

Sieci Komputerowe

Jakub Gałaszewski

March 13, 2024

1. Dla każdego z podanych poniżej adresów IP w notacji CIDR określ, czy jest to adres sieci, adres rozgłoszeniowy czy też adres komputera. W każdym przypadku wyznacz odpowiadający mu adres sieci, rozgłoszeniowy i jakiś adres IP innego komputera w tej samej sieci. adres rozgłoszeniowy (broadcast) to ostatni adres, natomiast adres sieci to pierwszy adres.

► **10.1.2.3/8**

adres sieci: **10.0.0.0**

adres rozgłoszeniowy: **10.255.255.255**

adres innego komputera: **10.1.1.1**

► **156.17.0.0/16**

adres sieci: **156.17.0.0**

adres rozgłoszeniowy: **156.17.255.255**

adres innego komputera: **156.17.1.1**

► **99.99.99.99/27**

inaczej: **01100011.01100011.01100011.01100011**

adres sieci: **99.99.99.96**

adres rozgłoszeniowy: **99.99.99.127**

adres innego komputera: **99.99.99.100**

► **156.17.64.4/30**

inaczej: **10011100.00010001.01000000.00000100**

adres sieci: **156.17.64.4**

adres rozgłoszeniowy: **156.17.64.7**

adres innego komputera: **156.17.64.5**

► **123.123.123.123/32**

adres sieci: **123.123.123.123**

adres rozgłoszeniowy: **123.123.123.123** (ponieważ pierwsza sieć jest równocześnie ostatnią)

adres innego komputera: **brak**

2. Podziel sieć 10.10.0.0/16 na 5 rozłącznych podsieci, tak aby każdy z adresów IP z sieci 10.10.0.0/16 był w jednej z tych 5 podsieci. Jak zmieniła się liczba adresów IP możliwych do użycia przy adresowaniu komputerów? Jaki jest minimalny rozmiar podsieci, który możesz uzyskać w ten sposób?

Aby podzielić sieć należy określić ile potrzeba bitów aby podzielić na 5 równych sieci. Potrzeba ich 3 ($2^3 = 8$).

rozpiszmy naszą sieć na system dwójkowy

10.10.0.0 = 00001010.00001010.00000000.00000000

zaznaczone wyżej bity będą odpowiadały za wybór naszej podsieci. Czyli adresy tych podsieci to będą:

- (a) 10.10.0.0/19
- (b) 10.10.32.0/19
- (c) 10.10.64.0/19
- (d) 10.10.96.0/19
- (e) 10.10.128.0/17

można zauważyć że ostatnia sieć posiada inną maskę. Chcemy tak, ponieważ musimy objąć wszystkie adresy. Gdybyśmy zostawili starą maskę (tj. 19), to nie mielibyśmy wykorzystanych adresów powyżej 10.10.159.255.

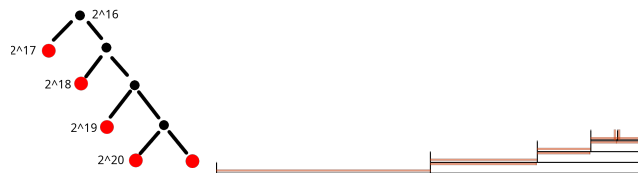
Dzięki temu uzyskaliśmy 5 rozłącznych podsieci, a minimalny rozmiar podsieci wynosi $2^{32-19} = 8192$

errata:

W zadaniu chodziło o najgłębsze drzewo. Tak więc poprawne odpowiedzi to:

- (a) 10.10.0.0/17
- (b) 10.10.128.0/18
- (c) 10.10.192.0/19
- (d) 10.10.224.0/20
- (e) 10.10.240.0/20

Warto napomknąć że 10 adresów będzie zajętych przez broadcast i adres sieci.
wizualizacja:



imo wygodniejszy sposób reprezentacji tego zadania to "kubelki" (ten po prawej).

3. Tablica routingu zawiera następujące wpisy (podsieć → dokąd wysłać):

- ▶ 0.0.0.0/0 → do routera A
- ▶ 10.0.0.0/23 → do routera B
- ▶ 10.0.2.0/24 → do routera B
- ▶ 10.0.3.0/24 → do routera B
- ▶ 10.0.1.0/24 → do routera C
- ▶ 10.0.0.128/25 → do routera B
- ▶ 10.0.1.8/29 → do routera B

- ▶ 10.0.1.16/29 → do routera B
- ▶ 10.0.1.24/29 → do routera B

Napisz równoważną tablicę routingu zawierającą jak najmniej wpisów

to co będziemy chcieli zrobić, to "zlepić" odpowiednie adresy, o ile są one sąsiadujące, oraz wykluczyć "nadmiarowe" adresy. Możemy zauważyć że:

koniec podsieci trzeciej (10.0.2.255) znajduje się obok sieci czwartej (10.0.3.0), tak więc możemy scalić (rozmiar maski maleje nam o 1);

koniec podsieci drugiej (10.0.1.255) znajduje się obok sieci trzeciej (10.0.2.0), tak więc możemy je złączyć (rozmiar maski maleje nam o 1);

podsieć szóstą w całości znajduje się w podsieci drugiej, tak więc możemy pominąć;

podsieć ósmą i dziewiątą (odpowiednio 10.0.1.23 i 10.0.1.24) można połączyć ze sobą;

podsieci siódmej nie możemy podłączyć, ponieważ nie potrafimy utworzyć takiej jednoznacznej maski.

Ostatecznie po wszystkich edycjach uzyskamy:

- ▶ 0.0.0.0/0 → do routera A
- ▶ 10.0.0.0/22 → do routera B
- ▶ 10.0.1.0/24 → do routera C
- ▶ 10.0.1.8/29 → do routera B
- ▶ 10.0.1.16/28 → do routera B

4. Wykonaj powyższe zadanie dla tablicy

- ▶ 0.0.0.0/0 → do routera A
- ▶ 10.0.0.0/8 → do routera B
- ▶ 10.3.0.0/24 → do routera C
- ▶ 10.3.0.32/27 → do routera B
- ▶ 10.3.0.64/27 → do routera B
- ▶ 10.3.0.96/27 → do routera B

Dwa ostatnie adresy możemy scalić, uzyskując 10.3.0.64/26

Mimo że może się zdawać, że nie potrafimy już bardziej tego złączyć, możemy zauważyć jedną rzecz. rozbijając router C na dwa osobne wpisy w tablicy routingu, będziemy mogli pozbyć aż trzy (!) ostatnie wpisy (lub dwa, jeżeli już się scalało). Niezależnie od tego czy scalaliśmy wcześniej, efekt jest zadowalający. tak więc adres C zakrywa odpowiednio od 10.3.0.0 do 10.3.0.31 i od 10.3.0.128 do 10.3.0.255. Tak więc nasza odpowiedź to:

- ▶ 0.0.0.0/0 → do routera A
- ▶ 10.0.0.0/8 → do routera B
- ▶ 10.3.0.0/27 → do routera C
- ▶ 10.3.0.128/25 → do routera C

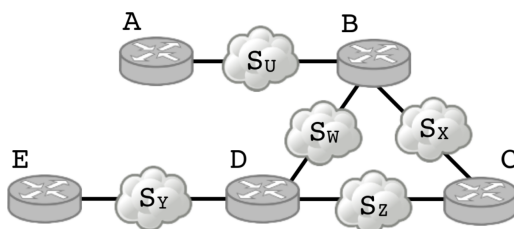
5. Jak uporządkować wpisy w tablicy routingu, żeby zasada najlepszego dopasowania odpowiadała wyborowi „pierwszy pasujący” (tj. przeglądaniu tablicy od początku do końca aż do momentu napotkania dowolnej pasującej reguły)? Odpowiedź uzasadnij formalnie.
 wiemy że tablice routingu kierują się regułą najdłuższego prefiksu dla wielu identycznych reguł. Tak więc nasze uporządkowanie listy powinny być w kolejności malejącej względem długości prefiksa.

Dowód: Załóżmy nie wprost, że zachłanny wynik z posortowanej tablicy daje nam wpis gorszy od najlepszego możliwego. rozpatrzmy trzy przypadki:

- prefiks najlepszej odpowiedzi jest dłuższy od zachłannej odpowiedzi.
to stwierdzenie nie może być prawdziwe, ponieważ zachłannie bierzemy prawidłowy wpis z najdłuższym możliwym prefiksem;
- prefiks najlepszej odpowiedzi jest równy wynikowi zachłannemu.
w takim przypadku odpowiedź najlepsza jest ta sama, co zachłanny wynik, ponieważ wpisy dla masek tej samej długości nie nachodzą na siebie (n pierwszych bitów jest "nieedytowalna");
- prefiks najlepszej odpowiedzi jest krótszy od wyniku zachłannego.
Nie jest to możliwe, ponieważ zgodnie z regułą wybieramy adres pasujący z najdłuższym możliwym prefiksem.

Dochodzimy do sprzeczności. Tak więc najlepszy wynik to zachłanny, co kończy nam dowód.

6. W podanej niżej sieci tablice routingu budowane są za pomocą algorytmu wektora odległości. Pokaż (krok po kroku), jak będzie się to odbywać. W ilu krokach zostanie osiągnięty stan stabilny?



Algorytm wektora odległości polega na okresowym powiadamianiu sąsiednich routerów o całej swojej tablicy przekazywania i aktualizowaniu na tej bazie.

Na początku uzupełnijmy wszystkich sąsiadów

	A	B	C	D	E
do S_u	1	1			
do S_w		1		1	
do S_x		1	1		
do S_z			1	1	
do S_y				1	1

	A	B	C	D	E
do S_u	1	1	2(via B)	2(via B)	
do S_w	2(via B)	1	2(via B)	1	2(via D)
do S_x	2(via B)	1	1	2(via C)	
do S_z		2(via C)	1	1	2(via D)
do S_y		2(via D)	2(via D)	1	1

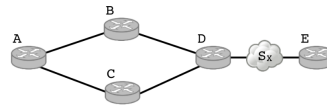
	A	B	C	D	E
do S_u	1	1	2(via B)	2(via B)	3(via D)
do S_w	2(via B)	1	2(via B)	1	2(via D)
do S_x	2(via B)	1	1	2(via C)	3(via D)
do S_z	3(via B)	2(via C)	1	1	2(via D)
do S_y	3(via B)	2(via D)	2(via D)	1	1

7. Załóżmy, że w powyższej sieci tablice routingu zostały już zbudowane. Co będzie się działo (krok po kroku), jeśli zostanie dodana sieć S_Q łącząca routery A i E ? po dodaniu w pierwszym kroku router A i E uzyskają informację że potrafią dotrzeć do S_Q ścieżką długości jeden. W drugim kroku tablica uzupełni się dla reszty routerów względem routera A lub E.

	A	B	C	D	E
do S_u	1	1	2(via B)	2(via B)	3(via D)
do S_w	2(via B)	1	2(via B)	1	2(via D)
do S_x	2(via B)	1	1	2(via C)	3(via D)
do S_z	3(via B)	2(via C)	1	1	2(via D)
do S_y	3(via B)	2(via D)	2(via D)	1	1
do S_Q	1				1

	A	B	C	D	E
do S_u	1	1	2(via B)	2(via B)	3(via D)
do S_w	2(via B)	1	2(via B)	1	2(via D)
do S_x	2(via B)	1	1	2(via C)	3(via D)
do S_z	3(via B)	2(via C)	1	1	2(via D)
do S_y	3(via B)	2(via D)	2(via D)	1	1
do S_Q	1	2(via A)	3(via A)	2(via E)	1

8. W przedstawionej poniżej sieci uszkodzeniu ulega połączenie między routerami D i E . Załóżmy, że w sieci działa algorytm wektora odległości wykorzystujący technikę zatrzymania ścieżki zwrotnej (poison reverse). Pokaż — opisując krok po kroku jakie komunikaty są przesyłane między routerami — że może powstać cykl w routingu.



poison reverse polega na wykorzystywaniu informacji skąd przybyliśmy aby zabezpieczyć się przed wieczną pętlą. Na początku tablica wygląda następująco:

	A	B	C	D	E
do S_x	3(via B)	2(via D)	2(via D)	1	1

S_x podana sieć będzie początkowo wyglądać:

	A	B	C	D	E
do S_x	3(via B)	2(via D)	2(via D)		

w kolejnym kroku B ustawi wartość dla D równą nieskończoność

	A	B	C	D	E
do S_x	3(via B)	2(via D)	2(via D)	inf	

następnie B i C zostaną ustawione na nieskończoność

	A	B	C	D	E
do S_x	3(via B)	inf	inf	inf	

aby na końcu ustawić w A nieskończoność

	A	B	C	D	E
do S_x	inf	inf	inf	inf	

9. Pokaż, że przy wykorzystaniu algorytmu stanu łączy też może powstać cykl w routingu. W tym celu skonstruuj sieć z dwoma wyróżnionymi, sąsiadującymi ze sobą routerami A i B. Załóż, że wszystkie routery znają graf całej sieci. W pewnym momencie łącze między A i B ulega awarii, o czym A i B od razu się dowiadują. Zalewają one sieć odpowiednią aktualizacją. Pokaż, że w okresie propagowania tej aktualizacji (kiedy dotarła ona już do części routerów a do części nie) może powstać cykl w routingu