بسيط لكيفية عمل إعلانات RIP. خذ في الاعتبار جزءاً من نظام AS كالموضح في الشكل 4-35. في هذا الشكل تدل الخطوط التي توصل بين الموجّهات على الشبكات الفرعية. فقط تم تسمية بعض الموجّهات المختارة (A، B، C، D) والشبكات الفرعية (w ، x, y ، z). تشير الخطوط المنقطة بأن AS ممتد؛ وهكذا يتكون هذا النظام المستقل ذاتياً من المزيد من الموجّهات والوصلات بالإضافة إلى تلك الموضحة بالشكل.



الشكل 4-35 جزء من نظام مستقل ذاتياً.

الشبكة الفرعية للوجهة	الموجّه التالي	عدد القفزات
w	A	2
y	B	2
z	B	7
x	—	1
...	...	...

الشكل 4-36 جدول التوجيه في الموجّه D قبل استلام الإعلان من الموجّه A.

يحتفظ كل موجّه بجدول RIP يُعرَف باسم جدول التوجيه. يشتمل جدول التوجيه لدى الموجّه على كلٍّ من متجه المسافة للموجّه وجدول التمرير للموجّه. يوضح الشكل 4-36 جدول التوجيه للموجّه D. لاحظ أن جدول التوجيه يتكون من ثلاثة أعمدة. يشير العمود الأول إلى شبكة الوجهة الفرعية، ويشير العمود الثاني إلى عنوان الموجّه التالي على طول المسار الأقصر إلى شبكة الوجهة الفرعية، ويشير العمود الثالث إلى عدد القفزات (أي عدد الشبكات الفرعية التي يجب أن تُعبر بما في ذلك شبكة الوجهة الفرعية) للوصول إلى شبكة الوجهة الفرعية على طول المسار الأقصر. في هذا المثال يبين الجدول أنه لكي ترسل رزمة بيانات من الموجّه D إلى شبكة الوجهة الفرعية *w*، يجب أن ترسل رزمة البيانات أولاً إلى الموجّه المجاور A؛ وأن شبكة الوجهة الفرعية *w* تبعد مسافة قفزتين على طول المسار الأقصر. بنفس الطريقة يبين الجدول أن الشبكة الفرعية *z* تبعد مسافة 7 قفزات عن طريق الموجّه B. من حيث المبدأ سيحتوي جدول التوجيه على صف واحد لكل شبكة فرعية في نظام AS ، رغم أن النسخة 2 من RIP تسمح بتجميع (تكتيل) مدخلات الشبكة الفرعية باستعمال تقنيات تشبه تقنيات تجميع المسارات التي تناولناها في الجزء 4-4. الجدول في الشكل 4-36 والجداول التي ستأتي لاحقاً هي جداول جزئية فقط.

افترض الآن أنه بعد 30 ثانية يستلم الموجّه D من الموجّه A الإعلان المبين في الشكل 4-37. لاحظ أن هذا الإعلان ما هو إلا معلومات جدول التوجيه من الموجّه A! تشير تلك المعلومات بشكلٍ خاص إلى أن الشبكة الفرعية z تبعد فقط أربع

قفزات عن الموجّه A. فور استلام الموجّه D لهذا الإعلان يدمج الإعلان (الشكل 4-37) بجدول التوجيه القديم (الشكل 4-36). وبشكل خاص يعرف الموجّه D أن هناك الآن مسار من خلال الموجّه A إلى الشبكة الفرعية z أقصر من المسار من خلال الموجّه B. وبالتالي يُحدّث الموجّه D جدول التوجيه لديه ليشمل المسار الأقصر كما هو موضح في الشكل 4-38. قد تتساءل: وكيف يصبح المسار الأقصر إلى الشبكة الفرعية z أقصر؟ من المحتمل أن خوارزمية متجه المسافة اللامركزية ما زالت في عملية التقارب (انظر الجزء 4-5-2)، أو ربما قد أضيفت وصلات جديدة أو موجّهات جديدة أو كلاهما إلى AS، ومن ثمَّ تغيّر المسار الأقصر في AS.

الشبكة الفرعية للوجهة	الموجّه التالي	عدد القفزات
z	C	4
w	—	1
x	—	1
…	…	…

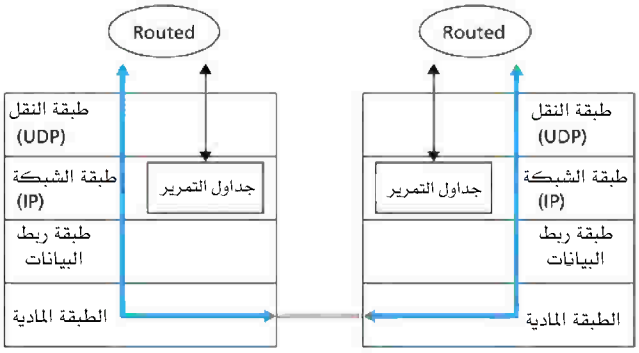
الشكل 4-37 الإعلان من الموجّه A.

الشبكة الفرعية للوجهة	الموجّه التالي	عدد القفزات
w	A	2
y	B	2
z	A	5
…	…	…

الشكل 4-38 جدول التوجيه في الموجه D بعد استلام الإعلان من الموجّه A.

دعنا نستعرض بضع سمات تطبيقية لبروتوكول RIP. تذكر أن موجّهات RIP تتبادل الإعلانات كل 30 ثانية تقريباً. إذا لم يسمع الموجّه من أحد جيرانه على الأقل مرة كل 180 ثانية فسيعتبر أن ذلك الجار لم يعد يستطيع الوصول إليه؛ أي أن ذلك الجار قد مات أو أن الوصلة بينهما قد انقطعت. عندما يحدث ذلك يعدّل بروتوكول RIP جدول التوجيه المحلي وبعد ذلك يذيع تلك المعلومات بإرسال الإعلانات إلى الموجّهات المجاورة (تلك التي ما زال في الإمكان الوصول إليها). يمكن أن يطلب موجّه أيضاً معلومات حول كلفة أحد الجيران للوصول لوجهة معينة باستخدام رسالة طلب RIP. ترسل الموجهات رسائل طلب RIP ورسائل ردّ RIP إلى بعضها البعض باستخدام منفذ UDP رقم 520. تُنقل قطعة UDP بين الموجّهات في رزمة بيانات IP قياسية. في الحقيقة إن RIP يستخدم بروتوكول طبقة النقل UDP فوق بروتوكول طبقة الشبكة IP لتنفيذ وظيفة طبقة الشبكة (خوارزمية التوجيه) قد يبدو ملتوياً بعض الشيء (وهو فعلاً كذلك!). ولو نظرنا بعمق أكثر إلى كيفية تحقيق RIP لاتضح لنا هذا اللبس.

يبين الشكل 4-39 كيف يُطبّق RIP عادةً في نظام التشغيل يونيكس (UNIX) (على سبيل المثال في محطة عمل يونيكس تعمل كموجّه). تُنفذ عملية تسمى routed (وتنطق روت - دي) بروتوكول RIP، أي تحتفظ بمعلومات التوجيه وتتبادل الرسائل مع عمليات التوجيه التي تعمل في الموجّهات المجاورة. نظراً لأن بروتوكول RIP يُنفّذ كعملية في طبقة التطبيقات (ولو أنها عملية خاصّة جداً حيث لديها القدرة على معالجة جداول التوجيه بداخل لب اليونيكس UNIX kernel)، يكون بوسع البروتوكول إرسال وتلقّي رسائل على مقبس قياسي وكذلك استخدام بروتوكول نقل قياسي. كما وضحنا يُنفّذ RIP كبروتوكول طبقة تطبيقات (انظر الفصل الثاني) يعمل فوق UDP.



الشكل 4-39 تطبيق RIP كعملية routed daemon في نظام التشغيل يونيكس.

4-6-2 التوجيه داخل AS في الإنترنت: بروتوكول OSPF

كما هو الحال مع بروتوكول RIP، يُستخدم بروتوكول OSPF على نطاق واسع للتوجيه في الإنترنت داخل أنظمة AS. يستعمل OSPF وبروتوكول آخر وثيق الصلة به (IS-IS) عادةً في شبكات موفري خدمة الإنترنت من الطبقة العليا (upper-tier)، بينما يستعمل RIP في شبكات موفري خدمة الإنترنت من الطبقة الدنيا (lower-tier) وشبكات الشركات. تدل كلمة open في اسم البروتوكول على أنه متوفر للجميع (ملكية عامة) (في مقابل الملكية الخاصة كما في حالة بروتوكول سيسكو EIGRP). تُعتبر النسخة 2 هي أحدث نسخة من OSPF وهي معرّفة في RFC 2328.

يعتبر OSPF أيضاً بمثابة الوريث لبروتوكول RIP ولذا فإنه يمتاز بعدة ميزات متقدّمة. ومع ذلك يعتبر OSPF في صميمه بروتوكول حالة الوصلة LS الذي يستعمل فيضان معلومات حالة الوصلة وخوارزمية Dijkstra لحساب المسارات الأدنى كلفة. باستخدام بروتوكول OSPF يبني الموجّه خريطة طوبوغرافية كاملة (أي رسماً بيانياً) للنظام المستقل ذاتياً AS بكامله. بعد ذلك يشغّل الموجّه خوارزمية Dijkstra محلياً لحساب شجرة أقصر المسارات إلى كل الشبكات الفرعية والتي يمثل الموجّه نفسه عقدة الجذر لها. يتم إعداد كلفة كل وصلة منفردة من قبل المشرف على الشبكة (انظر المبادئ والواقع العملي: إعداد أوزان الوصلات في بروتوكول

OSPF). قد يختار مشرف الشبكة قيمة الكلفة 1 لكل وصلة، ومن ثم ينجز توجيه أدنى القفزات، أو قد يختار تحديد أوزان الوصلة بطريقة تتناسب عكسياً مع سعة الوصلة لكي يقلل من مرور البيانات خلال الوصلات ذات سعة الإرسال المنخفضة. لا يشترط OSPF سياسة معينة لتحديد أوزان الوصلات (فتلك مهمة المشرف على الشبكة)، لكن بدلاً من ذلك يوفر OSPF الآليات (على شكل بروتوكول) لحساب المسارات الأدنى كلفة لمجموعة أوزان معطاة.

في بروتوكول OSPF يذيع الموجّه معلومات التوجيه لدعوة الموجّهات الأخرى في النظام المستقل ذاتياً (وليست الموجّهات المجاورة فقط). يذيع الموجّه معلومات حالة الوصلة حينما يكون هناك تغيّر في حالة وصلة (مثلاً تغيّر في الكلفة أو تغيّر في الحالة من كونها متعطّلة إلى شغّالة والعكس). أيضاً يذيع الموجّه حالة الوصلة بشكلٍ دوري (على الأقل مرة كل 30 دقيقة) حتى إذا لم يحدث تغيّر في حالة الوصلة. ينص RFC 2328 على أن "هذا التجديد الدوري بإعلانات حالة الوصلة يضيف متانة (موثوقية) لخوارزمية حالة الوصلة". توضع إعلانات OSPF في رسائل OSPF والتي تُنقل مباشرة من قِبَل IP حيث يستخدم الرقم 89 (أي OSPF) لتمثيل بروتوكول الطبقة الأعلى. وبالتالي يجب أن يُنفّذ بروتوكول OSPF نفسه الوظائف المختلفة كنقل الرسالة الموثوق فيه وإذاعة حالة الوصلة. يتأكد OSPF أيضاً من أن حالة الوصلة شغّالة (عن طريق رسالة HELLO "مرحباً" التي تُرسَل إلى أحد الجيران المرتبطين) وتسمح لموجّه OSPF بالحصول على قاعدة بيانات أحد الموجّهات المجاورة والتي تتضمن حالة الوصلات في كافة أنحاء الشبكة.

يتضمّن بروتوكول OSPF عدة خصائص متطورة من بينها:

- الأمن: يمكن توثيق تبادل المعلومات بين موجّهات OSPF (مثلاً لتحديث حالة الوصلة). ويسمح التوثيق للموجّهات الموثوق فيها فقط بالمشاركة ضمن بروتوكول OSPF داخل أنظمة AS وهكذا يُمنَع المتطفلون المؤذيون (أو طلاب الشبكات الذين يجربون ما تعلموه حديثاً للتسلية) من حقن معلومات غير صحيحة في جداول التوجيه. وبشكلٍ اعتيادي لا يستخدم OSPF التوثيق للتحقّق من رزم OSPF التي تنتقل بين الموجّهات ولذا يمكن تزييفها. يمكن

استخدام نوعين من التوثيق: بسيط وMD5 (انظر الفصل الثامن لمناقشة MD5 والتوثيق بصفة عامة). يعني التوثيق البسيط استخدام نفس كلمة السر على كل الموجّهات. فعندما يرسل موجّه OSPF رزمة يُضَمّن كلمة السر في النص الأصلي (plaintext). من الواضح أن التوثيق البسيط غير آمن تماماً. يعتمد MD5 على المفاتيح السرية المشتركة التي يتم إعدادها في كل الموجّهات. يحسب الموجّه ملخص MD5 (MD5 hash) لكل رزمة OSPF يرسلها بناءً على محتوى الرزمة والمفتاح السري الذي يذيلها (انظر مناقشة أكواد توثيق الرسائل في الفصل السابع). بعد ذلك يضع الموجّه قيمة الملخص الناتج ضمن رزمة OSPF. يستخدم موجّه الاستقبال المفتاح السري المعد من قبل ويحسب ملخص MD5 من الرزمة ويقارنه بقيمة ملخص MD5 الذي تحمله الرزمة، وهكذا يتحقّق من مصداقية الرزمة. تستخدم أيضاً الأرقام التسلسلية مع MD5 لتوفير الحماية ضدّ هجوم إعادة التشغيل (replay attack).

- استخدام مسارات متعدّدة بنفس الكلفة: عندما تكون هناك مسارات متعدّدة لها نفس الكلفة إلى وجهة ما يسمح OSPF باستعمال عدة مسارات (بمعنى أنه ليس من الضروري أن يختار مساراً وحيداً لنقل كل حركة مرور البيانات عندما توجد عدة مسارات ذات كلفة متساوية).
- الدعم المتكامل لتوجيه الإرسال الفردي والمتعدّد: يوفر بروتوكول MOSPF للإرسال المتعدد (multicast OSPF) [RFC 1584] امتدادات بسيطة لبروتوكول OSPF لتوجيه الإرسال المتعدد (سنغطّي هذا الموضوع بعمق أكثر في الجزء 4-7-2). يستخدم MOSPF قاعدة بيانات الوصلات الخاصة بـ OSPF ويضيف نوعاً جديداً من إعلان حالة الوصلة لآلية إذاعة حالة الوصلة في OSPF.
- دعم التوجيه الهرمي داخل نفس نطاق التوجيه: لعل أهم جوانب التقدم التي حققها بروتوكول OSPF هي قدرته على تنظيم نظامٍ مستقلٍ ذاتياً بشكلٍ هرمي. رأينا في الجزء 4-5-3 العديد من المزايا لتراكيب التوجيه الهرمي. سنغطّي تطبيق التوجيه الهرمي لـ OSPF في بقيّة هذا الجزء.

باستخدام OSPF يمكن تقسيم النظام المستقل ذاتياً إلى مناطق (areas). تستخدم كل منطقة خوارزمية توجيه حالة الوصلة الخاصة بها، ويذيع كل موجّه في منطقة حالة وصلاته إلى باقي الموجّهات في تلك المنطقة. بهذه الطريقة تبقى التفاصيل الداخلية لمنطقةٍ ما مخفية عن كل الموجّهات خارج تلك المنطقة. يتضمّن التوجيه داخل المنطقة (intra-area routing) فقط الموجّهات داخل المنطقة نفسها.

وضمن كل منطقة يضطلع واحدٌ أو أكثر من الموجّهات الموجودة على حدود المنطقة بمهمة توجيه الرزم خارج المنطقة. يتم تهيئة منطقة واحدة فقط من المناطق لكي تكون منطقة العمود الفقري. ويكون الدور الأساسي لمنطقة العمود الفقري هو توجيه مرور البيانات بين المناطق الأخرى في نظام AS. يحتوي العمود الفقري دائماً على موجّهات الحدود للمناطق في AS بالإضافة إلى احتمال وجود موجّهات أخرى غير حدودية.

يتطلّب التوجيه بين المناطق داخل AS توجيه الرزمة أولاً إلى موجّهٍ موجود على حدود منطقة المصدر (توجيه داخل منطقة)، ثمّ توجيهها عبر العمود الفقري إلى موجّه حدود المنطقة التي فيها الوجهة، ومن ثم توجّه داخل تلك المنطقة للوجهة النهائية.

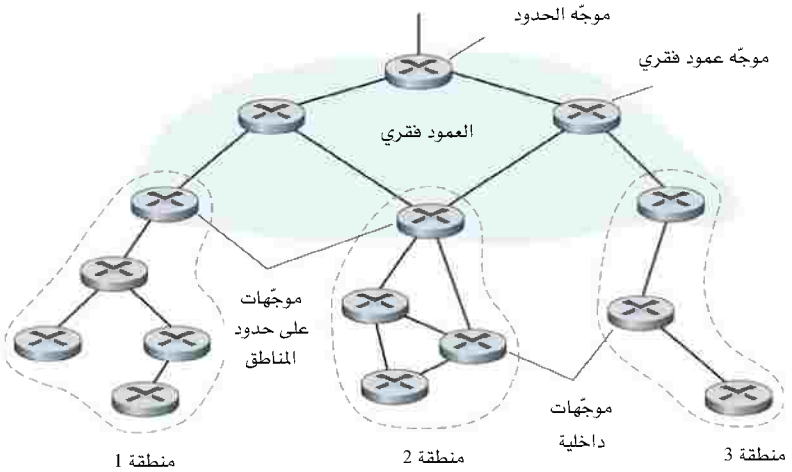
يوضح الشكل 4-40 مخططاً لتركيب هرمي لشبكة OSPF. يمكن أن نميّز أربعة أنواع من موجّهات OSPF في الشكل:

- موجّهات داخلية (internal routers): تقع تلك الموجّهات في مناطق غير العمود الفقري وتؤدّي مهمة التوجيه داخل AS فقط.
- موجّهات حدود منطقة (area border routers): تنتمي تلك الموجّهات إلى أحد المناطق بالإضافة إلى العمود الفقري.
- موجّهات عمود فقري (موجّهات غير حدودية) (backbone routers): تؤدّي تلك الموجّهات مهمة التوجيه داخل العمود الفقري، لكنها ليست في حد ذاتها موجّهات حدود منطقة. داخل منطقة غير العمود الفقري، تعرف الموجّهات الداخلية بوجود مسارات إلى المناطق الأخرى من المعلومات المذاعة داخل المنطقة من موجّهات العمود الفقري بها (بشكلٍ جوهري من إعلانات حالة

الوصلة، لكنها تعلن كلفة المسار إلى المنطقة الأخرى بدلاً من كلفة الوصلة).

- موجّهات حدود AS (boundary routers): تتبادل الموجّهات على حدود نظام AS معلومات التوجيه مع الموجّهات في الأنظمة الأخرى المستقلة ذاتياً. قد تستخدم تلك الموجّهات مثلاً بروتوكول BGP لعملية التوجيه بين الأنظمة المستقلة ذاتياً. من خلال مثل تلك الموجّهات تتعلّم الموجّهات الأخرى المسارات إلى الشبكات الخارجية.

يعتبر OSPF بروتوكولاً معقّداً نسبياً، وتغطيتنا له هنا كانت بالضرورة مقتضبة؛ يمكنك الاطلاع على تفاصيل إضافية في [Huitema 1998; Moy 1998; RFC 2328].



الشكل 4-40 تنظيم هرمي لنظام مستقل ذاتياً AS يتكون من أربع مناطق باستخدام بروتوكول OSPF.

المبادئ في الواقع العملي (Principles in Practice)

إعداد أوزان الوصلات في بروتوكول OSPF

افترضت مناقشتنا لتوجيه حالة الوصلة ضمنياً تهيئة أوزان الوصلة، وتشغيل خوارزمية توجيه مثل OSPF، وكذلك توجيه مرور البيانات طبقاً لجداول التوجيه المحسوبة بخوارزمية LS. من منظور السبب والتأثير، يتم إدخال أوزان الوصلات (بمعنى آخر تأتي بالمرتبة الأولى) وتنتج (عن طريق خوارزمية Dijkstra) مسارات التوجيه التي تقلّل الكلفة الإجمالية. من وجهة النظر هذه تعكس أوزان الوصلات كلفة استعمالها ويؤدي استخدام خوارزمية Disjkstra إلى تقليل الكلفة الإجمالية (وكمثال على ذلك إذا كان وزن الوصلة يتناسب عكسياً مع سعتها، فسيكون للوصلات ذات السعة العالية أوزان أصغر وبالتالي تصبح أكثر جاذبية من وجهة نظر التوجيه).

في الواقع العملي قد تُعكس علاقة السبب والتأثير بين أوزان الوصلات ومسارات التوجيه؛ بمعنى أنه قد يهيئ مشغلو الشبكة أوزان الوصلات للحصول على مسارات توجيه تحقق أهدافاً معينة لهندسة حركة المرور [Fortz 2000; Fortz 2002] كتوزيع الأحمال بشكل أفضل. افترض على سبيل المثال أن مشغل الشبكة يتوافر لديه تقديرٌ ما لتدفق حركة مرور البيانات التي تدخل الشبكة في كل نقطة دخول (ingress point) متجهة إلى كل نقطة خروج (egress point). عندئذٍ قد يريد المشغل تنفيذ خطة توجيه بعينها للتدفقات من نقاط الدخول إلى نقاط الخروج بحيث تقلل الاستخدام الأقصى للانتفاع بكل وصلات الشبكة. لكن مع خوارزمية توجيه مثل OSPF تُعتبر أوزان الوصلات بمثابة مقابض التحكم الرئيسية التي يقوم المشغل عن طريقها بضبط توجيه التدفق خلال الشبكة. وهكذا فللوصول لهدف تقليل الاستغلال الأقصى للوصلات لتوزيع أفضل للأحمال، يجب أن يوجد مُشغِّل الشبكة مجموعة أوزان الوصلات التي تحقق هذا الهدف. هذا عكس علاقة السبب والتأثير، حيث يكون توجيه التدفق معروفاً، ويكون المطلوب إيجاد أوزان الوصلات بحيث تؤدي خوارزمية التوجيه إلى ذلك التوجيه المطلوب للتدفق.

3-6-4 التوجيه بين أنظمة AS: بروتوكول BGP

عرفنا الآن كيف يستخدم موفرو خدمة الإنترنت بروتوكولات RIP وOSPF لتحديد المسارات المثلى بين أزواج المصادر والوجهات التي تقع ضمن نظام AS بعينه. دعنا نتناول الآن كيفية تحديد المسارات بين أزواج المصدر والوجهة التي تمر عبر عدة أنظمة AS. تعد النسخة 4 من بروتوكول BGP والموصوفة في RFC 4271 المعيار الواقعي de facto (انظر أيضاً [RFC 1772; RFC 1773]) في الإنترنت اليوم، ومن

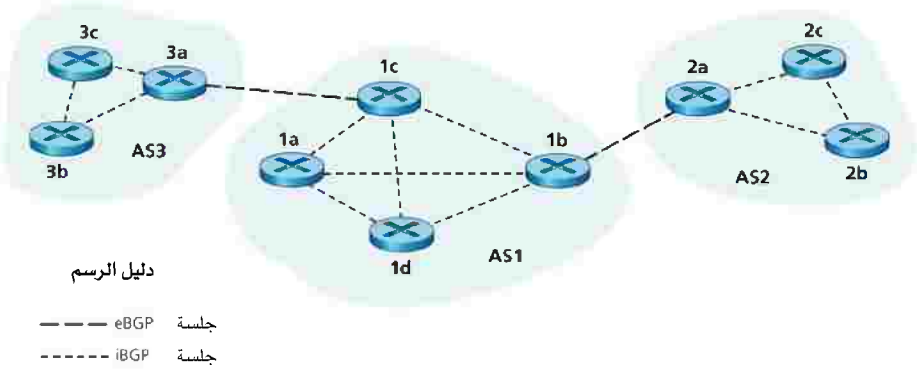
الشائع تسميته أيضاً باسم BGP4 أو ببساطة BGP. وكبروتوكول للتوجيه خارج AS (انظر الجزء 4-5-3)، فإنه يزوّد كل نظام AS بوسيلة لـ:

1. الحصول على معلومات الوصول للشبكات الفرعية من أنظمة AS المجاورة.
2. بث معلومات الوصول إلى كل الموجّهات داخل AS.
3. تحديد مسارات "جيدة" إلى الشبكات الفرعية طبقاً لمعلومات الوصول ولسياسة AS.

وبشكلٍ أكثر أهمية يسمح BGP لكل شبكة فرعية بالإعلان عن وجودها لبقيّة الإنترنت. تصرخ الشبكة الفرعية "أنا موجودة وأنا هنا"، ويتأكّد BGP من أن جميع أنظمة AS في الإنترنت تعلم بوجود الشبكة الفرعية وتعرف كيف تصل إليها. باختصار لولا بروتوكول BGP لأصبحت كل شبكة فرعية معزولة لوحدها ومجهولة لدى بقيّة الإنترنت.

أساسيات بروتوكول BGP

بروتوكول BGP معقّدٌ جداً؛ ولقد خصصت كتب بأكملها لهذا الموضوع وما زال العديد من القضايا غير مفهومة بشكلٍ واضح [Yannuzzi 2005]. علاوةً على ذلك وحتى بعد أن تقرأ الكتب وRFCs، قد تجد من الصعوبة أن تجيد BGP بالكامل بدون ممارسة لعدّة شهور (إن لم تكن سنوات) كمصمم أو كمشرف لشبكة موفر خدمة الإنترنت من الطبقات العليا. وعلى الرغم من ذلك ولكون BGP بروتوكولاً هاماً جداً للإنترنت (حيث يعتبر أساساً بمثابة الصمغ الذي يربط ما بين جميع أجزاء الشبكة)، فإننا نحتاج على الأقل لفهمٍ أولي لطريقة عمله. نبدأ بوصف كيفية عمل BGP ضمن سياق شبكة المثال البسيطة التي درسناها في وقت سابق في الشكل 4-32. في هذا الوصف نبني على مناقشتنا للتوجيه الهرمي في الجزء 4-5-3؛ وننصحك بمراجعة ذلك الموضوع.



الشكل 4-41 جلسات eBGP وiBGP.

في BGP تتبادل أزواج من الموجهات معلومات توجيه على توصيلات TCP شبه دائمة (semi-permanent) من خلال المنفذ 179. يبين الشكل 4-41 توصيلات TCP شبه الدائمة للشبكة الموجودة بالشكل 4-32. هناك عادةً توصيلة BGP TCP واحدة من هذا النوع لكل وصلة تربط مباشرة بين موجّهين في نظامي AS مختلفين؛ وهكذا ففي الشكل 4-41 توجد توصيلة TCP بين موجّهات البوّابة 3a وlc وتوصيلة TCP أخرى بين موجّهات البوّابة 1b و2a. هناك أيضاً توصيلات TCP نصف دائمة لبروتوكول BGP بين الموجّهات داخل نظام AS . وبشكلٍ محدد يعرض الشكل 4-41 إعداداً شائع الاستخدام يتضمن توصيلة TCP واحدة لكل زوج من الموجّهات التي بداخل AS مما يُشكِّل شبكة ربط من توصيلات TCP داخل كل AS. لكل توصيلة TCP يُطلق على الموجّهين في نهاية التوصيلة نظائر BGP، وتدعى توصيلة TCP مع كل رسائل BGP المرسلة عبرها "جلسة BGP". علاوةً على ذلك تُسمى جلسة BGP التي تغطّي اثنين من أنظمة AS جلسة BGP خارجية (eBGP)، وتُسمى جلسة BGP بين الموجهات في نفس AS جلسة BGP داخلية (iBGP). في الشكل 4-41 توضح جلسات eBGP بخطوط متقطعة بشُرَط طويلة؛ وجلسات iBGP بخطوط متقطعة بشُرَط قصيرة. لاحظ أن خطوط جلسات BGP في الشكل 4-41 لا تناظر بالضرورة الوصلات المادية في الشكل 4-32.

يسمح BGP لكل AS أن يحدد أي الوجهات يمكن الوصول إليها عن طريق أنظمة AS المجاورة له. في BGP هذه الوجهات ليست مضيفات ولكنها بادئات عناوين CIDR، حيث يمثل كلٌّ منها شبكة فرعية أو مجموعة شبكات فرعية. وهكذا فعلى سبيل المثال افترض أن هناك أربع شبكات فرعية متصلة بـ AS2: 138.16.64/24، 138.16.65/24، 138.16.66/24، 138.16.67/24. عندئذ يمكن أن يجمع AS2 البادئات لهذه الشبكات الفرعية الأربع ويستعمل BGP لإعلان بادئة واحدة 138.16.64/22 إلى AS1. وكمثال آخر افترض أن الشبكات الثلاث الأولى فقط من تلك الشبكات الفرعية الأربع موجودة في AS2 والشبكة الفرعية الرابعة 138.16.67/24 موجودة في AS3. وكما وصفنا في المبادئ والواقع العملي في الجزء 4-2-4، نظراً لأن الموجّهات تستخدم تطابق البادئة الأطول لتمرير رزم البيانات، يمكن أن يعلن AS3 لـ AS1 البادئة الأكثر تحديداً 138.16.67/24، ولايزال بوسع AS2 أن يعلن لـ AS1 البادئة المجمّعة 138.16.64/22.

دعنا نتناول الآن كيف يوزّع BGP معلومات الوصول للبادئات على جلسات BGP الموضحة في الشكل 4-41. كما قد تتوقّع، باستخدام جلسة eBGP بين موجهات البوّابة 3a و1c، يرسل AS3 إلى AS1 قائمة بالبادئات التي يمكن الوصول إليها من AS3؛ ويرسل AS1 إلى AS3 قائمة بالبادئات التي يمكن الوصول إليها من AS1. بنفس الطريقة يتبادل AS1 وAS2 معلومات الوصول خلال موجّهات البوّابة 1b و2a. أيضاً كما قد تتوقّع عندما يستلم موجّه بوّابة (في أي نظام AS) بادئات تم معرفتها عن طريق eBGP، يستخدم موجّه البوّابة جلسات iBGP لتوزيع البادئات إلى الموجّهات الأخرى في AS. وهكذا تتعرف كل الموجّهات في AS1 على بادئات AS3 بما في ذلك موجّه البوّابة 1b. يمكن أن يعيد موجّه البوّابة 1b (في AS1) إعلان بادئات AS3 إلى AS2. عندما يعرف موجّه (بوّابة أو غيره) بادئة جديدة فإنه ينشئ مدخلاً للبادئة في جدول التمرير لديه كما وصفنا في الجزء 4-5-3.

خواص المسار ومسارات BGP

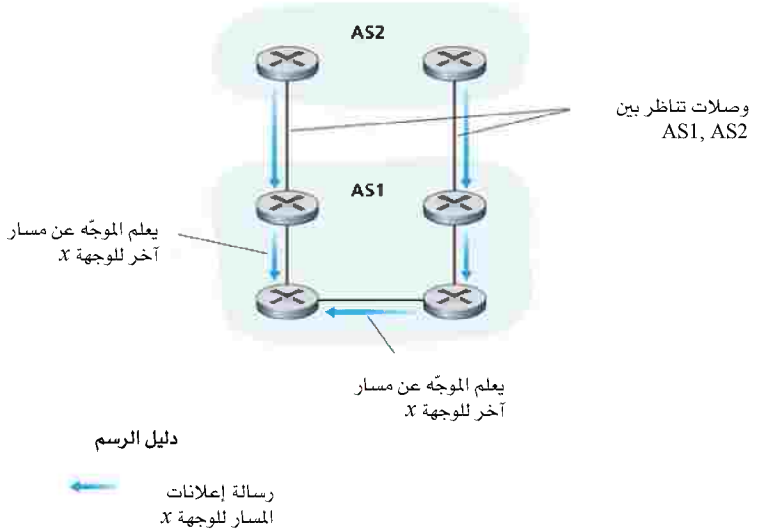
بعد هذا الفهم التمهيدي لـ BGP دعنا نغوص فيه بعمق بعض الشيء (مع مواصلة غض الطرف عن بعض التفاصيل الأقل أهمية). في BGP، يتم تمييز نظام مستقل ذاتياً (AS) برقم فريد عالمياً ((Autonomous System Number (ASN) [RFC 1930]. من الناحية الفنية ليس كل AS له رقم ASN. وبالتحديد فإن النظام الذي يطلق عليه نظام عقب stub AS (والذي يحمل فقط حركة مرور البيانات التي يكون هو مصدرها أو وجهتها) لن يكون له عادةً رقم ASN؛ وسنهمل هذا التفصيل في مناقشتنا لكي نتمكن من رؤية الغابة بدلاً من الأشجار. يتم تخصيص أرقام AS، مثلها في ذلك مثل عناوين IP، عن طريق مكاتب تسجيل ICANN الإقليمية [ICANN 2007].

عندما يعلن موجّه بادئة عبر جلسة BGP، يُضمّن مع البادئة عدداً من بارامترات BGP. في مفردات BGP يطلق على البادئة مع البارامترات الخاصة بها اسم "مسار" (route). وهكذا تعلن نظائر BGP المسارات لبعضها البعض. تُعتبر AS-PATH وNEXT-HOP من البارامترات الأكثر أهمية:

- بارامتر AS-PATH: يتضمن هذا البارامتر أنظمة AS التي مر عبرها إعلان البادئة. عندما تمر بادئة عبر نظام AS يضيف AS رقم ASN الخاص به إلى البارامتر AS-PATH. على سبيل المثال بالنظر إلى الشكل 4-41 وبافترض أن البادئة 138.16.64/24 أُعلنت أولاً من AS2 إلى AS1. إذا أعلن AS1 بعدئذٍ البادئة إلى AS3، فسيتضمن البارامتر AS-PATH AS2 AS1. تستخدم الموجّهات ذلك البارامتر لاكتشاف ومنع دوران الإعلانات في حلقات مفرغة؛ وبالتحديد إذا رأى موجّه أن نظام AS التي ينتمي له موجودٌ ضمن بارامتر AS-PATH فسيرفض الإعلان. كما سنناقش قريباً تستخدم الموجّهات AS-PATH أيضاً في الاختيار ما بين المسارات المتعدّدة إلى نفس البادئة.
- بارامتر NEXT-HOP: يلعب هذا البارامتر دوراً مهماً في الربط ما بين بروتوكولات التوجيه داخل نظام AS وبروتوكولات التوجيه خارجه. تمثل NEXT-HOP واجهة الموجّه الموجود في بداية AS-PATH. ولفهم هذا

البارامتر، دعنا نشير ثانية للشكل 4-41. لننظر ما يحدث عندما يعلن موجه البوّابة 3a الموجود في AS3 مساراً إلى موجّه البوّابة 1c في AS1 باستخدام eBGP. يتضمّن المسار البادئة المعلنة (والتي سنسميها x) والبارامتر AS-PATH للبادئة. يتضمّن هذا الإعلان أيضاً NEXT-HOP والذي يمثل عنوان IP لواجهة الموجّه 3a التي توصل إلى 1c. (تذكّر أن الموجّه له عدّة عناوين IP؛ واحد لكل واجهة من واجهاته) انظر الآن لما يحدث عندما يعرف الموجّه 1d عن المسار من iBGP. بعد العلم عن هذا المسار إلى x، قد يريد الموجّه 1d إرسال الرزم إلى x على طول المسار، أي قد يريد الموجّه 1d تضمين المدخل (x, l) في جدول التمرير لديه حيث تمثل l واجهته التي تبدأ المسار الأدنى كلفة من 1d نحو موجّه البوّابة 1c. ولتحديد l يزوّد 1d عنوان IP في خاصية NEXT-HOP للتوجيه داخل AS. لاحظ أن خوارزمية التوجيه داخل AS تحدد المسار الأدنى كلفةً إلى كل الشبكات الفرعية المتصلة بالموجّهات في AS1، ويتضمن ذلك الشبكة الفرعية للوصلة بين 1c و3a. من هذا المسار الأدنى كلفةً من 1d إلى الشبكة الفرعية 1c-3a يحدد 1d واجهته l التي تقع على بداية هذا المسار وبعد ذلك يضيف المدخل (x, l) إلى جدول التمرير لديه. الخلاصة: إنّ الموجّهات تستخدم البارامتر AS-PATH لتشكيل جداول التمرير لديها بشكلٍ صحيح.

يوضح الشكل 4-42 حالة أخرى نحتاج فيها إلى البارامتر AS-PATH. في هذا الشكل تتصل AS1 وAS2 بوصلتي نظير. يمكن أن يعرف موجّه في AS1 مسارين مختلفين إلى نفس البادئة x. يمكن أن يكون لهذين المسارين نفس البارامتر AS-PATH إلى x، لكن يمكن أن يكون لهما قيماً مختلفة للبارامتر NEXT-HOP تقابل وصلات النظير المختلفة. باستعمال قيم AS-PATH وخوارزمية التوجيه داخل AS، يمكن أن يحدد الموجّه كلفة المسار إلى كل وصلة نظير، وبعد ذلك يطبّق توجيه البطاطس الساخنة (انظر الجزء 4-5-3) لتحديد الواجهة الملائمة.



الشكل 4-42 تُستخدم بارامترات NEXT-HOP المتضمنة في الإعلانات لتحديد أي وصلة نظير سيتم استخدامها.

يتضمّن بروتوكول BGP أيضاً بارامترات تسمح للموجّهات بتخصيص معايير مفاضلة للمسارات وبارامتر يبين كيف تم إدخال البادئة إلى بروتوكول BGP في نظام AS المصدري. لمناقشة كاملة عن خواص المسارات اطّلِع على [Griffin 2002; Stewart 1999; Halabi 2000; Feamster 2004; RFC 4271].

عندما يستلم موجّه بوّابة إعلاناً من موجّه آخر فإنه يستعمل سياسته للاستيراد (import policy) لتقرير ما إذا كان سيقبل أو يستبعد المسار، وما إذا كان سيضع بعض البارامترات مثل معايير المفاضلة. قد تستبعد سياسة الاستيراد مساراً لأن AS لا يريد مرور البيانات على أحد أنظمة AS الموجودة في بارامتر AS-PATH للمسار. قد يستبعد موجّه البوّابة المسار أيضاً لأنه يعرف مساراً مفضّلاً إلى نفس البادئة.

اختيار مسار BGP

كما وصفنا في وقت سابق في هذا الجزء يستخدم BGP البروتوكول eBGP وiBGP لتوزيع المسارات إلى كل الموجّهات ضمن أنظمة AS. من هذا التوزيع قد

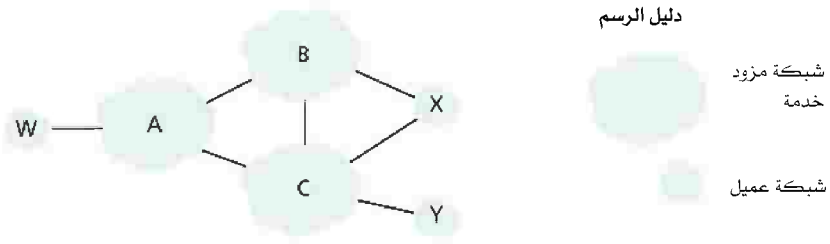
يتعرف موجّه على أكثر من مسار لبادئة معينة، في هذه الحالة يجب أن يختار الموجّه أحد تلك المسارات المحتملة. تشمل المدخلات لعملية اختيار المسار هذه مجموعة المسارات التي تم التعرف عليها وقبولها بالموجّه. في حالة وجود مسارين أو أكثر إلى نفس البادئة، عندئذ يطبق BGP قواعد الحذف التالية بشكلٍ متسلسل إلى أن يبقى مسار واحد:

- تخصص للمسارات قيم مفاضلة محلية كأحد البارامترات الخاصة بها. قد يتم ضبط المفاضلة المحلية للمسار من قِبَل الموجّه نفسه أو تكون مما تعلّمه موجّه آخر في نفس نظام AS. هذا القرار يتعلق بسياسة التشغيل ويترك لمشرف شبكة AS. (سنناقش قضايا سياسة BGP بعد قليل بشيء من التفصيل). يتم اختيار المسارات ذات قيم المفاضلة المحلية الأعلى.
- من المسارات الباقية (والتي لها جميعاً نفس قيم المفاضلة المحلية)، يتم اختيار المسار الذي له أقصر AS-PATH. إذا كانت تلك هي القاعدة الوحيدة لاختيار المسار، فإن BGP يكون في الواقع مستخدماً لخوارزمية DV لتحديد المسار، حيث يستخدم معيار المسافة عدد القفزات عبر أنظمة AS بدلاً من عدد القفزات عبر موجّهات.
- من بين المسارات المتبقية (والتي لها جميعاً نفس قيم المفاضلة المحلية ونفس طول AS-PATH)، يتم اختيار المسار الذي له أقرب NEXT-HOP. ونعني بـ "أقرب" هنا الموجّه بمسار له أصغر قيمة لأدنى كلفة حسب ما تحدده خوارزمية التوجيه داخل AS. كما نوقش في الجزء 4-5-3 تدعى هذه العملية توجيه البطاطس الساخنة.
- إذا تبقى أكثر من مسار واحد يستخدم الموجّه معرّفات BGP لاختيار المسار؛ انظر [Stewart 1999].

إن قواعد الحذف في الواقع أكثر تعقّيداً من تلك الموصوفة هنا. ولتفادي الكوابيس حول BGP من الأفضل أن نتعلم قواعد الاختيار في BGP بجرعات صغيرة!

سياسة التوجيه

دعنا نوضّح بعض المفاهيم الأساسية لسياسة التوجيه في BGP بمثال بسيط. يوضح الشكل 4-43 ستّة أنظمة مستقلة ذاتياً متصلة ببعضها: A، B، C، W، X، Y. من المهم ملاحظة أن A، B، C، W، X، Y أنظمة مستقلة ذاتياً وليست موجّهات. دعنا نفترض أن الأنظمة المستقلة ذاتياً W، X، Y شبكات عقب (stub) وأن A، B، C شبكات عمود فقري لموفر خدمة. سنفترض أيضاً أن A، B، C لها جمعياً وصلة نظير مع بعضها البعض، كما توفر معلومات BGP كاملةً إلى شبكات عملائها. يجب أن يكون كل المرور الداخل لشبكة عقب (stub) متّجه إلى تلك الشبكة، وأن يكون كل المرور الخارج من شبكة stub متولداً في تلك الشبكة. واضح أن W وY تُعتبر شبكات عقب. علاوةً على ذلك تُعتبر X شبكة عقب متعددة المنازل (multihomed) حيث توصّل إلى بقيّة الشبكة عن طريق موفري خدمة مختلفين (وهي طريقة أصبحت شائعة على نحو متزايد عملياً). ومع ذلك فمثلها مثل W وY يجب أن تكون X نفسها هي المصدر/الوجهة لكل المرور الخارج/الداخل لها.



الشكل 4-43 سيناريو BGP بسيط.

لكن كيف يمكن تطبيق وفرض هذا السلوك على شبكة العقب؟ كيف سيتم منع X من تمرير حركة المرور بين B وC؟ يمكن تحقيق ذلك بسهولة بالتحكم في الكيفية التي تعلن بها مسارات BGP. وبالتحديد ستعمل X كشبكة عقب إذا أعلنت (لجيرانها B وC) بأنها ليس لها مسارات إلى أي وجهات أخرى ماعدا

نفسها. أي أنه رغم كون X قد تعرف مساراً (مثلاً XCY والذي يصل إلى Y) إلا أن X لن تعلن عن هذا المسار لـ B. ولأن B لا تدري أن X عندها مسار إلى Y فإن B لن ترسل أبداً حركة بيانات متجهة إلى Y (أو C) عن طريق X. هذا المثال البسيط يوضح كيف يمكن أن تستخدم سياسة إعلان انتقائية للمسارات في تطبيق علاقات التوجيه بين العملاء وموفري الخدمة.

دعنا نركز على شبكة موفّر خدمة، مثلاً نظام B، وافترض أنه عرف (من A) أن A لديه مسار AW إلى W. يمكن أن يضيف B المسار BAW إلى قاعدة معلومات التوجيه لديه. من الواضح أن B يريد أيضاً أن يعلن المسار BAW إلى زبونه X لكي يعرف X أنه بوسعه التوجيه لـ W عن طريق B. لكن هل يجب أن يعلن B المسار BAW إلى C؟ إذا فعل ذلك يمكن أن يوجّه C المرور إلى W عن طريق CBAW. إذا كان كلٌّ من A، B، C شبكات عمود فقري عندئذ قد يشعر B – ومعه حق – بأنه لا يجب أن يتحمل عبئاً (وكلفة!) نقل حركة المرور بين A وC. قد يكون B محقاً في شعوره بأن هذه مهمة (وكلفة!) A وC في التأكد من أن C يمكنه التوجيه إلى زبائن A ومنهم عبر وصلة مباشرة بين A وC. لا توجد حالياً معايير رسمية تحكم كيفية التوجيه بين شبكات العمود الفقري لموفري خدمة الإنترنت. ومع ذلك فالطريقة المجرّبة والمتبعة من قِبَل موفري خدمة الإنترنت التجاريين هي أن أي حركة مرور تتدفّق عبر شبكة عمود فقري لموفر خدمة الإنترنت يجب أن يكون لها إمّا مصدر أو وجهة (أو كلاهما) في شبكة تُعتبر زبوناً لموفّر خدمة الإنترنت هذا؛ وإلا فإن حركة المرور ستحصل على "توصيلة مجانية" عبر شبكة موفر الخدمة. يتم التفاوض عادة على اتفاقيات ثنائية بين أزواج موفري خدمة الإنترنت (والتي تتناول تساؤلات كالمذكورة أعلاه)، وتكون تلك الاتفاقيات سرّية في أغلب الأحيان. يتضمن [Huston 1999a] مناقشة ممتعة للاتفاقيات بين النظائر. ولوصف مفصّل للكيفية التي تعكس بها سياسة التوجيه العلاقات التجارية بين موفري خدمة الإنترنت، راجع [Gao 2001]. لمناقشة حديثة لسياسات توجيه BGP من وجهة نظر موفر خدمة الإنترنت، اطّلِع على [Caesar 2005].

المبادئ في الواقع العملي (Principles in Practice)

لماذا توجد بروتوكولات مختلفة للتوجيه داخل أنظمة AS وفيما بينها؟

بعد أن درسنا الآن تفاصيل بروتوكولات محددة للتوجيه داخل أنظمة AS وفيما بينها في إنترنت اليوم، دعنا نختم بمحاولة للإجابة على سؤال قد يكون هو الأكثر أهمية والذي يمكن أن نسأله في المقام الأول عن تلك البروتوكولات (نأمل أن تكون قد تساءلت عن هذا دوماً ولم تغب عنك الغابة بالنظر للأشجار!): لماذا تستخدم بروتوكولات مختلفة للتوجيه داخل أنظمة AS وفيما بينها؟

تكمن الإجابة على هذا السؤال في الاختلافات الأساسية بين أهداف التوجيه داخل أنظمة AS وفيما بينها:

- سياسات التشغيل: تلعب سياسات التشغيل دوراً مهماً في عمليات التوجيه بين أنظمة AS. ربّما يكون من المهم منع حركة المرور التي تنشأ في AS معين من عبور AS آخر. وبالمثل قد يريد AS معين أن يسيطر على حركة المرور التي تعبر، إلى أنظمة AS الأخرى. رأينا أن بروتوكول BGP ينقل بارامترات المسارات ويسمح بالتحكم في توزيع معلومات التوجيه ليتسنى اتخاذ قرارات التوجيه المعتمدة على سياسة معينة. أما داخل نظام AS فيعتبر كل شيء اسمياً تحت نفس الرقابة الإدارية وبالتالي تلعب السياسة دوراً أقل أهمية بكثير في اختيار المسارات داخل AS.
- القدرة على التوسع: تُعتبر قدرة خوارزمية التوجيه وهياكل بياناتها على معالجة التوجيه إلى/بين أعداد كبيرة من الشبكات قضية حاسمة في التوجيه بين أنظمة AS، في حين تُعتبر القدرة على التوسع داخل AS أقل أهميةً. وأحد أسباب ذلك أنه إذا زاد حجم نطاق إداري جداً فيمكن دوماً تقسيمه إلى عدة أنظمة AS والقيام بعملية التوجيه بينها. (تذكّر أن بروتوكول OSPF يسمح بتكوين مثل هذا التدرج الهرمي بتقسيم AS إلى مناطق).
- الأداء: نظراً لأن التوجيه بين أنظمة AS يعتمد كثيراً على السياسة فإن معايير الجودة للمسارات المستعملة (كالأداء مثلاً) تُشكّل في أغلب الأحيان اهتماماً ثانوياً (بمعنى أنه قد يُفضّل مسارٌ أطول أو أكثر كلفة على مسار أقصر وأقل كلفة إذا كان الأول يحقق بعض معايير السياسة التي لا يحققها الثاني). في الحقيقة كما رأينا في التوجيه بين أنظمة AS ليس هناك ذكر للكلفة مرتبط بالمسارات (باستثناء عدد قفزات AS). في المقابل داخل نظام AS تكون مثل هذه المخاوف السياسية أقل أهميةً مما يسمح للتوجيه بالتركيز أكثر على مستوى الأداء الذي يمكن تحقيقه على المسار.

كما ذكرنا سابقاً يُعدّ BGP معياراً واقعياً (de facto standard) للتوجيه بين أنظمة AS في الإنترنت العامّة. ولرؤية محتويات جداول التوجيه المختلفة (والكبيرة!) لبروتوكول BGP والمستخلصة من موجّهات موفري خدمة الإنترنت في الطبقة الأولى (tier-1) راجع الموقع http://www.routeviews.org. في أغلب الأحيان تحتوي جداول توجيه BGP على عشرات الآلاف من البادئات والبارامترات المناظرة. توجد إحصائيات حول حجم وخصائص جداول توجيه BGP في [Huston 2001; Meng 2005]. وبهذا تكتمل مقدمتنا القصيرة عن بروتوكول BGP. إن فهم BGP مهم لأنه يلعب دوراً مركزياً في الإنترنت. ونشجّعك على الاطلاع على المراجع [Griffin 2002; Stewart 1999; Labovitz 1997; Halabi 2000; Huitema 1998; Gao 2001; Feamster 2004; Caesar 2005] للمزيد من المعلومات حول BGP.

4-7 التوجيه الإذاعي والمتعدّد (Broadcast and multicast routing)

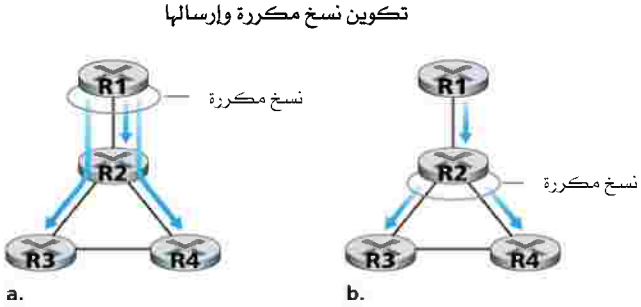
ركّزنا في هذا الفصل حتى الآن على بروتوكولات التوجيه التي تدعم الاتصال الفردي (unicast) (أي من نقطة إلى نقطة)، والذي فيه ترسل عقدة مصدر وحيدة الرزمة إلى عقدة وجهة وحيدة. في هذا الجزء سنحوّل انتباهنا لبروتوكولات توجيه الإذاعة والتوجيه المتعدد. في توجيه الإذاعة توفر طبقة الشبكة خدمة تسليم رزمة أرسلت من عقدة مصدر إلى كل العقد الأخرى في الشبكة؛ أما في التوجيه المتعدّد يمكّن لعقدة مصدر وحيدة إرسال نسخة من رزمة إلى مجموعة جزئية من عقد الشبكة الأخرى. في الجزء 4-7-1 سنناقش خوارزميات توجيه الإذاعة وتضمينها في بروتوكولات التوجيه، ثم نفحص التوجيه المتعدّد في الجزء 4-7-2.

4-7-1 خوارزميات توجيه الإذاعة

قد تكون أبسط طريقة لتحقيق اتصال إذاعي هي أن ترسل عقدة المصدر نسخة مستقلة من الرزمة إلى كل وجهة ممكنة (باستخدام توجيه إرسال أحادي unicast)، كما هو موضح في الشكل 4-44 (a). فإذا كان عدد عقد الوجهات N فإن عقدة المصدر ببساطة تُعد N نسخة من الرزمة؛ وتعنون كل نسخة منها إلى أحد الوجهات المختلفة ثم ترسلها إلى تلك الوجهات. هذه طريقة بسيطة للإذاعة (أي ما

يعرف بـ *N*-way unicast) ، ولا تحتاج إلى بروتوكولات جديدة لطبقة الشبكة ولا إلى إجراءات إضافية لنسخ الرزمة أو التمرير). ومع ذلك هناك عدّة عيوب لهذه الطريقة. العيب الأول عدم كفاءتها. إذا كانت عقدة المصدر موصّلة إلى بقيّة الشبكة عن طريق وصلة وحيدة، فإن عدد *N* نسخة من نفس الرزمة ستعبر هذه الوصلة الوحيدة. واضحٌ أن العملية ستكون أكثر كفاءة إذا أرسلنا نسخة واحدة فقط من الرزمة على تلك القفزة الأولى على أن تقوم العقدة على الطرف الآخر من تلك الوصلة بإرسال أي نسخ إضافية مطلوبة. أي سيكون الحل أكثر كفاءة إذا جعلنا عقد الشبكة نفسها (بدلاً من عقدة المصدر فقط) تنشئ نسخاً مضاعفة من الرزمة. على سبيل المثال في الشكل 4-44 (b) تعبر نسخة واحدة فقط الوصلة R1-R2. ثمّ تنسخ تلك الرزمة في R2 وترسل نسخة على كلٍّ من الوصلات R2-R3 وR2-R4.

إن العيوب الأخرى لطريقة *N*-way unicast ربما تكون أكثر دقة ولكنها ليست أقل أهمية. إن طريقة *N*-way unicast تفترض ضمناً أن عقد الاستقبال وعناوينها معروفة للمُرسِل. لكن كيف يحصل المُرسِل على تلك المعلومات؟ على الأغلب سنحتاج إلى آليات بروتوكولات إضافية (كبروتوكول عضوية الإذاعة أو بروتوكول التسجيل للوجهة). سيضيف هذا عبئاً إضافياً وبشكلٍ هام تعقيداً إضافياً للبروتوكول الذي بدا بسيطاً جداً في البداية. وعيب أخير لطريقة N-way unicast يتعلق بالأغراض التي من أجلها سيستخدم الإرسال الإذاعي. في الجزء 4-5 عرفنا أن بروتوكولات توجيه حالة الوصلة تستخدم الإرسال الإذاعي لنشر معلومات حالة الوصلة والتي تُستخدم لحساب مسارات الإرسال الفردي unicast. واضح أنه حين يُستخدم الإرسال الإذاعي في حساب وتحديث مسارات unicast يكون من غير المعقّول (في أحسن الأحوال!) الاعتماد على البنية التحتية لتوجيه unicast لإنجاز الإرسال الإذاعي.



الشكل 4-44 نسخ الرزمة عند المصدر في مقابل نسخها في الشبكة.

من هذه العيوب العديدة لطريقة N-way unicast تتضح أهمية الطرق التي تلعب فيها عقد الشبكة نفسها دوراً فاعلاً في نسخ الرزمة وتمريرها وحساب مسارات الإذاعة. سنتناول فيما يلي عدداً من تلك الطرق ونستخدم مرة أخرى اصطلاحات الرسم البياني التي قدّمناها في الجزء 4-5. نمثل الشبكة مرة أخرى كرسم بياني $G=(N, E)$ حيث N مجموعة العقد وE مجموعة الحافات، وحيث كل حافة هي زوج من العقد N. سنكون متساهلين نوعاً ما في استخدامنا للرموز وعندما لا يكون هناك مجال للخلط في الفهم سنستخدم N للإشارة إلى مجموعة العقد وأيضاً إلى حجم تلك المجموعة ($|N|$).

الفيض غير المحكوم

تُعتبر طريقة الفيض (flooding) أبسط الطرق لإنجاز الإرسال الإذاعي، حيث ترسل عقدة المصدر نسخة الرزمة إلى كل جيرانها. عندما تستلم عقدةٌ ما الرزمة المذاعة، فإنها تنسخ الرزمة وترسلها إلى كل جيرانها (ماعدا الجار الذي استلمت منه الرزمة). واضح أنه إذا كان الرسم البياني للشبكة متصلاً، فستؤدي هذه الطريقة إلى أن تتلقى كل عقدة في الرسم البياني في النهاية نسخةً من الرزمة المذاعة. وبالرغم من أن هذه الطريقة بسيطة ورائعة، إلا أن لها عيب قاتل (قبل أن تستمر في القراءة، فكّر لترى إذا كان يمكنك أن تفهم هذا العيب القاتل):

في حالة وجود حلقات مفرغة بالرسم البياني للشبكة، فعندئذ ستستمر نسخة أو أكثر من كل رزمة مذاعة في الدوران بشكلٍ غير محدود. على سبيل المثال في الشكل 4-44 ستفيض R2 على R3، وتفيض R3 على R4، وتفيض R4 على R2، ومرة أخرى تفيض R2 على R3، وهكذا. هذا السيناريو البسيط يؤدّي إلى دوران لانهائي لرزمتي إذاعة، أحداهما باتجاه دوران عقارب الساعة، والآخر بعكس اتجاه دوران عقارب الساعة.

ولكن يمكن أيضاً أن يكون هناك عيب آخر قاتل ومفجع بدرجة أكبر: عندما توصّل عقدة إلى أكثر من عقدتين أخريين، ستنشئ وترسل عدة نسخٍ من الرزمة المذاعة، وستقوم كلٌّ منها بدورها لإنشاء عدة نسخ (في العقد الأخرى التي لها أكثر من جارين)، وهكذا. سيكون نتيجة ذلك عاصفة من الرزم المذاعة والتي تقود في النهاية إلى جعل الشبكة عديمة الفائدة. (انظر إلى تمارين الواجب المنزلي في نهاية الفصل لإحدى المسائل التي يتم فيها تحليل المعدل الذي تنمو به مثل تلك العاصفة).

الفيض المحكوم

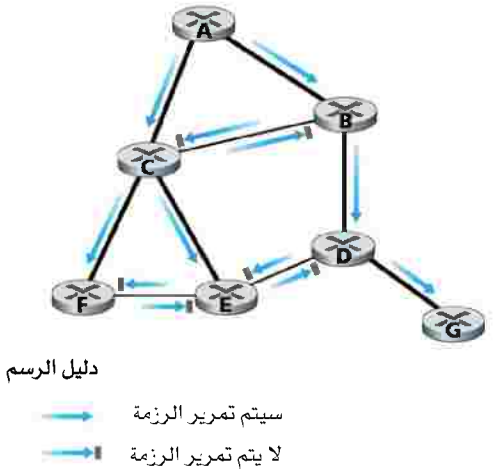
يكمن المفتاح لتجنب عاصفة الإذاعة تلك في جعل العقدة تختار بتعقّل متى تفيض برزمة ومتى تُحجم عن ذلك (مثلاً إذا سبق أن استلمت نسخة من نفس الرزمة وفاضت بها). عملياً يمكن أن يتم ذلك بعدّة طرق.

في الفيض المحكوم بالرقم المتسلسل تضيف عقدة المصدر عنوانها (أو معرّفاً فريداً آخر) بالإضافة إلى رقم متسلسل لرزمة الإذاعة، ثمّ ترسل الرزمة إلى كل جيرانها. تحتفظ كل عقدة بقائمة من عناوين المصدر الرقم المتسلسل لكل رزمة مذاعة تستلمها وتنسخها وترسلها. عندما تستلم عقدة رزمة مذاعة سوف تفحص القائمة أولاً لترى ما إذا كانت الرزمة موجودة بالقائمة. فإذا حدث ذلك فإنها تقوم بإسقاط الرزمة؛ وإلا فسوف تنسخها وترسلها لكل جيرانها (ماعدا العقدة التي استلمت منها الرزمة). يستخدم بروتوكول النظائر Gnutella (والذي ناقشناه في الفصل الثاني) الفيض المحكوم بالأرقام المتسلسلة لإذاعة الاستفسارات في الشبكة

الإضافية (overlay network). (إلا أن نسخ الرسائل وتمريرها في Gnutella يتم في طبقة التطبيقات بدلاً من طبقة الشبكة).

من الطرق الأخرى للتحكم في الفيض طريقة تعرف بتمرير المسار العكسي (Reverse Path Forwarding (RPF)) [Dalal 1978]، كما تُدعى أحياناً باسم إذاعة المسار العكسي (RPB). إن الفكرة وراء RPF بسيطة ورائعة. عندما يستلم موجّه رزمةً مذاعةً بعنوان مصدر معين فإنه يرسلها على كل وصلاته الخارجة (ماعدا تلك التي استلم منها الرزمة) ويتم هذا فقط إذا وصلت الرزمة على الوصلة الموجودة على أقصر مسار unicast من الموجّه إلى المصدر. فيما عدا ذلك يهمل الموجّه ببساطة الرزمة القادمة بدون تمريرها على أيٍّ من وصلاته الخارجة. يمكن إسقاط مثل تلك الرزمة لأن الموجّه يعرف أنه سوف يستلم أو قد استلم نسخة من تلك الرزمة على الوصلة التي على طريقه الأقصر الذي يعود إلى المُرسِل. (قد تريد إقناع نفسك بأن هذا سيحدث في الواقع وأنه لن يكون هناك مجال لحدوث دوران الرزم أوعاصفة الإرسال الإذاعي). لاحظ أن RPF في الحقيقة لا يستعمل توجيه unicast لتسليم رزمة إلى وجهة ما، ولا يتطلّب أن يعرف الموجّه المسار الأقصر بكامله بينه وبين المصدر. يحتاج RPF فقط لمعرفة الجار التالي على مسار unicast الأقصر إلى المُرسِل؛ حيث يستخدم هوية هذا الجار فقط لتحديد ما إذا كان سيرسل فيضاً من رزمة الإذاعة التي استلمها أم لا.

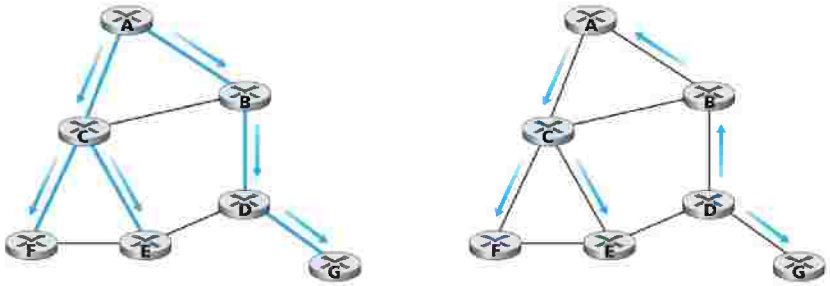
يوضح الشكل 4-45 بروتوكول RPF. افترض أن الوصلات المرسومة بخطوط سميكة تمثل مسارات أدنى كلفة من المستلمين إلى المصدر (A). تذيع العقدة A في البداية الرزمة التي مصدرها A إلى العقد C وB. ترسل العقدة B رزمة المصدر A التي استلمتها من A (لأن A على مسارأدنى كلفة من B إلى A) إلى كل من C وD. ستهمل B (تُسقط بدون تمرير) أي رزمة من المصدر A تتلقاها من أي عقدة أخرى (مثلاً C أو D). دعنا الآن نأخذ في الاعتبار عقدةً ولتكن C والتي تستلم رزمة من المصدر A مباشرة من A وكذلك من B. لأن B ليست على مسار C الأقصر الذي يعود إلى A، فإن C ستهمل أي رزمة من المصدر A تتلقاها من B. من ناحية أخرى عندما تستلم C رزمة من المصدر A مباشرة من A، فسترسل الرزمة إلى العقد B، E، F.



الشكل 4-45 تمرير المسار العكسي.

الإذاعة عبر الشجرة الممتدة (Spanning Tree Broadcast)

بينما يتفادى الفيض المحكوم بالأرقم المتسلسلة عاصفة الإذاعة، فإنه لا يتفادى بشكلٍ كامل إرسال رزم مذاعة غير ضرورية (زائدة عن الحاجة). على سبيل المثال في الشكل 4-46 تستلم العقد B، C، D، E، F رزمة أو رزمتين غير ضروريتين. وبشكلٍ مثالي يجب أن تستلم كل عقدة نسخة واحدة فقط من الرزمة المذاعة. بفحص الشجرة المكونة من العقد الموصلة بالخطوط السميكة في الشكل 4-46 (a)، يمكنك ملاحظة أنه إذا أُرسلت الرزم المذاعة فقط على طول الوصلات ضمن هذه الشجرة، فإن كل عقدة بالشبكة ستستلم نسخة واحدة فقط من الرزمة المذاعة (هذا بالضبط هو الحل الذي كنا نبحث عنه!). هذه الشجرة مثال للشجرة الممتدة (spanning tree)، وهي شجرة تحتوي كل العقد في الرسم البياني. وبشكلٍ أكثر رسمية الشجرة الممتدة للرسم البياني $G=(N, E)$ هي رسم بياني $G'=(N, E')$ حيث تمثل E' مجموعة جزئية من E، كما أن G' رسم بياني متصل يحتوي على كل العقد الأصلية في G ولا يحتوي على حلقات (cycles).



a. مصدر الإذاعة A

b. مصدر الإذاعة D

الشكل 4-46 البث الإذاعي عبر شجرة ممتدة.

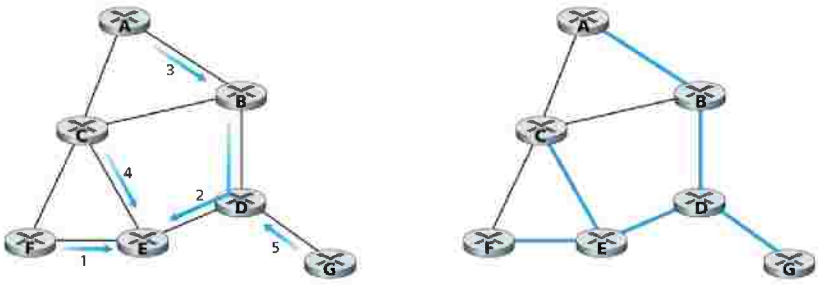
إذا كان لكل وصلة كلفة مصاحبة فإن كلفة الشجرة تساوي مجموع كلف وصلاتها، وعندئذ (بما لا يدعو للاستغراب) يطلق على الشجرة التي لها أدنى كلفة بين كل الأشجار الممتدة عبر الرسم البياني "الشجرة الممتدة بأدنى كلفة" (minimum-spanning tree).

ومن ثم فإن إحدى الطرق الأخرى للإرسال الإذاعي تتلخص في أن تبني عقد الشبكة أولاً شجرة ممتدة. عندما تريد عقدة المصدر أن تذيع رزمة، فإنها ترسل الرزمة على كل وصلاتها التي تنتمي للشجرة الممتدة. إذا استلمت عقدة رزمة مذاعة ترسل الرزمة إلى كل جيرانها في الشجرة الممتدة (ماعدا الجار الذي استلمت منه الرزمة). لا تتخلص الشجرة الممتدة فقط من إذاعة رزم غير ضرورية، ولكن ولأول مرة يمكن أن تستخدم الشجرة الممتدة من قِبَل أي عقدة لبدء إرسال إذاعي، كما هو موضح في الشكل 4-46. لاحظ أنه ليس من الضروري أن تكون العقدة على دراية كاملة بالشجرة؛ ببساطة تحتاج فقط لمعرفة أيٍّ من جيرانها في G هم جيران على الشجرة الممتدة.

التعقيد الأساسي لطريقة الشجرة الممتدة هو بناء وصيانة الشجرة. تم تطوير العديد من خوارزميات الشجرة الممتدة الموزّعة [Gallager 1983; Gartner 2003]. سنذكر هنا فقط خوارزمية بسيطة. تعتمد هذه الطريقة على وجود عقدة مركزية معرّفة – تُعرف أيضاً بنقطة الالتقاء (rendezvous point) أو نقطة القلب (core

(tree-join point) – لبناء شجرة ممتدة. تُرسل العقد رسائل التحاق بالشجرة (messages) معنونة إلى العقدة المركزية. يتم تمرير رسالة الالتحاق باستخدام توجيه unicast نحو المركز حتى تصل إلى عقدة تنتمي للشجرة الممتدة أو تصل إلى المركز. في كلتا الحالتين يحدد المسار الذي اتبعته رسالة الالتحاق الفرع من الشجرة بين العقدة الطرفية التي بدأت الرسالة والمركز. يمكن أن نعتبر هذا المسار الجديد على أنه "مُطعَّم" (ملحق) للشجرة الممتدة الموجودة.

يوضح الشكل 4-47 بناء شجرة ممتدة مركزية. افترض أن العقدة E اختيرت كمركز للشجرة، وافترض أن العقدة F هي العقدة الأولى التي تنضم للشجرة بإرسال رسالة التحاق إلى E. تصبح الوصلة الوحيدة EF هي الشجرة الممتدة الأولى. ثمّ تنضم العقدة B إلى الشجرة بإرسال رسالة التحاق نحو E. افترض أن المسار من B إلى E هو عن طريق D. في هذه الحالة ستؤدّي رسالة الالتحاق إلى أن يطعّم المسار BDE في الشجرة. بعد ذلك تنضم العقدة A بإرسال رسالة التحاق نحو E. إذا كان المسار من A إلى E يمر بـ B، ولأن B موجودة بالشجرة بالفعل فسيؤدي وصول رسالة A عند B إلى أن يطعّم المسار AB على الفور في الشجرة. تنضمّ العقدة C إلى الشجرة بإرسال رسالة نحو E. أخيراً لأن توجيه المسار من G إلى E يجب أن يكون عن طريق العقدة D، فإنه عندما ترسل G رسالة إلى E، ستضاف الوصلة GD إلى الشجرة.



a. البناء التدريجي للشجرة الممتدة

b. الشجرة الممتدة المبنية

الشكل 4-47 بناء شجرة ممتدة مركزية.

خوارزميات الإذاعة في الواقع العملي

تُستخدم بروتوكولات الإذاعة عملياً في طبقتي الشبكة والتطبيقات. كما أوردنا في الجزء 2-6 يستخدم بروتوكول Gnutella [Gnutella 2007] الإرسال الإذاعي في طبقة التطبيقات لإذاعة استفسارات المحتوى بين نظائر Gnutella. في هذه الحالة تُمثَّل الوصلة بين عمليتي نظير في مستوى التطبيقات موزّعتين في شبكة Gnutella في الواقع بتوصيلة TCP. يستخدم Gnutella الفيض المحكوم بالرقم المتسلسل حيث يستعمل 16 بتاً للمعرّف و16 بتاً لوصف نوع الحمل الآجر (والذي يعرّف نوع رسالة Gnutella) في اكتشاف ما إذا كان استفسار الإذاعة الذي تم استلامه قد سبق استلامه ونسخه وإرساله. كما ذكرنا أيضاً في الجزء 2-6 يستخدم بروتوكول Gnutella أيضاً حقل زمن فترة العمر (TTL) لتحديد عدد القفزات التي سيرسل عليها الاستفسار. عندما تستلم عملية Gnutella استفساراً وتنسخه فإنها تُنقِص حقل TTL قبل إرسال الاستفسار. وهكذا سيصل استفسار Gnutella فقط إلى النظائر التي تقع ضمن عدد معين (القيمة الأولية لـ TTL) من القفزات في مستوى التطبيقات ابتداءً من مصدر الاستفسار. ولذا يطلق أحياناً على آلية فيض Gnutella اسم فيض المجال المحدود.

يُستخدم أيضاً شكلٌ من الفيض المحكوم بأرقام متسلسلة لإذاعة إعلانات حالة الوصلة (LSAs) ضمن خوارزمية التوجيه في بروتوكول OSPF [RFC 2328; Perlman 1999] وبروتوكول IS-IS [RFC 1142; Perlman 1999]. يستعمل OSPF عدداً مكوناً من 32 بتاً كرقم متسلسل، بالإضافة إلى حقل العمر (age field) المكون من 16 بتاً لتمييز حالة الوصلة LSAs. تذكّر أن عقدة OSPF تذيع LSAs لوصلاتها الملحقة بشكلٍ دوري، أو عندما تتغيّر كلفة وصلة إلى عقدة مجاورة ، أو عندما تتغيّر حالة الوصلة إلى شغّالة أو متعطلة. تُستخدم الأرقام المتسلسلة لرزم LSA لاكتشاف تكرار LSAs، ولكنها تؤدي أيضاً وظيفة مهمة أخرى في بروتوكول OSPF. من المحتمل أثناء استخدام الفيض أن تصل رزمة LSA انشئت بالمصدر في الوقت t بعد رزمة LSA أخرى أُنشئت بنفس المصدر في الوقت

$t+\delta$. تسمح الأرقام المتسلسلة المستخدمة لدى عقدة المصدر بتمييز LSA الأقدم عن LSA الأحدث. يخدم حقل العمر غرضاً مشابهاً لقيمة TTL. تبدأ قيمة حقل العمر الأولية بصفر وتزاد في كل قفزة، كما تزداد أيضاً أثناء وجود الرزمة في ذاكرة الموجّه بانتظار فيضها. ورغم أننا وصفنا خوارزمية فيض LSA فقط سريعاً هنا، فإننا نلاحظ أن تصميم بروتوكولات إذاعة LSA يمكن أن يكون عملاً صعباً جدا في الواقع. يمكنك الاطلاع على حادثة موصوفة في [RFC 789؛ Perlman 1999] أدى فيها إرسال فيض LSAs بشكلٍ خاطئ من موجّهين معطوبين إلى تعطل شبكة ARPAnet بالكامل!

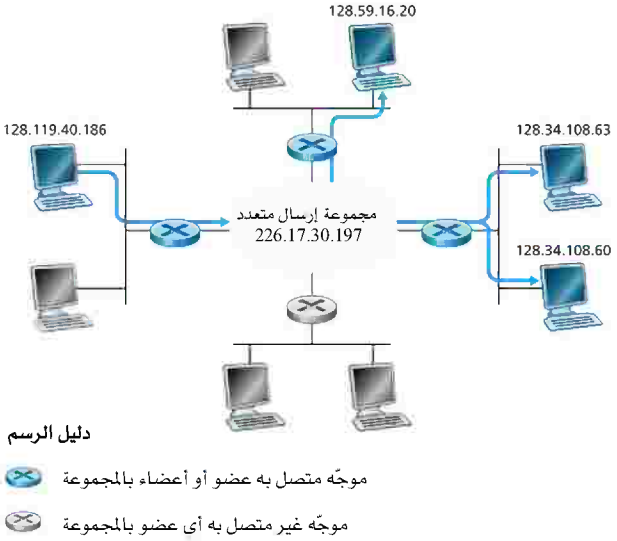
2-7-4 الإرسال المتعدد (الجماعي)

رأينا في الجزء السابق أن خدمة الإرسال الإذاعي توصِّل الرزم إلى كل عقدة في الشبكة. سنحوّل انتباهنا في هذا الجزء إلى خدمة الإرسال المتعدد (الجماعي) (multicast service) والتي يتم فيها توصيل الرزمة إلى مجموعة جزئية فقط من العقد الموجودة بالشبكة. يتطلب عددٌ من التطبيقات الحديثة للشبكة توصيل الرزم من مُرسِل واحد أو أكثر إلى مجموعة من المستلمين. تشمل هذه التطبيقات نقل كتل ضخمة من البيانات (كنقل ترقية للبرامج من مطوّرها إلى المستخدمين الذين يحتاجونها)، وكذلك العرض المستمر لمواد الوسائط المتعددة (كنقل التسجيل الصوتي والفيديو والنصوص لمحاضرة حية إلى مجموعة مشاركين موزعين في أماكن متفرقة)، وتطبيقات المشاركة في البيانات (مثل السبورة (whiteboard) والمؤتمرات عن بُعد (teleconferencing) بين مشاركين كُثر موزعين في أماكن متفرقة)، وكذلك مصادر تغذية البيانات (data feeds) (كأسعار الأسهم)، وتحديث ذاكرة الويب الوسيطة والمحادثات التفاعلية (كالبيئات الافتراضية التفاعلية الموزّعة أوالألعاب متعددة اللاعبين).

في الاتصالات الجماعية تواجهنا مباشرة مشكلتان: كيف يمكن تمييز المستلمين لرزمة جماعية، وكيف تُعنون رزمة مرسلة إلى هؤلاء المستلمين. في حالة الاتصال الفردي (unicast) يوضع عنوان IP مميز للمستلم (الوجهة) في كل رزمة

بيانات. أما في حالة الإرسال الإذاعي فتحتاج كل العقد لاستلام الرزمة المذاعة، ولذا لا نحتاج إلى عناوين للوجهة. لكن في حالة الإرسال الجماعي عندنا الآن مستلمون متعدّدون. هل يعقل أن تحمل كل رزمة جماعية عناوين IP لجميع المستلمين؟ ربما تكون هذه الطريقة عملية مع عدد صغير من المستلمين، لكنها غير قابلة للتوسع بطريقة جيدة للتعامل مع مئات أو آلاف المستلمين، حيث إن كمية معلومات العناوين في رزمة البيانات ستفوق بكثير كمية البيانات الموجودة حقيقةً في حقل الحمل الآجر للرزمة. يتطلب التعريف الصريح للمستلمين من قِبَل المُرسِل أيضاً أن يعرف المُرسِل هويّات وعناوين كل المستلمين. سنرى بعد قليل أن هناك حالات يكون فيها هذا المطلب غير مرغوب.

لهذه الأسباب تُعنوَن رزم الإرسال الجماعي في بنية الإنترنت (وبنى شبكات أخرى كمكائن سحب النقود [Black 1995]) باستخدام العنونة غير المباشرة. بمعنى أنه يكفي استعمال معرّف واحد لمجموعة المستلمين، وترسل نسخة واحدة من الرزمة معنونة إلى المجموعة باستعمال هذا المعرّف الوحيد إلى كل المستلمين المشتركين بتلك المجموعة. في الإنترنت يمثل المعرّف الوحيد لمجموعة من المستلمين بعنوان IP من الفئة D. يطلق على مجموعة المستلمين المرتبطة بعنوان IP من الفئة D مجموعة الإرسال الجماعي (multicast group). يوضح الشكل 4-48 المفهوم التجريدي لمجموعة الإرسال الجماعي، حيث ترتبط أربعة مضيفات (موضحة بلون مظلّل) بالعنوان الجماعي 226.17.30.197 وسوف تستلم كل وحدات البيانات المعنونة إلى تلك المجموعة. المشكلة التي لا يزال علينا تناولها هي حقيقة أن كل مضيف له عنوان IP فردي فريد ومستقل جداً عن عنوان المجموعة التي يشارك فيها.



الشكل 4-48 مجموعة الإرسال الجماعي: رزمة بيانات معنونة إلى المجموعة يتم تسليمها إلى كل أعضاء المجموعة.

بينما يبدو تجريد مجموعة الإرسال الجماعي أمراً بسيطاً فإنه يثير في الواقع العديد من التساؤلات. كيف تبدأ مجموعة وكيف تنتهي؟ كيف يُختار عنوان المجموعة؟ كيف تلتحق مضيفات جديدة بالمجموعة (كمُرسِلين أو كمستقبلين)؟ هل بالإمكان أن ينضم أي واحد إلى مجموعة ما (ويرسل لها أو يستقبل منها) أم أن عضوية المجموعة مقيّدة، وإذا كان الأمر كذلك فمن قِبَل مَنْ يتم ذلك التقييد؟ هل يعرف أعضاء المجموعة هويّات أعضاء المجموعة الآخرين كجزء من بروتوكول طبقة الشبكة؟ كيف تعمل عقد الشبكة مع بعضها البعض لتسليم رزمة بيانات إلى كل أعضاء المجموعة؟ في شبكة الإنترنت تكمن الإجابات على كل هذه الأسئلة في بروتوكول إدارة المجموعات للإنترنت ((IGMP) Internet Group Management Protocol) [RFC 3376]. لذا سنعرج سريعاً على بروتوكول IGMP ثم نعود بعد ذلك إلى تلك الأسئلة العامة.

الشكل 4-49 مكونا طبقة الشبكة للإرسال الجماعي في الإنترنت: بروتوكول IGMP وبروتوكولات التوجيه للإرسال الجماعي.

بروتوكول إدارة مجموعة للإنترنت (IGMP)

تعمل النسخة الثالثة من بروتوكول IGMP بين مضيف وموجّه ملحق به مباشرةً (وبشكلٍ غير رسمي يمكن أن نفكّر بالموجّه الملحق مباشرةً كموجه أول قفزة (first hop router) الذي يراه المضيف على المسار إلى أي مضيف آخر خارج شبكته المحلية الخاصة، أو موجّه القفزة الأخيرة على أي مسار إلى ذلك المضيف). يوضح الشكل 4-49 ثلاثة موجّهات قفزة أولى للإرسال الجماعي، كلٌّ منها مرتبط مع مضيفاته الملحقة عن طريق وصلة محلية خارجة واحدة. ترتبط هذه الوصلة المحلية بشبكة اتصالات محلية في هذا المثال. ورغم وجود العديد من المضيفات بكل شبكة اتصالات محلية إلا أنه غالباً ما ينتمي فقط عدد قليل من تلك المضيفات على أقصى تقدير إلى مجموعة معينة للإرسال الجماعي في أي لحظة.

يوفر بروتوكول IGMP وسائل للمضيف لإعلام الموجّه الملحق بأن تطبيقاً ما يجري تنفيذه على المضيف يريد الانضمام إلى مجموعة معينة للإرسال الجماعي (multicast group). ونظراً لأن مجال تفاعل IGMP محدود بين المضيف والموجّه الملحق به، فمن الواضح أن هناك حاجة لبروتوكول آخر للتنسيق بين موجّهات الإرسال الجماعي (بما في ذلك الموجّهات الملحقة) في كافة أنحاء الإنترنت لكي

توجه وحدات بيانات الإرسال الجماعي إلى وجهاتها النهائية. تتحقق هذه الوظيفة الأخيرة عن طريق خوارزميات طبقة الشبكة للتوجيه الجماعي (كتلك التي سنناقشها بعد قليل). وهكذا يتضمن الإرسال الجماعي في طبقة الشبكة في الإنترنت مكوّنين متكاملين: بروتوكول IGMP وبروتوكولات التوجيه للإرسال الجماعي.

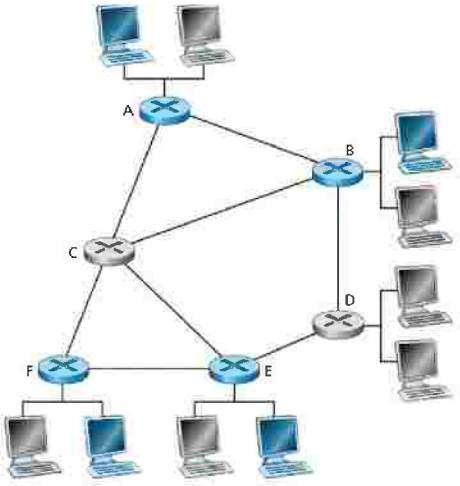
يتضمن بروتوكول IGMP ثلاثة أنواع فقط من الرسائل. كما هو الحال مع بروتوكول ICMP، تغلف رسائل IGMP ضمن وحدات بيانات IP وتحمل القيمة 2 في حقل رقم البروتوكول. ترسل رسالة استفسار العضوية (membership_query) من قِبَل موجّه إلى كل المضيفات على واجهة ملحقة (مثلاً إلى كل المضيفات على شبكة محلية) لتحديد المجموعات المختلفة التي انضمت إليها مضيفات على تلك الواجهة. ترد المضيفات على رسالة استفسار العضوية برسالة تقرير (membership_report) عن أعضاء IGMP. يمكن أيضاً إنشاء رسائل تقرير العضوية من قِبَل مضيف عند بداية انضمام تطبيق إلى مجموعة بدون انتظار لرسالة استفسار العضوية من الموجّه. النوع الثالث والأخير من رسائل IGMP هو رسالة مغادرة المجموعة (leave_group)، وهي رسالة اختيارية. لكن إذا كانت اختيارية فكيف يكتشف موجّه أن مضيفاً قد غادر مجموعة الاتصال الجماعي؟ الجواب على ذلك هو أن الموجّه يستنتج أن المضيف لم يعد في المجموعة إذا لم يرد على رسالة استفسار بالعنوان المحدد للمجموعة. هذا مثال لما يسمى أحياناً بـ "الحالة المرنة" (soft state) في بروتوكولات الإنترنت. في بروتوكول ذي "حالة مرنة" تُحذَف الحالة (في IGMP تشير الحالة إلى حقيقة أن هناك مضيفات ملحقة بمجموعة اتصالٍ جماعي معينة) عن طريق حدث "انتهاء الوقت" (timeout) (في هذه الحالة عن طريق رسائل دورية من الموجّه للاستفسار عن العضوية) إذا لم يتم تحديثها بشكلٍ واضح (في هذه الحالة عن طريق رسائل تقرير العضوية من مضيف ملحق). يرجع السبب في استخدام بروتوكولات الحالة المرنة إلى أنها تؤدي إلى تحكم أسهل من بروتوكولات الحالة المحددة (hard-state) والتي تتطلّب ليس فقط إضافة وإزالة الحالة بشكلٍ محدد ولكن أيضاً آليات للتعافي من الوضع الذي ينسحب فيه الكيان المسؤول عن إزالة

الحالة قبل الأوان أو يطرأ عليه عطب. يمكنك الاطلاع على مناقشات مفيدة عن الحالة المرنة في [Lui 2004; Raman 1999; Ji 2003].

خوارزميات التوجيه للإرسال الجماعي

يوضح الشكل 4-50 مشكلة التوجيه للإرسال الجماعي، وفيه تظهر المضيفات الملتحقة بمجموعة، وكذلك الموجّه الذي تلتحق عن طريقه مباشرة مظللة اللون. كما هو موضح في الشكل 4-50 تحتاج في الواقع مجموعة جزئية فقط من الموجّهات (تلك التي لديها مضيفات ملتحقة بمجموعة الإرسال الجماعي) لاستلام وحدات بيانات الإرسال الجماعي. في الشكل 4-50 تحتاج الموجّهات A، B، E، F فقط لاستلام وحدات بيانات الإرسال الجماعي. ولأنه لا يوجد أيٌّ من المضيفات المتصلة بالموجّه D ملتحق بمجموعة الإرسال الجماعي، ولأن الموجّه C لا يرتبط به أي مضيفات أصلاً، لذا لا يحتاج C ولا D لاستلام حركة مرور الإرسال الجماعي. إن هدف التوجيه للإرسال الجماعي إيجاد شجرة الوصلات التي تربط بين كل الموجّهات التي لديها مضيفات منضمة إلى مجموعة الإرسال الجماعي. بعد ذلك توجّه رزم الإرسال الجماعي خلال تلك الشجرة من المُرسِل إلى كل المضيفات المرتبطة بالشجرة. بالطبع قد تتضمن الشجرة موجّهات ليس لديها مضيفات ملتحقة بمجموعة الإرسال الجماعي (على سبيل المثال في الشكل 4-50 من المستحيل توصيل الموجّهات A، B، E، F في شجرة بدون المرور على الموجّه C أو الموجّه D).

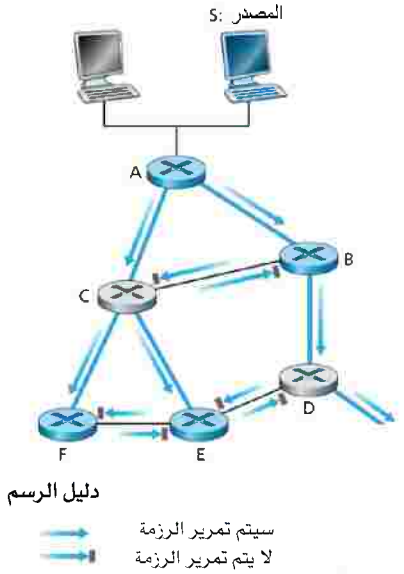
عملياً تم تبني طريقتين لتحديد شجرة توجيه الإرسال الجماعي، ولقد درسنا كلتيهما ضمن سياق توجيه البث الإذاعي، ولذا سنكتفي بذكرهما فقط هنا. تختلف الطريقتان طبقاً لنوع الشجرة المستخدمة لتوزيع وحدات بيانات الإرسال الجماعي: هل هي شجرة وحيدة مشتركة لكل المجموعة بغض النظر عن عدد المصادر (group-shared tree) أو أن كل مصدر يبني شجرة توجيه خاصة به للإرسال إلى المجموعة (source-specific routing tree).



الشكل 4-50 مضيفات الإرسال الجماعي والموجّهات الملحقة بها والموجّهات الأخرى.

- توجيه الإرسال الجماعي باستخدام شجرة مشتركة للمجموعة: كما في حالة الإذاعة عبر الشجرة الممتدة (spanning-tree broadcast)، يعتمد توجيه الإرسال الجماعي باستخدام شجرة مشتركة للمجموعة على بناء شجرة تتضمّن كل موجّهات الأطراف التي لديها مضيفات ملتحقة بمجموعة الإرسال الجماعي. عملياً تستخدم طريقة مركزية لبناء شجرة التوجيه وذلك بجعل موجّهات الأطراف التي لديها مضيفات ملتحقة تُرسِل (عن طريق الإرسال الفردي unicast) رسائل التحاق معنونة إلى عقدة المركز. كما في حالة الإذاعة توجّه رسالة الالتحاق عن طريق توجيه الإرسال الفردي نحو المركز حتى تصل الرسالة إلى موجّه ينتمي حالياً إلى الشجرة أو تصل إلى المركز. كل الموجّهات على طول الطريق الذي تعبر عليه رسالة الالتحاق سوف توجّه وحدات بيانات الإرسال الجماعي إلى موجهات الأطراف تلك التي بدأت رسالة الالتحاق. وهناك سؤال مهم لتوجيه الإرسال الجماعي المعتمد على شجرة مركزية ألا وهو: كيف يمكن اختيار المركز؟ توجد مناقشات حول خوارزميات اختيار المركز في [Wall 1980; Thaler 1997; Estrin 1997].
- توجيه الإرسال الجماعي باستخدام شجرة لكل مصدر: بينما يبنى توجيه الإرسال الجماعي باستخدام شجرة مشتركة للمجموعة (أي شجرة واحدة)

لتوجيه الرزم من كل المصادر، تستخدم الطريقة الثانية شجرة توجيه لكل مصدر في مجموعة الإرسال الجماعي. عملياً تستخدم خوارزمية RPF (بعقدة المصدر x) لبناء شجرة لتوجيه رزم البيانات المتولدة في المصدر x. تتطلب خوارزمية RPF للإذاعة التي درسناها سابقاً تعديلاً بسيطاً لاستخدامها في الإرسال الجماعي. ولتفهم سبب ذلك خذ في الاعتبار الموجّه D في الشكل 4-51. في حالة بروتوكول RPF للإذاعة يرسل الموجّه D الرزم إلى الموجّه G بالرغم من أن الموجّه G ليس لديه مضيفات ملحقة ضمن المجموعة. بينما هذا ليس أمراً سيئاً جداً في هذه الحالة حيث إن D عنده موجّه واحد فقط في اتجاه الإرسال (G)، تخيل ما يحدث إذا كان هناك آلاف الموجّهات في اتجاه الإرسال من D! ستستلم كل هذه الآلاف من الموجّهات رزماً غير مرغوبة. (هذا السيناريو ليس متكلّفاً كما قد يبدو. فقد عانت أول شبكة عالمية للإرسال الجماعي والمعروفة بـ MBone [Casner 1992; Macedonia 1994] في البداية من هذه المشكلة بالضبط). الحلّ لمشكلة استلام رزم غير مرغوبة عند استخدام RPF هو استخدام ما يُعرف بالتقليم (pruning). عندما يستلم موجّه رزم الإرسال الجماعي وليس لديه مضيفات ملحقة منضمّة لتلك المجموعة فإنه يرسل رسالة prune إلى الموجّه الأعلى (عكس اتجاه الإرسال). إذا استلم موجّه رسالة prune من كلٍّ من موجّهاته في اتجاه الإرسال فعندئذ يمكنه تمرير رسالة prune للموجّه الأعلى منه في عكس اتجاه الإرسال.



الشكل 4-51 تمرير المسار العكسي في حالة الإرسال الجماعي.

توجيه الإرسال الجماعي في الإنترنت

كان بروتوكول متجه المسافة لتوجيه الإرسال الجماعي (DVMRP) أول بروتوكول لتوجيه الإرسال الجماعي في الإنترنت [RFC 1075]. يستخدم بروتوكول DVMRP شجرةً لكل مصدر، كما يستخدم خوارزمية RPF مع التقليم (pruning) والتي ناقشناها من قبل. ربما يكون البروتوكول الأكثر استخداماً على نطاق واسع في الإنترنت لتوجيه الإرسال الجماعي هو بروتوكول PIM (Protocol Independent Multicast)، والذي يُعرِّف بشكلٍ صريح سيناريوهين لتوزيع وحدات البيانات. يعرف الأول بالنمط الكثيف (dense mode) [RFC 3973]، وفيه توجد مواقع أعضاء المجموعة بشكلٍ مكثف، وهذا يعني بالضرورة أن الكثير من الموجّهات الموجودة في منطقة أو أغلبها تشترك في توجيه وحدات البيانات. ويُعدّ نمط PIM الكثيف أسلوب تمرير مسار عكسي يستخدم الفيض والتقليم ويشبه من حيث المبدأ بروتوكول DVMRP.

يُعرَف النمط الثاني بالنمط المتناثر (sparse mode) [RFC 4601]، حيث يكون عدد الموجّهات المرتبطة بأعضاءٍ في المجموعة قليلاً مقارنةً بالعدد الكلي للموجّهات، ويكون أعضاء المجموعة متفرّقين على نحو واسع. يُستخدم بروتوكول PIM ذو النمط المتناثر نقاط الالتقاء (rendezvous points) لإعداد شجرة توزيع الإرسال الجماعي. أما في الإرسال الجماعي المحدد بمصدر معين ((Source-Specific (SSM) Multicast) [RFC 3569; RFC 4607] فيُسمَح لمُرسِل واحد فقط بالإرسال خلال الشجرة مما يبسّط إلى حدٍّ كبير إنشاء وصيانة الشجرة.

عند استخدام بروتوكولي PIM وDVMP ضمن نطاق معين يمكن أن يهيئ مشغل الشبكة موجّهات IP الموجودة داخل ذلك النطاق للإرسال الجماعي بنفس الطريقة تقريباً التي يهيئ بها بروتوكولات التوجيه الفردي داخل النطاق مثل RIP وIS-IS وOSPF. لكن ماذا يحدث عندما تكون مسارات الإرسال الجماعي مطلوبة بين النطاقات المختلفة؟ هل هناك بروتوكول إرسال جماعي مكافئ لبروتوكول BGP بين النطاقات؟ إن الجواب (بشكلٍ حرفي) نعم. في [RFC 4271] تم تعريف امتدادات متعددة لبروتوكول BGP للسماح له بنقل معلومات التوجيه للبروتوكولات الأخرى بما في ذلك معلومات الإرسال الجماعي. مثلاً يمكن استخدام بروتوكول اكتشاف مصدر الإرسال الجماعي (Multicast Source Discovery Protocol (MSDP)) [RFC 3618; RFC 4611] لتوصيل نقاط التقاء ببعضها البعض في نطاقات مختلفة من بروتوكول نمط PIM المتناثر.

دعنا ننهي مناقشتنا للإرسال الجماعي في الإنترنت بملاحظة أنه لا يزال أمامنا الكثير لتحقيق الاستفادة من الإرسال الجماعي في الإنترنت على نحوٍ كبير. يمكنك الاطلاع على المناقشات الممتعة حول النموذج الحالي لخدمة الإرسال الجماعي في الإنترنت وما يتعلق بقضايا انتشارها في [Diot 2000; Sharma 2003]. ومع ذلك ورغم عدم الانتشار الواسع إلا أن الإرسال الجماعي على مستوى الشبكة أبعد ما يكون من وصفه بأنه "عديم الفائدة". فحركة مرور بيانات الإرسال الجماعي موجودة على Internet 2 والشبكات المرتبطة بها منذ عدة سنوات [Internet2 Multicast 2007]. قامت إذاعة BBC في المملكة المتّحدة بتوزيع المحتوى

عن طريق الإرسال الجماعي على الإنترنت [BBC Multicast 2007]. في نفس الوقت وكما رأينا مع PPLive في الفصل الثاني وفي أنظمة النظائر الأخرى كالإرسال الجماعي بين الأنظمة الطرفية (End System Multicast) [ESM 2007] يوفر الإرسال الجماعي في مستوى طبقة التطبيقات خدمة الإرسال الجماعي للمحتوى بين النظائر باستخدام بروتوكولات الإرسال الجماعي في طبقة التطبيقات (بدلاً من طبقة الشبكة). هل ستطبّق خدمات الإرسال الجماعي في المُستقبل بشكلٍ أساسي في طبقة الشبكة (في قلب الشبكة) أم في طبقة التطبيقات (على أطراف الشبكة)؟ رغم أن الهوس الحالي لتوزيع المحتوى عن طريق أساليب النظائر يرجّح – على الأقل في المستقبل القريب – الكفّة تجاه تطبيق الإرسال الجماعي في طبقة التطبيقات، إلا أن التقدّم مازال مستمراً في تطبيق الإرسال الجماعي في طبقة الشبكة للإنترنت - وأحياناً يحرز قصب السبق في النهاية البطيئ الذي يسير بخطى ثابتة.

8-4 الخلاصة

بدأنا رحلتنا في هذا الفصل إلى صميم قلب الشبكة (network core) وعرفنا أن طبقة الشبكة توجد على كل مضيف وموجّه في الشبكة، ولذا تُعدّ بروتوكولات طبقة الشبكة من بين البروتوكولات الأكثر صعوبةً في رصّة البروتوكولات.

كما عرفنا أن الموجّه قد يحتاج لمعالجة الملايين من تدفقات الرزم بين أزواج مختلفة من المصدر والوجهة في نفس الوقت. للسماح لموجّه بمعالجة مثل هذا العدد الكبير من التدفقات، تعلم مصممو الشبكة على مر السنين أن مهام الموجّه يجب أن تكون بسيطة بقدر الإمكان. يمكن أن تؤخذ العديد من الإجراءات لتسهيل عمل الموجّه كاستخدام طبقة شبكة تتعامل مع وحدات البيانات بدلاً من طبقة شبكة ذات دوائر افتراضية، واستعمال ترويسة انسيابية وذات حجم ثابت (كما في IPv6)، وعدم السماح بتجزئة وحدة البيانات (وهذا أيضاً متاح في بروتوكول IPv6)، وتوفير خدمة وحيدة وفريدة تتمثل في "خدمة أفضل جهد". لعل الخدعة الأكثر أهمية هنا هي عدم التتبع الفردي لكل تدفق على حدة، وإنما جعل قرارات

التوجيه تعتمد فقط على عناوين وجهات منظمة بشكلٍ هرمي في وحدات البيانات. من المفيد ملاحظة أن الخدمة البريدية تستخدم هذه الطريقة منذ عدّة سنوات.

استعرضنا في هذا الفصل أيضاً المبادئ التي تُبنى عليها خوارزميات التوجيه، وتعلمنا كيف تجرّد خوارزميات التوجيه شبكة الحاسب إلى رسم بياني (graph) مكون من عقد ووصلات. بهذا التجريد يمكننا توظيف النظرية الغنية لأقصر مسار (shortest path) للتوجيه خلال الرسوم البيانية، والتي طوّرت خلال السنوات الأربعين الماضية في مجتمعات الخوارزميات وبحوث العمليات. رأينا أن هناك طريقتين رئيستين: الطريقة المركزية (العالمية) – وفيها تحصل كل عقدة على خريطة كاملة للشبكة وتطبق بشكلٍ مستقل خوارزمية توجيه المسار الأقصر؛ والطريقة اللامركزية – وفيها تحصل كل عقدة بشكلٍ فردي على صورة جزئية للشبكة، ورغم ذلك تعمل العقد سويةً لتوصيل الرزم على أقصر المسارات. درسنا أيضا كيف تُستخدم التراكيب الهرمية للتعامل مع مشكلة التوسع وذلك بتقسيم الشبكات الكبيرة إلى مناطق إدارية مستقلة تسمى النظم المستقلة ذاتياً (ASs). يقوم كل نظام AS بتوجيه وحدات البيانات داخله بشكلٍ مستقل تماماً كما تفعل كل دولة لتوجيه رسائل البريد داخلها بشكلٍ مستقل. وتعلمنا كيف تم تجسيد الأساليب المركزية واللامركزية والتركيب الهرمي في بروتوكولات التوجيه الرئيسة المستخدمة في الإنترنت مثل RIP وOSPF وBGP. وختمنا دراسة خوارزميات التوجيه بمناقشة توجيه البث الإذاعي (العام) والتوجيه المتعدد (الجماعي).

بهذا تكتمل دراستنا لطبقة الشبكة وتستمر رحلتنا في الفصل القادم بالنزول خطوة واحدة خلال رصّة البروتوكولات لمناقشة قضايا طبقة ربط البيانات. وكما هو الحال في طبقة الشبكة تُعدّ طبقة ربط البيانات أيضاً جزءاً من صميم قلب الشبكة. غير أننا سنرى أن طبقة ربط البيانات لها مهمة أكثر محلية لنقل الرزم بين العقد على نفس الوصلة أو شبكة الاتصالات المحلية. وبالرغم من أن هذه المهمة قد تظهر على السطح بأنها بسيطة بالمقارنة بمهام طبقة الشبكة، إلا أننا سنرى أن طبقة ربط البيانات تتضمّن عدداً من القضايا المهمة والممتعة والتي يمكن أن تستغرق منا وقتاً طويلاً لدراستها.

أسئلة وتمارين وتدريبات الفصل الرابع

❖ أسـئلة مراجعة

• الأجزاء 4-1 حتى 4-2

1. دعنا نراجع بعض المصطلحات المستخدمة في هذا الكتاب. تذكر أن رزمة طبقة النقل يطلق عليها "قطعة" (segment)، ورزمة طبقة ربط البيانات يطلق عليها "إطار" (frame). فماذا يطلق على رزمة طبقة الشبكة؟ وتذكر أنه يطلق على كلٍّ من الموجّهات ومحوّلات طبقة ربط البيانات " محوّلات الرزم" (packet switches)؛ فما الفرق الأساسي بين الموجّه ومحوّل طبقة ربط البيانات؟ وتذكر أيضاً أن نستخدم المصطلح "موجّه" لكلٍّ من شبكات وحدات البيانات وشبكات الدوائر الافتراضية.
2. ما الوظيفتان الأكثر أهمية لطبقة الشبكة في شبكات وحدات البيانات؟ وما تلك في شبكات الدوائر الافتراضية؟
3. ما الفرق بين التوجيه (routing) والتمرير (forwarding)؟
4. هل تستخدم الموجّهات في كلٍّ من شبكات وحدات البيانات وشبكات الدوائر الافتراضية جداول تمرير؟ وإذا كان الأمر كذلك فصف تلك الجداول في كلتا الحالتين.
5. صف بعض الخدمات المحتملة التي يمكن أن توفرها طبقة الشبكة لرزمة معينة، وكذلك التي توفرها لتدفق من الرزم (packet flow). هل أيٌّ من تلك الخدمات المحتملة متوفر في طبقة الشبكة للإنترنت؟ وهل أيٌّ منها متوفر في نموذج خدمة معدل البتات الثابت (CBR) لشبكة ATM؟
6. اذكر بعض التطبيقات التي يمكن أن تستفيد من نموذج خدمة معدل البتات الثابت (CBR) لشبكة ATM.

• الجزء 4-3

7. ناقش أسباب تخزين كل منفذ من منافذ المدخل (input ports) في الموجّه عالي السرعة نسخة ظلّ (shadow copy) من جدول التمرير.
8. تم مناقشة ثلاثة أنواع من أنسجة المحوّلات في الجزء 4-3؛ اذكر كل نوع مع وصفه باختصار.

9. صف كيف يمكن أن يحدث الفقد في الرزم عند منافذ المدخل، وكيف يمكن التخلص من ذلك (بدون استخدام مخازن مؤقتة بسعة لا نهائية).
10. صف كيف يمكن أن يحدث الفقد في الرزم عند منافذ المخرج.
11. ما المقصود بحجب HOL؟ وهل يحدث عند منافذ المدخل أم عند منافذ المخرج؟

- **الجزء 4-4**

12. هل يخصص للموجّهات عناوين IP؟ وإذا كان الأمر كذلك، فكم عددها؟
13. ما العدد الثنائي المؤلّف من 32 بتاً والمكافئ لعنوان IP التالي: 223.1.3.27؟
14. قم بزيارة أحد المضيفات الذي يستخدم بروتوكول DHCP للحصول على عنوان IP وقناع الشبكة والموجّه الافتراضي وعنوان خادم DNS المحلي، ومن ثم اكتب تلك القيم.
15. افترض وجود ثلاث موجّهات بين مضيف مصدر ومضيف الوجهة له. بإهمال تجزئة الرزمة (fragmentation) فكم عدد الواجهات (interfaces) التي تمر خلالها وحدة البيانات من المصدر إلى الوجهة؟ وكم عدد جداول التمرير التي سيتم البحث فيها لنقل وحدة البيانات من المصدر إلى الوجهة؟
16. افترض أن تطبيقاً يولّد قطع بيانات مؤلفة من 40 بايتاً كل 20 ميلي ثانية، وأن كل قطعة بيانات يتم تغليفها في قطعة TCP ثم في وحدة بيانات IP. ما النسبة المئوية من كل وحدة بيانات تُعتبر عبئاً إضافياً؟
17. افترض أن المضيف A يرسل إلى المضيف B قطعة TCP مُغلّفة في وحدة بيانات IP. عندما يستلم المضيف B وحدة البيانات فكيف تعرف طبقة الشبكة في المضيف B أنه يجب أن تُسلَم القطعة (أي الحمل الآجر لوحدة بيانات IP) لبروتوكول TCP وليس UDP أو أي شيء آخر؟
18. افترض أنك قمت بشراء موجّه لاسلكي وقمت بتوصيله بمودم الكبل لديك. وافترض أيضاً أن موفر خدمة الإنترنت لك يخصص بشكلٍ ديناميكي لجهازك الموصل (أي موجّه اللاسلكي) عنوان IP واحد. أيضاً افترض أن لديك خمسة حاسبات شخصية تستخدم بروتوكول 802.11 للاتصال بالموجّه لاسلكياً. كيف يتم تخصيص عناوين IP لتلك الحاسبات الخمسة؟ هل يستخدم الموجّه اللاسلكي NAT؟ بيّن السبب.
19. قارن بين حقول التريسة لبروتوكول IPv4 وIPv6؛ هل يوجد حقول مشتركة بينهما؟

20. قيل أن IPv6 يعمل أنفاق خلال موجّهات IPv4. هل توافق على أن IPv6 يتعامل مع أنفاق IPv4 تماماً كبروتوكولات طبقة الوصلة (طبقة ربط البيانات)؟ بين سبب الموافقة أو الرفض.

• الجزء 4-5

21. قارن بين خوارزمية التوجيه بحالة الوصلة وخوارزمية التوجيه بمتجه المسافة.
22. ناقش كيف سهّل التركيب الهرمي للإنترنت من إمكانية التوسع لتضم الملايين من المُستخدمين.
23. هل من الضروري أن يستخدم كل نظام مستقل ذاتياً (AS) نفس بروتوكول التوجيه بداخله؟ ما سبب ذلك؟

• الجزء 4-6

24. بالنظر إلى الشكل 4-35 وبدءاً من جدول التمرير الأصلي للموجّه D وبافترض أن D تلقّى من A الإعلان التالي:

الشبكة الفرعية للوجهة	الموجّه التالي	عدد القفزات إلى الوجهة
Z	C	10
W	—	1
X	—	1
....	....	....

فهل سيتغيّر جدول D؟ وإذا كان كذلك فكيف؟

25. قارن بين الإعلانات المستخدمة في كلٍّ من RIP وOSPF.
26. أكمل الجملة: عادةً يقوم RIP بإعلان عدد القفزات للوجهات المختلفة. من ناحية أخرى تقوم تحديثات BGP بالإعلان عن _____ للوجهات المختلفة.
27. لماذا تستخدم بروتوكولات مختلفة في الإنترنت للتوجيه داخل النظم المستقلة ذاتياً وللتوجيه فيما بينها؟
28. ما سبب أهمية اعتبارات السياسة لبروتوكولات التوجيه داخل النظم المستقلة ذاتياً (مثل OSPF وRIP) كما في بروتوكولات التوجيه فيما بينها (مثل BGP)؟
29. عرّف كلاً من المصطلحات التالية وبيّن العلاقة بينها: شبكة فرعية (subnet)، بادئة (prefix)، مسار BGP.
30. كيف يستخدم BGP خاصيّة NEXT-HOP؟ وكيف يستخدم خاصيّة AS-PATH؟

31. صف كيف يمكن أن يطبق مشرف شبكة موفر خدمة الإنترنت من الطبقة العليا سياسة معينة عند تهيئة بروتوكول BGP؟

- **الجزء 4-7**

32. ما الفرق الأساسي بين تحقيق البث الإذاعي (broadcast) عن طريق عدة إرسالات فردية وبين تحقيقها عن طريق شبكة ذات دعم للبث الإذاعي؟
33. بيّن إذا ما كانت كل جملة من الجمل التالية صحيحة أم خطأ لكلٍّ من الطرق الثلاث العامة التي درسناها للبث الإذاعي (الفيض غير المحكوم، الفيض المحكوم، الشجرة الممتدة للإذاعة) (يمكن افترض عدم فقد أيٍّ من الرزم نتيجة فيض المخازن المؤقتة وأن كل الرزم يتم تسليمها على الوصلة بنفس ترتيب إرسالها):
    a. يمكن أن تستلم عقدة عدة نسخ من نفس الرزمة.
    b. يمكن أن تعيد عقدة توجيه عدة نسخ من الرزمة على نفس وصلة المخرج.
34. عندما يلتحق مضيف بمجموعة للإرسال الجماعي فهل يجب أن تغيِّر عنوان IP لها إلى عنوان المجموعة التي تلتحق بها؟
35. ما الأدوار التي يلعبها كلٌّ من بروتوكول IGMP وبروتوكول التوجيه الإرسال الجماعي للمناطق الواسعة؟
36. ما الفرق بين الشجرة المشتركة للمجموعة (group-shared tree) والشجرة لكل مصدر (source-based tree) في سياق توجيه الإرسال الجماعي؟

❖ تمارين

1. لندرس بعض مميزات وعيوب شبكات وحدات البيانات وشبكات الدوائر الافترضية:
    a. افترض أنه في طبقة الشبكة كانت الموجّهات معرضة لظروف مجهدة والتي قد تُسبب تعطلها بشكلٍ متكرر. بشكلٍ عام ما الذي يجب فعله في هذه الحالة؟ هل هذا يعني أن شبكات وحدات البيانات أفضل من شبكات الدوائر الافترضية أم العكس في مثل تلك الظروف ؟
    b. افترض أنه لكي توفر ضماناً لمستوى الأداء (مثلاً زمن التأخير) الذي سيُرى خلال المسار من المصدر للوجهة تتطلب الشبكة من المُرسِل أن يحدد المعدل الأقصى لحركة بياناته. إذا كان المعدل المعلن والمعدلات الموجودة حالياً بحيث لا يمكن توصيل البيانات من المصدر إلى الوجهة بشكلٍ يحقق متطلبات التأخير، فإن

المصدر لا يسمح له بالوصول للشبكة. هل مثل هذا الأسلوب سهل التحقيق في بنية شبكات وحدات البيانات أم بنية شبكات الدوائر الافترَضية؟

2. افترض شبكة دائرة افترَضية وأن رقم الدائرة الافترَضية يتألف من 16 بتاً:
   a. ما أقصى عدد للدوائر الافترَضية التي يمكن حملها على وصلة ما؟
   b. افترض وجود عقدة مركزية تحدد أرقام المسارات والدوائر الافترَضية عند إنشاء التوصيلة. وافترض أن نفس رقم الدائرة الافترَضية يُستَخدم على مسار تلك الدائرة الافترَضية. صف كيف يمكن أن تحدد العقدة المركزية رقم الدائرة الافترَضية عند إنشاء التوصيلة. هل من الممكن عدم وجود رقم دائرة افترَضية مشترك عندما يكون عدد الدوائر الافترَضية الحالية أقل من العدد الأقصى المحدد في الجزء (a)؟
   c. افترض أنه على مسار الدائرة الافترَضية يُسمح بأرقام مختلفة على كل وصلة على المسار. أثناء إنشاء توصيلة وبعد تحديد مسار من طرف لطرف، صف كيف تختار الوصلات المختلفة أرقام الدوائر الافترَضية خلالها وكيف تهيئ جداول التمرير الخاصة بها بشكلٍ غير مركزي (أي بدون الاعتماد على عقدة مركزية).
3. يتكون جدول التمرير في شبكة الدائرة الافترَضية أساساً من أربعة أعمدة؛ ما معنى القيم في كلٍّ من هذه الأعمدة؟ ويتكون جدول التمرير في شبكة وحدات البيانات أساساً من عمودين؛ فما معنى القيم في كلٍّ من هذين العمودين؟
4. خذ في الاعتبار شبكة دائرة افترَضية يتكون حقل الدائرة الافترَضية لها من بتين. افترض أن الشبكة تريد أن تُنشئ دائرة افترَضية خلال أربع وصلات: A، B، C، D. افترض أن كلاً من هذه الوصلات تحمل حالياً دائرتين افترَضيتين وأرقامها كما هو مبين بالجدول التالي:

الوصلة A	الوصلة B	الوصلة C	الوصلة D
00	01	10	11
01	10	11	00

في إجابتك على الأسئلة التالية تذكَّر أن كلاً من الدوائر الافترَضية الحالية يمكن أن يمر خلال أحد الوصلات الأربعة فقط:

a. إذا كان من المطلوب أن تستخدم كل دائرة افترَضية نفس الرقم على كل الوصلات التي يتألف منها مسارها، فما رقم الدائرة الافترَضية الذي يمكن أن يخصص لدائرة افترَضية جديدة؟

b. إذا كان من المُصرَّح به أن تستخدم كل دائرة افترَاضية أرقاماً مختلفة على الوصلات المختلفة على مسارها (لكي يلزم جداول التمرير أن تقوم بترجمة رقم الدائرة الافترَاضية)، فما عدد التوافقات المختلفة للأربعة أرقام للدوائر الافترَاضية (رقم لكلٍّ من الوصلات الأربعة) التي يمكن أن تُستخدَم؟

5. استخدمنا المصطلح "خدمة توصيلية" (connection-oriented service) لوصف خدمة طبقة النقل، واستخدمنا المصطلح "خدمة توصيلة" (connection service) لوصف خدمة طبقة الشبكة. ما سبب هذا التفريق الدقيق في المصطلحات؟

6. في الجزء 4-3 لاحظنا أنه قد لا يوجد صف انتظار عند المدخل إذا كان نسيج التحويل n مرة أسرع من معدلات خطوط الدخل (بافترَاض وجود n خط دخل لكلٍّ منها نفس المعدل). وضح (بالوصف) سبب ذلك.

7. افترض موجّه بنسيج تحويل ومنفذي إدخال (A وB) ومنفذي إخراج (C وD). افترض أن سرعة نسيج التحويل 1.5 مرة سرعة الخط.

a. افترض (لسبب ما) أن كل الرزم من A متجهة إلى D، وكل الرزم من B متجهة إلى C. هل من الممكن أن يصمم نسيج تحويل بحيث لا يوجد صفوف انتظار لمنفذ الإدخال؟ وضح سبب موافقتك أو رفضك في جملة واحدة.

b. افترض الآن أن الرزم من A وB تتجه بشكلٍ عشوائي إلى C وD. هل من الممكن أن يصمم نسيج تحويل بحيث لا يوجد صفوف انتظار لمنفذ الإدخال؟ وضح سبب موافقتك أو رفضك في جملة واحدة.

8. افترض شبكة وحدات بيانات تستخدم 32 بتاً لعناوين المضيفات، وافترض موجّهاً بأربع وصلات مرقمة من 0 إلى 3 وأنه يلزم توجيه الرزم لواجهات الوصلات كما يلي:

مدى عناوين الوجهة	واجهة الوصلة
من 11100000 00000000 00000000 00000000 إلى 11100000 11111111 11111111 11111111	0
من 11100001 00000000 00000000 00000000 إلى 11100001 00000000 11111111 11111111	1
من 11100001 00000001 11111111 11111111 إلى 11100001 11111111 11111111 11111111	2
ما عدا ذلك	3

a. كوِّن جدول تمرير مؤلَّفاً من أربعة صفوف، ويستخدم قاعدة تطابق البادئة الأطول، ويرسل الرزم إلى واجهات الوصلات الصحيحة.

b. صف كيف يحدد جدول التمرير السابق (في جزء السؤال a-8) واجهة الوصلة الملائمة لرزم البيانات لكلٍّ من عناوين الوجهة التالية:

01010101 01010001 10010001 11001000
00111100 11000011 00000000 11100001
01110111 00010001 10000000 11100001

9. افرض أن شبكة وحدات البيانات تستخدم عناوين للمضيفات مؤلفة من 8 بتات. وافرض أن موجّهاً يستخدم قاعدة تطابق البادئة الأطول وله جدول التمرير التالي:

البادئة المطابقة	الواجهة
00	0
01	1
10	2
11	3

حدد المدى المناظر لعناوين مضيفات الوجهة وعدد العناوين في كل مدى، لكل وجهة من تلك الواجهات الأربع.

10. افرض أن شبكة وحدات البيانات تستخدم عناوين للمضيفات مؤلفة من 8 بتات. وافرض أن موجّهاً يستخدم قاعدة تطابق البادئة الأطول وله جدول التمرير التالي:

البادئة المطابقة	الواجهة
1	0
11	1
111	2
ما عدا ذلك	3

حدد المدى المناظر لعناوين مضيفات الوجهة وعدد العناوين في كل مدى، لكلٍّ من تلك الواجهات الأربعة.

11. افرض أن موجّهاً يربط بين ثلاث شبكات فرعية: S1، S2، S3. وافرض أن كل شبكة من تلك الشبكات الفرعية يجب أن تستخدم البادئة 223.1.17/24. وافرض أيضاً أن الشبكة S1 يجب أن تدعم 125 واجهة، وأن كل شبكة من الشبكات S1 وS2 يجب أن تدعم 60 واجهة. اذكر ثلاثة عناوين شبكات (بالصيغة a.b.c.d/x) تحقق تلك القيود.

12. ذكرنا مثالاً لجدول تمرير في الجزء 4-2-2 (يستخدم قاعدة تطابق البادئة الأطول). أعد كتابة جدول التمرير هذا مستخدماً الصيغة a.b.c.d/x بدلاً من صيغة الأرقام الثنائية.
13. طُلِب منك في التمرين 7 إعطاء جدول تمرير (يستخدم قاعدة تطابق البادئة الأطول). أعد كتابة جدول التمرير هذا مستخدماً الصيغة a.b.c.d/x بدلاً من صيغة الأرقام الثنائية.
14. افرض أن شبكة فرعية تستخدم البادئة 101.101.101.64/26. أعط مثالاً لعنوان IP (بالصيغة xxx.xxx.xxx.xxx) يمكن تخصيصه لتلك الشبكة. افرض أن كتلة عناوين موفر خدمة الإنترنت لها الشكل 101.101.128/17، وأنه يريد تشكيل أربع شبكات فرعية من هذه الكتلة بكلٍّ منها نفس عدد عناوين IP. فما هي البادئات (بالصيغة a.b.c.d/x) للشبكات الفرعية الأربع؟
15. افرض أن شبكة لها الشكل الطبوغرافي المبين في الشكل 4-17. ولنرمز للشبكات الفرعية الثلاث التي فيها مضيفات (في اتجاه عقارب الساعة من الوضع 12:00) بـ A، B، C؛ للشبكات الفرعية بدون مضيفات بـ D، E، F.
    a. خصص عنوان شبكة لكلٍّ من تلك الشبكات الفرعية الست بحيث: يجب أن تخصص كل العناوين من 214.97.254/23، ويجب أن يكون للشبكة A عناوين تكفي لدعم 250 واجهة، ويجب أن يكون للشبكة B عناوين تكفي لدعم 120 واجهة، ويجب أن يكون للشبكة C عناوين تكفي لدعم 120 واجهة. بالطبع يجب أن تكون كل شبكة من الشبكات D وE وF قادرة على دعم واجهتين. ويجب أن تأخذ العناوين الصيغة a.b.c.d/x أو e.f.g.h/y – a.b.c.d/x.
    b. حدد جداول التمرير (طبقاً لقاعدة تطابق البادئة الأطول) لكلٍّ من الموجّهات الثلاثة، مستخدماً إجابتك السابقة على الجزء (a).
16. افرض أنك ترسل وحدة بيانات مكونة من 3000 بايت إلى وصلة لها الحد الأقصى لوحدة النقل MTU يعادل 500 بايت. افرض أن وحدة البيانات الأصلية مختومة بالرقم التعريفي 422. كم عدد الرزم الجزئية (fragments) المولّدة؟ وما خصائصها؟
17. افرض أن حجم وحدة البيانات بين مضيف المصدر A ومضيف الوجهة B لا يزيد عن 1500 بايت (بما في ذلك التريسة). افرض أن حجم التريسة 20 بايتاً، فكم عدد وحدات البيانات اللازمة لإرسال ملف MP3 حجمه 4 مليون بايت؟
18. افرض أن موفر خدمة الإنترنت خصص العنوان 126.13.89.67 لموجّه للشبكة الموجودة في الشكل 4-22، وأن عنوان شبكة البيت 192.168/16.
    a. خصص عناوين لكل الواجهات التي في شبكة البيت.

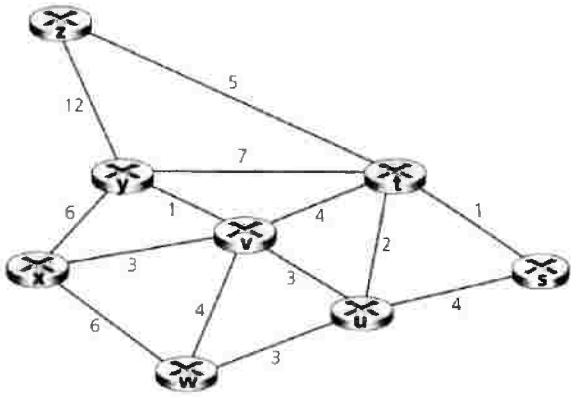
b. افرض أن كل مضيف له توصيلتان TCP وجميعها متجهة للمنفذ 80 على المضيف 128.119.40.86. حدد المدخلات الستة المناظرة في جدول ترجمة NAT.

19. في هذا التمرين سنفحص تأثير NAT على تطبيقات النظائر. افرض أن نظيراً باسم المستخدم آرنولد (Arnold) اكتشف من خلال الاستفسارات أن نظيراً آخر باسم المستخدم بيرنارد (Bernard) لديه ملف يريد تنزيله. وافرض أيضاً أن كلاً من بيرنارد وآرنولد وراء NAT. حاول ابتكار طريقة تسمح لآرنولد بتأسيس توصيلة TCP مع بيرنارد بدون إعداد NAT خاص بالتطبيق. إذا استصعب عليك الأمر لابتكار مثل هذه الطريقة، فناقش لماذا.

20. في الشكل 4-27 اذكر المسارات من v إلى y التي لا تتضمن حلقات (مسارات مغلقة).

21. أعد إجابة التمرين 20 للمسارات من x إلى w، ومن w إلى u، ومن z إلى x.

22. بالنظر إلى الشبكة التالية ذات كُلَف الوصلات المبينة، احسب أقصر مسار من x إلى كل عقدة من عقد الشبكة باستخدام خوارزمية Dijkstra. بيِّن خطوات الحل بإعداد جدول مشابه للجدول 4-3.



23. خذ في الاعتبار الشبكة المبينة بالتمرين 22، واستخدم خوارزمية Dijkstra مع بيان خطوات الحل (بإعداد جدول مشابه للجدول 4-3) لكلٍّ مما يلي:

a. احسب أقصر مسار من s إلى كل عقدة من عقد الشبكة.

b. احسب أقصر مسار من t إلى كل عقدة من عقد الشبكة.

c. احسب أقصر مسار من u إلى كل عقدة من عقد الشبكة.

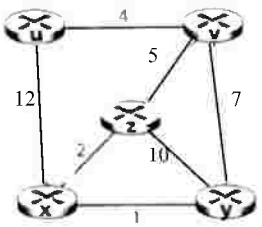
d. احسب أقصر مسار من v إلى كل عقدة من عقد الشبكة.

e. احسب أقصر مسار من w إلى كل عقدة من عقد الشبكة.

f. احسب أقصر مسار من y إلى كل عقدة من عقد الشبكة.

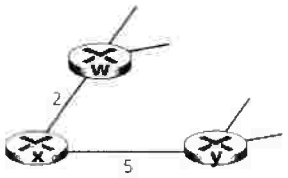
g. احسب أقصر مسار من z إلى كل عقدة من عقد الشبكة.

24. بفرض أن كل عقدة تعرف من البداية الكلفة إلى كلٍّ من جيرانها، استخدم خوارزمية متجه المسافة واحسب مدخلات جدول المسافات في العقدة z في الشبكة المبينة.



25. خذ في الاعتبار الشكل الطبوغرافي العام (أي ليس لشبكة معينة) واستخدم نسخة متزامنة من خوارزمية متجه المسافة. افرض أنه في كل تكرار تُرسل عقدة واحدة متجه المسافة لها إلى جيرانها وتستقبل متجهات المسافة من كلٍّ منهم. على فرض أنه عندما تبدأ الخوارزمية تعرِف كل عقدة الكُلَف إلى جيرانها المباشرين فقط، ما العدد الأقصى للتكرار اللازم لتقارب تلك الخوارزمية الموزّعة؟ بيّن السبب.

26. خذ في الاعتبار جزء الشبكة المبين بالشكل التالي، وفيه تتصل العقدة x بجارين w وy. افرض أن أدنى كلفة من w إلى الوجهة u تساوي 5، وأدنى كلفة من y إلى u تساوي 6. العقدة u والمسارات الكاملة من w وy إلى u (وبين w وy) غير مبينة بالشكل. افرض أيضاً أن جميع كلف الوصلات بالشبكة لها قيم صحيحة موجبة.



a. اذكر متجه المسافة لـ x للوجهات w وy وu.

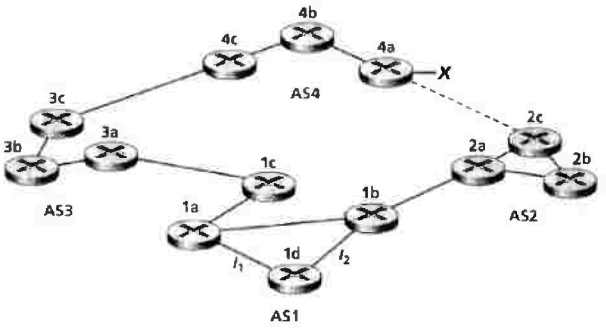
b. اذكر تغييراً في كلفة الوصلة $c(x, w)$ أو $c(x, y)$ ينتج عنه أن تخبر x جيرانها بمسار جديد للعقدة u كنتيجة لتنفيذ خوارزمية متجه المسافه.

c. اذكر تغييراً في كلفة الوصلة $c(x, w)$ أو $c(x, y)$ لا ينتج عنه أن تخبر x جيرانها بمسار جديد للعقدة u كنتيجة لتنفيذ خوارزمية متجه المسافه.

27. خذ في الاعتبار الطبوغرافية المبينة بالشكل 4-30 والمكونة من ثلاث عقد. لكن بدلاً من كُلَف الوصلات المبينة بالشكل افرض أن كلفة الوصلات كالتالي: $c(x,y) = 5$، $c(x,z) = 6$ ، $c(z,x) = 2$. احسب جداول المسافات بعد خطوة التهيئة وبعد كل خطوة من خطوات النسخة المتزامنة من خوارزمية متجه المسافة (كما فعلنا في المناقشة السابقة للشكل 4-30)

28. صف كيف يمكن الكشف عن وجود حلقات (مسارات مغلقة) في بروتوكول BGP.

29. خذ في الاعتبار الشبكة المبينة في الشكل التالي. افرض أن AS3 وAS2 يستخدمان بروتوكول OSPF، وأن AS1 وAS4 يَستخدمان بروتوكول RIP للتوجيه داخل النظام المستقل ذاتياً. افرض أن eBGP وiBGP يُستخدمان للتوجيه بين النظم المستقلة ذاتياً. في البداية افرض عدم وجود وصلة مادية بين AS2 وAS4.



a. من أي بروتوكول يعرف الموجّه 3c البادئة x هل هو OSPF أم RIP أم eBGP أم iBGP؟

b. من أي بروتوكول يعرف الموجّه 3a البادئة x؟

c. من أي بروتوكول يعرف الموجّه 1c البادئة x؟

d. من أي بروتوكول يعرف الموجّه 1d البادئة x؟

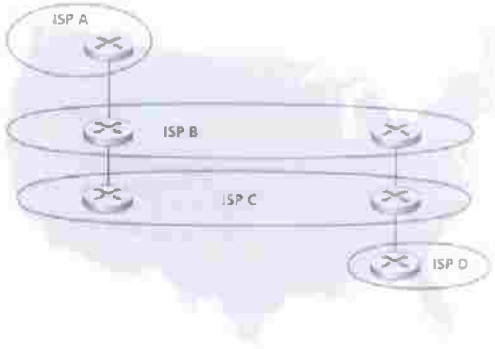
30. بالرجوع للتدريب السابق، بمجرد أن يعرف الموجّه 1d عن البادئة x سوف يضيف المُدخل (x, L) إلى جدول التمرير لديه.

a. هل تساوي L لهذا المُدخل l_1 أم l_2؟ وضح السبب في جملة واحدة.

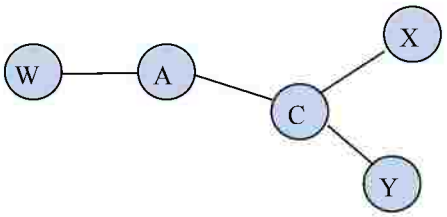
b. الآن افرض وجود وصلة مادية بين AS2 وAS4 والمبينة بالخط المنقوط. وافرض أن الموجّه 1d علم أنه يمكن الوصول لـ x عن طريق كل من AS2 وAS3. فهل تساوي L لهذا المُدخل I_1 أم I_2؟ وضح السبب في جملة واحدة.

c. الآن افرض وجود نظام مستقل ذاتياً آخر AS5 يقع على المسار بين AS2 وAS4 (غير مبين بالشكل). وافرض أن الموجّه 1d علم أنه يمكن الوصول لـ x عن طريق كلٍّ من AS2 AS5 AS4 وكذلك AS3 AS4. فهل تساوي L لهذا المُدخل I_1 أم I_2؟ وضح السبب في جملة واحدة.

31. خذ في الاعتبار الشبكة التالية. وفيها يوفر موفر خدمة الإنترنت B خدمة شبكة العمود الفقري القومي لموفر خدمة الإنترنت الإقليمي A، ويوفر موفر خدمة الإنترنت C خدمة شبكة العمود الفقري القومي لموفر خدمة الإنترنت الإقليمي D. بفرض أن كل موفر خدمة يتكون من نظام مستقل ذاتياً. يتصل B وC معاً من خلال وصلة ربط نظائر في موضعين باستخدام بروتوكول BGP. خذ في الاعتبار حركة مرور البيانات من A إلى D. سيفضل B أن يرسل حركة مرور البيانات إلى C من خلال وصلة الساحل الغربي (وبالتالي تتكلف C كلفة العبور خلال الولايات). في حين ستفضل C أن تستقبل حركة مرور البيانات من B من خلال وصلة الساحل الشرقي (وبالتالي تتكلف B كلفة العبور خلال الولايات). ما الآلية التي يستخدمها BGP لدى C حتى تقوم B بإرسال حركة مرور البيانات من خلال وصلة الساحل الشرقي؟ لكي تجيب على هذا السؤال ستحتاج للغوص في مواصفات BGP.



32. في الشكل 4-43 خذ في الاعتبار معلومات المسار التي تصل للشبكات الطرفية W وX وY. اعتماداً على المعلومات المتوفرة لكلٍّ من W وX، ما هو منظورهما لطبوغرافية الشبكة؟ علل إجابتك. مثلاً من منظور العقدة Y تكون طبوغرافية الشبكة كالتالي:



33. خذ في الاعتبار الشبكة المكونة من ثماني عقد (والمسماة بالحروف من s إلى z) في التمرين 21. بيّن الشجرة ذات أقل كلفة والتي جذرها s وتضم العقد u وv وw وy (كمضيفات طرفية). بشكلٍ تقريبي ناقش أسباب كون تلك الشجرة ذات أقل كلفة.
34. خذ في الاعتبار الطريقتين الأساسيتين اللتين ذكرناهما للإرسال الإذاعي: الأولى بمحاكاة الإرسال الفردي، والثانية بمعاونة الموجّه للإذاعة في طبقة الشبكة. افرض أننا استخدمنا الإرسال عبر الشجرة الممتدة للإذاعة في طبقة الشبكة، وافرض وجود مُرسِل وحيد و32 مُستقبل، وأن المُرسِل موصل بالمُستقبلين عن طريق شجرة ثنائية من الموجّهات. ما كلفة إرسال رزمة إذاعة في كلٍّ من الحالتين (أي عند محاكاة الإرسال الفردي وعند الإذاعة في طبقة الشبكة)؟ هنا في كل مرة تُرسَل رزمة (أو نسخة منها) على وصلة تتكلف وحدة التكلفة. ما الشكل الطبوغرافي لربط المُرسِل والمستقبلين والموجّهات والذي يجعل كلفة كلٍّ من تلك الطريقتين بعيدة جداً بقدر الإمكان عن الأخرى؟ يمكنك أن تختار أي عدد من الموجّهات.
35. خذ في الاعتبار طريقة عمل خوارزمية تمرير المسار العكسي (RPF) في الشكل 4-45. باستخدام نفس الطبوغرافية حدد مجموعة المسارات من جميع العقد إلى عقدة المصدر A (ووضح تلك المسارات في رسم بياني مستخدماً خطوط سميكة مظلّلة كما في الشكل 4-45) بحيث إذا كانت هذه المسارات تمثل المسارات ذات الكلفة الأقل، فعندئذ ستستقبل العقدة B نسخة من الرسالة المذاعة من العقدة A من العقد A وC وD.
36. خذ في الاعتبار الطبوغرافية المبينة في الشكل 4-45. افرض أن كلفة كل وصلة تعادل الوحدة وأن العقدة E هي مصدر الإذاعة. باستخدام أسهم كالمبينة بالشكل 4-45، وضح الوصلات التي سترسل الرزم خلالها باستخدام تمرير المسار العكسي (RPF)، وبيّن المسارات التي لن ترسل الرزم من خلالها.
37. خذ في الاعتبار الطبوغرافية المبينة في الشكل 4-47 وافترض أن كلفة كل وصلة تعادل الوحدة. افترض أن العقدة C اختيرت كمركز في خوارزمية توجيه متعدد

معتمدة على مركز (center-based multicast). على افترض أن كل موجّه ملحق يستخدم مسار أقل كلفة إلى العقدة C لإرسال رسائل الالتحاق (join messages) إلى C، ارسم شجرة التوجيه الناتجة. هل تلك الشجرة الناتجة شجرة أدنى كلفة؟ اذكر أسباب إجابتك.

38. في الجزء 4-5-1 درسنا خوارزمية Dijkstra لتوجيه حالة الوصلة، والتي تقوم بحساب مسارات الإرسال الفردي (unicast) ذات أقل كلفة من المصدر إلى كلٍّ من الوجهات بشكلٍ مستقل. وتُشكّل هذه المسارات مجتمعة شجرة مسارات أدنى كلفة (أو شجرة أقصر المسارات للإرسال الفردي إذا كانت كلف الوصلات متماثلة). بعرض مثال مضاد وضح أن شجرة مسارات أدنى كلفة ليست دائماً تماماً مثل الشجرة الممتدة بأدنى كلفة (minimum spanning tree).

39. خذ في الاعتبار شبكة فيها كل عقدة موصلة بثلاث عقد أخرى. في لحظة زمنية معينة يمكن أن تستقبل كل عقدة الرزم المذاعة المرسلة من جيرانها وتستنسخ تلك الرزم وترسلها إلى كلٍّ من جيرانها (ماعدا العقدة التي أرسلت تلك الرزمة). في اللحظة التالية يمكن أن تستقبل العقد المجاورة الرزم وتستنسخها وترسلها إلى جيرانهم، وهكذا. افترض أن الفيضان غير المحكوم يُستخدَم في الإرسال الإذاعي في مثل تلك الشبكة. في اللحظة t كم عدد النسخ للرزمة المذاعة ستُرسَل بافترض أنه أثناء الخطوة الزمنية 1 تم إرسال رزمة مذاعة وحيدة من عقدة المصدر إلى جيرانها الثلاثة؟

40. رأينا في الجزء 4-7 أنه لا يوجد بروتوكول بطبقة الشبكة يمكن استخدامه لتحديد المضيفات المشتركة في مجموعة إرسال متعدد. اشرح كيف تعرف تطبيقات الإرسال المتعدد هوية كلٍّ من تلك المضيفات المشتركة بالمجموعة.

41. صمم (بوصف خطوات الإجراء) بروتوكولاً بطبقة التطبيقات يحتفظ بعناوين المضيفات المشتركة في مجموعة إرسال متعدد. بالتحديد عيِّن نوع خدمة الشبكة التي يستخدمها هذا البروتوكول، هل هو إرسال فردي أو متعدد، ثم بين ما إذا كان هذا البروتوكول يرسل رسائله داخل النطاق (in-band) أو خارج النطاق (out-of-band) وذلك بالنسبة للبيانات المتدفقة من التطبيق بين المجموعة المشتركة، مع بيان الأسباب.

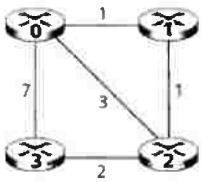
42. ما حجم فضاء عناوين الإرسال المتعدد؟ افرض الآن أن مجموعتين للإرسال المتعدد اختارتا عنواناً للإرسال المتعدد بشكلٍ عشوائي. ما احتمال اختيارهما لنفس العنوان؟ افرض أن 1000 مجموعة بدأت في نفس اللحظة اختيار العنوان بشكلٍ عشوائي، ما احتمال أن يحدث تداخل بين بعضهم البعض؟

❖ أسئلة للمناقشة

1. ابحث عن ثلاث شركات تبيع حالياً موجّهات بسرعات عالية، وقارن بين منتجاتها.
2. استخدم خدمة whois (من يكون؟) الموجودة في المسجِّل الأمريكي لأعداد الإنترنت (http://www.arin.net/whois) لتحديد كتل عناوين IP لثلاث جامعات. هل يمكن استخدام خدمة whois لتحديد بشكلٍ مؤكد الموقع الجغرافي لعنوان IP معين؟
3. هل من الممكن أن تكتب برنامج زبون ping (باستخدام رسائل ICMP) في لغة جافا؟ وضِّح السبب.
4. وضَّحنا في الجزء 4-4 أن انتشار بروتوكول IPv6 بطيء؛ بيّن أسباب ذلك؟ وما هو المطلوب لتسريع انتشاره؟
5. ناقش بعض مشاكل NAT المتعلقة بأمن IPSec (راجع كتاب [Phifer 2000])؟
6. قم بإعداد بحث عن بروتوكول UPnP. وبالتحديد صف الرسائل التي يستخدمها المضيف لإعادة تهيئة NAT.
7. افترض أن النظم المستقلة ذاتياً X وZ غير متصلة بشكلٍ مباشر وإنما متصلة عن طريق نظام آخر Y. وافترض أيضاً أن X لديه اتفاق نظائر (peering agreement) مع Y وأن Y لديه اتفاق نظائر مع Z. وأخيراً افترض أن Z يريد أن ينقل كل حركة بيانات Y ولا يريد أن ينقل حركة بيانات X. هل يسمح BGP لـ Z بتطبيق هذه السياسة؟
8. حددنا في الجزء 4-7 عدداً من تطبيقات الإرسال المتعدد (multicast)؛ أي من تلك التطبيقات يناسب بشكلٍ جيد نموذج خدمة الإرسال المتعدد في الإنترنت؟ وأي تطبيقات لا تناسب بشكلٍ جيد تحديداً نموذج الخدمة هذا؟ بين السبب.

❖ تدريبات على برمجة المقابس

في هذا التدريب ستكتب مجموعة من الإجراءات الموزّعة لتحقيق توجيه بمتجه المسافة لاتزامني موزّع للشبكة التالية:



يمكنك الحصول على التفاصيل الكاملة لهذا التمرين وكذلك أجزاء من البرنامج في لغة C وأيضاً في لغة جافا من خلال موقع الويب لهذا الكتاب http://www.awl.com/kurose-ross.

❖ تدريبات معملية على استخدام برنامج Ethereal

في موقع الويب الخاص بهذا الكتاب http://www.awl.com/kurose-ross ستجد تدريبين على استخدام برنامج Ethereal لهذا الفصل. يفحص التدريب الأول طريقة عمل بروتوكول IP وبشكلٍ محدد صيغة وحدة بيانات IP. أما التدريب الثاني فيفحص استخدام بروتوكول ICMP في أوامر ping (لاختبار اتصال الأجهزة المختلفة بالشبكة) وtraceroute (لتتبع المسارات إلى الأجهزة المختلفة).