Routing w sieciach IP

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UMCS

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Projekt „Programowa i strukturalna reforma systemu kształcenia na Wydziale Mat-Fiz-Inf”.
Projekt współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego.

Człowiek-najlepsza inwestycja

Uniwersytet Marii Curie-Skłodowskiej
Wydział Matematyki, Fizyki i Informatyki
Instytut Informatyki

Routing w sieciach IP

Karol Kuczyński
Rafał Stęgierski

Lublin 2011

Instytut Informatyki UMCS
Lublin 2011

Karol Kuczyński
Rafał Stęgierski

Routing w sieciach IP

Recenzent: Andrzej Bobyk

Opracowanie techniczne: Marcin Denkowski
Projekt okładki: Agnieszka Kuśmierska

Praca współfinansowana ze środków Unii Europejskiej w ramach
Europejskiego Funduszu Społecznego

Publikacja bezpłatna dostępna on-line na stronach
Instytutu Informatyki UMCS: informatyka.umcs.lublin.pl.

Wydawca
Uniwersytet Marii Curie-Skłodowskiej w Lublinie
Instytut Informatyki
pl. Marii Curie-Skłodowskiej 1, 20-031 Lublin
Redaktor serii: prof. dr hab. Paweł Mikołajczak
www: informatyka.umcs.lublin.pl
email: dyrii@hektor.umcs.lublin.pl

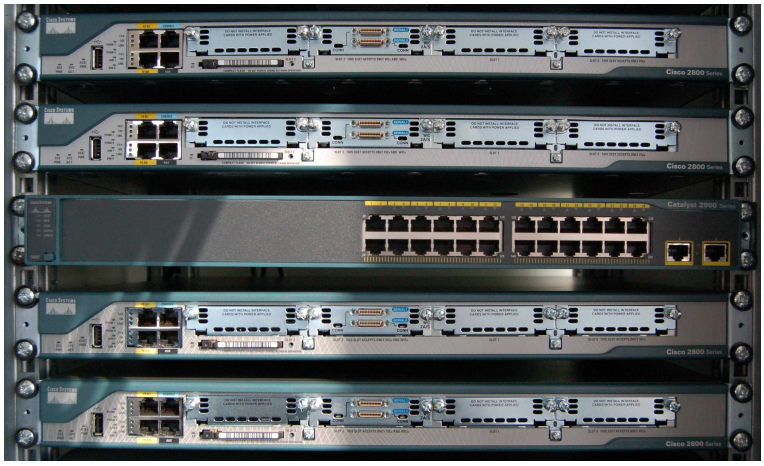
Druk
ESUS Agencja Reklamowo-Wydawnicza Tomasz Przybylak
ul. Ratajczaka 26/8
61-815 Poznań
www: www.esus.pl

ISBN: 978-83-62773-08-4

Spis treści

Przedmowa

Współczesny Internet funkcjonuje w oparciu o protokoły rodziny TCP/IP. Model warstwowy protokołów komunikacyjnych TCP/IP, jak również cały stos protokołów, został opracowany w latach siedemdziesiątych ubiegłego wieku, w Agencji Zaawansowanych Obronnych Projektów Badawczych Departamentu Obrony Stanów Zjednoczonych (DARPA). Celem projektu było stworzenie rozległej, rozproszonej sieci, która byłaby w stanie zapewnić łączność nawet w przypadku zniszczenia dużej części jej infrastruktury, w warunkach wojennych. Jednym z głównych problemów było opracowanie metod automatycznego, dynamicznego wyznaczania trasy do miejsca docelowego, z uwzględnieniem aktualnych warunków. Proces ten nazywany jest routingiem lub trasowaniem.



Rysunek 1. Routery Cisco i przełącznik Ethernet.

W dalszej części książki opisane są teoretyczne podstawy współcześnie wykorzystywanych protokołów routingu oraz ćwiczenia praktyczne umożliwiające opanowanie ich konfiguracji oraz prześledzenie działania w rzeczywistej sieci. Wykonanie ćwiczeń umożliwi zdobycie nie tylko wiedzy teoretycznej, lecz również umiejętności niezbędnych w pracy administratora

sieci komputerowej. Przykłady konfiguracyjne, o ile nie zaznaczono inaczej, dotyczą urządzeń (routerów i przełączników) Cisco wyposażonych w system IOS. Do wykonania ćwiczeń wystarczą 3 routery serii 1800, 2600 lub 2800 oraz dowolne przełączniki Ethernet (rys. 1). Jedno z ćwiczeń dotyczy implementacji routera linuksowego, z wykorzystaniem wyłącznie darmowego oprogramowania. W przypadku części zadań zamieszczono rozwiązania lub wskazówki.

Opisane protokoły (poza kilkoma wyjątkami) są powszechnie przyjętymi standardami (opisanymi w dokumentach RFC lub normach ISO). W związku z tym, zamieszczenie ćwiczeń praktycznych z wykorzystaniem sprzętu konkretnego producenta (Cisco Systems), nie stanowi istotnego ograniczenia ogólności prezentowanych rozważań. Różnice w konfigurowaniu urządzeń sieciowych różnych producentów są natury technicznej i mają w tym przypadku drugorzędne znaczenie. Użyte symbole urządzeń sieciowych są powszechnie przyjęte i przedstawiono je w Dodatku A.

Większość prezentowanych treści znajduje się na liście zagadnień obowiązujących na egzaminie Cisco Certified Network Associate (640-802 CCNA). Ponadto, opisano także kilka bardziej zaawansowanych problemów.

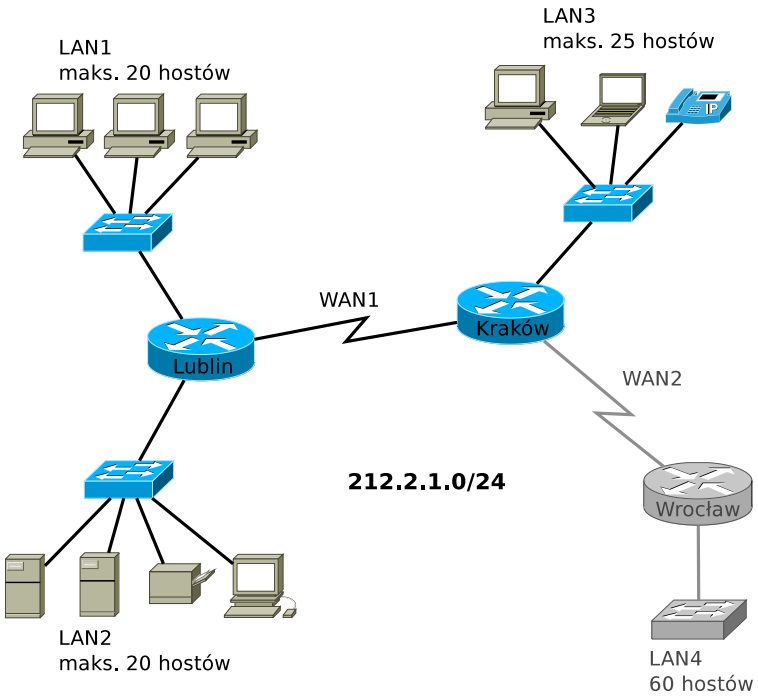
Wymagania wstępne obejmują znajomość następujących zagadnień:
— modele OSI oraz TCP/IP,
— podstawowe protokoły stosu TCP/IP,
— podstawy adresowania IPv4 i podziału na podsieci,
— konfigurowanie ustawień sieciowych hosta (w systemie Linux lub MS Windows),
— pojęcie routingu,
— podstawy technologii Ethernet,
— użytkowanie i podstawy administrowania systemem Linux (Rozdział 11).

Rozdział 1

Zaawansowane adresowanie IP – VLSM i CIDR

1.1. Podstawy teoretyczne

W bieżącym rozdziale zakłada się, że czytelnikowi znane są już podstawowe zagadnienia związane z adresowaniem IPv4:
— struktura adresu IP i maski,
— pojęcie klasy adresu,
— zasady przydzielania adresów IP urządzeniom,
— pojęcie bramy domyślnej,
— konfiguracja ustawień hosta, związanych z IP,
— podstawy podziału na podsieci.



Rysunek 1.1. Sieć, której należy przydzielić adresy IP z puli 212.2.1.0/24.

Podstawowym problemem wynikającym ze stosowania w publicznym Internecie protokołu IPv4 jest niewystarczająca obecnie liczba dostępnych adresów IP. Sytuacja ta wynika między innymi z klasowego podejścia do adresowania, jak również bardzo rozrzutnego gospodarowania adresami IP w początkowym okresie funkcjonowania Internetu.

Rozważmy sieć przedstawioną na rys. 1.1. Do dyspozycji jest pula adresów 212.2.1.0/24 (czyli sieć klasy C, z domyślną maską). Dołączenie routera Wrocław i sieci LAN4 jest planowane w dalszej przyszłości. Zgodnie z dokumentem RFC 950 [10], sieć może zostać podzielona na podsieci. Zatem,

dzięki podziałowi na podsieci, dwuwarstwowa, klasowa struktura adresu IP (numer sieci, numer hosta) zmienia się w trójwarstwową (numer sieci, numer podsieci, numer hosta). W scenariuszu z rys. 1.1 potrzebujemy 6 podsieci. "Pożyczenie" z części adresu przeznaczonej do zapisania numeru hosta 3 bitów (maska 255.255.255.224 lub /27) umożliwi stworzenie 8 podsieci z 30 użytecznymi adresami w każdej z nich. W tym przypadku, w sieci LAN4 nie będziemy dysponowali żądaną liczbą 60 adresów. W przypadku "pożyczenia" 2 bitów (maska 255.255.255.192 lub /26), powstaną natomiast tylko 4 podsieci z 62 adresami użytecznymi w każdej z nich. Należy zwrócić uwagę również na sieci WAN1 i WAN2. Są to połączenia punkt-punkt, wymagające jedynie 2 adresów użytecznych. W razie użycia podsieci z maską /26 lub /27, duża liczba adresów IP zostanie stracona, zwykle bezpowrotnie, ponieważ zmiana schematu adresowania w działającej sieci jest niezwykle kłopotliwa.

Tabela 1.1. Hierarchiczny podział sieci.

Sieć	Podsieci 1. poziomu	Podsieci 2. poziomu	Podsieci 3. poziomu
1	2	3	4
212.2.1.0/24	212.2.1.0/26 LAN4	X	X
	212.2.1.64/26	212.2.1.64/27 LAN1	X
		212.2.1.96/27 LAN2	X
	212.2.1.128/26	212.2.1.128/27 LAN3	X
		212.2.1.160/27	212.2.1.160/30 WAN1
			212.2.1.164/30 WAN2
			212.2.1.168/30
			212.2.1.172/30
			212.2.1.176/30
			212.2.1.180/30
			212.2.1.184/30
			212.2.1.188/30
	212.2.1.192/26		

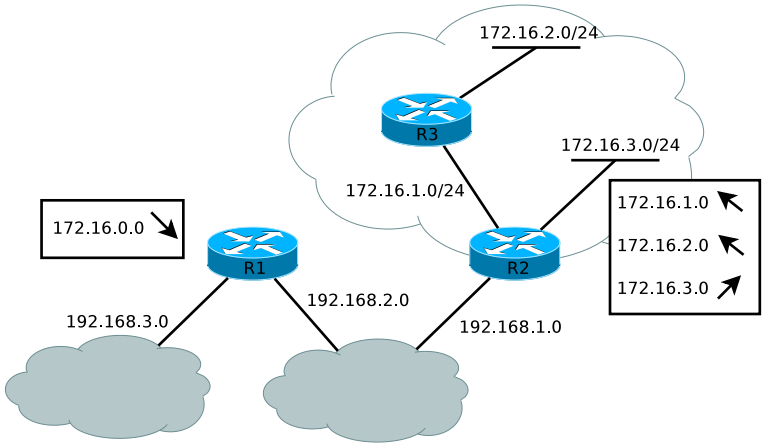
Rozwiązaniem problemu jest technika VLSM (ang. *Variable Length Subnet Masks*, podział na podsieci z maską o zmiennej długości) [1]. Umożliwia ona użycie więcej niż jednej maski podsieci, przy podziale danej sieci klasowej. Podsieć może zostać podzielona na mniejsze podsieci, które mogą

być dzielone na jeszcze mniejsze, itd., pod warunkiem, że w części adresu przeznaczonej dla hosta jest dostępna odpowiednia liczba bitów.

Planowanie adresowania należy rozpocząć od stworzenia największych podsieci, a następnie tworzyć coraz mniejsze (podejście hierarchiczne). Dzięki temu nie występuje nadmierna segmentacja przestrzeni adresowej, jak również możliwe jest dokonywanie agregacji tras (opisanej w dalszej części rozdziału). Przykład przedstawiony jest w tabeli 1.1. W naszym przypadku, w największej podsieci wymagane jest 60 adresów użytecznych, zatem należy użyć maski /26 (255.255.255.192). Można stworzyć 4 takie podsieci, a ich adresy zapisano w 2. kolumnie tabeli. Pierwsza z nich (212.2.1.0/26) zostanie użyta do zaadresowania sieci LAN4. Dwie kolejne (212.2.1.64/26 i 212.2.1.128/26) zostały podzielone na mniejsze podsieci z maską /27, natomiast ostatnia (212.2.1.192/26) pozostaje do wykorzystania w przyszłości (może zostać wykorzystana bezpośrednio do zaadresowania hostów lub dowolnie podzielona na podsieci). Trzy sieci z maską /27 zostaną użyte do zaadresowania LAN1, LAN2 i LAN3. Ostatnia (212.2.1.160/27) zostanie podzielona na podsieci z maską /30. Podsieci z taką maską zawierają po 2 adresy użyteczne i standardowo stosuje się je w przypadku połączeń punkt-punkt. W naszym przykładzie, 6 takich podsieci pozostaje do użycia w przyszłości.

Przy projektowaniu schematu adresowania w technice VLSM, możliwe jest popełnienie błędu polegającego na "zachodzeniu na siebie" dwóch podsieci. W naszym przykładzie, podsieć 212.2.1.64/27 została przeznaczona do zaadresowania hostów (LAN1). W związku z tym, nie może już zostać podzielona na mniejsze podsieci. Z kolei adresy z zakresu 212.2.1.161 – 212.2.1.190 (podsieć 212.2.1.160/27) nie mogą być wykorzystane do zaadresowania hostów wraz z maską /27, ponieważ ta podsieć została podzielona na mniejsze (z maską /30).

Przy routingu klasowym, o miejscu podziału adresu na część z numerem sieci i część z numerem hosta decyduje klasa adresu, którą z kolei można odczytać z pierwszego oktetu. Na rys. 1.2 zaprezentowano przykład, w którym sieć klasy B, 172.16.0.0/16, została podzielona na podsieci, z maską 24-bitową. Wszystkie te podsieci muszą stanowić spójny obszar – sieć nie może być nieciągła. W prostokątach schematycznie przedstawiono zawartość tablic routingu routerów. Na podstawie zawartych w niej informacji router, otrzymując pakiet zaadresowany do określonego miejsca docelowego, podejmuje decyzję o kierunku, w jakim pakiet zostanie przekazany w kolejnym kroku. Dokładniejsze informacje o strukturze i wykorzystywaniu tablicy routingu zawarte są w kolejnych rozdziałach. Router R1 nie jest świadomy istnienia podziału na podsieci. W jego tablicy routingu jest jedynie informacja o trasie do sieci 172.16.0.0, z domyślną maską. Informację o podsieciach posiada router R2, który ma z nimi bezpośrednie połączenie. Zakłada on,



Rysunek 1.2. Tablice routingu R1 i R2 w routingu klasowym (fragmenty dotyczące sieci 172.16.0.0).

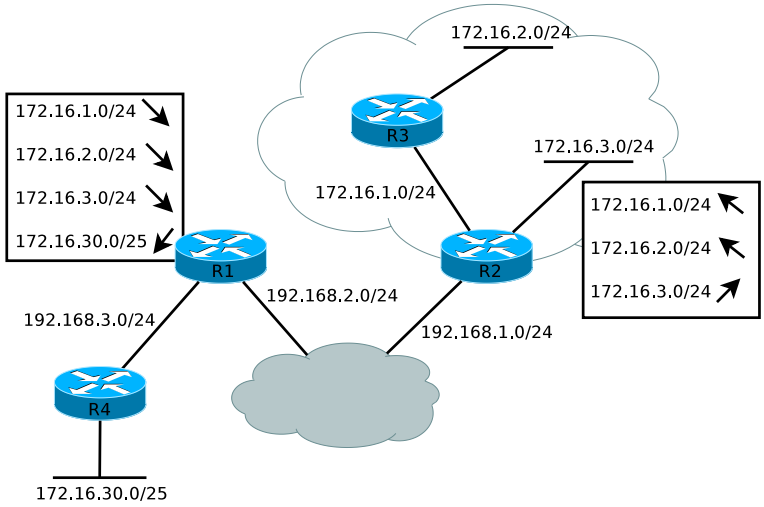
że maska wszystkich podsieci jest identyczna z maską adresu jego interfejsu stanowiącego połączenie z podsieciami.

Z punktu widzenia routingu klasowego, sytuacja przedstawiona na rys. 1.3 jest niepoprawna, ponieważ sieć 172.16.0.0 jest nieciągła. Co więcej, zastosowano w niej dwie różne maski (/24 i /25). Rozwiązaniem jest routing bezklasowy, w którym, oprócz informacji o adresie sieci, przekazywana musi być także informacja o masce. Klasa adresu nie jest wówczas istotna. We współczesnym Internecie adresowanie VLSM jest powszechnie przyjęte – standardem jest routing bezklasowy. Zjawisko nieciągłych sieci, lub bardziej ogólnie, nieciągłych przestrzeni adresowych, nie jest pożądane, ale często nieuniknione.

W tym momencie, aby sieć była w pełni funkcjonalna, tablica routingu R1 musi zawierać aż cztery wpisy o trasach do poszczególnych podsieci 172.16.0.0. W przypadku trzech pierwszych, sposób postępowania z pakietami jest identyczny i polega na wysłaniu ich poprzez interfejs dołączony do sieci 192.168.2.0. Liczba wpisów może zostać w takiej sytuacji zredukowana dzięki mechanizmowi agregacji (lub podsumowania) tras (ang. *route aggregation, route summarization*). Zapisując binarnie trzeci oktet adresów podsieci 172.16.1–3.0/24, dołączonych do routera R2 otrzymujemy:

$$\begin{array}{ll} \textbf{172.16.}\textbf{000000}01_2.0 & /24 \\ \textbf{172.16.}\textbf{000000}10_2.0 & /24 \\ \textbf{172.16.}\textbf{000000}11_2.0 & /24 \end{array}$$

Okazuje się, że najbardziej znaczące 22 bity powyższych adresów (zaznaczone pogrubioną czcionką) są identyczne. Agregacji dokonujemy poprzez



Rysunek 1.3. Tablice routingu R1 i R2 w routingu bezklasowym (fragmenty dotyczące sieci 172.16.0.0).

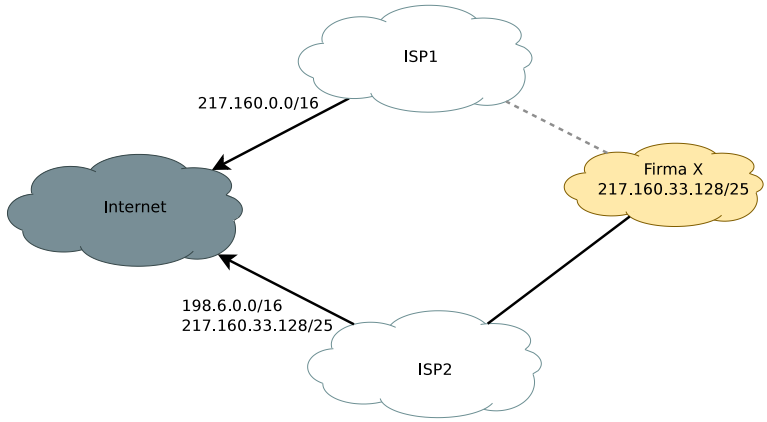
przepisanie początkowej, wspólnej części adresów, wypełnienie pozostałych bitów zerami i zastosowanie maski równej długości wspólnej części. W tym przypadku wynikiem będzie: 172.16.0.0/22. Ostatecznie, w tablicy routingu routera R1 z rys. 1.3, trzy pierwsze trasy mogą być zastąpione trasą do 172.16.0.0/22. Ten zapis, oprócz trzech powyższych podsieci, obejmuje także 172.16.0.0/24, która na rysunku nie występuje.

Agregacja może dotyczyć nie tylko adresów podsieci, ale również sieci klasowych, jeżeli ich najbardziej znaczące bity są identyczne. Wynikowa maska jest wówczas krótsza niż domyślna maska dla danej klasy adresu, co w przypadku routingu klasowego byłoby niedopuszczalne. Tradycyjna koncepcja adresów sieciowych klasy A, B i C traci zatem znaczenie.

Dzięki agregacji tras, możliwe jest bardzo znaczące zmniejszenie ilości informacji o routingu rozsyłanych między routerami i liczby wpisów w tablicach routingu. Jednocześnie łatwiejsze staje się zarządzanie tymi informacjami. Jest to szczególnie istotne w przypadku routerów stanowiących szkielet Internetu. Przy obecnej liczbie istniejących w nim sieci, sprawne funkcjonowanie bez agregacji tras byłoby prawdopodobnie niemożliwe (wystąpiłby problem skalowalności). Należy jednak zaznaczyć, że jej implementacja wymaga konsekwentnego stosowania hierarchicznego adresowania, powiązanego logicznie z topologią sieci. Przykładowo, problem stanowią bloki adresów przydzielone w przeszłości, przed rozpowszechnieniem modelu hierarchicznego. Strukturę adresowania hierarchicznego zakłóca również zmia-

na operatora dostępu do Internetu przez klienta posiadającego określoną pulę adresów IP.

Sytuację tę ilustruje przykład z rys. 1.4. Operatorzy ISP1 i ISP2 wykupili pule adresów 217.160.0.0/16 i 198.6.0.0/16, odpowiednio. Następnie sprzedają je swoim klientom. W związku z tym, korzystając z odpowiedniego protokołu routingu (bardziej szczegółowo opisanego w dalszej części podręcznika), ISP1 i ISP2 rozsyłają informacje o dysponowaniu trasami do 217.160.0.0/16 i 198.6.0.0/16. Firma X, będąca początkowo jednym z wielu klientów operatora ISP1, nabyła od niego niewielką pulę adresów 217.160.33.128/25. Przejście firmy X do operatora ISP2 skutkuje dla niego koniecznością generowania dodatkowego komunikatu, o dysponowaniu również trasą do sieci 217.160.33.128/25. Duża liczba tego typu wyjątków zaburza hierarchiczny model adresowania w środowisku bezklasowym i uniemożliwia efektywną agregację tras. Lepszym rozwiązaniem, z punktu widzenia operatorów i funkcjonowania publicznego Internetu, byłoby zakupienie przez firmę puli adresów od nowego operatora i dokonanie przeadresowania. To jednak wiąże się z koniecznością poniesienia dodatkowych kosztów i nie zawsze jest akceptowalne.



Rysunek 1.4. Problem zmiany operatora dostępu do Internetu.

Powyższe mechanizmy określane są mianem bezklasowego routingu międzydomenowego (ang. *Classless Inter-Domain Routing*, CIDR) [7, 14, 4, 17]. Wraz z VLSM, CIDR we współczesnym Internecie poprawia efektywność gospodarowania adresami IPv4, umożliwiając przydzielanie praktycznie dowolnej wielkości bloków adresów, zamiast pełnych sieci klasy A, B lub C. Ostatecznym rozwiązaniem problemu niedoboru adresów pozostaje jednak migracja do protokołu IPv6. Mechanizmy routingu w IPv6, w porównaniu z IPv4 z CIDR, pozostają niemal niezmienione. Podstawową różnicą jest długość adresu (128 zamiast 32 bitów).

1.2. Zadanie

Dana jest pula adresów IP: 13.1.0.0/16. Zaplanuj hierarchiczny schemat adresowania, spełniający warunki z tabeli 1.2. Nie przydzielaj więcej adresów niż jest to konieczne i nie pozostawiaj zbędnych przerw. Wypełnij tabele 1.3 – 1.7.

Tabela 1.2. Wymagana liczba hostów w poszczególnych podsieciach.

Lp.	Nazwa podsieci	Liczba hostów
1	Podsieć I	500
2	Podsieć II	1000
3	Podsieć III	100
4	Podsieć IV	31
5	Podsieć V	1000

Tabela 1.3. Podsieć I.

Adres podsieci i maska	
Adres rozgłoszeniowy	
Adresy hostów	
Liczba użytecznych adresów hostów	

Tabela 1.4. Podsieć II.

Adres podsieci i maska	
Adres rozgłoszeniowy	
Adresy hostów	
Liczba użytecznych adresów hostów	

Tabela 1.5. Podsieć III.

Adres podsieci i maska	
Adres rozgłoszeniowy	
Adresy hostów	
Liczba użytecznych adresów hostów	

Tabela 1.6. Podsieć IV.

Adres podsieci i maska	
Adres rozgłoszeniowy	
Adresy hostów	
Liczba użytecznych adresów hostów	

Tabela 1.7. Podsieć V.

Adres podsieci i maska	
Adres rozgłoszeniowy	
Adresy hostów	
Liczba użytecznych adresów hostów	

1.3. Rozwiązanie

Uwaga: Zawsze zaczynamy od tworzenia największych podsieci, a następnie tworzymy coraz mniejsze.

1. Stwórz największe podsieci (1000 hostów), z nową maską /22 (tabela 1.8).
2. Podziel podsieć 13.1.8.0/22, aby stworzyć Podsieć I (500 hostów), używając nowej maski /23 (tabela 1.9).
3. Podziel podsieć 13.1.10.0/23 aby stworzyć Podsieć III (100 hostów), używając nowej maski /25 (tabela 1.10).
4. Podziel podsieć 13.1.10.128/25 aby stworzyć Podsieć IV (31 hostów), używając nowej maski /26 (tabela 1.11). W tym przypadku konieczne jest stworzenie podsieci zawierającej 62 adresy użyteczne, ponieważ mniejsza, 30-adresowa jest niewystarczająca.

Tabela 1.8. Podsieci dla 1000 hostów (1022 adresy użyteczne).

Nr	Adres podsieci	Adresy hostów	Adres rozgłoszeniowy	Uwagi
1	13.1.0.0/22	13.1.0.1/22 – 13.1.3.254/22	13.1.3.255/22	**Podsieć II**
2	13.1.4.0/22	13.1.4.1/22 – 13.1.7.254/22	13.1.7.255/22	**Podsieć V**
3	13.1.8.0/22	13.1.8.1/22 – 13.1.11.254/22	13.1.11.255/22	**do podziału**
4	13.1.12.0/22	13.1.12.1/22 – 13.1.15.254/22	13.1.15.255/22	wolna
5	13.1.16.0/22	13.1.16.1/22 – 13.1.19.254/22	13.1.19.255/22	wolna
...	...	...	...	...

Tabela 1.9. Podsieci dla 500 hostów (510 adresów użytecznych), uzyskane z podziału 13.1.8.0/22.

Nr	Adres podsieci	Adresy hostów	Adres rozgłoszeniowy	Uwagi
1	13.1.8.0/23	13.1.8.1/23 – 13.1.9.254/23	13.1.9.255/23	**Podsieć I**
2	13.1.10.0/23	13.1.10.1/23 – 13.1.11.254/23	13.1.11.255/23	**do podziału**

Tabela 1.10. Podsieci dla 100 hostów (126 adresów użytecznych), uzyskane z podziału 13.1.10.0/23.

Nr	Adres podsieci	Adresy hostów	Adres rozgłoszeniowy	Uwagi
1	13.1.10.0/25	13.1.10.1/25 – 13.1.10.126/25	13.1.10.127/25	**Podsieć III**
2	13.1.10.128/25	13.1.10.129/25 – 13.1.10.254/25	13.1.10.255/25	**do podziału**
3	13.1.11.0/25	13.1.11.1/25 – 13.1.11.126/25	13.1.11.127/25	wolna
4	13.1.11.128/25	13.1.11.129/25 – 13.1.11.254/25	13.1.11.255/25	wolna

Tabela 1.11. Podsieci dla 31 hostów (62 adresy użyteczne), uzyskane z podziału 13.1.10.128/25.

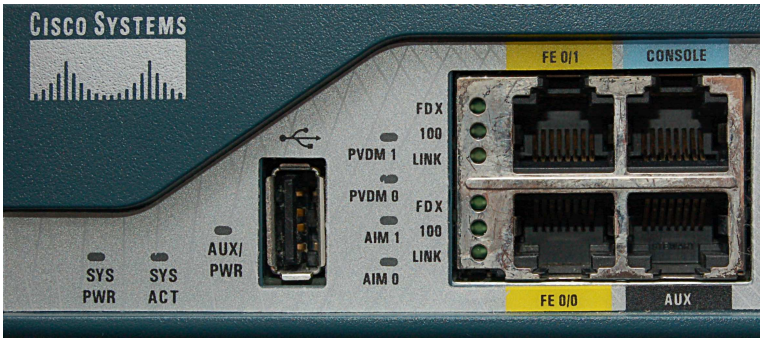
Nr	Adres podsieci	Adresy hostów	Adres rozgłoszeniowy	Uwagi
1	?	?	?	**Podsieć IV**
2	?	?	?	wolna

Rozdział 2

Podstawowa konfiguracja routera

2.1. Wstęp

Kolejne rozdziały zawierają zadania polegające na konfigurowaniu różnych protokołów routingu, z wykorzystaniem routerów Cisco. W każdym przypadku, przed przystąpieniem do zasadniczej części zadania, konieczne jest przeprowadzenie konfiguracji podstawowych mechanizmów bezpieczeństwa oraz interfejsów sieciowych urządzenia. Implementacja mechanizmów bezpieczeństwa powinna stać się czynnością rutynową.



Rysunek 2.1. Interfejsy routera: Fast Ethernet (FE0/0, FE0/1), konsolowy (Console), AUX, USB.

Początkową konfigurację routera przeprowadza się korzystając z portu konsolowego (Console, rys. 2.1). Interfejsy sieciowe domyślnie są wyłączone. Potrzebny jest odpowiedni kabel konsolowy (ang. *rollover*), ze złączami RJ45 (po stronie routera) i DB9 (do połączenia z interfejsem RS-232 komputera PC). Na komputerze należy uruchomić program emulujący terminal, np. Minicom, PuTTY, HyperTerminal. Standardowe dla urządzeń Cisco parametry połączenia szeregowego to: 9600 baud, 8 bitów danych, 1 bit stopu, brak parzystości.

2.2. Konfiguracja

W przypadku, gdy uruchamiany jest nowy router lub router z przywróconymi ustawieniami fabrycznymi, po uruchomieniu systemu operacyjnego (IOS) na konsoli wyświetlany jest komunikat:

```
Continue with configuration dialog? [yes/no]:
```

Odpowiedź twierdząca powoduje wejście w tryb, w którym router zadaje administratorowi szereg pytań (*setup mode*) i na podstawie odpowiedzi generuje plik konfiguracyjny. Możliwości konfiguracji są w tym trybie bar-

dzo ograniczone. W razie przypadkowego uruchomienia, można przerwać go kombinacją klawiszową Ctrl+C. Następnie router zgłasza się w trybie EXEC użytkownika:

```
Router>
```

Jest w nim dostępna ograniczona liczba poleceń umożliwiających jedynie weryfikację niektórych aspektów pracy urządzenia. Listę poleceń dostępnych w danym trybie poleceń można uzyskać po wpisaniu "?". Znak "?" umożliwia również przejrzenie listy poleceń rozpoczynających się od zadanego ciągu znaków (`początek_polecenia?`) lub dostępnych, kolejnych opcji polecenia (`polecenie ?`, następnie `polecenie opcja1 ?`, itd.).

Polecenie `enable` powoduje przejście do uprzywilejowanego trybu EXEC (w przeciwną stronę działa `disable`). Znak zgłoszenia zmienia się na:

```
Router#
```

Dostępne są w nim te same polecenia, co w poprzednim trybie oraz dodatkowo bardziej zaawansowane polecenia diagnostyczne (w szczególności szereg poleceń rozpoczynających się od słowa `show`) i konfiguracyjne. Polecenie `configure terminal` powoduje przejście do trybu konfiguracji globalnej:

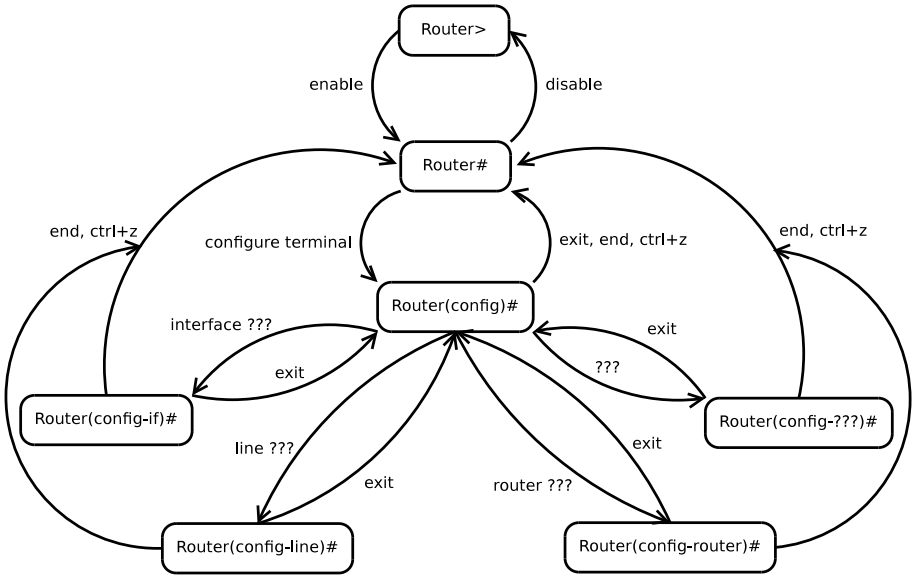
```
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
```

Można w nim modyfikować pewne ustawienia konfiguracyjne lub przejść do zagnieżdżonych, bardziej szczegółowych trybów konfiguracyjnych. Rys. 2.2 przedstawia schemat najważniejszych trybów wiersza poleceń. Zapis `Router(config-???)` symbolizuje pozostałe tryby, nieuwzględnione na schemacie. Wewnątrz niektórych trybów konfiguracyjnych występują kolejne poziomy zagnieżdżenia. W większości przypadków kolejność konfigurowania poszczególnych ustawień jest dowolna, istnieją jednak pewne zalecenia, związane z dobrymi praktykami. W sytuacji, gdy jest to jednoznaczne, można operować skrótami poleceń, np. można użyć polecenia `config t` zamiast `configure terminal`.

W pierwszej kolejności, zwłaszcza gdy konfigurowanych jest kilka urządzeń, powinno się nadać im nazwy, np.:

```
Router(config)#hostname R1
R1(config)#
```

Nazwa jest lokalna i nie ma wpływu na pracę routera w sieci. Jest wyświetlana przed znakiem zgłoszenia systemu i w przypadku konfigurowania



Rysunek 2.2. Tryby wiersza poleceń IOS.

wielu urządzeń, służy administratorowi do ich identyfikacji. Jak widać na powyższym przykładzie, efekt wydania polecenia jest natychmiastowy. Hasło zabezpieczające tryb uprzywilejowany konfiguruje się poleceniem:

```
enable password hasło
```

lub

```
enable secret hasło
```

W pierwszym przypadku hasło jest przechowywane w pliku konfiguracyjnym w postaci jawnego tekstu, natomiast w drugim (rekomendowanym) będzie zaszyfrowane algorytmem MD5. W razie skonfigurowania obu, router będzie korzystał z zaszyfrowanego. Przed nieuprawnionym dostępem należy zabezpieczyć również port konsolowy i wirtualne połączenia terminalowe (zdalny dostęp poprzez telnet lub ssh):

```
1 R1(config)#line console 0
  R1(config-line)#password hasło
3 R1(config-line)#login
  R1(config-line)#exit
5 R1(config)#line vty 0 15
  R1(config-line)#password hasło
7 R1(config-line)#login
  R1(config-line)#exit
```

```
9 R1(config)#
```

W wierszu nr 5, w miejscu "15" należy wpisać "?", a następnie wprowadzić maksymalną możliwą dla danego urządzenia wartość, by identycznie zabezpieczyć wszystkie połączenia, które router może jednocześnie obsługiwać. Powyższy przykład pokazuje również, że z trybu konfiguracji globalnej (`config`) można przejść do bardziej szczegółowych trybów konfiguracyjnych, np. trybu konfiguracji połączenia konsolowego (`config-line`). Przy pomocy polecenia `exit` można powrócić o jeden poziom wyżej. Polecenie `end` lub kombinacja klawiszowa `ctrl+z` powoduje powrót do uprzywilejowanego trybu EXEC. Wydanie polecenia `service password-encryption` (w trybie konfiguracji globalnej) spowoduje, że hasła, które są domyślnie przechowywane w postaci jawnego tekstu zostaną zaszyfrowane. Jednak algorytm szyfrowania jest bardzo łatwy do złamania (bez problemu można znaleźć opis algorytmu, jak również gotowe narzędzia), w przeciwieństwie do MD5 używanego do szyfrowania hasła `enable secret`.

Po uruchomieniu podstawowych mechanizmów zabezpieczających, można przystąpić do zasadniczej części zadania, czyli konfiguracji interfejsów sieciowych routera. Najpierw jednak należy uzyskać informacje o ich oznaczeniach. Można do tego celu użyć polecenia `show interfaces` lub wariantu `show ip interface brief`:

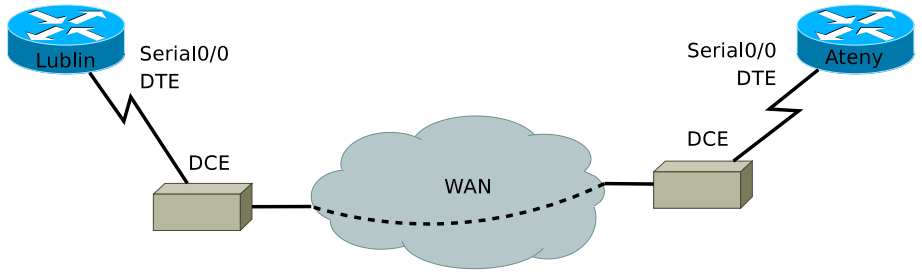
```
R1#show ip interface brief
Interface        IP-Address OK? Method Status                Pro-
                                                             tocol

FastEthernet0/0  unassigned YES unset  administratively down
                                       down
FastEthernet0/1  unassigned YES unset  administratively down
                                       down
Serial0/0/0      unassigned YES unset  administratively down
                                       down
```

Nazwa interfejsu składa się z wyrazu określającego jego typ (np. FastEthernet, Serial) i jednej, dwóch lub trzech liczb, zależnie od konstrukcji routera. Zapis `administratively down` w kolumnie Status oznacza, że interfejs jest wyłączony poleceniem `shutdown`. Jest to domyślne ustawienie. Potrzebne interfejsy należy zatem włączyć i nadać im odpowiednie adresy IP z maskami podsieci, np.:

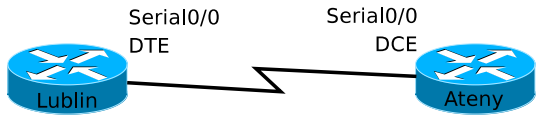
```
R1(config)#interface FastEthernet0/0
R1(config-if)#no shutdown
R1(config-if)#ip address 192.168.0.1 255.255.255.0
```

Polecenie `no shutdown` powoduje usunięcie z konfiguracji routera polecenia

`shutdown`. Użycie słowa `no` jest standardowym sposobem usuwania z konfiguracji niepożądanych wpisów. W analogiczny sposób konfiguruje się pozostałe interfejsy. Interfejsy niewykorzystywane powinny pozostać wyłączone.



Rysunek 2.3. Zastosowanie interfejsów szeregowych routerów.



Rysunek 2.4. Model sieci rozległej w warunkach laboratoryjnych.

Interfejsy szeregowe (Serial) routerów służą do połączenia z siecią rozległą (WAN, *Wide Area Nerwork*) – rys. 2.3. Do tego celu należy użyć odpowiedniego dla danej technologii WAN urządzenia dostępowego DCE (*Data Communications Equipment*). Router pracuje wówczas w roli urządzenia DTE (*Data Terminal Equipment*). Laboratoryjnym modelem sieci z rys. 2.3 jest sytuacja przedstawiona na rys. 2.4. Wówczas jeden z routerów musi pracować w nienaturalnej dla siebie roli urządzenia DCE. O tym, które z dwóch urządzeń będzie pracowało w roli DCE, decyduje typ podłączonego do niego kabla (DCE lub DTE). Można to sprawdzić wizualnie lub poleceniem `show controllers` (informacja w czwartym wierszu) z nazwą odpowiedniego interfejsu, np.:

```
1 R1#show controllers serial 0/0/0
  Interface Serial0/0/0
3 Hardware is PowerQUICC MPC860
  DCE V.35, no clock
5 idb at 0x81081AC4, driver data structure at 0x81084AC0
  [...]
```

W ustawieniach interfejsu szeregowego DCE konieczne jest skonfigurowanie sygnału zegara, np.:

```
R1(config-if)#clock rate 56000
```

Wartość liczbowa określa szybkość pracy interfejsu i jest wyrażona w bitach na sekundę. Wpisując "?" można uzyskać listę wartości akceptowanych przez dane urządzenie.

Po przeprowadzeniu konfiguracji routera, należy powrócić do uprzywilejowanego tryby EXEC i zweryfikować wprowadzone ustawienia. Wszelkie powyższe modyfikacje odnoszą się do pliku konfiguracyjnego o nazwie `running-config`, przechowywanego w pamięci RAM routera. Jego zawartość można wylistować poleceniem `show running-config`. Znajdują się tam polecenia wprowadzone przez administratora, jak również szereg wpisów domyślnych. Aby nie utracić ustawień w momencie restartu urządzenia lub awarii zasilania, należy wydać polecenie:

```
R1#copy running-config startup-config
```

Plik `startup-config`, do którego skopiowane zostaną ustawienia konfiguracyjne, jest przechowywany w nieulotnej pamięci NVRAM. Po uruchomieniu routera, jego zawartość jest automatycznie zapisywana w `running-config`. Oczywiście działa również polecenie `show startup-config`.

Dysponując serwerem TFTP w sieci, można w prosty sposób przesłać do niego kopię pliku konfiguracyjnego oraz systemu operacyjnego IOS, korzystając z poleceń `copy running-config tftp` oraz `copy flash tftp` (obraz systemu operacyjnego zwykle przechowywany jest w pamięci flash routera).

W przypadku budowy nowej sieci lub przed wykonywaniem ćwiczeń praktycznych, zalecane jest przywrócenie fabrycznych ustawień routera. Procedurę tę pokazuje poniższy listing:

```
1 R1#erase startup-config
  Erasing the nvram filesystem will remove all configuration
3 files! Continue? [confirm]
  [OK]
5 Erase of nvram: complete
  %SYS-7-NV_BLOCK_INIT: Initialized the geometry of nvram
7 R1#reload
  Proceed with reload? [confirm]
```

Polecenie `erase startup-config` usuwa plik z konfiguracją startową, natomiast w wyniku restartu (polecenie `reload`) zostanie utracona konfiguracja bieżąca. W razie wyświetlenia pytania o zapisanie bieżących ustawień, należy udzielić odpowiedzi odmownej (`No`).

2.3. Podstawy wykrywania problemów i śledzenia pracy routera

Współczesne sieci komputerowe są niezwykle rozbudowanymi systemami. Nieuchronne są więc różnego rodzaju problemy, związane z ich funkcjonowaniem. Błędy są popełniane są już na etapie wstępnego konfigurowania urządzeń sieciowych przez administratora. IOS Cisco udostępnia szereg mechanizmów ułatwiających proces wykrywania problemów.

W trybie EXEC użytkownika dostępne są polecenia `ping` i `traceroute` [22], będące odpowiednikami poleceń `ping` i `traceroute/tracert` znanych z systemów Linux/Unix i Windows. Ping korzysta z protokołu ICMP (*Internet Control Message Protocol*) i umożliwia sprawdzenie, czy dany host działa i jest osiągalny poprzez sieć. Informuje także o opóźnieniach i utracie pakietów na trasie między dwoma hostami. Polecenie `ping` powoduje wysłanie komunikatu ICMP *echo request*, a następnie przez pewien czas oczekuje na odpowiedź – *echo reply*. Przykład:

```
R1#ping 130.13.0.7

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 130.13.0.7,
timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip
min/avg/max = 9/12/14 ms
```

Znak "!" oznacza poprawne odebranie komunikatu *echo reply*.

Polecenie `traceroute` pozwala odtworzyć przebieg trasy, którą przesyłane są pakiety od źródła do celu. Urządzenie (np. router) wysyła sekwencję pakietów zawierających segmenty UDP (*User Datagram Protocol*) zaadresowane niepoprawnym numerem portu. Najpierw wysyłane są 3 pakiety z wartością TTL (*Time-to-live*) równą 1. W związku z tym, czas życia tych pakietów kończy się w momencie dotarcia do pierwszego routera, który powinien wówczas odesłać komunikat ICMP *Time Exceeded Message* (TEM). Kolejne pakiety wysyłane są z coraz większymi wartościami TTL, dzięki czemu można odebrać komunikaty TEM od coraz odleglejszych routerów na trasie do celu. Ostatni z nich będzie pochodził od hosta docelowego. Przykład:

```
R1#traceroute 130.13.0.7
Type escape sequence to abort.
Tracing the route to 130.13.0.7

  1   212.182.1.2     12 msec    3 msec    5 msec
  2   219.17.100.3    5 msec     7 msec    5 msec
```

```
3    130.13.0.7        12 msec    8 msec     11 msec
```

Olbrzymią grupę stanowią polecenia rozpoczynające się od słowa `show`. Niektóre z nich zosały już przedstawione [22]:

— `show interfaces`
— `show ip interface brief`
— `show controllers`
— `show running-config`
— `show startup-config`

Kolejne, często stosowane polecenia to [22]:

— `show version` wyświetla podstawowe informacje o konfiguracji sprzętowej i software'owej urządzenia, w tym o wartości rejestru konfiguracji, wykorzystywanego podczas procedury odzyskiwania hasła (Dodatek B).
— `show arp` wyświetla zawartość tablicy ARP routera, zawierającej adresy MAC urządzeń dołączonych do poszczególnych interfejsów.
— `show ip route` wyświetla zawartość tablicy routingu (polecenie zostanie bardziej szczegółowo omówione w kolejnych rozdziałach).
— `show history` wyświetla listę uprzednio wprowadzonych poleceń.
— `show flash:` wyświetla zawartość i informacje o ilości wolnego miejsca w pamięci flash routera, w której standardowo zapisany jest obraz systemu operacyjnego (IOS) routera.
— `show cdp neighbors` (z opcjonalnym parametrem `detail`) wyświetla informacje o sąsiednich urządzeniach, uzyskane dzięki działaniu protokołu CDP (*Cisco Discovery Protocol*). Protokół jest obsługiwany wyłącznie przez urządzenia Cisco i domyślnie jest włączony. Jest on użyteczny na etapie budowy i konfigurowania sieci. Umożliwia odtworzenie schematu połączeń między urządzeniami w sposób prostszy niż poprzez bezpośrednią, wizualną analizę okablowania. W momencie, gdy protokół CDP przestaje być potrzebny, powinien zostać wyłączony (poleceniem `no cdp run` w trybie konfiguracji globalnej), ponieważ jego stałe funkcjonowanie obniża poziom bezpieczeństwa sieci.
— `show environment all` pozwala uzyskać informacje o warunkach pracy routera – temperaturze i funkcjonowaniu wentylatorów.

Kolejną rozbudowaną grupę stanowią polecenia rozpoczynające się od słowa `debug`. W odróżnieniu od poleceń `show`, pozwalają one śledzić wybrane typy zdarzeń bezpośrednio w chwili ich wystąpienia. Stosując debugowanie w routerach obsługujących sieci produkcyjne należy zachować szczególną ostrożność, ponieważ stanowi ono znaczące obciążenie urządzenia. Nie należy jednocześnie debugować zbyt wielu typów zdarzeń, natomiast gdy przestanie być potrzebne, należy je wyłączyć (`no debug all` lub `undebug all`). Domyślnie komunikaty debugowania dostępne są tylko poprzez połączenie

konsolowe. Aby debugować poprzez połączenie zdalne (telnet lub ssh) należy dodatkowo użyć polecenia `terminal monitor`.

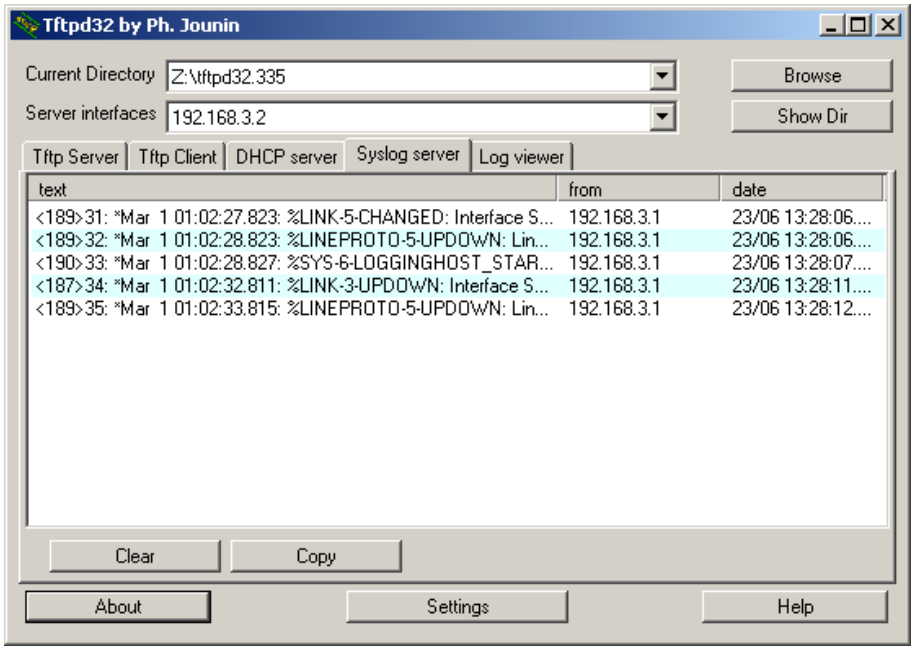
Polecenie `debug ip packet [detail]` wyświetla na bieżąco informacje o pakietach IP wysyłanych, odbieranych i przekazywanych dalej przez router. Generowana jest duża liczba informacji, co może w znacznym stopniu zakłócić pracę routera, zwłaszcza przy dużym ruchu IP. Szczególnie niebezpieczne jest użycie tego polecenia przy zdalnym połączeniu z routerem. Debugowane byłyby wówczas pakiety przekazujące komunikaty debugowania, co spowodowałoby zapętlenie. Poniższy listing przedstawia przykład śledzenia działania polecenia `ping`. Widoczne są informacje o wysyłanych i odbieranych wiadomościach ICMP. `ICMP type=8` oznacza komunikat *echo*, natomiast `ICMP type=0` – *echo-reply*.

Listing 2.1. Sesja debugowania polecenia ping.

```
R1#debug ip packet detail
IP packet debugging is on (detailed)
R1#ping 192.168.3.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.1,
 timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip
 min/avg/max = 8/15/24 ms
R1#
*Mar 1 00:06:37.923: IP: tableid=0, s=172.16.0.1 (local),
  d=192.168.3.1 (Serial0), routed via FIB
*Mar 1 00:06:37.923: IP: s=172.16.0.1 (local),
 d=192.168.3.1 (Serial0), len 100, sending
*Mar 1 00:06:37.923:     ICMP type=8, code=0
*Mar 1 00:06:37.939: IP: tableid=0, s=192.168.3.1 (Serial0),
 d=172.16.0.1 (Serial0), routed via RIB
*Mar 1 00:06:37.939: IP: s=192.168.3.1 (Serial0),
 d=172.16.0.1 (Serial0), len 100, rcvd 3
*Mar 1 00:06:37.939:     ICMP type=0, code=0
*Mar 1 00:06:37.943: IP: tableid=0, s=172.16.0.1 (local),
 d=192.168.3.1 (Serial0), routed via FIB
*Mar 1 00:06:37.943: IP: s=172.16.0.1 (local),
 d=192.168.3.1 (Serial0/0), len 100, sending
*Mar 1 00:06:37.943:     ICMP type=8, code=0
*Mar 1 00:06:37.951: IP: tableid=0, s=192.168.3.1 (Serial0),
 d=172.16.0.1 (Serial0), routed via RIB
*Mar 1 00:06:37.951: IP: s=192.168.3.1 (Serial0),
 d=172.16.0.1 (Serial0), len 100, rcvd 3
*Mar 1 00:06:37.951:     ICMP type=0, code=0
*Mar 1 00:06:37.951: IP: tableid=0, s=172.16.0.1 (local),
 d=192.168.3.1 (Serial0), routed via FIB
```

```
35 [...]

37 R1#
   R1#no debug all
39 All possible debugging has been turned off
```

Powyżej przedstawiono tylko niewielką część poleceń `show` i `debug`. Polecenia służące do weryfikacji działania poszczególnych protokołów routingu zostaną zaprezentowane w kolejnych rozdziałach. Należy mieć świadomość tego, że nie wszystkie polecenia są obsługiwane przez poszczególne wersje IOS, a ich działanie może się w pewnym stopniu różnić od przedstawionego w przykładach.



Rysunek 2.5. Program Tftpd32 w roli serwera syslog.

Informacje o zdarzeniach systemowych domyślnie dostępne są tylko poprzez konsolę. Rozwiązanie to jest słabo skalowalne i staje się uciążliwe w przypadku zarządzania dużą liczbą urządzeń. Lepszym rozwiązaniem, również ze względów bezpieczeństwa, jest przechowywanie logów systemowych na innej maszynie. Routery Cisco mogą przesyłać komunikaty na serwer syslog. Można w tym celu wykorzystać standardowy, linuksowy serwer syslog, lub w przypadku systemów Windows, np. darmowy program Tftpd32

[1]. Konfiguracja routera polega wówczas na wprowadzeniu w trybie konfiguracji globalnej poleceń:

```
service timestamps log datetime msec
logging 192.168.3.2
```

Pierwszy wiersz uruchamia usługę dodawania do komunikatów informacji o dokładnym czasie wystąpienia zdarzenia. W drugim wierszu należy podać adres IP hosta, na którym działa serwer syslog. Poziom szczegółowości komunikatów można skonfigurować poleceniem `logging trap`:

```
R2(config)#logging trap ?
<0-7>          Logging severity level
alerts         Immediate action needed           (severity=1)
critical       Critical conditions               (severity=2)
debugging      Debugging messages                (severity=7)
emergencies    System is unusable                (severity=0)
errors         Error conditions                  (severity=3)
informational  Informational messages            (severity=6)
notifications  Normal but significant conditions (severity=5)
warnings       Warning conditions                (severity=4)
<cr>
```

Im wyższy poziom, tym bardziej szczegółowe wiadomości są logowane. Domyślną wartością dla sysloga jest 6 (*informational*), natomiast dla konsoli 7 (*debugging*). Aktualny status logowania można sprawdzić poleceniem `show logging`, które wyświetla następujące informacje:

```
Syslog logging: enabled (1 messages dropped, 0 messages
                rate-limited, 0 flushes, 0 overruns,
                xml disabled, filtering disabled)

No Active Message Discriminator.

No Inactive Message Discriminator.

Console logging: level debugging, 34 messages logged,
    xml disabled, filtering disabled
Monitor logging: level debugging, 0 messages logged,
    xml disabled, filtering disabled
Buffer logging:  disabled, xml disabled,
                 filtering disabled
Logging Exception size (4096 bytes)
Count and timestamp logging messages: disabled
Persistent logging: disabled
Trap logging: level informational,
              38 message lines logged
```

[1] `http://tftpd32.jounin.net`

```
    Logging to 192.168.3.2  (udp port 514,
21          audit disabled, authentication disabled,
            encryption disabled, link up),
23          8 message lines logged,
            0 message lines rate-limited,
25          0 message lines dropped-by-MD,
            xml disabled, sequence number disabled
27          filtering disabled
```

Dość istotną kwestią jest synchronizacja zegarów czasu rzeczywistego poszczególnych urządzeń w sieci. Przykładowo, w razie ataku na sieć, gdy dysponujemy serwerem przechowującym logi z wielu urządzeń i ich zegary są zsynchronizowane, łatwiejsze będzie odtworzenie sekwencji zdarzeń, które doprowadziły do złamania zabezpieczeń. Do synchronizacji czasu w sieci wykorzystuje się protokół NTP [9].

Zalecane jest posiadanie w sieci własnego serwera NTP. W tej roli może być wykorzystany również router Cisco. W przypadku ręcznego ustawiania czasu, stosuje się polecenie `clock set` (w uprzywilejowanym trybie EXEC) z odpowiednimi parametrami. Bieżący czas podaje polecenie `show clock`. Router staje się źródłem czasu dla innych urządzeń po wprowadzeniu polecenia konfiguracyjnego:

```
ntp master [stratum]
```

Parametr *stratum* z zakresu od 1 do 15 jest miarą odległości od źródła czasu (domyślnie 8). Na innych routerach, które zsynchronizują czas z serwerem, wartość tego parametru będzie o 1 większa. Konfigurując klienta NTP, należy podać adres serwera, np.:

```
ntp server 172.16.0.2
```

Po włączeniu debugowania zdarzeń związanych z protokołem NTP, poleceniem `debug ntp events`, można prześledzić proces synchronizacji czasu z serwerem, np.:

```
1 *Mar 1 02:32:05.299: NTP: Initialized interface Serial0/0
  *Mar 1 02:32:05.299: NTP: Initialized interface Serial0/1
3 *Mar 1 02:32:05.299: NTP: Initialized interface Serial0/2
  *Mar 1 02:32:05.315: NTP: system restart
5 *Jun 23 14:59:24.807: NTP: peer stratum change
  *Jun 23 14:59:24.807: NTP: clock reset
7 Jun 23 14:59:25.835: NTP: 172.16.0.2 synced to new peer
  Jun 23 14:59:25.835: NTP: sync change
9 Jun 23 14:59:25.835: NTP: peer stratum change
  Jun 23 14:59:25.835: NTP: 172.16.0.2 reachable
```

Poniższe listingi pokazują wynik działania polecenia `show ntp status` na serwerze (z czasem ustawionym ręcznie i stratum 4) i kliencie (zsynchronizowanym z serwerem 172.16.0.2), odpowiednio:

Listing 2.2. Status serwera NTP.

```
Clock is synchronized, stratum 4, reference is 127.127.7.1
nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz,
 precision is 2**18
reference time is CFCCA2AC.C2897C28 (15:11:40.759
 UTC Wed Jun 23 2010)
clock offset is 0.0000 msec, root delay is 0.00 msec
root dispersion is 0.02 msec, peer dispersion is 0.02 msec
```

Listing 2.3. Status klienta NTP.

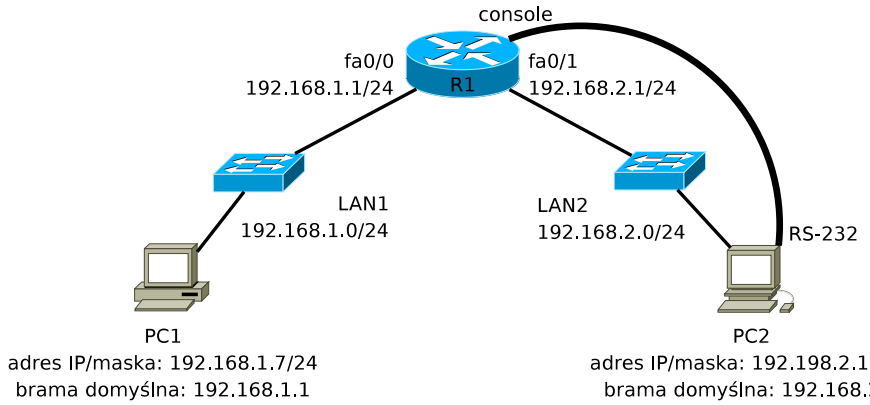
```
Clock is synchronized, stratum 5, reference is 172.16.0.2
nominal freq is 250.0000 Hz, actual freq is 250.0003 Hz,
 precision is 2**18
reference time is CFCCA2DB.CB10041A (15:12:27.793
 UTC Wed Jun 23 2010)
clock offset is 15.1543 msec, root delay is 0.44 msec
root dispersion is 38.42 msec, peer dispersion is 23.24 msec
```

Powiązanie klienta z serwerem (tu 172.16.0.2) można prześledzić wydając na kliencie polecenie `show ntp associations detail`:

```
172.16.0.2 configured, our_master, sane, valid, stratum 4
ref ID 127.127.7.1, time CFCCA3EC.C28C350A (15:17:00.759
 UTC Wed Jun 23 2010)
our mode client, peer mode server, our poll intvl 64,
 peer poll intvl 64
root delay 0.00 msec, root disp 0.03, reach 377,
 sync dist 24.521
delay 7.92 msec, offset -9.9441 msec, dispersion 20.54
precision 2**18, version 3
org time CFCCA41B.C5B1B620 (15:17:47.772 UTC Wed Jun 23 2010)
rcv time CFCCA41B.C94126FF (15:17:47.786 UTC Wed Jun 23 2010)
xmt time CFCCA41B.C739086F (15:17:47.778 UTC Wed Jun 23 2010)
filtdelay = 7.92  20.14   7.92 11.93 28.23   7.92 24.22   3.92
filtoffset=-9.94 -11.29 -43.89 -7.70  2.50 -9.04  5.13  1.78
filterror = 0.02   0.99   1.97  2.94  3.92  4.90  5.87  6.85
```

2.4. Zadanie

Uwaga: Podstawowa konfiguracja routera, zgodna z poniższym schematem, jest wymagana w zadaniach laboratoryjnych z kolejnych rozdziałów. Oznaczenia interfejsów sieciowych routerów w laboratorium mogą się różnić od używanych w zadaniach. Należy to sprawdzić (`show ip interface brief`).



Rysunek 2.6. Schemat topologii logicznej sieci.

1. Połącz urządzenia zgodnie ze schematem (rys. 2.6). Skonfiguruj ustawienia sieciowe (adres IP, maska podsieci, adres bramy domyślnej) komputerów PC.
2. Skonfiguruj i uruchom połączenie konsolowe z routerem.
3. Jeżeli router jest już skonfigurowany, przywróć ustawienia fabryczne. W razie zabezpieczenia routera nieznanym hasłem, przeprowadź procedurę odzyskiwania, opisaną w Dodatku B.
4. Nadaj routerowi nazwę, np. R1.
5. Zabezpiecz hasłami wejście do trybu uprzywilejowanego, połączenie konsolowe i telnetowe. Jako hasło użyj słowa `cisco`.
6. Uruchom i skonfiguruj interfejsy sieciowe routera. Użyj adresowania zgodnego z rys. 2.6.
7. Przejrzyj plik konfiguracyjny. Skoryguj ewentualne błędy. Które hasła przechowywane są w postaci jawnego tekstu? Spowoduj ich zaszyfrowanie.
8. Zapisz konfigurację do pamięci NVRAM.
9. Przejrzyj informacje o interfejsach (polecenia `show interfaces` oraz `show ip interface brief`).
10. Przetestuj komunikację między sieciami LAN1 i LAN2 (ping).
11. Przetestuj połączenie telnetowe z routerem. Czy router faktycznie wymaga podania uprzednio skonfigurowanego hasła?

12. Na jednym z komputerów PC uruchom serwer TFTP. Prześlij do niego kopię pliku konfiguracyjnego.
13. Przywróć ustawienia fabryczne routera lub wykonaj zadanie z kolejnego rozdziału.

2.5. Rozwiązanie

Uwaga: Poniższy plik konfiguracyjny stanowi przykładowe rozwiązanie. Mogą wystąpić różnice wynikające z użytego sprzętu (w szczególności inne oznaczenia interfejsów) oraz wersji IOS.

Listing 2.4. Plik konfiguracyjny routera R1.

```
1 !
  version 12.4
3 no service timestamps log datetime msec
  no service timestamps debug datetime msec
5 service password-encryption
  !
7 hostname R1
  !
9 !
  !
11 enable secret 5 $1$mERr$hx5rVt7rPNoS4wqbXKX7m0
  !
13 !
  !
15 !
  !
17 !
  !
19 !
  !
21 !
  !
23 !
  !
25 !
  !
27 !
  interface FastEthernet0/0
29  ip address 192.168.1.1 255.255.255.0
   duplex auto
31  speed auto
  !
33 interface FastEthernet0/1
   ip address 192.168.2.1 255.255.255.0
35  duplex auto
   speed auto
```

```
!
interface Serial0/0/0
 no ip address
 clock rate 56000
 shutdown
!
interface Vlan1
 no ip address
 shutdown
!
ip classless
!
!
!
!
!
!
!
line con 0
 password 7 0822455D0A16
 login
line vty 0 4
 password 7 0822455D0A16
 login
line vty 5 15
 password 7 0822455D0A16
 login
!
!
!
end
```

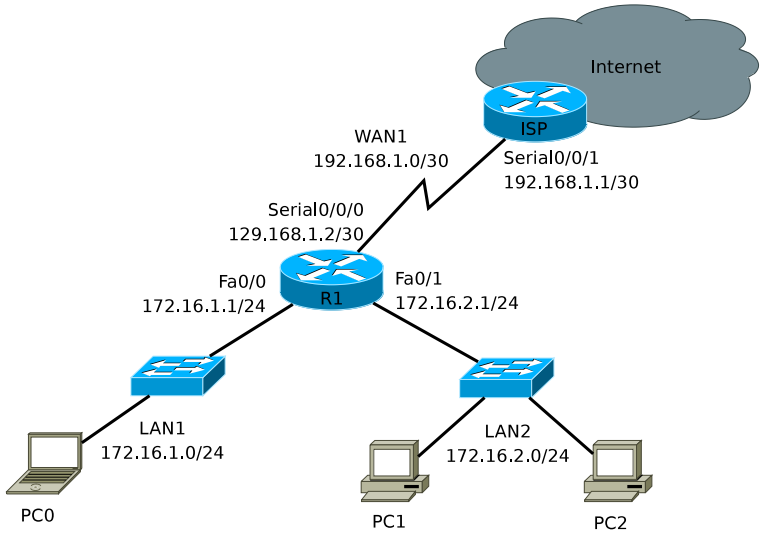
Rozdział 3

Routing statyczny

3.1. Zasady działania i konfiguracji routingu statycznego

Zasady przekazywania pakietów przez router można sformułować następująco [19]:

1. Każdy router podejmuje samodzielne decyzje, kierując się informacją zapisaną w jego tablicy routingu.
2. Fakt posiadania przez router określonej informacji w tablicy routingu nie oznacza, że inne routery również posiadają tę informację.
3. Aby była możliwa komunikacja między dwiema sieciami, konieczne jest zapewnienie poprawnych tras w obu kierunkach. Nie muszą one przebiegać tą samą drogą.



Rysunek 3.1. Bezpośrednio dołączone sieci routera R1: LAN1, LAN2, WAN1.

Zadaniem administratora jest zapewnienie, by informacje zapisane w tablicach routingu poszczególnych routerów były spójne i umożliwiały poprawną komunikację między sieciami. Dysponując adresami IP swoich interfejsów, wraz z maskami podsieci, router jest w stanie rozpoznać sieci, do których jest bezpośrednio dołączony. Jeżeli interfejsy są poprawnie skonfigurowane i działają, odpowiednie wpisy automatycznie pojawiają się w tablicy routingu. Router może przekazywać pakiety do bezpośrednio dołączonych sieci, bez dodatkowej konfiguracji routingu (o ile tylko nie wydano polecenia `no ip routing`). W przypadku sieci przedstawionej na rys. 3.1, w wyniku skonfigurowania adresów IP interfejsów ethernetowych routera R1 i ich włączenia (interfejs szeregowy, stanowiący połączenie z operatorem dostępu do internetu, pozostaje na razie wyłączony), uzyskujemy następującą zawar-

tość tablicy routingu routera R1 (można ją zobaczyć używając polecenia `show ip route`):

Listing 3.1. Zawartość tablicy routingu routera R1.

```
1 Codes: C - connected, S - static, I - IGRP, R - RIP,
  [...]
3
  Gateway of last resort is not set
5
      172.16.0.0/24 is subnetted, 2 subnets
7 C      172.16.1.0 is directly connected, FastEthernet0/0
  C      172.16.2.0 is directly connected, FastEthernet0/1
```

Litery C na początku wierszy 7 i 8 oznaczają sieci bezpośrednio podłączone. Gdy do routera zostanie przekazany pakiet z adresem docelowym należącym do sieci 172.16.1.0/24, wówczas zostanie on wysłany poprzez interfejs FastEthernet0/0. Analogicznie router będzie postępował z pakietami przeznaczonymi dla sieci 172.16.2.0/24. Pod warunkiem poprawnego skonfigurowania hostów (w szczególności adresu bramy domyślnej), powinna być możliwa komunikacja między wszystkimi hostami z rysunku 3.1. Pakiety przeznaczone dla jakichkolwiek innych sieci będą odrzucane, ponieważ router nie posiada w tablicy routingu wpisów umożliwiających podjęcie innej decyzji.

W celu uzupełnienia tablicy routingu o kolejne wpisy, niezbędna jest ingerencja administratora. Do dyspozycji jest routing statyczny, polegający na podaniu routerowi wprost informacji o sposobie postępowania z pakietami przeznaczonymi dla określonej sieci docelowej, oraz routing dynamiczny, w którym routery automatycznie wymieniają między sobą informacje o dostępnych trasach, zgodnie z określonym protokołem routingu. Routing statyczny jest stosunkowo prosty do implementacji i nie zwiększa zapotrzebowania na pasmo w sieci oraz zasoby sprzętowe. Jest to również rozwiązanie bezpieczne (luki w zabezpieczeniach i błędy protokołów routingu są potencjalnym zagrożeniem bezpieczeństwa sieci). Jednak, w przypadku większych sieci, rozwiązanie to jest słabo skalowalne. Ponadto, każda zmiana (np. pojawienie się nowej sieci lub awaria któregoś segmentu) wymaga ingerencji administratora, w celu przywrócenia poprawnej komunikacji. Dlatego routing statyczny jest stosowany w niewielkich sieciach, w których nie występują trasy alternatywne, a także w połączeniu z routingiem dynamicznym.

Trasy statyczne definiuje się poleceniem `ip route`. Jego podstawowa składnia jest następująca [23]:

`ip route` *prefix mask* {*ip-address* | *interf-type interf-number*} [*distance*]

prefix mask to adres i maska sieci docelowej. Następnie podaje się adres IP sąsiedniego routera (interfejsu należącego do tej samej sieci, co interfejs naszego routera), do którego należy przekazać pakiet, lub pełną nazwę interfejsu wyjściowego routera (np. Serial0/0/0), na którym konfigurujemy trasę. Ostatni, opcjonalny parametr to tzw. odległość administracyjna (*administrative distance*), omówiona w kolejnym rozdziale. Konfigurując trasy statyczne, należy pamiętać o zapewnieniu transmisji pakietów w dwie strony.

Specyficznym typem trasy statycznej jest trasa domyślna, konfigurowana następująco:

```
ip route 0.0.0.0 0.0.0.0 {ip-address | interf-type interf-number}
```

Trasa domyślna jest wykorzystywana, w razie gdy żaden inny wpis w tablicy routingu nie pozwala na podjęcie decyzji o dalszych losach pakietu. Pozwala ona na znaczące ograniczenie rozmiaru tablic routingu. W przypadku sieci pokazanej na rys. 3.1, po uruchomieniu interfejsu Serial0/0/0, trasę domyślną moglibyśmy skonfigurować poleceniem:

```
ip route 0.0.0.0 0.0.0.0 Serial0/0/0
```

lub

```
ip route 0.0.0.0 0.0.0.0 192.168.1.1
```

Pozornie oba sposoby są równoważne. W pierwszym przypadku tablica routingu routera R1 będzie miała postać:

Listing 3.2. Zawartość tablicy routingu routera R1 z trasą domyślną zawierającą nazwę interfejsu wyjściowego.

```
1  Codes: C - connected, S - static, I - IGRP, R - RIP,
   [...]
3         * - candidate default,
   [...]
5
   Gateway of last resort is 0.0.0.0 to network 0.0.0.0
7
        172.16.0.0/24 is subnetted, 2 subnets
9  C       172.16.1.0 is directly connected, FastEthernet0/0
   C       172.16.2.0 is directly connected, FastEthernet0/1
11       192.168.1.0/30 is subnetted, 1 subnets
   C       192.168.1.0 is directly connected, Serial0/0/0
13 S*   0.0.0.0/0 is directly connected, Serial0/0/0
```

W razie użycia drugiej metody, w tablicy routingu zobaczymy:

Listing 3.3. Zawartość tablicy routingu routera R1 z trasą domyślną zawierającą adres sąsiedniego routera.

```
1 Codes: C - connected, S - static, I - IGRP, R - RIP,
  [...]
3         * - candidate default,
  [...]
5
  Gateway of last resort is 192.168.1.1 to network 0.0.0.0
7
       172.16.0.0/24 is subnetted, 2 subnets
9 C       172.16.1.0 is directly connected, FastEthernet0/0
  C       172.16.2.0 is directly connected, FastEthernet0/1
11     192.168.1.0/30 is subnetted, 1 subnets
  C       192.168.1.0 is directly connected, Serial0/0/0
13 S*   0.0.0.0/0 [1/0] via 192.168.1.1
```

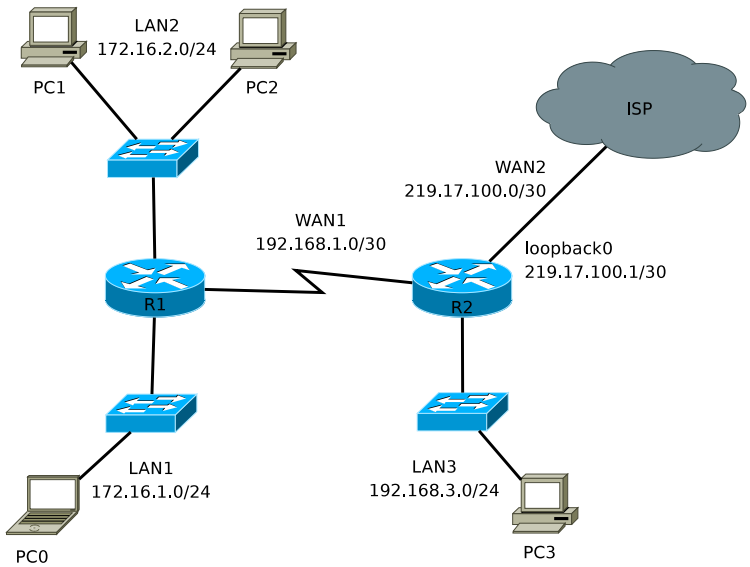
W przypadku listingu 3.2, wiersz 13 zawiera kompletną informację umożliwiającą przekazanie pakietu dalej. W razie skorzystania przez router z ostatniego wpisu z listingu 3.3, konieczny jest jeszcze jeden krok, polegający na wybraniu interfejsu wyjściowego dla pakietu, który ma być przekazany do sąsiedniego routera 192.168.1.1. Zostanie do tego celu wykorzystana informacja z wiersza 12. Zatem, przynajmniej teoretycznie, konfiguracja trasy statycznej z wykorzystaniem nazwy interfejsu wyjściowego umożliwia routerowi szybsze podjęcie decyzji. W rzeczywistości, zwykle różnice są pomijalne. Z drugiej strony, w przypadku połączeń innych niż punkt-punkt, nazwa interfejsu wyjściowego nie określa w sposób jednoznaczny kolejnego routera, któremu należy przekazać pakiet. Wówczas (np. w przypadku połączeń ethernetowych między routerami), definiując trasę statyczną, należy podawać adres IP sąsiedniego routera.

W tym momencie router R1 będzie przekazywał pakiety adresowane do innych miejsc niż sieci LAN1, LAN2 i WAN1 do routera ISP. Zakładając, że router ISP zna trasę do sieci LAN1 i LAN2 (np. została ona skonfigurowana na nim w sposób statyczny), konfigurację routingu możemy uznać za kompletną.

Sposób wykorzystania trasy domyślnej zależy od klasowego lub bezklasowego zachowania routera. We współczesnych wersjach IOS (a dokładniej, począwszy od 11.3), domyślne jest zachowanie bezklasowe (w pliku konfiguracyjnym routera automatycznie pojawia się polecenie `ip classless`). Załóżmy, że do routera z tablicą routingu przedstawioną na listingu 3.2 zostanie przekazany pakiet adresowany do 172.16.16.16. Wiersz nr 8 z listingu możemy odczytać następująco: *sieć 172.16.0.0/16 (z domyślną maską) została podzielona na podsieci przy użyciu maski /24. Znane są dwie podsieci.* Wpis ten daje szansę na znalezienie informacji pozwalającej przekazać

pakiet dalej. W kolejnym kroku router analizuje więc wiersze nr 9 i 10. Okazuje się jednak, że adres docelowy nie należy do podsieci 172.16.1.0/24 ani 172.16.2.0/24. Pakiet zostanie zatem przesłany trasą domyślną. Router dopuszcza więc możliwość nieciągłości sieci 172.16.0.0/16. W naszym przypadku, inne jej podsieci (np. 172.16.16.0/24) mogą znajdować się wewnątrz chmury na rys. 3.1, od której sieci LAN1 i LAN2 są oddzielone zupełnie inną podsiecią (192.168.1.0/30). W przypadku zachowania klasowego, pakiet ten zostałby odrzucony, pomimo zdefiniowania trasy domyślnej. W przypadku pakietu adresowanego np. do hosta 212.182.1.111, w obu przypadkach (tzn. klasowym i bezklasowym) zachowanie routera byłoby identyczne i polegałoby na skorzystaniu z trasy domyślnej, ponieważ w tablicy routingu nie ma jakiegokolwiek wpisu odnoszącego się do sieci, do której należy ten adres.

3.2. Zadanie



Rysunek 3.2. Schemat topologii logicznej sieci.

1. Zbuduj sieć zgodnie ze schematem (rys. 3.2).
2. Na routerze R1 wydaj polecenie `debug ip routing` (w trybie uprzywilejowanym), aby śledzić proces budowy tablicy routingu.
3. Przeprowadź podstawową konfigurację routerów (nazwy, hasła, interfejsy sieciowe) oraz komputerów PC. Interfejsom sieciowym routerów przy-

dziel najniższe możliwe adresy, natomiast komputerom PC – dowolne. Interfejs Loopback0 routera R2 symuluje połączenie z dostawcą usług internetowych (chmurą ISP). Loopback jest interfejsem wirtualnym, który konfigurujemy podobnie jak interfejsy fizyczne (`interface loopback`). Nie jest konieczne polecenie `no shutdown` ponieważ, w przeciwieństwie do interfejsów fizycznych, Loopback jest domyślnie włączony. Router może posiadać wiele interfejsów Loopback.

4. Czy możliwa jest komunikacja (`ping`) między urządzeniami podłączonymi do tego samego interfejsu routera (np. PC1 i PC2)? Jeżeli nie, zidentyfikuj i usuń błędy.
5. Przejrzyj zawartość tablic routingu (`show ip route`). Czy możliwa jest komunikacja między PC0 i PC3? Dlaczego?
6. Na routerze R1 zdefiniuj trasę domyślną w kierunku routera R2 (używając nazwy interfejsu wyjściowego routera R1 lub adresu IP interfejsu sąsiedniego routera). Czy teraz możliwy jest ping 219.17.100.1 z komputera PC0?
7. Na routerze R2 skonfiguruj domyślną trasę poprzez interfejs Loopback0 oraz trasy statyczne do sieci LAN1 i LAN2. Zweryfikuj tablice routingu routerów. Upewnij się, że możliwa jest komunikacja między wszystkimi urządzeniami w sieci.
8. Prześledź trasę pakietów wysyłanych z hosta PC0 na adres 212.182.1.111 oraz 172.16.100.100 (użyj polecenia `traceroute/tracert` na PC0 lub `debug ip packet` na R2). Następnie wymuś klasowe zachowanie routera R1 (`no ip classless`, może być konieczne również wyłączenie mechanizmu CEF[1]: `no ip cef`). Zaobserwuj zmianę w postępowaniu routera R1 z pakietami przeznaczonymi dla 172.16.100.100. Przywróć domyślne zachowanie bezklasowe (`ip classless`).
9. Wyłącz wszystkie procesy debugowania (polecenie `no debug all` lub `undebug all`). Wykonaj kopie plików konfiguracyjnych routerów.

3.3. Rozwiązanie

Listing 3.4. Istotne fragmenty konfiguracji routera R1.

```
1 hostname R1
  enable secret 5 $1$mERr$hx5rVt7rPNoS4wqbXKX7m0
3 interface FastEthernet0/0
   ip address 172.16.1.1 255.255.255.0
5  duplex auto
   speed auto
```

[1] *Cisco's Express Forwarding*, zaawansowana technologia szybkiego przełączania w trzeciej warstwie.

```
 7 interface FastEthernet0/1
    ip address 172.16.2.1 255.255.255.0
 9  duplex auto
    speed auto
11 interface Serial1/0
    ip address 192.168.1.1 255.255.255.252
13  clock rate 64000
   ip classless
15 ip route 0.0.0.0 0.0.0.0 Serial1/0
   line con 0
17  password cisco
    login
19 line vty 0 4
    password cisco
21  login
   end
```

Listing 3.5. Istotne fragmenty konfiguracji routera R2.

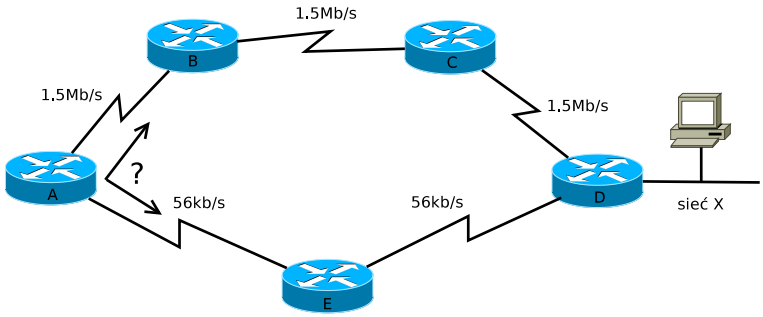
```
   hostname R2
 2 enable secret 5 $1$mERr$hx5rVt7rPNoS4wqbXKX7m0
   interface Loopback0
 4  ip address 219.17.100.1 255.255.255.252
   interface FastEthernet0/0
 6  ip address 192.168.3.1 255.255.255.0
    duplex auto
 8  speed auto
   interface Serial1/0
10  ip address 192.168.1.2 255.255.255.252
   ip classless
12 ip route 172.16.2.0 255.255.255.0 Serial1/0
   ip route 172.16.1.0 255.255.255.0 Serial1/0
14 ip route 0.0.0.0 0.0.0.0 Loopback0
   line con 0
16  password cisco
    login
18 line vty 0 4
    password cisco
20  login
   end
```

Rozdział 4

Routing dynamiczny

4.1. Podstawy teoretyczne

Jak już wspomniano w poprzednim rozdziale, routing statyczny jest dobrym rozwiązaniem jedynie w przypadku niewielkich sieci. W przypadku bardziej złożonych sieci, potrzebny jest mechanizm automatycznej wymiany informacji między routerami, wyznaczania tras i adaptacji do zmieniających się warunków. Jest to realizowane przy pomocy protokołów routingu. W razie istnienia więcej niż jednej trasy do danej sieci docelowej, dąży się do tego, by została wybrana najlepsza.



Rysunek 4.1. Niedoskonałość metryki liczby skoków.

Miarą jakości trasy jest metryka. Każdy protokół routingu charakteryzuje się własnym sposobem określania metryki. W najprostszym przypadku może to być liczba skoków (*hop count*), czyli liczba routerów, przez które pakiet musi przejść na drodze do sieci docelowej. Niedoskonałość tej metryki ilustruje rys. 4.1. Z perspektywy routera A, optymalna trasa do sieci X prowadzi poprzez router E (2 skoki). Alternatywna trasa, poprzez router B, charakteryzuje się 3 skokami. Oczywiście w rzeczywistości górna trasa byłaby szybsza. Bardziej rozbudowane metryki mogą brać pod uwagę szerokość pasma, opóźnienie, obciążenie łącza, niezawodność i inne czynniki.

Na routerze może jednocześnie działać wiele protokołów routingu, jak również routing statyczny. Może zdarzyć się, że router będzie dysponował informacjami o dwóch różnych trasach do danej sieci docelowej, pochodzącymi od dwóch różnych protokołów routingu. W większości przypadków nie jest możliwe sensowne przeliczenie metryki protokołu routingu na metrykę innego protokołu, więc nie da się na tej podstawie dokonać wyboru lepszej trasy. Dlatego stosowany jest kolejny parametr – tzw. odległość administracyjna (*administrative distance*), informujący o wiarygodności danego protokołu routingu. Przykładowe, domyślne wartości, stosowane w przypadku urządzeń Cisco, podane są w tabeli 4.1 [39] (mogą one zostać zmodyfikowane przez administratora). Zatem, w przypadku dostępności informacji o kilku trasach do danej sieci, w pierwszej kolejności wybiera się najbardziej

wiarygodne źródło (tzn. charakteryzujące się najmniejszą wartością odległości administracyjnej), a następnie trasę o najniższej wartości metryki. Najlepsza trasa (ewentualnie więcej niż jedna, w przypadku dostępności tras równorzędnych) jest umieszczana w tablicy routingu.

Tabela 4.1. Przykładowe, domyślne wartości odległości administracyjnej różnych protokołów routingu, dla routerów Cisco[39].

Źródło trasy	Odległość administracyjna
Sieci bezpośrednio podłączone	0
Trasy statyczne	1
EIGRP – trasa sumaryczna	5
BGP – trasa zewnętrzna	20
EIGRP – trasa wewnętrzna	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
EIGRP – trasa zewnętrzna	170
BGP – trasa wewnętrzna	200
Nieznane	255

Poniższy listing przedstawia fragment tablicy routingu, w której znajdują się wpisy pochodzące z czterech różnych źródeł: sieci bezpośrednio podłączone (C), trasy statyczne (S), protokół routingu EIGRP (D) oraz protokół routingu OSPF (O). W wierszach z wpisami pochodzącymi od protokołów routingu (w tym przykładzie 16 i 19) w nawiasie kwadratowym umieszczona jest para liczb (tu: [90/2172416] oraz [110/65]). Pierwsza z nich to odległość administracyjna, natomiast druga to wartość metryki dla danej trasy.

Listing 4.1. Przykładowa zawartość tablicy routingu.

```
1 Codes: C - connected, S - static, I - IGRP, R - RIP,
         M - mobile, B - BGP, D - EIGRP, EX - EIGRP external,
3        O - OSPF, IA - OSPF inter area, N1 - OSPF NSSA
         external type 1, N2 - OSPF NSSA external type 2,
5        E1 - OSPF external type 1, E2 - OSPF external type 2,
         E - EGP, i - IS-IS, L1 - IS-IS level-1,
7        L2 - IS-IS level-2, ia - IS-IS inter area
         * - candidate default, U - per-user static route,
9        o - ODR, P - periodic downloaded static route

11 Gateway of last resort is not set

13      10.0.0.0/24 is subnetted, 1 subnets
```

```
   S        10.0.0.0 is directly connected, Serial0/3/0
15       172.16.0.0/24 is subnetted, 3 subnets
   D        172.16.0.0 [90/2172416] via 192.168.1.1, 00:03:34,
17                                                  Serial0/3/0
   C        172.16.1.0 is directly connected, FastEthernet0/0
19 O        172.16.3.0 [110/65] via 192.168.1.6, 00:03:25,
                                                    Serial0/3/1
21       192.168.1.0/30 is subnetted, 2 subnets
   C        192.168.1.0 is directly connected, Serial0/3/0
23 C        192.168.1.4 is directly connected, Serial0/3/1
```

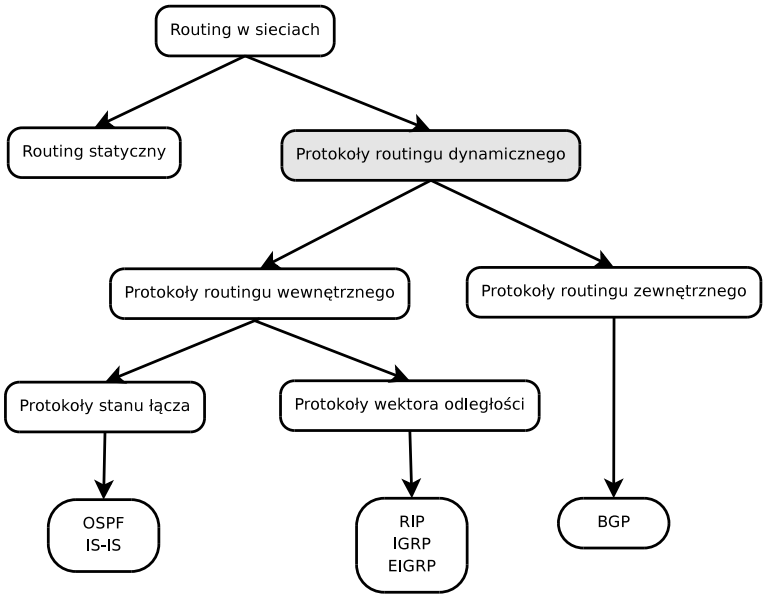
Z tabeli 4.1 wynika, że router w pierwszej kolejności korzysta z tras statycznych (i oczywiście sieci bezpośrednio podłączonych). Jednak, definiując trasę statyczną w sposób podany w poprzednim rozdziale, można samodzielnie podać inną niż domyślna wartość odległości administracyjnej. Załóżmy, że mamy w sieci skonfigurowany protokół OSPF (odległość administracyjna 110) oraz trasy statyczne z odległością administracyjną 130. Wówczas router skorzysta z tras statycznych jedynie w przypadku sieci, o których informacji nie dostarczy OSPF, lub gdy z jakichś powodów OSPF zawiedzie. Zatem trasy statyczne mogą być również trasami zapasowymi na wypadek awarii protokołu routingu.

4.2. Ogólna klasyfikacja protokołów routingu

Niektóre spośród algorytmów wykorzystywanych we współczesnych protokołach routingu zostały wynalezione już w początkowym okresie powstawania stosu TCP/IP i nadal są rozwijane. Rys. 4.2 przedstawia ogólną klasyfikację protokołów routingu.

Grupę sieci, którą zarządza ta sama jednostka, nazywamy systemem autonomicznym. Przykładem są sieci należące do jednej firmy. Wówczas wybór strategii routingu wewnątrz systemu autonomicznego zależy od administratora. System autonomiczny zwykle ma połączenie z innymi systemami autonomicznymi, z którymi również należy wymieniać informacje o dostępnych trasach. Protokoły routingu, które stosuje się wewnątrz systemu autonomicznego, nazywamy protokołami routingu wewnętrznego (lub protokołami bramy wewnętrznej, ang. *Interior Gateway Protocol*), w odróżnieniu od protokołów routingu zewnętrznego (lub protokołów bramy zewnętrznej, ang. *Exterior Gateway Protocol*), stosowanych między systemami autonomicznymi. W tym momencie powszechnie stosowanym protokołem routingu zewnętrznego jest BGP (*Border Gateway Protocol*).

W przypadku protokołów routingu wewnętrznego, możemy mieć do czynienia z sytuacją, w której każdy router dysponuje kompletną informacją o sieciach, połączeniach między routerami i ich parametrach. Na jej podsta-



Rysunek 4.2. Ogólna klasyfikacja protokołów routingu.

wie można stworzyć dokładną mapę połączeń w sieci. Protokoły tej grupy nazywamy protokołami stanu łącza (*link state*) i należą do niej między innymi OSPF (*Open Shortest Path First*) i IS-IS (*Intermediate System to Intermediate System*). Każdy router buduje identyczną bazę danych, opisującą topologię systemu autonomicznego. Składają się na nią informacje o stanie poszczególnych routerów, tzn. ich interfejsach i sąsiadach (sąsiednich routerach), z którymi mają połączenie. Informacje o własnym stanie router rozsyła do wszystkich pozostałych routerów w systemie autonomicznym (tzw. rozsyłanie zalewowe, ang. *flooding*). Na każdym spośród routerów pracuje również ten sam algorytm, zgodnie z którym budowane jest drzewo (graf), którego korzeniem jest dany router. Następnie, korzystając z algorytmu znajdowania najkrótszych ścieżek w grafie, autorstwa E. Dijkstry [3] (ang. *Shortest Path First*), router znajduje najkrótsze trasy do wszystkich miejsc docelowych wewnątrz systemu autonomicznego. Najlepsze trasy umieszczane są w tablicy routingu. W razie istnienia kilku równoważnych tras do danego miejsca docelowego, ruch może być dzielony miedzy nimi.

W prostszym przypadku, do wybrania optymalnych tras wystarcza routerowi znajomość jedynie odległości do sieci docelowych (metryki) oraz kierunku, w jakim należy wysyłać pakiety (interfejsu wyjściowego lub adresu sąsiada). Nie są natomiast znane informacje o układzie i parametrach poszczególnych połączeń między routerami. Protokoły takie nazywamy protokołami wektora odległości (*distance vector*). Najbardziej znane to: RIP

(*Routing Information Protocol*), IGRP (*Interior Gateway Routing Protocol*), EIGRP (*Enhanced Interior Gateway Routing Protocol*).

Starsze protokoły routingu (np. RIPv1, IGRP) nie rozsyłają informacji o masce podsieci wraz z adresem miejsca docelowego, zakładając, że jest ona zgodna z maską domyślną dla danej klasy adresu, którą z kolei można stwierdzić na podstawie wartości pierwszego oktetu. Protokoły takie nazywamy protokołami klasowymi. Protokoły bezklasowe (RIPv2, EIGRP, OSPF, IS-IS, BGP) rozsyłają informacje o masce. Protokoły klasowe zawodzą w przypadku, gdy jest stosowany podział na podsieci ze zmienną maską (VLSM), lub gdy występują nieciągłe podsieci (co jest powszechnym zjawiskiem we współczesnym Internecie). Osobną grupę stanowią tu odmiany protokołów przeznaczone dla IPv6 (RIPng, OSPFv3, BGPv4 dla IPv6 itd.).

Istotna jest także kwestia skalowalności protokołów routingu. Ze względu na ograniczenia (omówione w kolejnych rozdziałach), takie protokoły jak RIP lub IGRP nie nadają się do stosowania w rozbudowanych sieciach. Z kolei bardziej zaawansowane protokoły (np. OSPF) mają znacznie większe wymagania odnośnie sprzętu (w szczególności ilości pamięci i szybkości procesora zainstalowanego w routerze). W pewnych sytuacjach nie bez znaczenia może również być stopień wykorzystania pasma sieciowego na potrzeby przesyłania komunikatów protokołu routingu. Ważny jest także czas osiągania zbieżności, tzn. stanu, gdy wszystkie routery posiadają spójny i kompletny obraz sieci w swoich tablicach routingu. Im szybciej routery dostosują się do zmiany, jaka wystąpiła w sieci, tym krócej sieć może zachowywać się w sposób nieprzewidywalny.

4.3. Zasady konfiguracji protokołów routingu

W przypadku podstawowej konfiguracji poszczególnych protokołów routingu na routerach Cisco, ogólne zasady są podobne. W trybie konfiguracji globalnej dostępne jest polecenie `router`, którego parametrem jest nazwa protokołu routingu, który ma zostać włączony. W przypadku niektórych protokołów, konieczne są również dodatkowe parametry. Polecenie `router` uruchamia tryb konfiguracji protokołu routingu (znak zgłoszenia uzyskuje postać `(config-router)#`). Następnie, przy pomocy polecenia `network`, określa się interfejsy, na których ma działać protokół routingu. Polecenie `redistribute` pozwala protokołowi routingu rozsyłać informacje o trasach pozyskane z innych źródeł (tzw. redystrybucja tras). Większość pozostałych ustawień konfiguracyjnych jest już specyficzna dla konkretnego protokołu routingu.

Istotną kwestią jest weryfikacja działania skonfigurowanego protokołu. Podstawowe polecenie diagnostyczne, wyświetlające zawartość tablicy ro-

utingu (`show ip route`) zostało już przedstawione. Jak widać na listingu 4.1, w przypadku poszczególnych tras, na początku wiersza podawane jest źródło informacji (w tym przypadku litery S, D, C, O). Każdorazowo pokazywane są również objaśnienia stosowanych skrótów.

Polecenie `show ip protocols` wyświetla podstawowe informacje o protokołach routingu działających na routerze. W praktyce, należy unikać jednoczesnego uruchamiania więcej niż jednego protokołu routingu na urządzeniu, ponieważ stanowi to znaczące obciążenie. Dwa (a w rzadkich przypadkach więcej) protokoły routingu uruchamia się na routerach odpowiedzialnych za redystrybucję tras między domenami routingu. Pozostałe polecenia diagnostyczne (`show` i `debug`) są omówione w rozdziałach dotyczących poszczególnych protokołów routingu.

Rozdział 5

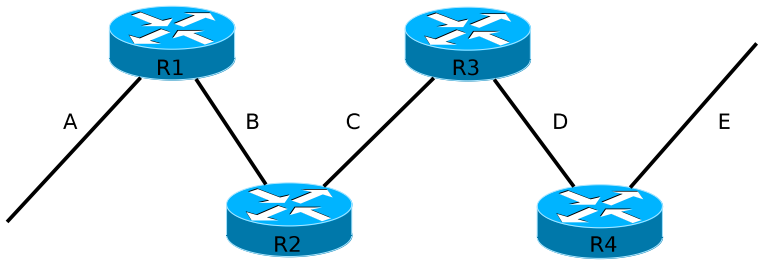
RIP

5.1. Podstawy teoretyczne protokołu

Routing Information Protocol (RIP)[6] działa w oparciu o algorytm autorstwa Bellmana-Forda, który był wykorzystywany w sieciach komputerowych już od czasów wczesnego ARPANETu. Jest protokołem wektora odległości, przeznaczonym do pracy wewnątrz systemu autonomicznego (protokół routingu wewnętrznego), w sieciach IP. Metryką jest liczba skoków (*hop count*), czyli liczba routerów przez które pakiet musi przejść, by dotrzeć do sieci docelowej.

RIP ma szereg ograniczeń, które w przypadku współczesnych sieci zwykle przesądzają o wyborze innego protokołu routingu:

1. Maksymalna długość trasy wynosi 15 skoków. Sieci położone w większej odległości są niedostępne.
2. Pomimo stosowania szeregu środków zapobiegawczych, występuje istotne ryzyko powstawania pętli routingu.
3. Bardzo prosta metryka często nie odzwierciedla faktycznej jakości trasy.
4. Protokół działa w sposób klasowy. W informacjach o trasach rozsyłanych między routerami nie jest podawana maska podsieci. W ramach jednej sieci, gdy informacja dotyczy podsieci, router przyjmuje maskę interfejsu wejściowego. Gdy komunikat dotyczy sieci, z którą router nie ma bezpośredniego połączenia, przyjmowana jest domyślna maska dla danej klasy adresu. W związku z tym, RIP nie jest kompatybilny z mechanizmami VLSM i CIDR. W razie podziału sieci, wszystkie jej podsieci muszą posiadać tę sama maskę. RIP zawodzi także w przypadku nieciągłych podsieci.
5. Routery periodycznie (domyślnie co 30 sekund) rozsyłają rozgłoszenia zawierające całą lub prawie całą zawartość tablicy routingu, co może stanowić znaczące obciążenie łącz sieciowych.
6. Osiągnięcie stanu zbieżności przez sieć jest czasochłonne.



Rysunek 5.1. Przykładowa sieć z zaimplementowanym protokołem RIP.

Działanie protokołu RIP zostanie przeanalizowane na przykładzie sieci z rys. 5.1. Zawartość tablic routingu w kolejnych etapach osiągania zbież-

Tabela 5.1. Zawartości tablic routingu w kolejnych etapach osiągania zbieżności sieci z protokołem RIP.

Krok	R1	R2	R3	R4
0	A 0 ⇐	B 0 ⇐	C 0 ⇐	D 0 ⇐
	B 0 ⇒	C 0 ⇒	D 0 ⇒	E 0 ⇒
1	A 0 ⇐	A 1 ⇐	B 1 ⇐	C 1 ⇐
	B 0 ⇒	B 0 ⇐	C 0 ⇐	D 0 ⇐
	C 1 ⇒	C 0 ⇒	D 0 ⇒	E 0 ⇒
		D 1 ⇒	E 1 ⇒	
2	A 0 ⇐	A 1 ⇐	A 2 ⇐	B 2 ⇐
	B 0 ⇒	B 0 ⇐	B 1 ⇐	C 1 ⇐
	C 1 ⇒	C 0 ⇒	C 0 ⇐	D 0 ⇐
	D 2 ⇒	D 1 ⇒	D 0 ⇒	E 0 ⇒
		E 2 ⇒	E 1 ⇒	
3	A 0 ⇐	A 1 ⇐	A 2 ⇐	A 3 ⇐
	B 0 ⇒	B 0 ⇐	B 1 ⇐	B 2 ⇐
	C 1 ⇒	C 0 ⇒	C 0 ⇐	C 1 ⇐
	D 2 ⇒	D 1 ⇒	D 0 ⇒	D 0 ⇐
	E 3 ⇒	E 2 ⇒	E 1 ⇒	E 0 ⇒

ności jest przedstawiona w tabeli 5.1. Liczby przy nazwach sieci (A, B, C, D, E) oznaczają wartość metryki, natomiast strzałki wskazują kierunek, w którym należy wysłać pakiet, by dotarł do sieci docelowej. Po uruchomieniu, router automatycznie rozpoznaje i umieszcza w swojej tablicy routingu trasy do sieci, które są do niego bezpośrednio dołączone (krok 0), niezależnie od skonfigurowanego na nim protokołu routingu (oczywiście pod warunkiem poprawnego skonfigurowania i włączenia interfejsów sieciowych). Następnie, zgodnie z protokołem RIP, router wysyła rozgłoszenia zawierające informacje ze swojej tablicy routingu. Trafiają one do jego sąsiadów. W naszym przykładzie zawartość tablicy routingu routera R1 trafia do routera R2, tablica routera R2 trafia do R1 i R3, itd. Dzięki tym informacjom, każdy router może dodać do swojej tablicy routingu trasy, którymi dysponuje sąsiad, po zwiększeniu o 1 wartości metryki. W kroku 1. router R1 otrzymał od R2 informację o trasach do sieci B i C. Sieć C została dopisana do tablicy routingu R1 z metryką o 1 większą niż w tablicy routingu R2. Trasa do sieci B została zignorowana, ponieważ R1 już posiada lepszą trasę do tej sieci (jest bezpośrednio dołączona – metryka 0). Analogiczne procedury zastosowały wszystkie routery (krok 1.). Po raz kolejny informacje z tablic routingu zostaną rozesłane po 30 sekundach (krok 2.). Dopiero po 3 cyklach aktualizacji wszystkie routery będą dysponowały trasami do wszystkich pięciu sieci. Widać zatem, że wraz ze wzrostem rozmiarów sieci istotnie wzrasta

także czas potrzebny na rozpropagowanie aktualnych informacji do wszystkich routerów. W rzeczywistości aktualizacje nie są rozsyłane dokładnie co 30 sekund, lecz czas ten jest zmieniany o niewielką, losową wartość (*RIP jitter*), by uniknąć sytuacji, w której wszystkie routery wysyłają rozgłoszenia RIP w tym samym momencie.

W protokole RIP (który jest najbardziej typowym przedstawicielem protokołów wektora odległości) routery nie uzyskują kompletnej informacji o strukturze sieci. W związku z tym, w pewnych sytuacjach może powstać tzw. pętla routingu, w której pakiety krążą między kilkoma routerami, nigdy nie docierając do celu. Ostatecznie pakiety takie są eliminowane dzięki mechanizmowi TTL (*time-to-live*). Przy każdym przejściu pakietu przez router, wartość zapisana w polu TTL pakietu jest zmniejszana o 1, a gdy osiągnie wartość zerową, pakiet jest odrzucany. Nie zmienia to jednak faktu, że pętle routingu są zjawiskiem niepożądanym.

Załóżmy, że sieć z rys. 5.1 jest w pełni funkcjonalna, a zawartość tablic routingu jest zgodna z wierszem nr 3 tabeli 5.1. W pewnym momencie sieć E przestaje być dostępna wskutek wyłączenia interfejsu routera R4, do którego jest podłączona. Wówczas router R4 natychmiast usuwa wpis o bezpośrednio dołączonej sieci E ze swojej tablicy routingu. Jednak pozostałe routery nadal uważają trasy do sieci E, które mają w swoich tablicach routingu, za poprawne. Ponieważ komunikacja w sieci nie jest całkowicie niezawodna, router zaczyna uważać trasę za niepoprawną dopiero gdy informacja o niej nie zostanie odświeżona przez 180 sekund, mimo że oczekuje się aktualizacji co 30 sekund. Gdy router R4 otrzyma tablicę routingu routera R3, z wpisem o trasie do sieci E z metryką 1, będzie mógł dodać tę trasę do swojej tablicy routingu, z metryką 2, w kierunku routera R3 (traktując ją jako alternatywę dla utraconej ścieżki). Doprowadzi to do sytuacji, w której pakiety zaadresowane do sieci E będą krążyć między routerami R3 i R4, bez szansy dotarcia do celu. Błędna informacja zostanie rozpropagowana również do pozostałych routerów. Ponieważ router R3 nie będzie już otrzymywał od routera R4 informacji o trasie do sieci E z metryką 0, po 180 sekundach uzna tę trasę za niepoprawną. Może zastąpić ją trasa prowadząca poprzez router R2 (z gorszą metryką). Możemy tu zaobserwować proces pozornego, stopniowego "oddalania się" w rzeczywistości niedostępnej sieci E, nazywany zliczaniem do nieskończoności (ang. *count to infinity*). Oczywiście również to zjawisko nie jest pożądane.

Aby zminimalizować ryzyko wystąpienia powyższych, niepożądanych zjawisk, wprowadzono do protokołu RIP szereg dodatkowych mechanizmów:

1. Aby zatrzymać proces zliczania do nieskończoności, określono maksymalną możliwą wartość metryki: 16 oznacza niedostępną sieć.
2. Licznik wstrzymania (ang. *holddown timer*) przez pewien, określony czas powstrzymuje router przed akceptowaniem informacji o trasach alterna-

tywnych, gorszych od utraconej ścieżki (w naszym przypadku powinno to powstrzymać router R4 przed skorzystaniem z trasy do sieci E zaoferowanej przez R3).
3. Reguła podzielonego horyzontu (ang. *split horizon rule*) zabrania wysyłania informacji poprzez interfejs, przez który uprzednio ta informacja została uzyskana (zgodnie z nią router R3 nie będzie informował routera R4 o trasie do sieci E, ponieważ sam otrzymał ją właśnie od routera R4).
4. Zatruwanie tras (ang. *route poisoning*) polega na rozsyłaniu informacji o niedostępnej trasie z metryką 16 w celu jawnego poinformowania sąsiednich routerów o jej niedostępności. Zwykle jest stosowane w połączeniu z regułą podzielonego horyzontu.
5. Aktualizacje wyzwalane (ang. *triggered updates*) polegają na rozsyłaniu zawartości tablic routingu natychmiast po zmianie którejkolwiek z tras, bez czekania na standardową aktualizację (co 30 sekund). Przyspiesza to w znacznym stopniu uzyskiwanie przez sieć stanu zbieżności i jednocześnie zmniejsza ryzyko wystąpienia pętli routingu.

5.2. Podstawowa konfiguracja

W pierwszej kolejności należy upewnić się, że router ma prawidłowo skonfigurowane interfejsy i może komunikować się z bezpośrednio podłączonymi sieciami. Następnie, będąc w trybie konfiguracji globalnej, należy uruchomić tryb konfiguracji protokołu routingu, poleceniem[26]:

```
router rip
```

Dla każdej bezpośrednio dołączonej sieci, w której ma działać protokół RIP, wydajemy polecenie:

```
network network-address
```

gdzie *network-address* to klasowy adres sieci. Jeżeli router posiada 3 interfejsy sieciowe, np.: 172.16.1.1/24, 172.16.2.2/30 oraz 192.168.1.1/24, należy wydać polecenia:

```
router rip
 network 172.16.0.0
 network 192.168.1.0
```

W razie wydania polecenia:

```
 network 172.16.1.0
```

w pliku konfiguracyjnym znajdzie się wpis `network 172.16.0.0` (adres podsieci zostanie zastąpiony adresem sieci z domyślną maską dla danej klasy), bez żadnego dodatkowego komunikatu ze strony routera.

Działanie polecenia `network` jest podwójne. Uruchamia ono komunikację w protokole RIP poprzez interfejsy należące do wskazanej sieci, jak również powoduje, że router będzie informował sąsiadów o posiadaniu trasy do wskazanej sieci bezpośrednio do niego podłączonej. W razie gdy poprzez któryś z interfejsów routera nie jest osiągalny inny router z protokołem RIP (lecz np. sieć LAN zawierająca wyłącznie hosty), wówczas istotna jest tylko druga funkcja polecenia `network`, natomiast wysyłanie komunikatów protokołu RIP poprzez taki interfejs jest niewskazane (ze względów bezpieczeństwa oraz by nie generować zbędnego ruchu w sieci). Tego typu interfejsy należy skonfigurować jako pasywne (w trybie konfiguracji protokołu routingu):

```
Router(config-router)# passive-interface interface
```

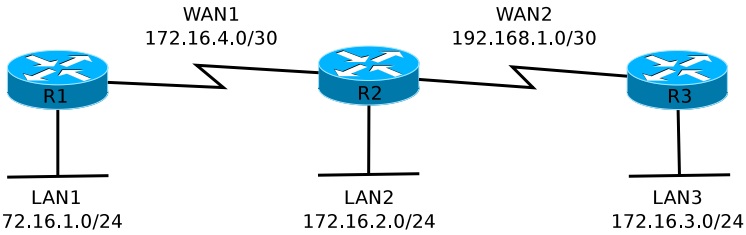
Jeżeli sieć posiada połączenie z Internetem, wówczas na wszystkich routerach należałoby skonfigurować trasę domyślną, informującą router o sposobie postępowania z pakietami przeznaczonymi dla nieznanych sieci (które chcemy przekazywać dostawcy usług internetowych). Jednak informacja o trasie domyślnej może być również automatycznie rozesłana do wszystkich routerów. Wystarczy skonfigurować ją na routerze brzegowym (mającym bezpośrednie połączenie z providerem) i wydać polecenie:

```
Router(config-router)# default-information originate
```

Domyślna trasa powinna automatycznie pojawić się w tablicach routingu pozostałych routerów dzięki protokołowi RIP.

5.3. Protokół RIPv2 – teoria i konfiguracja

RIPv2 jest bezpośrednim następcą protokołu RIPv1 omówionego powyżej (pisząc RIP często milcząco przyjmuje się, że chodzi o RIPv1). Różnica polega na tym, że oprócz adresów sieci docelowych w komunikatach rozsyłanych między routerami, podawane są również maski podsieci. RIPv2 jest dzięki temu protokołem bezklasowym, zgodnym z CIDR i VLSM. Do komunikacji między sąsiednimi routerami wykorzystywany jest adres multicastowy (224.0.0.9), a nie rozgłoszenia, jak w przypadku RIPv1. RIPv2 umożliwia też skonfigurowanie mechanizmów uwierzytelniania komunikacji między routerami. Poza tym, posiada wszystkie wady i ograniczenia swojego poprzednika.



Rysunek 5.2. Przykładowa sieć do implementacji protokołu RIPv2.

Załóżmy, że w sieci z rys. 5.2 został skonfigurowany protokół RIPv1. Nie zostanie uzyskany stan zbieżności, ponieważ sieć 172.16.0.0/16 jest nieciągła (przedzielona podsiecią 192.168.1.0/30) oraz zastosowano w niej zmienną maskę (24- i 30-bitową). Migracji do wersji drugiej protokołu dokonujemy poprzez wydanie na każdym z routerów polecenia (w trybie konfiguracji globalnej):

```
router rip
 version 2
```

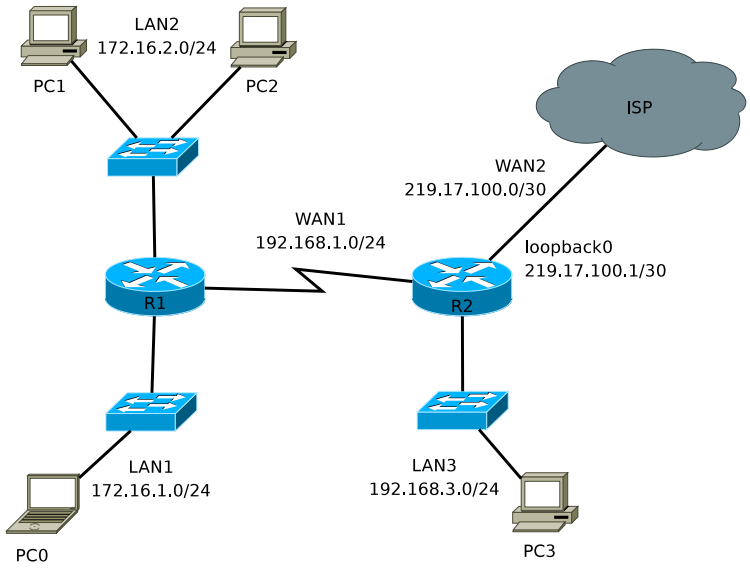
Powinno to rozwiązać problem różnych masek w obszarze LAN1 – WAN1 – LAN2. Jednak domyślnym zachowaniem protokołu RIPv2, podobnie jak RIPv1, jest dokonywanie automatycznego podsumowania na granicy sieci (klasowej). Zatem router R2 będzie informował router R3 o trasie do sieci 172.16.0.0/16 (a nie 172.16.1.0/24, 172.16.2.0/24 i 172.16.4.0/30). R3 nie skorzysta z tej trasy, ponieważ sam dysponuje lepszą trasą do sieci 172.16.0.0 (jest do niej bezpośrednio dołączony). Co więcej, będzie on również informował router R2 o swojej trasie do tej sieci. W tym przypadku mechanizm automatycznego podsumowania należy wyłączyć na routerach R2 i R3:

```
router rip
 no auto-summary
```

W wyniku tego działania, w tablicach routingu wszystkich routerów powinno być wyszczególnionych 5 tras (do wszystkich podsieci).

5.4. Zadanie 1 – RIPv1

1. Zbuduj sieć zgodnie ze schematem (rys. 5.3). Chmura jest jej wirtualną częścią (symuluje połączenie z dostawcą usług internetowych).
2. Skonfiguruj adresy IP urządzeń (interfejsów fizycznych i Loopback routerów oraz komputerów PC). Przejrzyj zawartość tablic routingu (polece-



Rysunek 5.3. Schemat topologii logicznej sieci.

nie `show ip route`). W przypadku korzystania z konfiguracji z poprzedniego ćwiczenia (routing statyczny), usuń wszystkie trasy statyczne.
3. Na routerze R1 skonfiguruj protokół RIP dla sieci LAN1, LAN2 i WAN1.
4. Na routerze R2 skonfiguruj protokół RIP dla sieci LAN3 i WAN1.
5. Upewnij się, że trasy do sieci LAN1, LAN2, LAN3, WAN1 znalazły się w tablicach routingu obu routerów.
6. Przeanalizuj informacje podawane przez polecenie `show ip protocol`, odnoszące się do protokołu RIP.
7. Przeanalizuj działanie protokołu RIP (`debug ip rip`). Jakie informacje są przesyłane między routerami? W którym momencie następuje inkrementacja metryki? Czy działa reguła podzielonego horyzontu? Czy router R1 dokonuje podsumowania tras? Wyłącz debugowanie.
8. Skonfiguruj odpowiednie interfejsy jako pasywne i sprawdź, czy faktycznie w taki sposób działają.
9. Na routerze R2 skonfiguruj trasę domyślną poprzez interfejs Loopback0. Użyj polecenia `default-information originate`. Czy trasa domyślna pojawiła się również w tablicy routingu routera R2? Sprawdź, czy jest wykorzystywana.
10. Wykonaj kopie plików konfiguracyjnych routerów.

5.5. Zadanie 1 – Rozwiązanie

Listing 5.1. Istotne fragmenty konfiguracji routera R1.

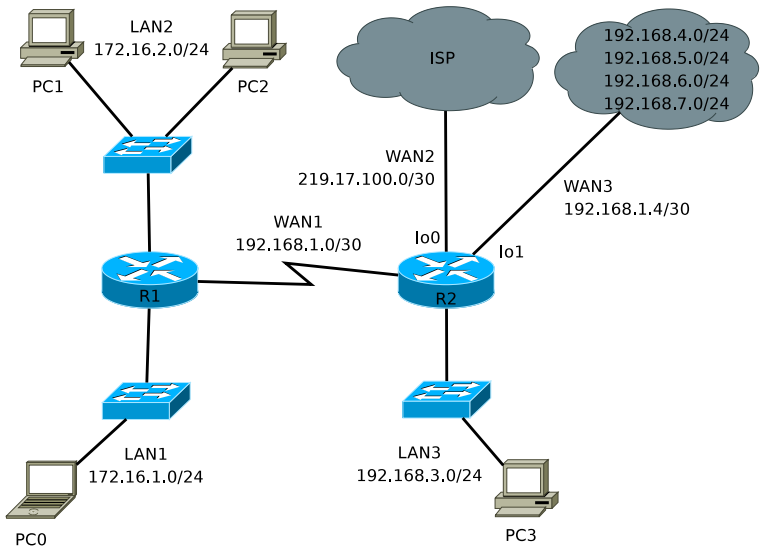
```
hostname R1
!
interface FastEthernet0/0
 ip address 172.16.1.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/1
 ip address 172.16.2.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/0/0
 ip address 192.168.1.1 255.255.255.0
 clock rate 64000
!
router rip
 passive-interface FastEthernet0/0
 passive-interface FastEthernet0/1
 network 172.16.0.0
 network 192.168.1.0
!
ip classless
!
end
```

Listing 5.2. Istotne fragmenty konfiguracji routera R2.

```
hostname R2
!
interface Loopback0
 ip address 219.17.100.1 255.255.255.252
!
interface FastEthernet0/0
 ip address 192.168.3.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/0/0
 ip address 192.168.1.2 255.255.255.0
!
router rip
 passive-interface FastEthernet0/0
 network 192.168.1.0
 network 192.168.3.0
 default-information originate
```

```
19 !
   ip classless
21 ip route 0.0.0.0 0.0.0.0 Loopback0
   !
23 end
```

5.6. Zadanie 2 – RIPv2



Rysunek 5.4. Schemat topologii logicznej sieci.

1. Zbuduj sieć zgodnie ze schematem (rys. 5.4). Jako punkt startowy wykorzystaj sieć z poprzedniego zadania. Należy zmienić maski w sieci WAN1 oraz skonfigurować dodatkowy interfejs Loopback1 (z adresem 192.168.1.5/30) na routerze R2. Interfejs Loopback1 symuluje połączenie z sieciami 192.168.4.0/24 – 192.168.7.0/24.
2. Dokonaj migracji protokołu RIP do wersji 2 na obu routerach.
3. Przejrzyj zawartość tablic routingu. Zaobserwuj działanie protokołu RIP (`show ip protocol`, `debug ip rip`). Czy informacje o maskach podsieci są przesyłane między routerami? Czy routery dokonują automatycznego podsumowania (*auto-summary*)?
4. Wyłącz automatyczne podsumowanie na R1 (`no auto-summary`). Kontynuuj obserwację wiadomości RIP (na routerze R1) oraz zawartości ta-

blicy routingu routera R2 przez co najmniej 3 minuty (do chwili pojawienia się wszystkich zmian wynikających z polecenia `no auto-summary`).
5. Na routerze R2 skonfiguruj trasę statyczną do sieci 192.168.4.0/24 – 192.168.7.0/24, poprzez interfejs Loopback1, stosując odpowiednie podsumowanie i jedno polecenie `ip route` (trasa do supersieci, z wykorzystaniem mechanizmu agregacji).
6. Spowoduj, by router R2 rozesłał informację o supersieci (korzystając z polecenia `redistribute static`). Zweryfikuj zawartość tablicy routingu routera R1.
7. Zachowaj kopie konfiguracji routerów R1 i R2. Przywróć ustawienia fabryczne routerów.

5.7. Zadanie 2 – rozwiązanie

Listing 5.3. Istotne fragmenty konfiguracji routera R1.

```
hostname R1
!
interface FastEthernet0/0
 ip address 172.16.1.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/1
 ip address 172.16.2.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/0/0
 ip address 192.168.1.1 255.255.255.252
 clock rate 64000
!
router rip
 version 2
 network 172.16.0.0
 network 192.168.1.0
 no auto-summary
!
ip classless
!
end
```

Listing 5.4. Istotne fragmenty konfiguracji routera R2.

```
hostname R2
!
```

```
3 interface Loopback0
   ip address 219.17.100.1 255.255.255.252
5 !
  interface Loopback1
7  ip address 192.168.1.5 255.255.255.252
  !
9 interface FastEthernet0/0
   ip address 192.168.3.1 255.255.255.0
11 duplex auto
   speed auto
13 !
  interface Serial0/0/0
15 ip address 192.168.1.2 255.255.255.252
  !
17 router rip
   version 2
19 redistribute static
   network 192.168.1.0
21 network 192.168.3.0
   default-information originate
23 !
  ip classless
25 ip route 0.0.0.0 0.0.0.0 Loopback0
  ip route 192.168.4.0 255.255.252.0 Loopback1
27 !
  end
```

Rozdział 6

EIGRP

6.1. Wstęp

EIGRP jest następcą protokołu IGRP (Interior Gateway Routing Protocol) [20], niewspieranego już przez nowsze wersje IOS Cisco (począwszy od 12.3). Opracowanie IGRP wynikało z braku prostego protokołu routingu typu wektora odległości, który mógłby być alternatywą dla przestarzałego protokołu RIP. EIGRP (Enhanced Interior Gateway Routing Protocol) [30, 31] jest bezklasowym protokołem routingu wewnętrznego, charakteryzującym się dobrą skalowalnością (istnieją implementacje liczące nawet ponad 1000 routerów) i szybkim osiąganiem stanu zbieżności, wykorzystującym niewielką część pasma sieci. Jest on protokołem zastrzeżonym przez Cisco (podobnie jak IGRP), co ogranicza zakres jego stosowalności do sieci zbudowanych na bazie routerów tej firmy. EIGRP jest protokołem typu wektora odległości, jednak posiada również pewne cechy protokołów stanu łącza, w związku z czym bywa określany mianem protokołu hybrydowego. Jednak charakter informacji o sieciach, gromadzonych przez routery, pozwala na jednoznaczne zakwalifikowanie go do grupy protokołów wektora odległości. EIGRP działa niezależnie od wykorzystywanego w sieci protokołu warstwy trzeciej. Po zainstalowaniu odpowiednich modułów (ang. *protocol-dependent modules*, PDM), może pracować dla AppleTalk, IP oraz IPX.

6.2. Podstawy działania EIGRP

W przypadku standardowego protokołu routingu typu wektora odległości (jakim jest RIP, opisany w poprzednim rozdziale), rozpropagowanie informacji o dostępnych sieciach do wszystkich routerów może wymagać wielu cykli periodycznego rozsyłania zawartości tablicy routingu między sąsiadami. Router, dysponując informacją o trasie do danej sieci docelowej, ignoruje informacje o alternatywnych trasach, charakteryzujących się gorsza metryką. W razie awarii któregokolwiek odcinka sieci, rozpropagowanie informacji o jego niedostępności i wyznaczenie alternatywnych tras jest również czasochłonne.

W protokole EIGRP aktualizacje informacji o routingu nie są rozsyłane periodycznie, lecz wyłącznie gdy wystąpi wymagająca tego zmiana (np. pojawi się nowa sieć lub istniejąca przestanie działać). Aktualizacja dotyczy wówczas jedynie informacji o trasie, która uległa zmianie. Router nie ignoruje informacji, które aktualnie nie są potrzebne, lecz gromadzi je w tablicy topologii. Oprócz najlepszej trasy do danej sieci docelowej (w terminologii EIGRP – sukcesor, ang. *successor*), wyznaczane są trasy zapasowe (sukcesor dopuszczalny, ang. *feasible successor*), które w razie awarii mogą natychmiast przejąć rolę uprzednio działającej trasy. Nie ma wówczas potrzeby

oczekiwania na nowe informacje ani przeliczania tras. Jeżeli jednak router nie posiada trasy zapasowej, wysyła do sąsiednich routerów zapytania w celu zgromadzenia brakujących informacji, umożliwiających wyznaczenie nowej drogi. Wymiana informacji między routerami odbywa się z wykorzystaniem wiarygodnego protokołu (RTP, ang. *Reliable Transport Protocol*), w sposób, który zapewnia, że routery podejmują decyzje o routingu dysponując kompletem informacji. To z kolei zapewnia spójność routingu i minimalizuje ryzyko wystąpienia pętli. Działanie poszczególnych segmentów sieci jest weryfikowane poprzez częste, periodyczne wysyłanie pakietów typu Hello między sąsiednimi routerami. Dzięki temu możliwa jest szybka reakcja wszystkich routerów na zmianę topologii.

6.3. Wymiana informacji między sąsiednimi routerami

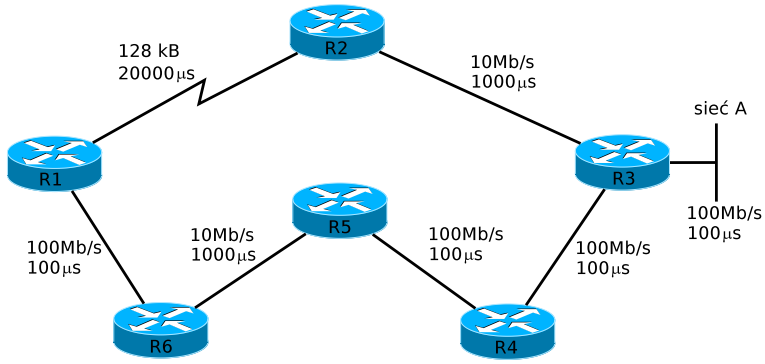
W standardowych protokołach routingu typu wektora odległości, informacje o routingu są rozsyłane periodycznie, z dużą częstotliwością. Brak informacji o istniejącej uprzednio trasie w kilku kolejnych aktualizacjach oznacza, że trasa stała się niedostępna i należy usunąć ją z tablic routingu. W protokole EIGRP informacje o routingu przesyłane są rzadko (tzn. tylko w razie zmiany w topologii sieci), w związku z czym potrzebny jest inny mechanizm testowania poszczególnych połączeń.

Jest do tego celu wykorzystywany podprotokół Hello. Przy jego pomocy router rozpoznaje swoich sąsiadów (tzn. routery, z którymi posiada bezpośrednie połączenie) i informacje u nich umieszcza w specjalnej tablicy (*neighbor table*). Następnie, do sąsiadów periodycznie wysyłane są pakiety typu Hello. Router oczekuje również pakietów Hello przychodzących od sąsiadów. Standardowo pakiety te są wysyłane na adres multicastowy 224.0.0.10, co 5 sekund (w sieciach rozgłoszeniowych takich jak Ethernet, połączeniach punkt-punkt i połączeniach wielopunktowych o przepustowości powyżej T1, np. Frame Relay) lub co 60 sekund (w wolniejszych połączeniach wielopunktowych). Czas ten określa, konfigurowalny w ustawieniach interfejsu, parametr *hello interval*. Jeżeli router nie otrzyma od sąsiada żadnego pakietu Hello w czasie równym parametrowi *hold time*, przyjmuje że połączenie z sąsiadem przestało być dostępne. Domyślnie parametr *hold time* jest trzy razy większy niż *hello interval*. W przeciwieństwie do informacji o routingu, pakiety Hello są niewielkie i, pomimo częstego przesyłania, nie stanowią istotnego obciążenia sieci. Ich otrzymanie nie jest dodatkowo potwierdzane.

Sąsiednie routery nie wymieniają między sobą informacji zawartych w tablicy routingu (jak w przypadku protokołu RIP), lecz w tablicy topologii (ang. *topology table*). Są w niej umieszczane informacje o wszystkich, znanych routerowi, trasach do wszystkich miejsc docelowych. Wymiana tych

informacji między sąsiadami odbywa się przy pomocy pakietów typu Update. Są one przesyłane w sposób wiarygodny, potwierdzane przy pomocy pakietów typu Acknowledgment. W pewnych sytuacjach stosuje się też pakiety typu Query i Reply (pytanie i odpowiedź, również potwierdzane przy pomocy Acknowledgment), co jest omówione w dalszej części rozdziału. Najlepsze trasy z tablicy topologii przepisywane są do tablicy routingu. Nazwa tablicy topologii sugeruje podobieństwo z topologiczną bazą danych, stosowaną w protokołach typu stanu łącza, na podstawie której można odtworzyć dokładną strukturę sieci. Skojarzenie to jest mylne. Tablica topologii zawiera jedynie dane umożliwiające obliczenie metryki i informacje o kierunku, w którym należy wysyłać pakiety.

6.4. Metryka EIGRP



Rysunek 6.1. Parametry wykorzystywane do obliczania metryki EIGRP.

Domyślnie, do obliczania metryki w protokole EIGRP wykorzystuje się najmniejszą szerokość pasma na trasie do sieci docelowej oraz całkowite opóźnienie. Rys. 6.1 przedstawia przykładową sytuację. W przypadku routera R1 i metryki do sieci A, dla trasy R1, R2, R3, szerokość pasma będzie wynosiła 128 kB/s, natomiast opóźnienie 21100 μs. Analogicznie, dla trasy R1, R6, R5, R4, R3, szerokość pasma wynosi 10 Mb/s, natomiast opóźnienie 1400 μs.

Do obliczenia wartości metryki, szerokość pasma (ang. *bandwidth*), wyrażona w kilobitach na sekundę, jest skalowana zgodnie z formułą [30] (gdzie znak "=" oznacza operację przypisania):

```
bandwidth = (10000000/bandwidth(i)) * 256
```

natomiast opóźnienie (ang. *delay*), wyrażone w dziesiątkach milisekund:

```
delay = delay * 256
```

Ostatecznie, wartość metryki obliczana jest zgodnie z formułą [30]:

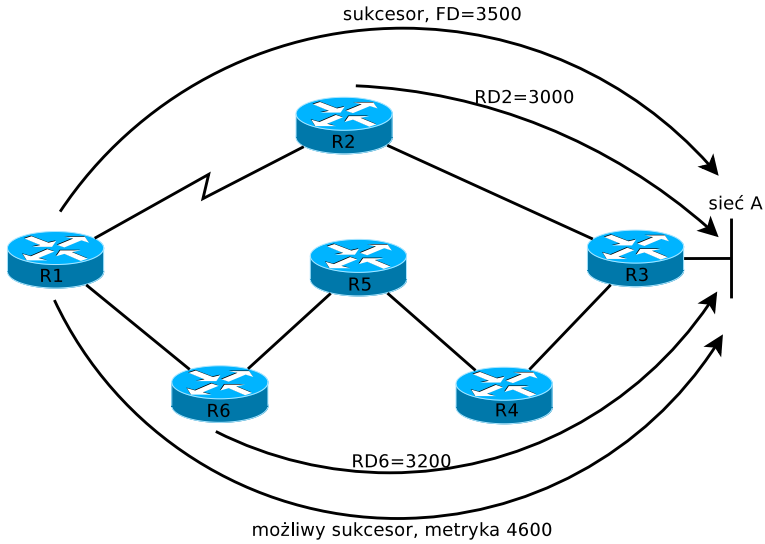
```
metric = [K1 * bandwidth + (K2 * bandwidth) / (256 - load)
        + K3 * delay] * [K5 / (reliability + K4)]
```

Oprócz szerokości pasma i opóźnienia, metryka może uwzględniać również obciążenia łącza (ang. *load*) oraz jego niezawodność (ang. *reliability*). Domyślnie jednak tylko wagi K1 i K3 mają wartość 1, natomiast pozostałe zerową. W przeciwieństwie do szerokości pasma i opóźnienia, obciążenie i niezawodność łącza mogą zmieniać się dynamicznie podczas pracy sieci, co z kolei może skutkować częstymi aktualizacjami informacji o routingu, przeliczaniem tras i niestabilną pracą sieci. Dlatego też ich wykorzystywanie nie jest zalecane. W większości przypadków pozostawienie standardowych wartości wag jest optymalnym rozwiązaniem. Ich modyfikowanie wymaga szczególnej uwagi. Istotne jest wówczas skonfigurowanie identycznych wag na wszystkich routerach w całej domenie routingu.

6.5. Wyznaczanie tras

Do wyznaczania najkrótszych tras do poszczególnych miejsc docelowych w sieci wykorzystywany jest algorytm DUAL (*Diffused Update Algorithm*). Jego celem jest znalezienie możliwie najlepszej, wolnej od pętli, trasy do każdego miejsca docelowego.

Sposób wyboru tras zostanie przedstawiony na przykładzie z rys. 6.2. Sytuacja zostanie przeanalizowana z punktu widzenia routera R1. Router R1 otrzymał od routera R2 informację, że dysponuje on trasą do sieci A, z metryką 3000 (RD2). Odległość, o jakiej informuje inny router, jest określana mianem odległości ogłaszanej (ang. *reported distance*, RD). R1 otrzymał informację o trasie do sieci A również od routera R6, z odległością ogłaszaną 3200 (RD6). Dla obu tych tras R1 oblicza własne wartości metryki (pakiety typu Update zawierają informacje o parametrach niezbędnych do obliczenia metryki, w szczególności o opóźnieniach i szerokości pasma). Trasa charakteryzująca się najmniejszą metryką staje się sukcesorem i jest umieszczana w tablicy routingu. Metryka sukcesora jest określana mianem odległości dopuszczalnej (ang. *feasible distance*, FD). W tym przypadku sukcesorem jest trasa R1, R2, R3. W razie gdyby trasa ta przestała być dostępna, np. z powodu awarii któregoś segmentu, istnieje trasa alternatywna: R1, R6, R5, R4, R3. Router R1 nie zna jednak jej przebiegu. Istnieje ryzyko, że



Rysunek 6.2. Trasy do sieci A z punktu widzenia routera R1.

trasa proponowana przez R6 mogłaby mieć następujący przebieg: R1, R6, R1, R2, R3 (przy założeniu wyłączenia reguły podzielonego horyzontu). W razie jej użycia, wystąpiłaby pętla routingu. Dlatego też trasą zapasową, określaną mianem dopuszczalnego sukcesora (ang. *faesible successor*) może być jedynie trasa, dla której odległość ogłaszana (w naszym przypadku RD6) jest mniejsza, niż metryka dotychczas wykorzystywanego sukcesora (FD). Jak widać, warunek ten jest spełniony ($RD6 < FD$). Zatem, w razie awarii sukcesora, jego rolę natychmiast przejmie dopuszczalny sukcesor (zajmie miejsce sukcesora w tablicy routingu). Nie będą potrzebne żadne dodatkowe przeliczenia ani wymiana informacji między routerami, więc czas niestabilnego działania sieci będzie krótki.

Załóżmy teraz, że na rys. 6.2 $RD6 = 4000$ (przy pozostałych wartościach niezmienionych). Teraz warunek $RD6 < FD$ nie jest spełniony. Nie każda trasa alternatywna spełnia warunek bycia dopuszczalnym sukcesorem. W takiej sytuacji, w razie awarii sukcesora, trasa zostanie przełączona w stan aktywny (pożądanym stanem, w którym trasa może być wykorzystywana do przesyłania pakietów, jest stan pasywny). Router poinformuje swoich sąsiadów o niedostępności trasy prowadzącej od niego do sieci A poprzez router R2. Uzyskawszy potwierdzenia (Acknowledgment), że informacja dotarła, roześle do nich zapytania (Query) o dostępność tras do sieci A. Router R6 nie zaproponuje trasy prowadzącej poprzez R1, ponieważ uprzednio uzyskał od R1 informację o jej niedostępności. Odpowiedź jest udzielana przy wykorzystaniu pakietów Reply. Jeżeli sąsiad nie dysponuje informacjami

pozwalającymi mu odpowiedzieć na zapytanie, jest ono przekazywane dalej. Żadne zapytanie nie może pozostać bez odpowiedzi. Gdy R1 uzyska pożądane informacje, nowy sukcesor będzie mógł być dodany do tablicy topologii i tablicy routingu, a trasa zostanie ponownie przełączona w stan pasywny.

W pewnych, niekorzystnych okolicznościach, przy mocno obciążonych routerach i niestabilnie działającej sieci, może się zdarzyć, że czas oczekiwania na odpowiedź na zapytanie będzie bardzo długi. Mówi się wówczas o problemie tras zablokowanych w stanie aktywnym (ang. *stuck in active routes*). Innym problemem związanym z protokołami routingu typu wektora odległości jest ryzyko wystąpienia pętli routingu. W przypadku EIGRP jest ono bardzo niewielkie. Oprócz mechanizmów omówionych powyżej, stosowane są zabezpieczenia przedstawione w rozdziale dotyczącym protokołu RIP: reguła podzielonego horyzontu i zatruwanie tras wstecz (*split horizon*, *poison reverse*).

6.6. Konfiguracja EIGRP

Podstawowa konfiguracja EIGRP jest podobna do konfiguracji protokołów RIP i OSPF. W przypadku rozbudowanych sieci, jego poprawna implementacja jest znacznie łatwiejsza niż implementacja OSPF.

Protokół EIGRP uruchamia się wydając w trybie konfiguracji globalnej polecenie [25]:

```
router eigrp AS-number
```

Wymaganym parametrem jest numer systemu autonomicznego (*AS-number*). Ponieważ EIGRP jest protokołem routingu wewnętrznego, nie musi to być numer zarejestrowany. W rzeczywistości, nazwa wynika ze względów historycznych, a liczbę tę należy traktować jako numer procesu (na jednym routerze można uruchomić maksymalnie 30 procesów EIGRP). Aby routery mogły komunikować się w protokole EIGRP, muszą mieć skonfigurowany ten sam numer systemu autonomicznego. W przeciwnym razie, konieczne jest skonfigurowanie redystrybucji. W przypadku, gdy w sieci występują jeszcze routery z protokołem IGRP, jeżeli w IGRP i EIGRP skonfigurowano ten sam numer systemu autonomicznego, redystrybucja tras między tymi dwoma protokołami jest realizowana automatycznie.

Następnie, podobnie jak przy konfigurowaniu protokołu RIP, dla każdej bezpośrednio dołączonej sieci, w której ma działać EIGRP, należy wydać polecenie:

```
network network-address
```

Możliwe jest również wyspecyfikowanie konkretnych podsieci, poprzez podanie dodatkowo maski blankietowej, podobnie jak przy konfiguracji OSPF (opisanej w Rozdziale 7.6).

OSPF, podobnie jak RIP, dokonuje automatycznego podsumowania na granicy sieci (klasowej), w związku z czym w przypadku nieciągłych sieci konieczne jest wydanie polecenia:

```
no auto-summary
```

Odpowiedniego podsumowania można natomiast dokonać samodzielnie, przy pomocy polecenia:

```
ip summary-address eigrp AS-number address mask
```

wydanego w trybie konfiguracji interfejsu (nie protokołu routingu, jak poprzednie polecenia). Konfigurując interfejsy szeregowe, należy podać ich szerokość pasma (w kilobitach na sekundę), poleceniem `bandwidth`, np.:

```
interface Serial0/0/0
 bandwidth 128
```

Skonfigurowana wartość nie ma żadnego wpływu na pracę samego interfejsu. Jest to jedynie informacja wykorzystywana przez protokoły routingu do obliczania metryki. Aby możliwe było wybranie optymalnych tras, należy konfigurować wartości zgodne z faktycznymi przepustowościami interfejsów.

Parametry *hello interval* i *hold time* można skonfigurować na poszczególnych interfejsach poleceniami:

```
ip hello-interval eigrp AS-number seconds
ip hold-time eigrp AS-number seconds
```

W przeciwieństwie do analogicznych ustawień w protokole OSPF, wartości tych parametrów, skonfigurowane na komunikujących się interfejsach sąsiednich routerów, nie muszą być identyczne.

Informację o trasie domyślnej (np. połączeniu z Internetem), skonfigurowanej statycznie na jednym z routerów, można automatycznie rozesłać do pozostałych poleceniem:

```
router eigrp AS-number
 redistribute static
```

W przypadku istnienia kilku alternatywnych tras do danego miejsca docelowego, charakteryzujących się jednakową metryką, domyślnie maksymalnie cztery mogą zostać umieszczone w tablicy routingu i jest wówczas realizowane rozkładanie obciążenia na trasy równorzędne (ang. *equal-cost*

load balancing). EIGRP umożliwia także rozkładanie obciążenia na trasy różniące się metryką (*unequal-cost load balancing*). Przy pomocy polecenia `variance` można ustalić, ile razy metryka wykorzystywanych tras może być większa, niż metryka najlepszej trasy. Przykładowo, jeżeli w trybie konfiguracji EIGRP zostanie wydane polecenie

```
variance 2
```

a najlepsza trasa posiada metrykę 1300, wówczas w tablicy routingu mogą znaleźć się również trasy alternatywne z metryką mniejszą niż 2600.

W typowej sytuacji, komunikacja między routerami w protokole EIGRP nie stanowi istotnego obciążenia sieci. Jednak w dużych sieciach, w chwili startu protokołu lub gdy sieć zachowuje się niestabilnie, zapotrzebowanie EIGRP na zasoby sieciowe może być znacznie większe. Przy pomocy polecenia:

```
ip bandwidth-percent eigrp AS-number percent
```

można, w trybie konfiguracji interfejsu, procentowo określić, jaka część dostępnego pasma może być wykorzystana przez EIGRP. Domyślnie jest to 50 procent.

Włączenie mechanizmów uwierzytelniania zapewnia, że router będzie akceptował komunikaty EIGRP wyłącznie od routerów, które mają skonfigurowany identyczny klucz. Najpierw, w trybie konfiguracji globalnej, na wszystkich routerach należy stworzyć zestaw kluczy (ang. *keychain*) i dodać do niego klucz, np.:

```
key chain MojeKlucze
 key 1
  key-string KluczDlaEIGRP
  end
```

Następnie, w trybie konfiguracji interfejsów, między którymi komunikacja EIGRP ma być szyfrowana, należy wydać polecenia:

```
ip authentication mode eigrp 4 md5
ip authentication key-chain eigrp 4 MojeKlucze
```

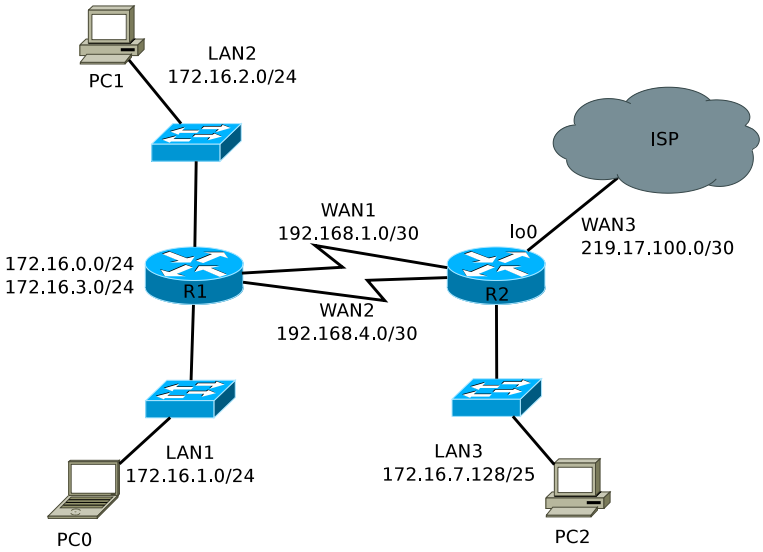
“4“ w powyższym przykładzie jest numerem systemu autonomicznego.

Do weryfikacji działania protokołu EIGRP dostępnych jest kilka poleceń `show`:

— `show ip protocols`
— `show ip eigrp interfaces` wyświetla informacje o konfiguracji EIGRP na poszczególnych interfejsach.

— `show ip eigrp neighbors` informuje o sąsiednich routerach zidentyfikowanych przez EIGRP.
— `show ip eigrp topology` wyświetla zawartość tablicy topologii routera. Domyślnie pokazywane są tylko trasy pełniące rolę sukcesora lub dopuszczalnego sukcesora. Parametr `all-links` skutkuje wyświetlaniem informacji o wszystkich trasach. Jest również podawana informacja o aktywnym lub pasywnym stanie trasy.
— `show ip eigrp traffic` wyświetla informacje o liczbie wysłanych i odebranych pakietów EIGRP poszczególnych typów.

6.7. Zadanie



Rysunek 6.3. Schemat topologii logicznej sieci.

1. Zbuduj sieć zgodnie ze schematem (rys. 6.3). Przeprowadź podstawową konfigurację routerów (nazwy, hasła, adresy IP interfejsów) i komputerów PC. Podłączenie sieci 172.16.0.0/24 i 172.16.3.0/24 do routera R1 jest planowane w przyszłości. Podaj identyczną szerokość pasma obu połączeń szeregowych (`bandwidth`). Aktualnie ustawioną wartość można sprawdzić poleceniem `show interface serial x/y`.
2. Na routerze R2 skonfiguruj trasę domyślną poprzez interfejs Loopback0.
3. Skonfiguruj protokół EIGRP dla sieci LAN1, LAN2, LAN3, WAN1, WAN2. Na razie nie wyłączaj automatycznego podsumowania.

4. Przejrzyj zawartość tablic routingu. Czy pojawiły się trasy sumaryczne Null0? Czy jest możliwa komunikacja między LAN1 i LAN3?
5. Wyłącz automatyczne podsumowanie (`no auto-summary`) na obu routerach. Zaobserwuj zmiany w tablicach routingu.
6. Na routerze R1 skonfiguruj na obu interfejsach szeregowych ręczne podsumowanie (agregację) dla sieci 172.16.0.0/24 – 172.16.3.0/24 (używając polecenia: `ip summary-address`). Zaobserwuj zmiany w tablicy routingu R2.
7. Na routerze R2 skonfiguruj redystrybucję trasy domyślnej (użyj polecenia `redistribute static`). Przejrzyj zawartość tablicy routingu R1.
8. Czy w tablicach routingu występują trasy alternatywne?
9. Zapoznaj się z poleceniami diagnostycznymi protokołu EIGRP:
   — `show ip protocols`,
   — `show ip eigrp interfaces`,
   — `show ip eigrp neighbors`,
   — `show ip eigrp topology`,
   — `show ip eigrp traffic`.
   Czy wszystkie trasy znajdują się w stanie pasywnym?
10. Zmień informację o przepustowości (`bandwidth`) jednego z połączeń szeregowych (alternatywna, gorsza trasa powinna zniknąć z tablicy routingu). Przejrzyj tablicę topologii (`show ip eigrp topology` z opcjonalnym parametrem `[all-links]`). Czy widzisz trasy pełniące rolę sukcesora i dopuszczalnego sukcesora? Czy warunek dopuszczalności dla tras alternatywnych jest spełniony?
11. Przy pomocy polecenia `variance` skonfiguruj rozkładanie obciążenia na trasy nierównorzędne (tak, aby wykorzystywać oba połączenia szeregowe, pomimo różnych metryk).
12. Zachowaj kopie konfiguracji routerów R1 i R2.

6.8. Rozwiązanie

Listing 6.1. Istotne fragmenty konfiguracji routera R1.

```
hostname R1
!
interface FastEthernet0/0
 ip address 172.16.1.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/1
 ip address 172.16.2.1 255.255.255.0
 duplex auto
```

```
 speed auto
!
interface Serial1/0
 bandwidth 56
 ip address 192.168.1.1 255.255.255.252
 ip summary-address eigrp 1 172.16.0.0 255.255.252.0
 clock rate 56000
!
interface Serial1/1
 bandwidth 64
 ip address 192.168.1.5 255.255.255.252
 ip summary-address eigrp 1 172.16.0.0 255.255.252.0
 clock rate 64000
!
router eigrp 1
 variance 2
 network 172.16.0.0
 network 192.168.1.0
 no auto-summary
!
ip classless
!
end
```

Listing 6.2. Istotne fragmenty konfiguracji routera R2.

```
hostname R2
!
interface Loopback0
 ip address 219.17.100.1 255.255.255.252
!
interface FastEthernet0/0
 ip address 172.16.7.129 255.255.255.128
 duplex auto
 speed auto
!
interface Serial1/0
 bandwidth 56
 ip address 192.168.1.2 255.255.255.252
!
interface Serial1/1
 bandwidth 64
 ip address 192.168.1.6 255.255.255.252
!
!
router eigrp 1
 variance 2
 redistribute static
 network 192.168.1.0
 network 172.16.0.0
```

```
 no auto-summary
!
ip classless
ip route 0.0.0.0 0.0.0.0 Loopback0
!
end
```

Rozdział 7

OSPF

7.1. Wstęp

OSPF (ang. *Open Shortest Path First*) jest protokołem routingu wewnętrznego, typu stanu łącza (*link-state*). Podobnie jak RIP, jest otwartym standardem (o czym wprost mówi słowo *open* w nazwie), opisanym w RFC 2328 [11]. Obecnie powszechnie stosowana jest 2. wersja. Został opracowany przez grupę związaną z IETF (*Internet Engineering Task Force*[1]). Powodem wprowadzenia OSPF była potrzeba zastąpienia protokołu RIP bardziej wyrafinowanym mechanizmem. OSPF jest przeznaczony wyłącznie dla środowiska TCP/IP. Jest protokołem bezklasowym, zgodnym z mechanizmami VLSM i CIDR (adresy miejsc docelowych są przekazywane wraz z maskami). Do komunikacji między routerami wykorzystywane są pakiety multicastowe. Jest możliwe również uwierzytelnianie. OSPF szybko reaguje na zmiany topologii sieci – stan zbieżności jest osiągany w krótkim czasie. W przypadku konfiguracji zgodnej z dobrymi praktykami, ruch związany z protokołem routingu nie powinien stanowić istotnego obciążenia infrastruktury sieciowej. Protokół charakteryzuje się również dobrą skalowalnością. Może sprawnie funkcjonować w sieciach liczących setki lub nawet tysiące routerów. Warunkiem koniecznym jest jednak poprawne przeprowadzenie dość skomplikowanej konfiguracji uwzględniającej specyfikę konkretnej sieci.

W protokołach stanu łącza, jakakolwiek zmiana topologii sieci (np. wyłączenie interfejsu, pojawienie się nowej sieci, itp.) powoduje rozesłanie informacji o niej do wszystkich routerów wewnątrz danego systemu autonomicznego. W przypadku rozbudowanych sieci, rozsyłanie takich komunikatów może pochłonąć znaczącą część przepustowości sieci. Ponadto, po otrzymaniu informacji o zmianie topologii, wszystkie routery muszą dokonać przeliczenia tras. Podczas tego procesu sieć może zachowywać się w sposób nieprzewidywalny. Dlatego protokół OSPF oferuje możliwość grupowania sieci w tzw. obszary (ang. *area*). Szczegóły topologii danego obszaru są wówczas ukryte przed pozostałą częścią systemu autonomicznego. Pozwala to na znaczące zredukowanie ruchu generowanego przez OSPF i poprawę stabilności działania sieci.

7.2. Podstawowe pojęcia

Do opisu działania OSPF stosuje się zestaw pojęć, spośród których część jest charakterystyczna tylko dla tego protokołu. Najważniejsze z nich są przedstawione w tym podrozdziale [11].

[1] `http://www.ietf.org/`

Identyfikator routera (*Router ID*) jest 32-bitową liczbą, zapisywaną identycznie jak adresy IPv4, unikalną wewnątrz systemu autonomicznego. Identyfikatory są niezbędne do działania protokołu OSPF.

Protokół OSPF rozróżnia kilka typów sieci i w zależności od niego, wykazuje pewne specyficzne zachowania. Sieć typu punkt-punkt (*point-to-point network*) łączy ze soba dwa routery. Zwykle jest to łącze szeregowe. Sieć rozgłoszeniowa (*broadcast network*) umożliwia fizyczne przesłanie tej samej wiadomości do wielu urządzeń, czyli działa w niej mechanizm rozgłoszeń. Jest on wykorzystywany przez protokół OSPF. Przykładem jest sieć Ethernet. Sieć nierozgłoszeniowa (*non-broadcast network*) umożliwia połączenie więcej niż dwóch routerów, jednak bez możliwości przesyłania rozgłoszeń. Wiadomości, które w przypadku sieci rozgłoszeniowych są wysyłane jako multicast, w sieciach nierozgłoszeniowych muszą być oddzielnie wysyłane do każdego odbiorcy. Przykładem jest Frame Relay. W sieciach nierozgłoszeniowych OSPF może symulować działanie sieci rozgłoszeniowej (tryb *non-broadcast multi-access*, NBMA) lub traktować taką sieć jak zbiór połączeń typu punkt-punkt (tryb *Point-to-MultiPoint*).

Sąsiednie routery (*neighboring routers*) to routery, które posiadają interfejsy dołączone do tej samej sieci. Para sąsiednich routerów może (w określonej sytuacji) nawiązać relację przylegania (*adjacency*), w celu wymiany między sobą informacji o routingu.

Ogłoszenie stanu łącza (*link state advertisement*, LSA) jest to jednostka informacji o stanie routera (tzn. jego interfejsów i sąsiadów w relacji przylegania) lub sieci. Informacje LSA są rozsyłane z wykorzystaniem mechanizmu rozsyłania zalewowego (*flooding*) i zbierane przez wszystkie routery w celu zapisania w bazie danych o stanach łącz (*link state database*).

Protokół Hello służy do nawiązywania i zarządzania relacjami sąsiedztwa z innymi routerami. Umożliwia weryfikację poprawnego działania poszczególnych połączeń w sieci.

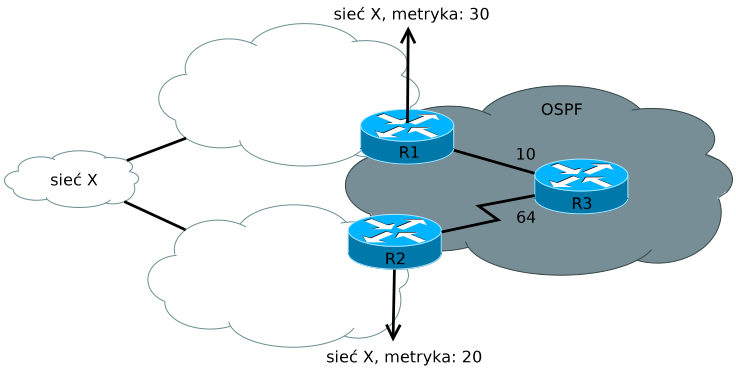
Router desygnowany (*Designated Router*) w sieci rozgłoszeniowej lub NBMA jest odpowiedzialny między innymi za generowanie LSA sieci. Mechanizm ten jest szerzej omówiony w dalszej części rozdziału.

7.3. Metryka protokołu

Metryką stosowaną w protokole OSPF jest koszt. Jest on związany z każdym interfejsem wyjściowym routera i może być konfigurowany przez administratora. Koszt całej trasy jest sumą algebraiczną kosztów przypisanych kolejnym interfejsom, przez które wysyłany jest pakiet. Preferowane są trasy charakteryzujące się najmniejszym kosztem. Protokół nie precyzuje sposobu określania kosztów. W przypadku implementacji stosowanej przez Cisco

przy domyślnej konfiguracji, koszt jest odwrotnie proporcjonalny do pasma, zgodnie z formułą [33]:

$$koszt = \frac{10^8}{pasmo[b/s]} \tag{7.1}$$



Rysunek 7.1. Trasy zewnętrzne w OSPF (liczby obok routera R3 informują o kosztach przypisanych interfejsom).

OSPF może również rozgłaszać informacje o routingu pochodzące z zewnątrz, tzn. od innego protokołu routingu, trasy statyczne lub domyślne. W przykładzie z rys. 7.1, informacja o trasie do sieci X jest rozgłaszana w systemie autonomicznym z protokołem OSPF (ciemna chmura) przez dwa routery brzegowe: R1 i R2, z kosztem 30 i 20 odpowiednio. W przypadku tras zewnętrznych, stosuje się dwa typy metryk.

Metryka zewnętrzna typu 1 jest traktowana identycznie, jak zwykła metryka OSPF. Z punktu widzenia routera R3, sieć X będzie osiągalna poprzez sąsiada R1 trasą o koszcie 40 lub poprzez sąsiada R2, trasą o koszcie 84. Całkowity koszt trasy do sieci zewnętrznej jest tu sumą kosztu, o którym informuje router brzegowy (R1 lub R2) i kosztu trasy od danego routera (tu R3) do routera brzegowego. W tablicy routingu routera R3 znajdzie się trasa o niższym koszcie całkowitym (tzn. poprzez R1).

W razie zastosowania metryki typu 2, jedynym kryterium wyboru trasy do sieci X przez router R3 jest wartość metryki rozgłaszanej przez routery brzegowe, z pominięciem kosztu trasy od R3 do routera brzegowego. W tym przypadku router R3 wybierze trasę poprzez sąsiada R2. Przy stosowaniu metryki typu 2 zakłada się, że główny koszt transmisji pakietu jest związany z routingiem między systemami autonomicznymi, a koszt transmisji wewnątrz własnego systemu jest pomijalnie mały. W razie dostępności tras zewnętrznych typu 1 i 2 do tej samej sieci docelowej, preferowane są trasy z metryką typu 1.

7.4. Komunikacja między routerami

OSPF korzysta bezpośrednio z protokołu IP do wymiany informacji między routerami. Wyróżnia się 5 typów pakietów OSPF, przedstawionych w tabeli 7.1.

Tabela 7.1. Typy pakietów OSPF [11].

Typ	Nazwa pakietu	Funkcja
1	Hello	Nawiązywanie i utrzymywanie relacji z sąsiadami
2	Database Description	Streszczenie informacji w bazie danych o stanach łącz
3	Link State Request	Żądanie informacji o stanie łącz
4	Link State Update	Informacja o stanie łącz
5	Link State Ack	Potwierdzenie otrzymania wiadomości

Podprotokół Hello jest odpowiedzialny za nawiązywanie i podtrzymywanie relacji między sąsiednimi routerami, jak również zapewnienie dwukierunkowej komunikacji między nimi. Pakiety Hello są periodycznie wysyłane poprzez wszystkie interfejsy, jako pakiety typu multicast, na adres 224.0.0.5 (adresatem są wszystkie routery OSPF – AllSPFRouters). Wysyłając pakiety, router informuje sąsiadów o swojej obecności, natomiast odbierając je, weryfikuje poprawną pracę poszczególnych łącz. W przypadku urządzeń Cisco, w sieciach wielodostępnych i punkt-punkt, pakiety Hello są domyślnie wysyłane co 10 sekund – czas ten określa parametr *HelloInterval*. Jeżeli router nie odbierze pakietu Hello od któregoś z sąsiadów przez czas określony parametrem *RouterDeadInterval*, wówczas przyjmuje, że przestał on być dostępny. *RouterDeadInterval* jest zwykle cztery razy dłuższy niż *HelloInterval*. Oba parametry muszą mieć identyczną wartość na sąsiednich interfejsach. Jak widać, pakiety Hello są rozsyłane dość często, jednak ze względu na małą wielkość nie stanowią istotnego obciążenia sieci. Ich odebranie nie jest potwierdzane pakietami Link State Ack.

W przypadku wszystkich protokołów routingu typu stanu łącza, istotna jest synchronizacja baz danych o stanach łącz. Na podstawie informacji zawartej w pakiecie Database Description, router jest w stanie stwierdzić, że jego sąsiad posiada nowszą kopię bazy danych niż on sam. Wysyła wówczas żądanie przesłania brakujących informacji (pakiet Link State Request), w odpowiedzi otrzymując Link State Update. Ze względu na zasadnicze znaczenie synchronizacji baz danych, komunikacja ta musi być realizowana w sposób wiarygodny, z wykorzystaniem potwierdzeń – pakietów Link State Ack. W protokole OSPF synchronizacja baz danych dotyczy wyłącznie są-

siednich routerów, które są ze sobą w tzw. relacji przyległości (ang. *adjacent routers*).

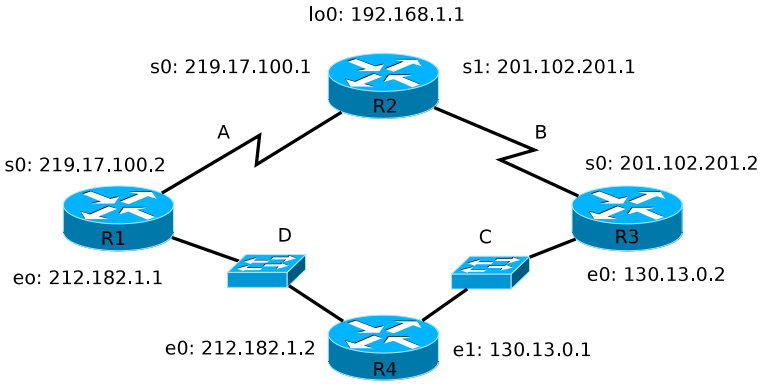
W przypadku sieci typu wielodostępnego może zaistnieć sytuacja, w której do jednej sieci są dołączone więcej niż dwa routery (np. połączone ze sobą poprzez przełącznik Ethernet). Procedura synchronizacji bazy danych każdego z routerów ze wszystkimi jego sąsiadami dostępnymi poprzez ten sam interfejs generowałaby wówczas dużą liczbę pakietów. Dlatego w sieciach wielodostępnych wybierany jest jeden router o specjalnym przeznaczeniu – tzw. router desygnowany (ang. *Designated Router*, DR). Pozostałe routery (tzw. DROther) nawiązują wówczas relację przyległości z routerem desygnowanym, natomiast z pozostałymi sąsiadami już nie.

Podczas ustanawiania relacji przyległości z sąsiadem, interfejs routera przechodzi poprzez sekwencję kilku stanów. Docelowo interfejs powinien znaleźć się w jednym z poniższych stanów:

— Wyłączony (*Down*) – nie otrzymano żadnych informacji w protokole OSPF w tym segmencie sieci.
— Dwukierunkowy (*Two-way*) – nawiązano dwukierunkową komunikację z sąsiadem, router odnalazł informację o sobie w pakietach Hello przesyłanych przez sąsiada. W przypadku sieci wielodostępnych jest to docelowy stan interfejsu w stosunku do sąsiadów, którzy nie pełnią roli DR ani BDR (DROthers).
— Pełny (*Full*) – pełna relacja przyległości.

Wybór routera desygnowanego jest dokonywany przy użyciu protokołu Hello. Pakiet Hello posiada pole, w którym zapisywany jest priorytet routera - nadawcy (*Router Priority*), konfigurowalny dla każdego interfejsu. Po uruchomieniu interfejsu routera, dokonywane jest sprawdzenie, czy w sieci już znajduje się router desygnowany. Jeżeli tak, sytuacja ta jest akceptowana, niezależnie od skonfigurowanych priorytetów. Routerem desygnowanym zostaje więc urządzenie, które zostało włączone jako pierwsze. W razie jednoczesnego lub prawie jednoczesnego uruchomienia kilku routerów, desygnowanym routerem zostaje urządzenie posiadające najwyższą wartość priorytetu. Jeżeli nie jest możliwe rozstrzygnięcie na podstawie priorytetu (np. gdy pozostawiono identyczne, domyślne wartości), zwycięża router o najwyższej wartości identyfikatora (*Router ID*). Kolejny w rankingu router otrzymuje rolę zapasowego routera desygnowanego (ang. *Backup Designated Router*, BDR). BDR odbiera komunikaty adresowane do DR (natomiast nie generuje własnych LSA) i razie awarii routera desygnowanego przejmuje jego rolę. W razie awarii obu, konieczne jest ponowne przeprowadzenie procedury wyborów DR i BDR. Routery DR i BDR odbierają pakiety wysyłane na adres multicastowy 224.0.0.6 (AllDRouters).

W przypadku routerów Cisco, identyfikator może zostać ręcznie skonfigurowany przez administratora lub stworzony automatycznie. Przy automa-



Rysunek 7.2. Przykładowa sieć z protokołem OSPF.

Tabela 7.2. Indetyfikatory routerów z rys. 7.2.

Router	Router ID
R1	219.17.100.2
R2	192.168.1.1
R3	201.102.201.2
R4	212.182.1.2

tycznym tworzeniu, identyfikatorem staje się najwyższy co do wartości adres IP interfejsu typu Loopback. W razie braku interfejsu Loopback, identyfikatorem staje się najwyższy co do wartości adres IP działającego interfejsu fizycznego routera. Ostatnie rozwiązanie nie jest rekomendowane, ponieważ wyłączenie interfejsu skutkuje utratą identyfikatora i chwilowo niestabilną pracą sieci (zmiana identyfikatora wymaga restartu procesu OSPF na routerze).

Procedura wyboru DR i BDR zostanie prześledzona na przykładzie sieci z rys. 7.2. Zakładamy, że identyfikatory routerów (Router ID) nie zostały skonfigurowane ręcznie, a wszystkie priorytety interfejsów OSPF mają wartości domyślne. Tab. 7.2 przedstawia identyfikatory, które zostały automatycznie przydzielone routerom, zgodnie z procedurą opisaną powyżej.

Tabela 7.3. Rouery DR i BDR w sieciach z rys. 7.2.

Sieć	DR	BDR
A	brak	brak
B	brak	brak
C	R4	R3
D	R1	R4

Wyniki wyborów DR i BDR przedstawia tab. 7.3. A i B są połączeniami typu punkt-punkt, w związku z czym w tych sieciach DR i BDR nie występują. Jak widać, router R4 pełni rolę DR dla sieci C i jednocześnie rolę BDR dla sieci D. Jest to sytuacja normalna, ponieważ wybory sa przeprowadzane w każdej sieci oddzielnie. Należy natomiast mieć świadomość faktu, że wyniki wyborów mogą być inne, jeżeli routery lub ich interfejsy nie zostaną uruchomione jednocześnie.

7.5. Podział systemu autonomicznego na obszary

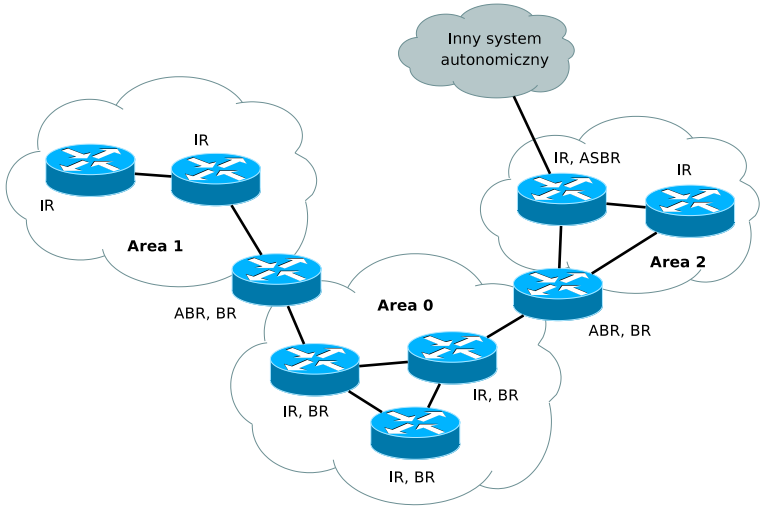
W dotychczasowych rozważaniach dotyczących protokołu OSPF zakładano, że działa on wewnątrz całego systemu autonomicznego. Wszystkie routery mają wówczas identyczne bazy danych o stanach łącz i posiadają kompletną wiedzę o dostępnych sieciach, routerach i połączeniach między nimi. W razie jakiejkolwiek zmiany w sieciach (np. uruchomienia nowego interfejsu, awarii połączenia itp.), musi nastąpić synchronizacja bazy między wszystkimi routerami i wszystkie routery muszą ponownie wyznaczyć optymalne trasy. W przypadku rozbudowanych sieci, liczba zdarzeń wymuszających resynchronizację może być na tyle duża, że praca sieci stanie się niestabilna i mało wydajna. Dlatego też protokół OSPF oferuje możliwość podziału systemu autonomicznego na tzw. obszary (ang. *area*). Routery należące do tego samego obszaru mają identyczne bazy danych i znają szczegółową topologię swojego obszaru. Nie znają natomiast topologii sieci spoza własnego obszaru (informacja o trasach do sieci w innych obszarach, którą dysponują, jest podobna jak w przypadku protokołów typu wektora odległości). Izolacja obszarów umożliwia znaczące zredukowanie ruchu sieciowego związanego z protokołem routingu. Routery brzegowe (ang. *border routers*), tzn. należące do więcej niż jednego obszaru, zapewniają komunikację między obszarami.

Istotną kwestią przy projektowaniu sieci jest określenie rozsądnej wielkości obszaru. Protokół OSPF charakteryzuje się dużą skalowalnością i zwykle wspomniane powyżej problemy nie pojawiają się w obszarach liczących mniej niż kilkaset routerów. W zaleceniach projektowych Cisco [33] nie są podawane konkretne liczby, natomiast zwraca się uwagę na fakt, że maksymalna liczba routerów w jednym obszarze zależy od wielu czynników. Są to między innymi:

— platforma sprzętowa routerów, w szczególności wydajność procesora oraz ilość pamięci,
— rodzaj medium sieciowego,
— typ sieci,
— ilość zewnętrznych informacji redystrybuowanych w obszarze,

— efektywność zastosowanej agregacji adresów sieci.

W razie implementacji wieloobszarowego OSPF, szczególną rolę pełni obszar zerowy (*Area 0*). Jest on szkieletem, odpowiedzialnym za dystrybucję informacji o routingu między pozostałymi obszarami. W przypadku konfiguracji jednoobszarowej, wykorzystuje się tylko obszar zerowy. Wszystkie routery brzegowe (tzn. routery należące do więcej niż jednego obszaru), muszą należeć również do obszaru zerowego. Obszar zerowy musi być ciągły. Jeżeli nie jest możliwe zapewnienie ciągłości fizycznej, należy skonfigurować odpowiednie połączenia wirtualne (mechanizm protokołu OSPF). Ruch między dwoma różnymi obszarami (innymi niż zerowy) jest transmitowany poprzez obszar zerowy. Trasa pakietu składa się z odcinka od źródła do routera brzegowego (wewnątrz tego samego obszaru), odcinka łączącego dwa obszary poprzez szkielet i trasy od routera brzegowego do celu w obszarze zawierającym miejsce docelowe. Algorytm wyszukuje trasę charakteryzującą się najniższym kosztem.



Rysunek 7.3. Kategorie routerów w OSPF.

W kontekście wieloobszarowej konfiguracji protokołu OSPF, wyróżnia się cztery kategorie routerów (przy czym router może równocześnie należeć do kilku kategorii), przedstawione na rys. 7.3:

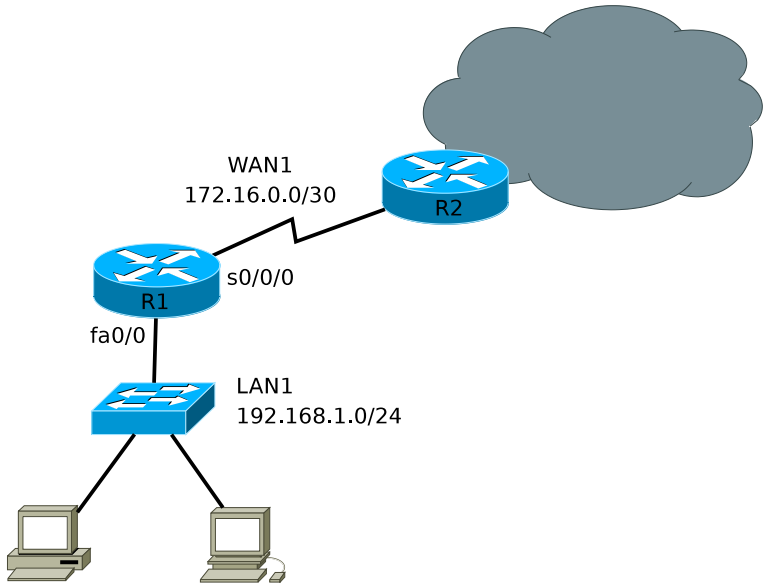
— Routery wewnętrzne (ang. *internal routers*, IR) to routery, których wszystkie interfejsy są dołączone do sieci należących do tego samego obszaru.
— Routery brzegowe (ang. *area border routers*, ABR) są dołączone do więcej niż jednego obszaru. Dla każdego obszaru działa na nich oddzielna kopia podstawowego algorytmu OSPF.
— Routery szkieletowe (*backbone routers*, BR) posiadają co najmniej je-

den interfejs w obszarze szkieletowym. Routery brzegowe są routerami szkieletowymi, natomiast routery szkieletowe nie muszą być jednocześnie brzegowymi.

— Routery brzegowe systemu autonomicznego (ang. *autonomous system boundary routers*, ASBR) wymieniają informacje dotyczące routingu z routerami należącymi do innych systemów autonomicznych (tzn. realizują redystrybucję tras), np. korzystając z protokołu BGP. Kategoria ta jest niezależna od pozostałych trzech, tzn. router tego typu może jednocześnie być wewnętrznym, brzegowym lub szkieletowym.

7.6. Konfiguracja OSPF

Podstawowa, jednoobszarowa konfiguracja protokołu OSPF na routerach Cisco jest prosta i zbliżona do konfiguracji protokołów routingu opisanych w poprzednich rozdziałach. W przypadku rozbudowanych sieci i konfiguracji wieloobszarowej, wymagane jest doświadczenie, dogłębna znajomość sieci, jak również zrozumienie zaawansowanych mechanizmów protokołu OSPF.



Rysunek 7.4. Prosta sieć, w której zostanie zaimplementowany OSPF.

Konfiguracja protokołu OSPF zostanie przedstawiona na przykładzie routera R1 w sieci z rys. 7.4. Zakładamy, że router posiada podstawową konfigurację, w szczególności zostały już skonfigurowane i włączone (poleceniem `no shutdown`) interfejsy sieciowe:

```
interface FastEthernet0/0
 ip address 192.168.1.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/0/0
 ip address 172.16.0.1 255.255.255.252
 clock rate 128000
!
```

Do uruchomienia procesu OSPF służy polecenie [33]:

```
router ospf process-id
```

Numer procesu (*process-id*) ma znaczenie lokalne (komunikujące się routery nie muszą mieć identycznych numerów procesu, chociaż jest to dobrą praktyką) i umożliwia uruchomienie kilku procesów OSPF na jednym routerze. Nie jest to jednak rekomendowane, ze względu na duże obciążenie maszyny.

Następnie należy określić interfejsy routera, które mają działać w protokole OSPF i przypisać je do obszaru, poleceniem:

```
network ip-address wildcard-mask area id
```

Realizuje się to pośrednio, podając adresy sieci do których dołączone są interfejsy, wraz z tzw. maską blankietową (ang. *wildcard mask*). W masce 0 w zapisie binarnym oznacza bity, które muszą być zgodne, natomiast 1 – bity ignorowane. Parametr *id* jest numerem obszaru, do którego ma zostać przypisany interfejs. Może on być zapisany jako liczba z przedziału od 0 do 4294967295 lub w konwencji, w jakiej są podawane adresy IPv4.

W przypadku routera R1 z rys. 7.4, konfiguracja będzie miała postać:

```
router ospf 10
 network 192.168.1.0 0.0.0.255 area 0
 network 172.16.0.0 0.0.0.3 area 0
```

W konfiguracji interfejsów szeregowych należy ponadto określić ich szerokość pasma poleceniem `bandwidth`, podobnie jak w przypadku protokołu EIGRP:

```
interface Serial0/0/0
 bandwidth 128
```

Wartość kosztu jest obliczana zgodnie z formułą 7.1. Jak łatwo sprawdzić, w przypadku interfejsu w technologii Fast Ethernet (100Mb/s) koszt jest równy 1, co jest najniższą dopuszczalną wartością. Problematyczne staje się przypisanie poszczególnym łączom wartości kosztu odzwierciedlają-

cych ich przepustowość w przypadku występowania połączeń szybszych niż 100Mb/s. Przy pomocy polecenia:

```
ip ospf cost cost
```

w trybie konfiguracji interfejsu można samodzielnie określić wartość kosztu. W przypadku wielu routerów staje się to kłopotliwe. Alternatywą jest polecenie:

```
ospf auto-cost reference-bandwidth ref-bw
```

dostępne w trybie konfiguracji protokołu OSPF (`(config-router)#`). Umożliwia ono zmodyfikowanie wartości znajdującej się w liczniku formuły 7.1. *ref-bw* staje się nową wartością, wyrażoną w Mb/s (gdzie $Mb = 10^6$ bitów). Domyślną wartością jest 100. W celu zachowania spójności konfiguracji, identyczna *ref-bw* powinna zostać ustawiona na wszystkich routerach.

W celu zapewnienia stabilnej pracy sieci, zalecane jest ręczne przypisanie routerom identyfikatorów (*Router ID*). Można do tego celu wykorzystać polecenie trybu konfiguracji protokołu routingu:

```
router-id id
```

gdzie *id* jest identyfikatorem zapisanym w takim formacie jak kares IP. Należy zapewnić unikalność identyfikatorów wszystkich routerów w systemie autonomicznym. Alternatywnym sposobem jest skonfigurowanie na routerze interfejsu typu Loopback. Jak już wspomniano wcześniej, o ile istnieją interfejsy Loopback, adres jednego z nich (najwyższy co do wartości) staje się identyfikatorem. Ponieważ Loopback jest interfejsem wirtualnym, nie zostanie w sposób niezamierzony wyłączony i nie ma zagrożenia utratą identyfikatora przez router.

W przypadku segmentów sieci, w których wybierane są routery DR i BDR, o ile administrator nie zaingeruje w procedurę wyborów, może się zdarzyć, że ten sam router będzie pełnił rolę DR w dwóch segmentach, do których jest dołączony. Sytuacji takiej należy unikać, ponieważ funkcja DR wiąże się ze znacznym, dodatkowym obciążeniem. Na wynik procedury wyborów DR i BDR można wpłynąć odpowiednio dobierając identyfikatory routerów lub ustalając priorytety (które w pierwszej kolejności decydują o wyniku wyborów). Metoda polegająca na włączaniu routerów w odpowiedniej kolejności (DR, BDR, następnie pozostałe routery) nie jest rekomendowana, ponieważ po przypadkowym restarcie urządzeń odbędą się ponowne wybory, zgodnie ze standardową procedurą. Priorytet routera ustala się w trybie konfiguracji interfejsu poleceniem:

```
ip ospf priority value
```

Domyślną wartością priorytetu (*value*) jest 1. Wartość 0 powoduje, że router nie może pełnić roli DR ani BDR w segmencie, do którego dołączony jest interfejs.

Ze względów bezpieczeństwa sieci, zalecane jest skonfigurowanie mechanizmu uwierzytelniania wymiany informacji między routerami w protokole routingu. Domyślnie jest on wyłączony. Możliwe jest proste uwierzytelnianie z użyciem hasła oraz uwierzytelnianie z algorytmem MD5. W przypadku prostego uwierzytelniania, hasło jest przesyłane jawnie i może zostać przechwycone.

Na routerze R1 z rys. 7.4 proste uwierzytelnianie może zostać skonfigurowane następująco:

```
1 interface Serial0/0/0
  ip ospf authentication-key password
3
  router ospf 10
5 area 0 authentication
```

gdzie *password* jest hasłem użytym na wszystkich routerach należących do danego obszaru. Router R2 (i pozostałe routery należące do tego samego obszaru) należy skonfigurować analogicznie.

W razie użycia uwierzytelniania z algorytmem MD5, sam klucz nie jest przesyłany między routerami, co znacząco zwiększa bezpieczeństwo. W trybie konfiguracji interfejsu stosuje się polecenie:

```
ip ospf message-digest-key keyid md5 key
```

natomiast w trybie konfiguracji protokołu routingu:

```
area 0 authentication message-digest
```

key jest kluczem, natomiast *keyid* jego identyfikatorem (liczbą). W razie zmiany klucza na nowy, należy użyć wyższej wartości *keyid*. Router będzie wówczas wysyłał kopie tych samych pakietów, uwierzytelnionych starszym i nowszym kluczem, aż do chwili, gdy stwierdzi, że nowe hasło zostało zaadaptowane przez wszystkich jego sąsiadów. Dzięki temu mechanizmowi, możliwa jest zmiana klucza w sposób niezakłócający działania protokołu OSPF.

W przypadku konfiguracji routera brzegowego systemu autonomicznego (ABSR), często pożądane jest wymuszenie na nim rozsyłania informacji o trasie domyślnej pozostałym routerom w domenie OSPF. Jeżeli router ten posiada już trasę domyślną, skonfigurowaną poleceniem

```
ip route 0.0.0.0 0.0.0.0 {ip-address | interf-type interf-number}
```

lub pochodzącą od innego protokołu routingu, polecenie

```
default-information originate
```

wydane w trybie konfiguracji protokołu routingu spowoduje rozesłanie informacji o trasie domyślnej. Przy pomocy dodatkowych parametrów powyższego polecenia, można określić także wartość i typ metryki (pierwszy lub drugi). Słowo kluczowe `always` spowoduje natomiast rozesłanie informacji o trasie domyślnej, nawet jeżeli sam router ABSR jej w danej chwili nie posiada (w związku z czym należy tu zachować ostrożność).

Do redystrybucji informacji o trasach statycznych można użyć polecenia w postaci:

```
redistribute static metric metric metric-type type subnets
```

W razie pominięcia słowa kluczowego `subnets`, redystrybuowane będą wyłącznie trasy do sieci, w przypadku których nie zastosowano podziału na podsieci.

Wartości parametrów *HelloInterval* i *RouterDeadInterval* można skonfigurować w trybie konfiguracji interfejsu (pamiętając, że w przypadku interfejsów sąsiednich routerów muszą one byc identyczne), korzystając z poleceń [33]:

```
ip ospf hello-interval seconds
ip ospf deadinterval seconds
```

Zmniejszenie ich wartości spowoduje szybszą reakcję sieci na ewentualne zmiany, jednak kosztem większego ruchu generowanego przez protokół OSPF. Zwykle zachowuje się konwencję, zgodnie z którą *RouterDeadInterval* jest cztery razy dłuższy niż *HelloInterval*.

7.7. Weryfikacja działania protokołu OSPF

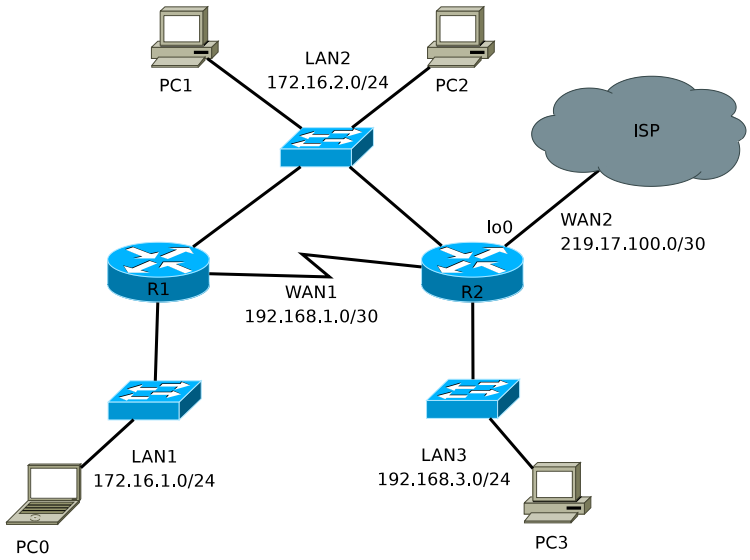
OSPF jest skomplikowanym protokołem, w związku z czym ważna jest znajomość mechanizmów służących do weryfikacji jego działania i wykrywania ewentualnych problemów. Oprócz uniwersalnych poleceń diagnostycznych informujących o funkcjonowaniu protokołów routingu, przedstawionych w podrozdziale 4.3, IOS oferuje również szereg poleceń udostępniających szczegółowe informacje o OSPF. Najważniejsze z nich omówione są poniżej [24]:

— `show ip ospf` wyświetla ogólne, podstawowe informacje o procesie routingu OSPF, uruchomionym na routerze. Opcjonalnie, jako parametr,

można podać numer interesującego procesu, co może być użyteczne w razie działania kilku jednocześnie.
— `show ip ospf database` udostępnia informacje o zawartości bazy danych o stanach łącz routera. Umożliwia zatem szczegółowe przeanalizowanie komunikatów LSA, które router zgromadził.
— `show ip ospf interface` wyświetla informacje o aspektach działania protokołu OSPF związanych z interfejsami (lub jednym interfejsem, zadanym jako parametr). Podawane są między innymi informacje o typie sieci, koszcie połączenia, priorytecie, sąsiednich routerach (w szczególności DR i BDR), parametrach czasowych (HelloInterval, RouterDeadInterval).
— `show ip ospf neighbor` wyświetla listę sąsiednich routerów wraz z podstawowymi informacjami o nich, między innymi o rodzaju relacji z danym sąsiadem.
— `show ip ospf summary-address` wyświetla informacje o trasach sumarycznych skonfigurowanych w protokole OSPF.
— W trybie konfiguracji protokołu routingu można użyć (domyślnie włączonego) polecenia `log-adjacency-changes` z opcjonalnym parametrem `detail`. Włącza ono generowanie komunikatów sysloga przy zmianach statusu sąsiadów.
— Podobne informacje można uzyskać również przy pomocy debugowania, poleceniem `debug ip ospf adjacency`. Pozwala ono prześledzić również proces uwierzytelniania między routerami.
— `debug ip ospf events` udostępnia informacje o większości zdarzeń związanych z protokołem OSPF, takich jak nawiązywanie relacji, wysyłanie informacji, wybory DR i BDR, kalkulacja tras.
— `debug ip ospf packet` wyświetla zestaw informacji o każdym odebranym przez router pakiecie OSPF.
— `debug ip ospf spf statistic` wyświetla informacje o charakterze statystycznym, dotyczące działania algorytmu znajdowania najkrótszych tras.

7.8. Zadanie

1. Zbuduj sieć zgodnie ze schematem (rys. 7.5). Przeprowadź podstawową konfigurację routerów (nazwy, hasła, adresy IP interfejsów) i komputerów PC.
2. Na routerze R2 skonfiguruj trasę domyślną poprzez interfejs Loopback0.
3. Skonfiguruj informację o przepustowości (polecenie `bandwidth`) na interfejsach szeregowych.
4. Skonfiguruj proces OSPF dla wszystkich sieci oprócz WAN2.



Rysunek 7.5. Schemat topologii logicznej sieci.

5. Jakie są identyfikatory *router ID* routerów R1 i R2? Które routery pełnią rolę DR (routera desygnowanego) i BDR (zapasowego routera desygnowanego) w sieci LAN2? Użyj poleceń `show ip ospf neighbor`, `show ip ospf interface`.
6. Spowoduj, by funkcję DR dla sieci LAN2 pełnił router R1. Użyj w tym celu jednej z metod zapewniających identyczny wybór DR również po restarcie urządzeń. Zaobserwuj proces wyboru (`debug ip ospf adj`). Konieczne będzie użycie polecenia `clear ip ospf process` lub restart urządzeń.
7. Przejrzyj tablice routingu, sprawdź połączenia między dowolnymi miejscami w sieci.
8. Skonfiguruj redystrybucję trasy domyślnej na routerze R2 (użyj polecenia `default-information originate`). Sprawdź, czy pojawiła się w tablicy routingu R1.
9. Która trasa jest wykorzystywana do przesyłania pakietów z sieci LAN1 do LAN3? Zmodyfikuj koszty połączeń na odpowiednich interfejsach (`ip ospf cost`) tak, aby w tablicy routingu R1 znalazły się obie trasy. Zaobserwuj rozkładanie obciążenia na trasy równorzędne (użyj polecenia `debug ip packet`). Czy obciążenie jest rozkładane dla poszczególnych pakietów (*per packet*) czy dla poszczególnych celów (*per destination*)? Czy można to zmienić?
10. Skonfiguruj uwierzytelnianie MD5 w protokole OSPF.
11. Zweryfikuj działanie protokołu OSPF używając poleceń:

— `show ip protocols`,
— `show ip ospf`,
— `show ip ospf interface` (Jaka jest wartość parametrów *HelloInterval* i *RouterDeadInterval*? Który router pełni rolę DR/BDR?),
— `show ip ospf neighbor` (Jaki jest stan poszczególnych interfejsów?),
— `debug ip ospf adj`, `debug ip ospf events` (Komunikacja między sąsiadami, co się stanie w przypadku różnych wartości *HelloInterval* i *RouterDeadInterval*?)
— `show ip ospf database` (Jakie informacje o sieciach posiada OSPF?).

12. Wyłącz debugowanie, o ile nadal jest uruchomione. Zachowaj kopie konfiguracji routerów R1 i R2.

7.9. Rozwiązanie

Listing 7.1. Istotne fragmenty konfiguracji routera R1

```
hostname R1
!
interface FastEthernet0/0
 ip address 172.16.1.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/1
 ip address 172.16.2.1 255.255.255.0
 ip ospf message-digest-key 1 md5 H@sl01
 ip ospf cost 1562
 ip ospf priority 200
 duplex auto
 speed auto
!
interface Serial0/0/0
 bandwidth 64
 ip address 192.168.1.1 255.255.255.252
 ip ospf message-digest-key 1 md5 H@sl0
 clock rate 64000
!
router ospf 1
 log-adjacency-changes
 area 0 authentication message-digest
 network 172.16.1.0 0.0.0.255 area 0
 network 172.16.2.0 0.0.0.255 area 0
 network 192.168.1.0 0.0.0.3 area 0
!
ip classless
!
end
```

Listing 7.2. Istotne fragmenty konfiguracji routera R2

```
hostname R2
!
interface Loopback0
 ip address 219.17.100.1 255.255.255.252
!
interface FastEthernet0/0
 ip address 192.168.3.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/1
 ip address 172.16.2.2 255.255.255.0
 ip ospf message-digest-key 1 md5 H@sl01
 duplex auto
 speed auto
!
interface Serial0/0/0
 bandwidth 64000
 ip address 192.168.1.2 255.255.255.252
 ip ospf message-digest-key 1 md5 H@sl0
!
router ospf 1
 log-adjacency-changes
 area 0 authentication message-digest
 network 172.16.2.0 0.0.0.255 area 0
 network 192.168.3.0 0.0.0.255 area 0
 network 192.168.1.0 0.0.0.3 area 0
 default-information originate
!
ip classless
ip route 0.0.0.0 0.0.0.0 Loopback0
!
end
```

Rozdział 8

IS-IS

8.1. Wstęp

IS-IS (ang. *Intermediate System to Intermediate System*) jest otwartym protokołem routingu wewnętrznego, typu stanu łącza. Podobnie jak OSPF, działa w oparciu o algorytm Dijkstry. Jest opisany w RFC 1142 [12]. Protokół został opracowany w firmie Digital Equipment Corporation, a następnie stał się standardem ISO (DP 10589). Początkowo był on protokołem routingu dla CLNS (ang. *Connectionless Network Service*), który jest konkurencyjnym dla IP rozwiązaniem w 3 warstwie modelu OSI. Dopiero w późniejszym okresie został rozbudowany o wsparcie dla protokołu IP. IS-IS jest protokołem starszym niż dominujący współcześnie, zwłaszcza w dużych sieciach korporacyjnych, OSPF. Nadal jednak, ze względu na stabilność i skalowalność, IS-IS jest wykorzystywany przez niektórych dużych operatorów internetowych.

8.2. CLNS

CLNS (*Connectionless Network Service*) jest usługą warstwy sieci modelu OSI, niewymagającą nawiązywania połączenia do transmisji danych. CLNP (*Connection-Less Network Protocol*) jest z kolei protokołem implementującym tę usługę. Rozróżnienia takiego nie stosuje się w przypadku protokołu IP, dla którego CLNS wraz z CLNP stanowi alternatywne rozwiązanie. Zostało ono, podobnie jak model OSI, opracowane przez ISO (*International Organization for Standardization*[1])

We wczesnych latach Internetu, CLNS (wraz z protokołem routingu IS-IS) był powszechnie wykorzystywany, zwłaszcza w dużych sieciach. Obecnie jest rozwiązaniem niszowym. Jednak, ponieważ stosowane są w nim długie adresy (maksymalnie 20-bajtowe) o zmiennej długości, jego użycie było jednym z proponowanych rozwiązań problemu niewystarczającej liczby adresów w IPv4 (jeszcze przed opracowaniem IPv6 przez IETF) [13]. Początkowo nie było natomiast planowane wykorzystanie IS-IS w charakterze protokołu routingu w środowisku IP. Stało się to w momencie, gdy RIP był już zbyt ograniczony, a OSPF jeszcze niewystarczająco dopracowany, by mógł być wdrożony w sieciach największych operatorów. IS-IS był już wówczas dojrzałym protokołem, implementowanym między innymi w routerach Cisco [32].

W terminologii stosowanej do opisu architektury OSI, hosty określane są mianem systemów końcowych (ang. *end systems*), natomiast routery – systemów pośrednich (ang. *intermediate systems*). Jest to wyjaśnienie pochodzenia nazwy protokołu IS-IS. Komunikacja między hostami a route-

[1] `http://www.iso.org/`

rami odbywa się zgodnie z protokołem *End System-to-Intermediate System* (ES-IS). Adresy CLNS określane są mianem punktów dostępu usługi sieciowej (ang. *network service access point*, NSAP). Są one przypisywane węzłom sieci (tzn. ruterom lub hostom), a nie poszczególnym interfejsom, jak w przypadku IP. NSAP jest odpowiednikiem adresu IP wraz z numerem portu protokołu warstwy transportowej, w związku z czym jedno urządzenie zwykle posiada wiele takich adresów. Adres NSAP, w którym pole NSEL (*NSAP selector*) ma wartość zerową, jest identyfikatorem samego urządzenia (*network entity title*, NET).

CLNS nie jest już raczej używany do transmisji danych. Jednak w przypadku konfigurowania IS-IS, adresy CLNS, a dokładniej adresy NET, są konieczne do identyfikacji routerów, podobnie jak adresy IP wykorzystywane jako identyfikatory routerów przez OSPF. CLNS nie jest natomiast wykorzystywany do przesyłania pakietów IS-IS.

8.3. IS-IS a OSPF

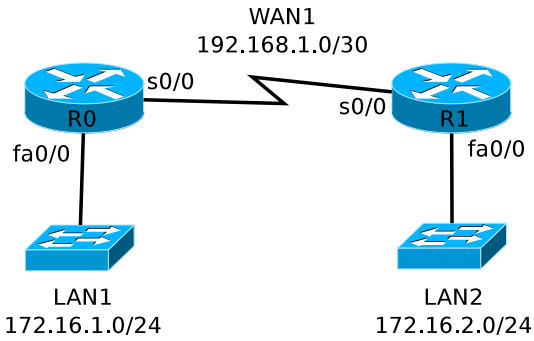
IS-IS wzbogacony o możliwości routingu IP jest określany mianem zintegrowanego IS-IS (*Integrated IS-IS*) [2]. Sposób funkcjonowania, możliwości i pojęcia stosowane do opisu protokołów IS-IS i OSPF są bardzo zbliżone.

Występują różnice w sposobie definiowania obszarów i realizowania routingu między nimi. IS-IS wyróżnia routery poziomu 1 (wewnątrz obszaru), poziomu 2 (między obszarami) i poziomu 1-2 (wewnątrz i między obszarami). Routery poziomu 1 i 2 wymieniają informacje o routingu wyłącznie z innymi routerami tego samego poziomu. Routery poziomu 1-2 wymieniają informacje z routerami obu poziomów. Stanowią one połączenie między routerami wewnątrz obszaru a routerami międzyobszarowymi. Granice między obszarami przebiegają pomiędzy routerami. Inaczej niż w OSPF, nie ma routerów brzegowych, należących do więcej niż jednego obszaru. Zamiast obszaru zerowego, szkielet stanowią połączone routery poziomu 2. Do nich dołączane są routery poziomu 1-2, a następnie routery poziomu 1 w poszczególnych obszarach.

Różnicą natury podstawowej jest możliwość funkcjonowania OSPF wyłącznie w środowisku IP. IS-IS jest natomiast w pełni samodzielnym protokołem trzeciej warstwy modelu OSI (podobnie jak CLNS i IP). IS-IS może przesyłać informacje o routingu z wykorzystaniem dowolnych adresów sieciowych. Dzięki temu dostosowanie go do IPv6 nie było problemem. W przypadku OSPF, konieczne były głębsze modyfikacje, w wyniku których opracowano nową wersję – OSPFv3. OSPF posiada większą liczbę różnego rodzaju funkcji dodatkowych i rozszerzeń. Z kolei IS-IS charakteryzuje się lepszą skalowalnością i jest w stanie obsługiwać wewnątrz jednego obszaru

więcej routerów. W związku z tym bywa preferowany przez dużych operatorów. Problem stanowi jednak obecnie niewielka liczba inżynierów sieciowych posiadających dogłębną wiedzę i doświadczenie w jego konfigurowaniu.

8.4. Podstawowa konfiguracja IS-IS



Rysunek 8.1. Sieć, w której zostanie zaimplementowany Integrated IS-IS.

Każdy router musi posiadać unikalny adres NET. W przypadku urządzeń Cisco, składa się on z numeru obszaru (identycznego dla wszystkich routerów w danym obszarze) o zmiennej długości, 6-bajtowego identyfikatora systemu i 1-bajtowego pola NSEL o wartości zerowej [28, 27]. Przykład konfiguracji zostanie przedstawiony dla sieci z rys. 8.1. Podstawowa konfiguracja IS-IS dla routera R0 może mieć postać:

```
router isis
 net 48.0001.1111.1111.1111.00
!
interface FastEthernet0/0
 ip router isis
!
interface Serial0/0
 ip router isis
```

i analogicznie dla R1:

```
router isis
 net 48.0001.2222.2222.2222.00
!
interface FastEthernet0/0
 ip router isis
!
interface Serial0/0
 ip router isis
```

Polecenie `net` w trybie konfiguracji protokołu routingu służy do nadania identyfikatora. W tym przypadku numer obszaru to 48.0001, natomiast identyfikatory systemów to 1111.1111.1111 i 2222.2222.2222. Częstą praktyką jest stosowanie adresów MAC w charakterze identyfikatorów systemów, w celu zapewnienia ich unikalności. Polecenie `ip router isis`, wydawane w trybie konfiguracji interfejsu, określa interfejsy biorące udział w procesie routingu.

O ile konfiguracja została wykonana poprawnie, w tablicach routingu powinny pojawić się nowe wpisy z literą "i":

```
R0#show ip route
Codes: C - connected, [...]
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1
       L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.1.0 is directly connected, FastEthernet0/0
i L1    172.16.2.0 [115/20] via 192.168.1.2, Serial0/0
     192.168.1.0/30 is subnetted, 1 subnets
C       192.168.1.0 is directly connected, Serial0/0
```

8.5. Zadanie

1. Wykorzystaj sieć zbudowaną w zadaniu z Rozdziału 7 (rys. 7.5). Zastąp OSPF protokołem IS-IS. Dokonanie pełnej migracji może wymagać skorzystania z dokumentacji Cisco [27].
2. Znajdź i przetestuj polecenia `show` służące do weryfikacji działania protokołu IS-IS.

Rozdział 9

BGP

Najnowsza, obowiązująca wersja protokołu BGP (protokół bramy brzegowej, ang. *Border Gateway Protocol*) jest standardem opisanym w dokumencie RFC 4271 [15] (który zastąpił starsze wersje RFC 1771 i RFC 1654). We współczesnym Internecie BGP-4 jest jedynym, powszechnie przyjętym protokołem routingu zewnętrznego. Jego zadaniem jest wymiana informacji o dostępności tras do poszczególnych sieci, między systemami autonomicznymi. Informacje te zawierają listę systemów autonomicznych, poprzez które odbywa się transmisja, która z kolei wystarcza do skonstruowania grafu połączeń między systemami autonomicznymi. Służy on następnie między innymi do wykrycia i wyeliminowania pętli routingu. BGP-4 wspiera technologię CIDR (Classless Inter-Domain Routing). Rozsyłane są informacje o masce, eliminując pojęcie klasy adresu i umożliwiając agregację tras.

Do działania protokołu BGP niezbędne są unikalne numery systemów autonomicznych. W przypadku sieci nie mających połączenia z publicznym Internetem, można użyć prywatnych numerów z zakresu 64512–65535 [5]. W przeciwnym razie, należy uzyskać własny numer przydzielany przez organizację IANA (*Internet Assigned Numbers Authority*). Z powodu wyczerpywania się 16-bitowych numerów, w 2007 roku wprowadzono numery 32-bitowe i zaproponowano odpowiednie modyfikacje BGP [18, 16].

Jeżeli protokół BGP działa między routerami należącymi do różnych systemów autonomicznych, mówi się o EBGP (ang. *External BGP*, zewnętrzny BGP). Zwykle routery takie są ze sobą bezpośrednio połączone. Wewnątrz jednego systemu autonomicznego może pracować wiele, niekoniecznie bezpośrednio połączonych ze sobą, routerów BGP. Wówczas, w celu zapewnienia spójnego obrazu świata zewnętrznego, wszystkie powinny komunikować się ze sobą zgodnie z IBGP (ang. *Internal BGP*, wewnętrzny BGP). Wszystkie routery BGP, należące do tego samego systemu autonomicznego, posiadają wówczas identyczne informacje o zewnętrznych systemach autonomicznych. Zapewnienie spójności routingu wewnątrz systemu autonomicznego jest szczególnie istotne, gdy świadczy on usługi tranzytowe, tzn. są poprzez niego przesyłane pakiety transmitowane pomiędzy innymi systemami autonomicznymi.

Z punktu widzenia BGP, trasa jest zdefiniowana jako jednostka informacji, w której miejscu docelowemu przypisany jest pewien zbiór atrybutów, jakimi charakteryzuje się ścieżka do celu. Informacje te przechowywane są w bazie informacji o routingu (ang. *Routing Information Base*, RIB). Składa się ona z trzech części (co jednak nie oznacza, że implementacja BGP musi zawierać trzy oddzielne tablice – protokół nie precyzuje szczegółów implementacji):

— Adj-RIBs-In przechowuje informacje o trasach uzyskane od innych routerów BGP.
— Loc-RIB zawiera informacje o trasach wybranych spośród tras przecho-

wywanych w Adj-RIBs-In, zgodnie z określonymi założeniami, z których router będzie korzystał. W przypadku każdej z nich, kolejny krok musi być osiągalny, dzięki informacjom z tablicy routingu.
— Adj-RIBs-Out zawiera informacje przeznaczone do rozgłaszania w protokole BGP, przy pomocy komunikatów typu Update.

Router może rozgłaszać wyłącznie informacje o trasach, z których sam korzysta. BGP zapewnia również mechanizm, przy pomocy którego router może informować inne routery, że trasa o której informacje uprzednio rozgłaszał, nie jest już dostępna. Aktualizacje rozsyłane są w razie wystąpienia zmiany w tablicy routingu, nie stosuje się natomiast periodycznego odświeżania.

Komunikacja między routerami w protokole BGP odbywa się w sposób połączeniowy, z wykorzystaniem TCP, co wyeliminowało potrzebę implementacji mechanizmów zapewniających niezawodność transmisji w samym BGP. Routery BGP prowadzą nasłuch na porcie TCP 179. Każdy komunikat BGP posiada nagłówek złożony z 3 pól:
— Znacznik (ang. *Marker*, 16 bajtów) – obecnie niewykorzystywany, występuje dla zachowania zgodności ze starszymi wersjami. Pole wypełnione jest jedynkami.
— Długość (ang. *Length*, 2 bajty) – całkowita długość komunikatu, wraz z nagłówkiem, wyrażona w bajtach. Minimalna długość komunikatu to 19, natomiast maksymalna 4096.
— Typ (ang. *Type*) informuje o rodzaju wiadomości:
    1 – OPEN (otwarcie)
    2 – UPDATE (aktualizacja)
    3 – NOTIFICATION (zawiadomienie)
    4 – KEEPALIVE (sprawdzanie połączenia)

Komunikat może składać się z samego nagłówka lub nagłówka i danych, zapisanych w formacie odpowiednim dla danego typu wiadomości.

OPEN jest pierwszym komunikatem wysyłanym po nawiązaniu połączenia TCP. Odpowiedzią na niego jest KEEPALIVE. Komunikat OPEN zawiera podstawowe informacje o nadawcy i jego systemie autonomicznym. UPDATE służy do przekazywania informacji o routingu między systemami autonomicznymi. Wiadomości KEEPALIVE umożliwiają cykliczne sprawdzanie dostępności połączenia z sąsiednimi routerami. NOTIFICATION informuje o wystąpieniu błędu i skutkuje przerwaniem połączenia BGP.

Każda trasa w protokole BGP charakteryzuje się zbiorem atrybutów, na podstawie których wybierana jest możliwie najlepsza spośród alternatywnych dróg do danego miejsca docelowego. Są one odpowiednikiem metryki w protokołach routingu wewnętrznego. Jednak problem wyboru optymalnej trasy pomiędzy różnymi systemami autonomicznymi jest zagadnieniem znacznie bardziej złożonym niż routing wewnątrz systemu. W protokołach

routingu wewnętrznego, metryka trasy wyznaczana jest na podstawie co najwyżej kilku parametrów (zwykle jednego lub dwóch). W BGP natomiast stosuje się wiele złożonych atrybutów. Część z nich jest klasyfikowana jako dobrze znane atrybuty. Wśród nich z kolei znajduje się grupa atrybutów obowiązkowych, które muszą być obecne we wszystkich aktualizacjach. Dobrze znane atrybuty muszą być obsługiwane przez wszystkie implementacje BGP, natomiast pozostałe są opcjonalne. Przed rozesłaniem informacji o trasie, którą router uprzednio otrzymał, jej atrybuty mogą zostać dodane lub zmodyfikowane. Przykładowe, obowiązkowe atrybuty to:

— ORIGIN (źródło) – atrybut jest generowany przez router, który jest źródłem informacji o routingu. Atrybut ten nie powinien być następnie modyfikowany przez kolejne routery BGP.
— AS_PATH zawiera listę systemów autonomicznych, poprzez które była przesyłana bieżąca informacja. Przekazując informację dalej (do innego systemu autonomicznego), router dodaje do listy numer swojego systemu autonomicznego. Umożliwia to wykrycie pętli routingu poprzez znalezienie na otrzymanej liście numeru własnego systemu autonomicznego.
— NEXT_HOP (następny skok) – adres routera stanowiącego następny skok dla danego celu.

Sposób wyboru najlepszych tras nie jest w sposób ścisły zdefiniowany w protokole BGP. Są natomiast określone pewne ogólne zalecenia. Powinna zostać zdefiniowana funkcja, której argumentami są atrybuty trasy, a zwracana nieujemna liczba całkowita jest stopniem preferencji danej trasy lub jest pewną wartością specjalną, informującą o wykluczeniu danej trasy. Wartość zwracana przez funkcję dla danej trasy nie powinna zależeć od istnienia innych tras ani ich atrybutów.

W procesie decyzyjnym, spośród tras dostępnych w tablicy Adj-RIBs-In wybierane są najlepsze i umieszczane w tablicy Loc-RIB. Spośród nich z kolei wybierane są trasy przeznaczone do rozgłaszania innym routerom BGP (Adj-RIBs-Out).

Oprócz komercyjnych implementacji protokołu BGP (np. Cisco) istnieje również szereg rozwiązań alternatywnych, np. BIRD[1], OpenBGPD[2], Quagga[3], XORP[4]. Niezależnie od implementacji, poza najprostszymi przypadkami, konfiguracja BGP nie jest zadaniem banalnym. Przykładowo, sieć pewnej firmy może posiadać dwa połączenia z Internetem, wykupione od dwóch różnych dostawców. Implementacja BGP umożliwi dzielenie ruchu wychodzącego z firmy do Internetu między dwa łącza. Jednak, przy nieod-

[1] `http://bird.network.cz/`
[2] `http://www.openbgpd.org/`
[3] `http://www.quagga.net/`
[4] `http://www.xorp.org/`

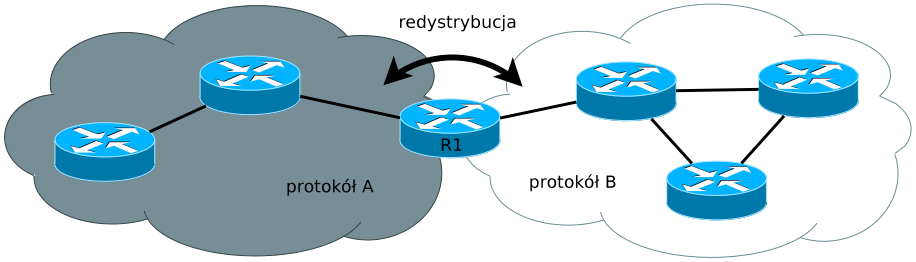
powiedniej konfiguracji, istnieje niebezpieczeństwo udostępnienia łącz firmy na potrzeby ruchu tranzytowego pomiędzy dwoma operatorami.

Rozdział 10

Podstawy redystrybucji między protokołami routingu

10.1. Wstęp

Rekomendowanym rozwiązaniem jest zastosowanie spójnej strategii routingu (tzn. implementacja jednego protokołu routingu) w całym obszarze systemu autonomicznego. Problemy pojawiają się np. w przypadku fuzji dwóch przedsiębiorstw – konieczne jest wówczas połączenie sieci, w których zastosowano różne protokoły routingu. Zmiana protokołu routingu jest zwykle zadaniem trudnym, czasochłonnym i kosztownym. Może np. wymagać wymiany części sprzętu, który nie obsługuje docelowego protokołu routingu. Przykładowo, wskazówki odnośnie migracji z protokołu EIGRP do OSPF można znaleźć w [29]. Niejednokrotnie, przynajmniej w okresie przejściowym, nieuniknione jest współistnienie różnych protokołów routingu wewnątrz jednego systemu autonomicznego. Należy wówczas zapewnić wymianę informacji między nimi. Rozgłaszanie przez protokół routingu informacji o trasach dostarczonych przez inny protokół routingu (lub routing statyczny) nazywamy redystrybucją. Zadanie nie jest banalne, ponieważ różne protokoły routingu charakteryzują się różnymi sposobami obliczania metryk, różnymi wartościami odległości administracyjnej i działaniem klasowym lub bezklasowym.



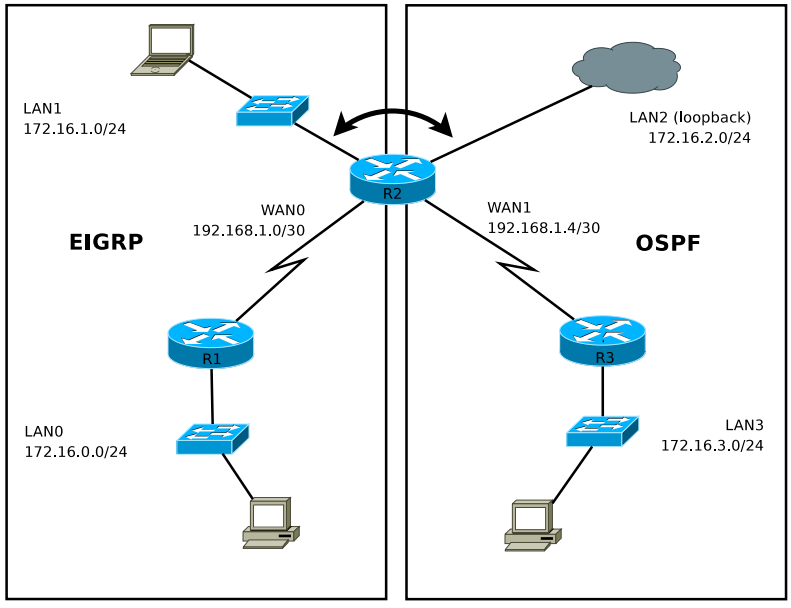
Rysunek 10.1. Redystrybucja między protokołami routingu.

W przykładzie przedstawionym na rys. 10.1, w dwóch obszarach zostały skonfigurowane protokoły routingu A i B. Router R1 jest jedynym urządzeniem, na którym uruchomiono oba protokoły routingu. Dysponuje zatem informacjami o trasach z obu obszarów. W razie gdyby router R1 otrzymał informacje o trasie do pewnej sieci od obu protokołów (co akurat w przypadku sieci pokazanej na rysunku nie wystąpi), wówczas w jego tablicy routingu znalazłaby się trasa pochodząca od protokołu charakteryzującego się niższą wartością odległości administracyjnej. Zadaniem routera R1 będzie przekazywanie informacji o trasach z obszaru A do obszaru B i odwrotnie. W przypadku urządzeń Cisco, redystrybucja może dotyczyć wyłącznie tras, które znajdują się w tablicy routingu. Ogólny schemat konfiguracji redystrybucji (na routerze R1) wygląda następująco [36]:

```
router A
 redistribute B [parametry]

router B
 redistribute A [parametry]
```

Ze względu na różne sposoby wyliczania metryki, podczas redystrybucji konieczne jest zdefiniowane metryki w sposób zrozumiały dla protokołu, któremu przekazywane są informacje. Wykorzystuje się do tego celu odpowiednie zestawy parametrów (jak powyżej) lub polecenie `default-metric`. Dla poszczególnych par protokołów routingu mogą być potrzebne pewne specyficzne ustawienia konfiguracyjne. Przykłady konfiguracji można znaleźć w [36]. Szczególną uwagę należy zachować w przypadku, gdy na styku obszarów A i B znajduje się więcej niż jeden router realizujący redystrybucję. Występuje wówczas zwiększone ryzyko wystąpienia pętli routingu. Przy nieodpowiednim doborze metryk redystrybuowanych tras może zdarzyć się również, że np. router z obszaru A będzie przesyłał pakiety przeznaczone dla sieci z obszaru A poprzez obszar B.

10.2. Zadanie



Rysunek 10.2. Schemat topologii logicznej sieci.

1. Zbuduj sieć zgodnie ze schematem (rys. 10.1). Interfejsom sieciowym routerów przydziel najniższe możliwe adresy, natomiast komputerom PC – dowolne. Sieć LAN2 jest symulowana przy pomocy interfejsu Loopback.
2. Uruchom protokół EIGRP na routerach R1 i R2 oraz OSPF na R2 i R3. Konfigurując protokół EIGRP uwzględnij wyłącznie interfejsy pracujące w sieciach LAN0, LAN1 i WAN0. Konfigurując OSPF uwzględnij wyłącznie interfejsy w sieciach LAN2, LAN3 i WAN1. W tablicy routingu R2 powinny pojawić się wszystkie sieci, w tablicy routingu R1 – wyłącznie sieci z obszaru EIGRP, natomiast w tablicy routingu R3 – wyłącznie sieci z obszaru OSPF.
3. Przetestuj połączenie między LAN0 i LAN1 oraz między LAN2 i LAN3 (`ping`).
4. Na routerze R2 skonfiguruj redystrybucję tras między protokołami OSPF i EIGRP (w obu kierunkach). W razie problemów, znajdź odpowiedni przykład w dokumentacji Cisco [36].
5. Zweryfikuj zawartość wszystkich tablic routingu. Czy informacje dodane dzięki redystrybucji można odróżnić od pozostałych? Przetestuj połączenie między LAN0 i LAN3.
6. Wykonaj kopie konfiguracji routerów.

10.3. Rozwiązanie

Listing 10.1. Istotne fragmenty konfiguracji routera R1

```
interface FastEthernet0/0
 ip address 172.16.0.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/3/0
 ip address 192.168.1.1 255.255.255.252
 clock rate 56000
!
router eigrp 1
 network 172.16.0.0
 network 192.168.1.0
 no auto-summary
!
ip classless
```

Listing 10.2. Istotne fragmenty konfiguracji routera R2

```
interface Loopback0
 ip address 172.16.2.1 255.255.255.0
```

```
!
interface FastEthernet0/0
 ip address 172.16.1.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/3/0
 ip address 192.168.1.2 255.255.255.252
!
interface Serial0/3/1
 ip address 192.168.1.5 255.255.255.252
 clock rate 56000
!
router eigrp 1
 redistribute ospf 1 metric 10000 100 255 1 1500
 network 192.168.1.0 0.0.0.3
 network 172.16.1.0 0.0.0.255
 no auto-summary
!
router ospf 1
 log-adjacency-changes
 redistribute eigrp 1 subnets
 network 172.16.2.0 0.0.0.255 area 0
 network 192.168.1.4 0.0.0.3 area 0
!
ip classless
```

Listing 10.3. Istotne fragmenty konfiguracji routera R3

```
interface FastEthernet0/0
 ip address 172.16.3.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/3/0
 ip address 192.168.1.6 255.255.255.252
!
router ospf 1
 log-adjacency-changes
 network 172.16.3.0 0.0.0.255 area 0
 network 192.168.1.4 0.0.0.3 area 0
!
ip classless
```

Rozdział 11

Budowa routera linuksowego

11.1. Wstęp

Budując sieć na bazie dedykowanych, sprzętowych routerów dowolnego spośród wiodących producentów sprzętu sieciowego, uzyskuje się stabilnie działającą i dobrze udokumentowaną platformę, jak również gwarantowany dostęp do aktualizacji i wsparcia technicznego. Są to jednak rozwiązania kosztowne, adresowane głównie do klientów korporacyjnych. W przypadku zastosowań z sektora SOHO (ang. *Small Office/Home Office*, małe biuro, biuro przydomowe), jedną z alternatyw jest wykorzystanie rozwiązań bazujących na darmowym oprogramowaniu i tanim sprzęcie.

Rolę routera może pełnić komputer PC, z zainstalowanym jednym z uniksowych systemów operacyjnych (w szczególności Linux, FreeBSD, Solaris, NetBSD). Przy założeniu odpowiedniej konfiguracji jądra systemu, do zrealizowania prostego routingu wystarczają polecenia `ifconfig` oraz `route` (wydawane z uprawnieniami administratora). Bardziej zaawansowany routing wymaga instalacji dodatkowego oprogramowania. Jednym z popularnych pakietów jest Quagga[1]. Jest to darmowa (na licencji GNU GPL) implementacja protokołów routingu RIP, OSPF, BGP i IS-IS (w niepełnym zakresie), dla IPv4, jak również IPv6. Quagga[2] wywodzi się z nierozwijanego od kilku lat projektu GNU Zebra[3].

System Quagga składa się z kilku współdziałających demonów. Demon `zebra` jest odpowiedzialny za zarządzanie tablicą routingu i redystrybucję tras między różnymi protokołami routingu. W razie potrzeby, konfiguruje się i uruchamia kolejne demony, odpowiedzialne za poszczególne protokoły routingu, np.: `ripd` (RIP), `ospfd` (OSPF), `bgpd` (BGP). Dodanie nowego protokołu routingu (poprzez dodanie kolejnego demona) nie zaburza zatem struktury całego systemu. Poszczególne demony mogą działać na tej samej lub na różnych maszynach. Co więcej, na jednej maszynie może jednocześnie działać kilka demonów tego samego protokołu. Każdy demon posiada swój własny plik konfiguracyjny, jak również możliwość nawiązania z nim połączenia terminalowego (poprzez telnet). Zatem, aby skonfigurować np. adresy interfejsów i trasy statyczne, należy połączyć się z demonem `zebra`. Aby natomiast skonfigurować protokół RIP, należy połączyć się z demonem RIP. Może być to uciążliwe, w związku z tym dostępna jest także powłoka `vtysh`, oferująca zintegrowany dostęp (proxy) do wszystkich demonów. Sposób konfigurowania jest bardzo zbliżony do IOS Cisco. Podobna jest filozofia pracy z wierszem poleceń, a znaczna część poleceń identyczna.

[1] `http://www.quagga.net/`
[2] łac. *Equus quagga* - zebra właściwa
[3] `http://www.zebra.org/`

11.2. Konfiguracja protokołu RIP w Quagga

W dalszej części zakłada się wykorzystywanie dystrybucji Linuksa Ubuntu Server 10.04 (zainstalowanej na komputerze z dwoma interfejsami sieciowymi, eth0 i eth1) i pakietu Quagga 0.99.15. W przypadku innych dystrybucji lub wersji, mogą wystąpić różnice w porównaniu z przedstawionymi przykładami. Pakiet Quagga, wraz z dokumentacją, można zainstalować poleceniem:

```
sudo apt-get install quagga quagga-doc
```

W pliku `/etc/services` powinny pojawić się informacje o numerach portów wykorzystywanych przez poszczególne demony pakietu Quagga:

```
zebrasrv        2600/tcp        # zebra service
zebra           2601/tcp        # zebra vty
ripd            2602/tcp        # ripd vty (zebra)
ripngd          2603/tcp        # ripngd vty (zebra)
ospfd           2604/tcp        # ospfd vty (zebra)
bgpd            2605/tcp        # bgpd vty (zebra)
ospf6d          2606/tcp        # ospf6d vty (zebra)
isisd           2608/tcp        # ISISd vty (zebra)
```

Aby skonfigurować routing z wykorzystaniem protokołu RIP, konieczne będzie uruchomienie demona `zebra` oraz `ripd`. Do ich działania niezbędne są pliki konfiguracyjne `zebra.conf` oraz `ripd.conf`, zapisane w katalogu `/etc/quagga`. Początkowa zawartość `zebra.conf` może mieć postać [8]:

```
hostname Router
password zebra
enable password zebra
log stdout
```

natomiast `ripd.conf`:

```
hostname ripd
password zebra
log stdout
```

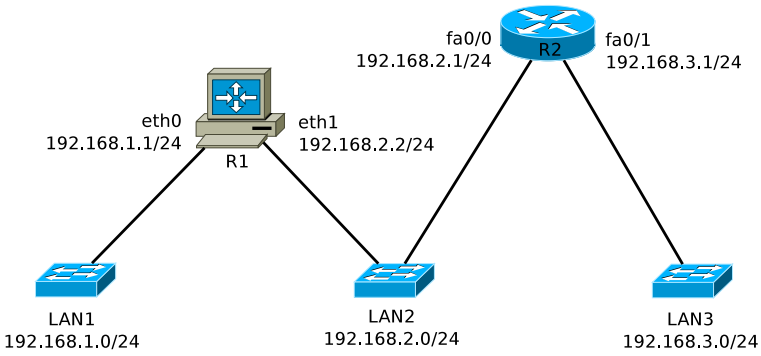
Polecenie `password` służy do ustalenia hasła wymaganego przy połączeniu terminalowym z demonem. W razie jego braku, nawiązanie połączenia nie będzie możliwe. `log stdout` włącza wyświetlanie komunikatów na standardowym wyjściu. Polecenia `hostname` i `enable password` są odpowiednikami poleceń IOS Cisco. Więcej przykładowych plików konfiguracyjnych można znaleźć w katalogu `/usr/share/doc/quagga/examples`.

W tym momencie powinno być możliwe uruchomienie skonfigurowanych demonów:

```
/etc/init.d/quagga start
```

O sukcesie informuje komunikat:

```
Starting Quagga daemons (prio:10): zebra ripd.
```



Rysunek 11.1. Przykładowa sieć z routerem linuksowym.

Dalsza konfiguracja będzie realizacją scenariusza przedstawionego na rys. 11.1. Zakładamy, że router R2 (np. Cisco 2600) został uprzednio poprawnie skonfigurowany i uruchomiono na nim protokół RIPv2 dla obu dołączonych sieci LAN. Celem jest skonfigurowanie routera linuksowego R1 w celu skomunikowania wszystkich trzech sieci. Aby skonfigurować podstawowe ustawienia sieciowe, należy nawiązać połączenie z demonem `zebra` (podając uprzednio skonfigurowane hasło "zebra"):

```
telnet localhost 2601
```

Następnie należy wydać polecenia:

```
enable
configure terminal
ip forwarding
interface eth0
 ip address 192.168.1.1/24
 no shutdown
interface eth1
 ip address 192.168.2.2/24
 no shutdown
 exit
copy running-config startup-config
```

```
exit
```

Konfiguracja powinna zostać automatycznie zapisana w pliku `zebra.conf`. Odpowiednikiem polecenia `ip forwarding`, włączającego przekazywanie pakietów IP między interfejsami routera, byłoby:

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

wydane z poziomu systemu operacyjnego. Z kolei konfiguracja interfejsów ethernetowych mogłaby być przeprowadzona przy pomocy linuksowego polecenia `ifconfig`.

Konfiguracja protokołu RIP wymaga połączenia z demonem `ripd`:

```
telnet localhost 2602
```

a następnie wydania poleceń:

```
enable
configure terminal
router rip
 version 2
 network 192.168.1.0/24
 network eth1
exit
copy running-config startup-config
```

Konfiguracja zostanie zapisana w pliku `ripd.conf`. Wiersze nr 5 i 6 na powyższym listingu prezentują dwa alternatywne sposoby określenia bezpośrednio dołączonej sieci dla protokołu RIP. Należy tu zwrócić uwagę na pewne różnice między konfiguracją Quagga a IOS Cisco.

Wygodnym sposobem konfigurowania jest uruchomienie powłoki VTY shell (polecenie `vtysh` z poziomu systemu operacyjnego). Można wówczas wydawać polecenia różnych demonów (np. `zebra` i `ripd` w prezentowanym przykładzie), bez potrzeby samodzielnego łączenia się z nimi. O ile konfiguracja została przeprowadzona poprawnie, wynikiem polecenia

```
show ip rip
```

powinno być wyświetlenie listy sieci znanych protokołowi RIP:

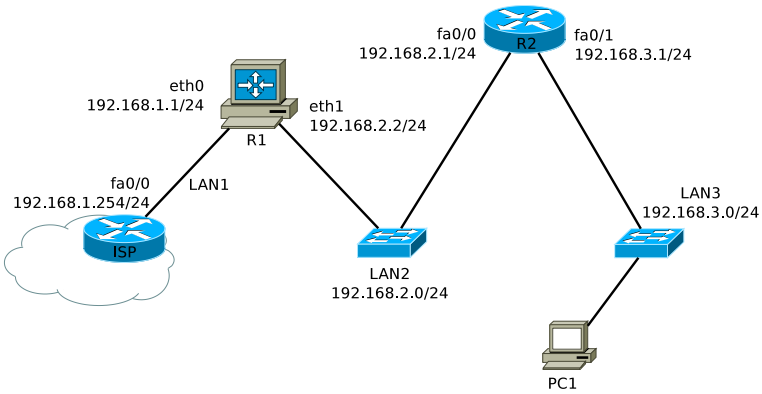
```
Codes: R-RIP, C-connected, S-Static, O-OSPF, B-BGP
Sub-codes:
 (n)-normal, (s)-static, (d)-default, (r)-redistribute,
 (i)-interface

     Network         Next Hop      Metric From     Tag Time
C(i) 192.168.1.0/24  0.0.0.0       1 self          0
```

```
  C(i) 192.168.2.0/24 0.0.0.0       1 self          0
9 R(n) 192.168.3.0/24 192.168.2.1   2 192.168.2.1   0 02:49
```

Tablica routingu (polecenie `show ip route`) powinna być zbliżona do:

```
1 Codes: K - kernel route, C - connected, S - static, R - RIP,
         O - OSPF, I - ISIS, B - BGP, > - selected route,
3        * - FIB route

5 C>* 127.0.0.0/8 is directly connected, lo
  C>* 192.168.1.0/24 is directly connected, eth0
7 C>* 192.168.2.0/24 is directly connected, eth1
  R>* 192.168.3.0/24 [120/2] via 192.168.2.1, eth1, 00:16:49
```

11.3. Zadanie



Rysunek 11.2. Schemat topologii logicznej sieci.

1. Zbuduj sieć zgodnie ze schematem z rys. 11.2. R1 jest komputerem PC z dwiema kartami sieciowymi Ethernet, z zainstalowanym Linuksem i pakietem Quagga. Będzie on pełnił rolę routera z protokołem RIPv2. R2 i ISP są routerami Cisco. ISP symuluje router operatora dostępu do Internetu.
2. Skonfiguruj router R2. Uruchom protokół RIPv2 dla obu dołączonych sieci.
3. Na routerze ISP skonfiguruj podstawowe ustawienia oraz trasy statyczne do sieci LAN2 i LAN3.
4. Przeprowadź konfigurację ustawień sieciowych routera R1. Uruchom protokół RIPv2 w sieci LAN2.

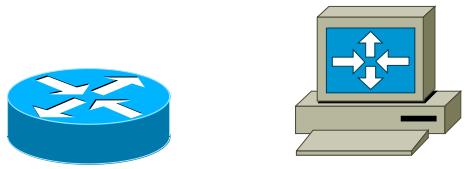
5. Upewnij się, czy sieć LAN3 pojawiła się w tablicy routingu R1.
6. Na routerze R1 skonfiguruj trasę domyślną do ISP i spraw, by informacja o niej została rozesłana poprzez RIP. Sprawdź, czy trasa domyślna pojawiła się w tablicy routingu R2.
7. Zweryfikuj działanie protokołu RIP na R1 i R2 przy pomocy odpowiednich poleceń `show`.
8. Skonfiguruj ustawienia sieciowe komputera PC1. Przetestuj połączenie z ISP (`ping`, `traceroute/tracert`).
9. *Zastąp RIP protokołem OSPF.*[4]

[4] Zadanie wymaga samodzielnego zapoznania się z odpowiednim fragmentem dokumentacji Quagga [8].

Dodatek A

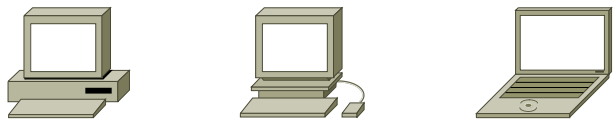
Symbole urządzeń

Na poniższych rysunkach przedstawione są używane w książce symbole urządzeń. Są to symbole powszechnie przyjęte i stosowane na schematach topologii sieciowych.

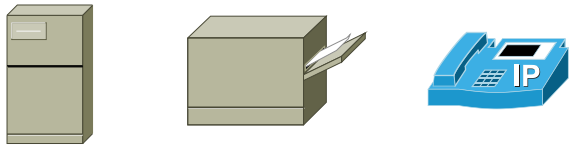


Rysunek A.1. Router sprzętowy, router zbudowany na bazie komputera PC.

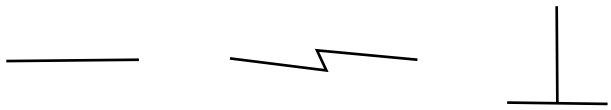
Rysunek A.2. Hub (koncentrator), przełącznik Ethernet.



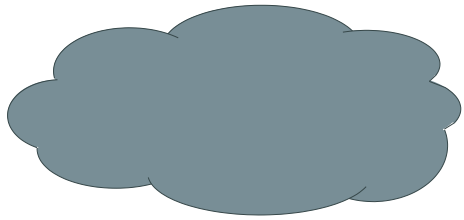
Rysunek A.3. Komputery PC.



Rysunek A.4. Serwer, drukarka sieciowa, telefon IP.



Rysunek A.5. Połączenie Ethernet, połączenie szeregowe, sieć Lan.



Rysunek A.6. Chmura – nieistotny w danym kontekście zbiór połączonych urządzeń sieciowych.

Dodatek B

Procedura odzyskiwania hasła

Konfiguracja routera może okazać się niemożliwa ze względu na uprzednie zabezpieczenie dostępu do konsoli lub trybu uprzywilejowanego nieznanym hasłem. Trzeba wówczas przeprowadzić procedurę odzyskiwania hasła. Procedury te są różne dla różnych urządzeń i opisane w dokumentacji [35]. Do ich przeprowadzenia niezbędny jest fizyczny dostęp do urządzenia. W związku z tym, w sieciach produkcyjnych ważne jest fizyczne zabezpieczenie urządzeń sieciowych przed dostępem osób nieuprawnionych. Poniższy opis dotyczy routerów Cisco serii 2600 i 2800 [34].

Odzyskanie kontroli nad urządzeniem zabezpieczonym nieznanym hasłem jest możliwe poprzez wykonanie poniższej sekwencji czynności:

1. Należy w standardowy sposób zestawić połączenie konsolowe z routerem.
2. Router należy wyłączyć i włączyć ponownie przy pomocy wyłącznika na obudowie.
3. W ciągu 60 sekund trzeba wysłać do routera sygnał *Break* (w przypadku HyperTerminala i PuTTY stosuje się kombinację klawiszową Ctrl+Break). Wskutek tego router zostanie uruchomiony w trybie ROMMON (*ROM monitor*). Jest to specjalny tryb pracy służący do niskopoziomowego rozwiązywania problemów. Znak zgłoszenia ma wówczas postać:

   ```
   rommon 1>
   ```

   Parametry dotyczące sposobu uruchamiania systemu operacyjnego IOS routera są zakodowane w rejestrze konfiguracji, przechowywanym w pamięci nieulotnej. Jego standardowa wartość to 0x2102. Wpisanie wartości 0x2142 spowoduje, że podczas startu urządzenia zostanie pominięty plik konfiguracyjny (w którym zapisane są hasła). Następnie router należy zrestartować. Służą do tego celu polecenia:

   ```
   confreg 0x2142
   reset
   ```

4. Po restarcie routera należy przejść do wiersza poleceń (udzielając odpowiedzi odmownej na pytanie u uruchomienie dialogu konfiguracyjnego) i trybu uprzywilejowanego (polecenie `enable`). Jeżeli zależy nam na zachowaniu istniejącej konfiguracji routera, należy wydać polecenie:

   ```
   copy startup-config running-config
   ```

   Następnie można zmodyfikować konfigurację (w standardowy sposób), ustalając nowe hasła lub usuwając je. Jeżeli natomiast istniejąca konfiguracja routera nie będzie już potrzebna, zamiast powyższego polecenia można użyć:

   ```
   erase startup-config
   ```

5. Należy przywrócić domyślną wartość rejestru konfiguracji, wydając w trybie konfiguracji globalnej (`configure terminal`) polecenie:

```
config-register 0x2102
```

6. Należy zapisać aktualną konfigurację:

```
copy running-config startup-config
```

Dodatek C

Security Device Manager

Wiersz poleceń IOS jest podstawowym, ale nie jedynym sposobem komunikowania się administratora z routerem. Cisco SDM (*Cisco Router and Security Device Manager*) jest narzędziem graficznym, uruchamianym w przeglądarce WWW, które umożliwia zarówno konfigurację podstawowych aspektów pracy routera, jak również wielu zaawansowanych mechanizmów, w większości związanych z bezpieczeństwem sieci.

Użycie SDM jest możliwe pod warunkiem posiadania jednego ze wspieranych przez niego routerów z odpowiednią wersją oprogramowania IOS. Nie są obsługiwane przełączniki. SDM jest napisany w języku Java, więc niezbędne jest również oprogramowanie Java Runtime Environment (JRE), w odpowiedniej wersji (np. dla SDM w wersji 2.5 zalecane jest JRE 1.6.0_02 lub 1.6.0_03). SDM współpracuje z przeglądarkami WWW Firefox, Netscape i Internet Explorer (również w odpowiedniej wersji).

W przypadku routera ze standardowymi ustawieniami konfiguracyjnymi, nie jest możliwe natychmiastowe użycie SDM. Pewne ustawienia muszą zostać wprowadzone w tradycyjny sposób (poprzez połączenie konsolowe). Trzeba zapewnić połączenie sieciowe między routerem a komputerem PC, na którym będzie działał SDM, a także skonfigurować zdalny dostęp do routera poprzez http lub https i telnet lub ssh [38].

Jeżeli komputer PC i router znajdują się w tej samej podsieci (10.0.0.0/24) i router jest do niej dołączony interfejsem FastEthernet0/0, należy wprowadzić polecenia konfiguracyjne:

```
1 interface FastEthernet0/0
  ip address 10.0.0.1 255.255.255.0
3 no shutdown
```

Na routerze należy założyć konto użytkownika – administratora posiadającego pełne uprawnienia (słowo `cisco` jest w tym przypadku hasłem użytkownika `admin`):

```
username admin privilege 15 secret 0 cisco
```

Należy włączyć usługę serwera http (lub https)[38]:

```
1 ip http server
  ip http secure-server
3 ip http authentication local
  ip http timeout-policy idle 60 life 86400 requests 10000
```

W powyższym listingu niezbędne są wiersze 1 lub 2, natomiast pozostałe są zalecane. Zdalny dostęp do routera poprzez telnet lub ssh konfiguruje się poleceniami:

```
line vty 0 15
```

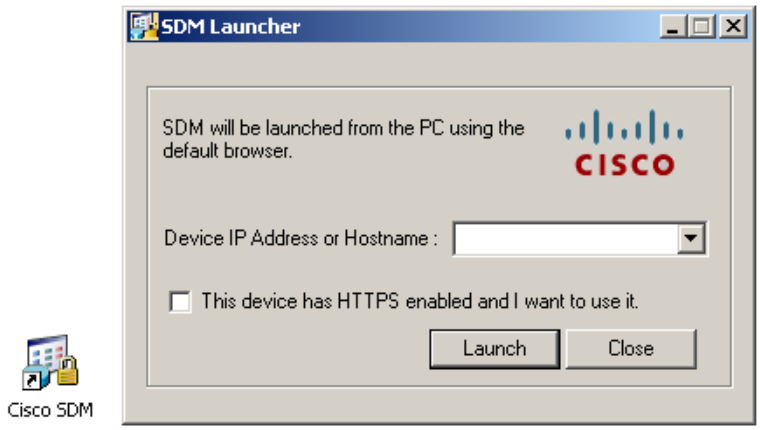
```
2 privilege level 15
  login local
4 transport input telnet
  transport input telnet ssh
```

Polecenie podane w 3. wierszu wymusza logowanie z użyciem loginu i hasła zdefiniowanego uprzednio przy pomocy `username`.

Rysunek C.1. Instalacja oprogramowania SDM.

Instalacja oprogramowania SDM przebiega w sposób typowy dla aplikacji MS Windows. W przypadku wybrania instalacji na komputerze (rys. C.1), możliwe będzie konfigurowanie z użyciem SDM dowolnego routera, przygotowanego zgodnie z procedurą opisaną powyżej. Po uruchomieniu programu SDM Launcher (rys. C.2), można wprowadzić adres IP żądanego routera. W razie instalacji na routerze, SDM zostanie zainstalowany w pamięci flash wybranego urządzenia i uruchomi się przy próbie połączenia z nim przeglądarką WWW. W obu przypadkach SDM uruchamia się w przeglądarce WWW komputera PC.

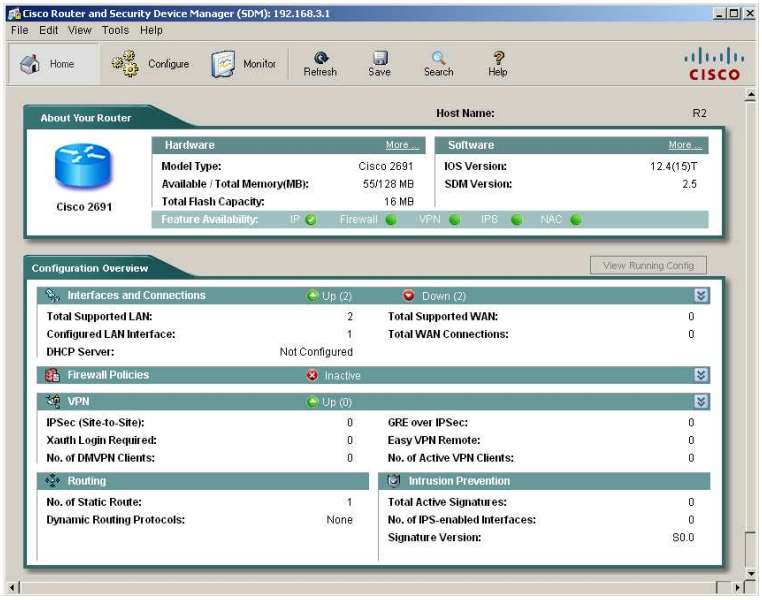
Rys. C.3 przedstawia główne okno SDM, prezentujące podstawowe informacje o routerze. Sekcja *Configure* (rys. C.4) umożliwia konfigurację podstawowych elementów (ustawień interfejsów, routingu) jak również zaawansowanych mechanizmów (firewall, VPN, system wykrywania intruzów, QoS). Warto zwrócić uwagę na opcję *Security Audit* (audyt bezpieczeństwa, rys. C.5). Opcja *One-step lockdown* umożliwia początkującemu administratorowi skonfigurowanie podstawowych mechanizmów bezpieczeństwa routera,



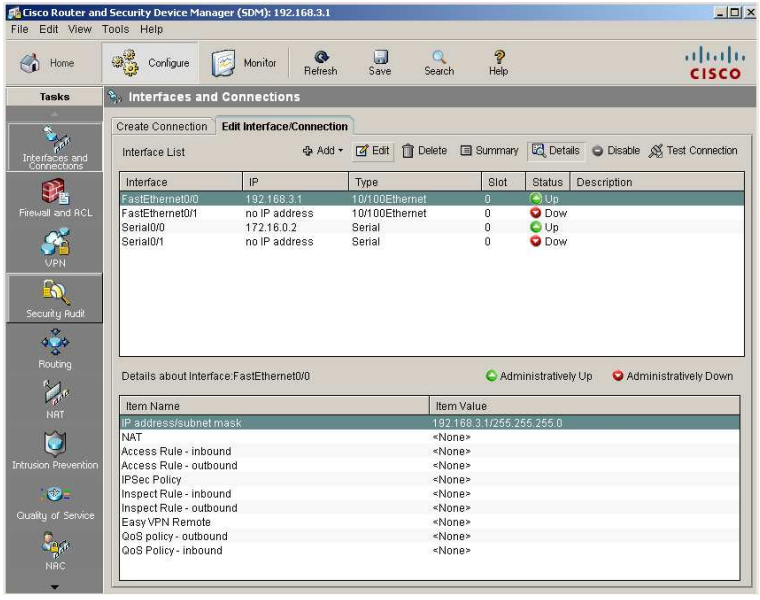
Rysunek C.2. Uruchamianie SDM zainstalowanego na komputerze PC.

bez konieczności posiadania szczegółowej wiedzy na ich temat. Opcja *Perform security audit* generuje raport o potencjalnych lukach bezpieczeństwa urządzenia (rys. C.6). W kolejnym kroku można zlecić programowi SDM wyeliminowanie wszystkich lub tylko wybranych zagrożeń (rys. C.7).

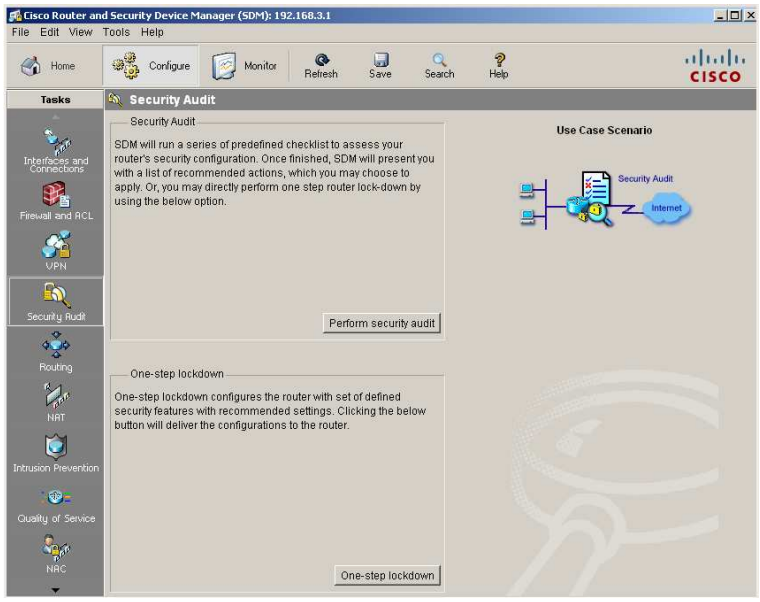
SDM generuje odpowiednie wpisy w pliku konfiguracyjnym routera, które można przejrzeć przed wysłaniem. Korzystanie z niego zwykle nie koliduje z tradycyjnym sposobem konfiguracji, poprzez wiersz poleceń. SDM znacząco przyspiesza uruchamianie złożonych mechanizmów (np. tuneli VPN), zmniejszając przy tym ryzyko popełnienia trudnych do wykrycia błędów. Z drugiej strony, jego używanie może skutkować trudniejszą dla administratora interpretacją znaczenia poszczególnych wpisów w pliku konfiguracyjnym.



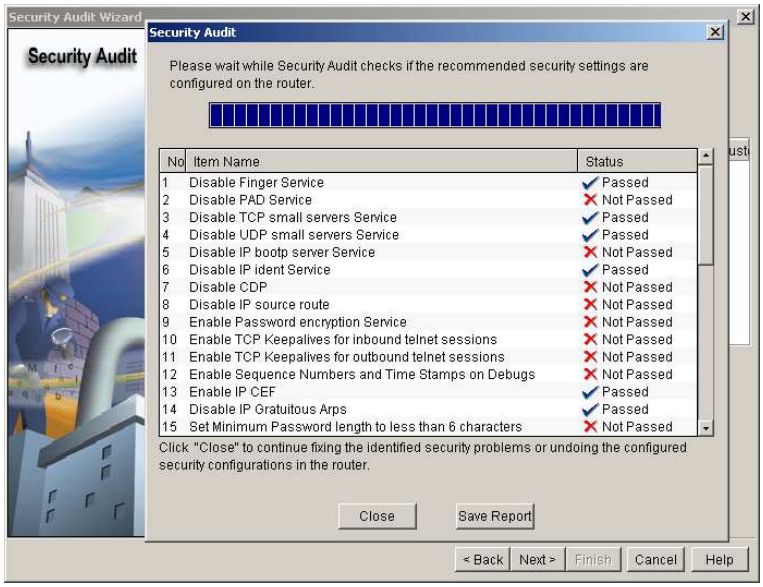
Rysunek C.3. Główne okno SDM.



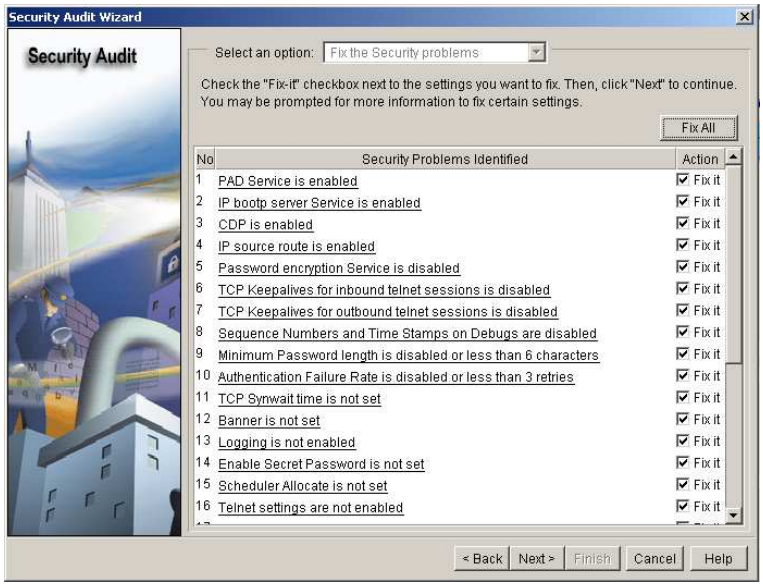
Rysunek C.4. Okno konfiguracji routera SDM.



Rysunek C.5. SDM – audyt bezpieczeństwa.



Rysunek C.6. SDM – przykładowy wynik audytu bezpieczeństwa.

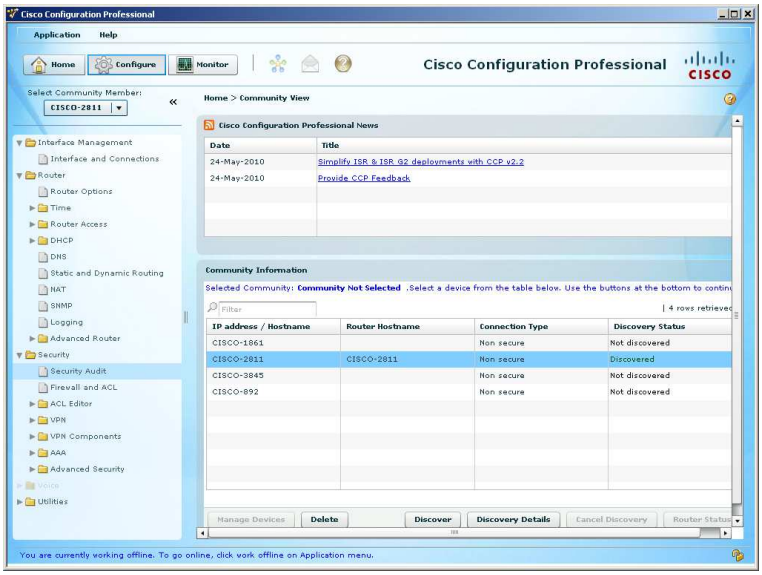


Rysunek C.7. SDM – okno wyboru zabezpieczeń routera.

Dodatek D

Cisco Configuration Professional

Cisco Configuration Professional (*Cisco CP*, CCP) [21, 37] jest kolejnym narzędziem do konfiguracji routerów Cisco za pośrednictwem interfejsu graficznego (rys. D.1). CCP posiada możliwości narzędzia SDM (Dodatek C) oraz szereg dodatkowych funkcji, np. narzędzia związane z telefonią IP. W zamierzeniu producenta, ma on zastąpić SDM. Pełna wersja CCP może być zainstalowana wyłącznie na komputerze PC i ma znacznie większe wymagania niż SDM (w przypadku wersji 2.2: min. 1 GB RAM, JRE 1.5.0_11 – 1.6.0_17, Adobe Flash Player v. 10.0 lub nowszy). Większa jest również grupa obsługiwanych urządzeń. *Cisco CP Express* jest wersją CCP o mniejszych możliwościach i zajmującą mniej miejsca, przeznaczoną do instalacji w pamięci flash routera.



Rysunek D.1. Jedno z okien CCP.

Procedura przygotowania routera do współpracy z CPP i sama instalacja przebiega podobnie jak w przypadku SDM. Szczegółowe informacje można znaleźć w dokumentacji [21].

Słownik angielsko-polski

administrative distance	odległość administracyjna
access control list (ACL)	lista kontroli dostępu
adjacency	relacja przylegania
authentication	uwierzytelnianie
autonomous system	system autonomiczny
backup designated router (BDR)	zapasowy router desygnowany
bandwidth	szerokość pasma
boundary router	router brzegowy
bounded update	aktualizacja z ograniczeniami
broadcast	rozgłoszenie
classful routing	routing klasowy
Classless Inter-Domain Routing	bezklasowy routing międzydomenowy
classless routing	routing bezklasowy
convergence	zbieżność
count to infinity	zliczanie do nieskończoności
default gateway	brama domyślna
default route	trasa domyślna
designated router (DR)	router desygnowany
distance-vector protocol	protokół wektora odległości
end system	system końcowy
equal-cost load balancing	rozkładanie obciążenia na trasy równorzędne
equal-cost metric	metryka równorzędna
feasibility condition (FC)	warunek dopuszczalności
feasible distance (FD)	odległość dopuszczalna
feasible successor (FS)	dopuszczalny sukcesor
flapping link	niestabilne łącze
flash memory	pamięć flash
flooding	rozsyłanie zalewowe

frame	ramka
holddown timer	licznik wstrzymania
hop count	liczba skoków
hub	koncentrator
intermediate system	system pośredni
ISP	dostawca usług internetowych
link state advertisement	ogłoszenie stanu łącza
link-state protocol	protokół stanu łącza
local area network (LAN)	sieć lokalna
loopback	pętla zwrotna
manual summarization	ręczne podsumowanie
multicast	multicast
neighboring routers	sąsiednie routery
network service access point	punkt dostępu usługi sieciowej
OSPF area	obszar OSPF
partial update	częściowa aktualizacja
point-to-point	punkt-punkt
poison reverse	zatruwanie wstecz
privileged EXEC mode	uprzywilejowany tryb EXEC
Quality of Service (QoS)	jakość usług
reported distance (RD)	odległość ogłaszana
route aggregation	agregacja tras
route poisoning	zatruwanie tras
route summarization	podsumowanie tras
routed protocol	protokół routowany, routowalny
router	router
routing	routing, trasowanie
Routing Information Base (RIB)	baza informacji o routingu
routing loop	pętla routingu
routing protocol	protokół routingu
routing table	tablica routingu
split horizon rule	reguła podzielonego horyzontu
stuck in active route	trasa zablokowana w stanie aktywnym
subnet mask	maska podsieci
subnetting	podział na podsieci
successor	sukcesor
summary route	trasa sumaryczna
supernet	supersieć
switch	przełącznik
triggered update	aktualizacja wyzwalana

unequal-cost load balancing	rozkładanie obciążenia na trasy nierównorzędne
user EXEC mode	tryb EXEC użytkownika
Variable Length Subnet Masks	podział na podsieci z maską o zmiennej długości
Wide Area Nerwork (WAN)	sieć rozległa
wildcard mask	maska blankietowa

Bibliografia

[1] R. Braden, J. Postel, Requirements for Internet Gateways, Request for Comments 1009, Network Working Group, 1987, http://tools.ietf.org/html/rfc1009.

[2] R. Callon, Use of OSI IS-IS for Routing in TCP/IP and Dual Environments, Request for Comments 1195, Network Working Working Group, 1990, http://tools.ietf.org/html/rfc1195.

[3] E.W. Dijkstra, A note on two problems in connexion with graphs, Numerische Mathematik 1: 269-271, 1959.

[4] V. Fuller, T. Li, J. Yu, K. Varadhan, Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy, Request for Comments 1519, Network Working Group, 1993, http://tools.ietf.org/html/rfc1519.

[5] J. Hawkinson, T. Bates, Guidelines for creation, selection, and registration of an Autonomous System (AS), Request for Comments 1930, Network Working Group, 1996, http://tools.ietf.org/html/rfc1930.

[6] C. Hedrick, Routing Information Protocol, Request for Comments 1058, Network Working Group, 1988, http://tools.ietf.org/html/rfc1058.

[7] R. Hinden, Applicability Statement for the Implementation of Classless Inter-Domain Routing (CIDR), Request for Comments 1517, Network Working Group, 1993, http://tools.ietf.org/html/rfc1517.

[8] K. Ishiguro, A routing software package for TCP/IP networks Quagga 0.99.4, 2006, http://www.quagga.net.

[9] D. Mills, U. Delaware, J. Martin, J. Burbank, W. Kasch, Network Time Protocol Version 4: Protocol and Algorithms Specification, Request for Comments 5905, Internet Engineering Task Force (IETF), 2010, http://tools.ietf.org/html/rfc5905.

[10] J. Mogul, J. Postel, Internet Standard Subnetting Procedure, Request for Comments 950, Network Working Group, 1985, http://tools.ietf.org/html/rfc950.

[11] J. Moy, OSPF Version 2 Request for Comments 2328, Network Working Group, 1998, http://tools.ietf.org/html/rfc2328.

[12] D. Oran, OSI IS-IS Intra-domain Routing Protocol, Request for Comments 1142, Network Working Group, 1990, http://tools.ietf.org/html/rfc1142.

[13] D. Piscitello, Use of ISO CLNP in TUBA Environments, Request for Comments 1561, Network Working Group, 1993, http://tools.ietf.org/html/rfc1561.

[14] Y. Rekhter, T. Li, An Architecture for IP Address Allocation with CIDR,

Request for Comments 1518, Network Working Group, 1993, http://tools.ietf.org/html/rfc1518.
[15] Y. Rekhter, T. Li, S. Hares, A Border Gateway Protocol 4 (BGP-4), Request for Comments 4271, Network Working Group, 2006, http://tools.ietf.org/html/rfc4271.
[16] Y. Rekhter, S. Sangli, D. Tappan, 4-Octet AS Specific BGP Extended Community, Request for Comments 5668, Network Working Group, 2009, http://tools.ietf.org/html/rfc5668.
[17] Y. Rekhter, C. Topolcic, Exchanging Routing Information Across Provider Boundaries in the CIDR Environment, Request for Comments 1520, Network Working Group, 1993, http://tools.ietf.org/html/rfc1520.
[18] Q. Vohra, E. Chen, BGP Support for Four-octet AS Number Space, Request for Comments 4893, Network Working Group, 2007, http://tools.ietf.org/html/rfc4893.
[19] A. Zinin, Cisco IP Routing: Packet Forwarding and Intra-domain Routing Protocols, Addison-Wesley Professional, 2001.
[20] An Introduction to IGRP, Document ID: 26825, http://cisco.com.
[21] Cisco Configuration Professional Quick Start Guide, April 21 2010, http://cisco.com.
[22] Cisco IOS Configuration Fundamentals Command Reference, http://cisco.com.
[23] Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2, http://cisco.com.
[24] Cisco IOS IP Command Reference, Volume 2 of 4: Routing Protocols, Release 12.3, http://cisco.com.
[25] Cisco IOS IP Routing: EIGRP Configuration Guide, Release 12.4, http://cisco.com.
[26] Cisco IOS Release 12.0 Network Protocols Command Reference, Part 1, http://cisco.com.
[27] Configuring Integrated IS-IS, Cisco IOS IP and IP Routing Configuration Guide, Release 12.1, http://cisco.com.
[28] Configuring ISO CLNS, 2010 http://cisco.com.
[29] EIGRP to OSPF Migration Strategies, White Paper, Juniper Networks, 2005, http://www.juniper.net/solutions/literature/white_papers/350053.pdf.
[30] Enhanced Interior Gateway Routing Protocol, Document ID: 16406, http://cisco.com.
[31] Introduction to EIGRP, Document ID: 13669, http://cisco.com.
[32] Introduction to Intermediate System-to-Intermediate System Protocol, 2002, http://cisco.com.
[33] OSPF Design Guide, Document ID: 7039, http://cisco.com.
[34] Password Recovery Procedure for the Cisco 2600 and 2800 Series Routers, Document ID: 22188, http://cisco.com.
[35] Password Recovery Procedures, Document ID: 6130, http://cisco.com.
[36] Redistributing Routing Protocols, Document ID: 8606, http://cisco.com.
[37] Release Notes for Cisco Configuration Professional 2.2, May 26 2010, http://cisco.com.

[38] Release Notes for Cisco Router and Security Device Manager 2.5, January 21 2008, `http://cisco.com`.
[39] What Is Administrative Distance?, Document ID: 15986, `http://cisco.com`.

Skorowidz