MINISTERSTVO ŠKOLSTVA, VEDY, VÝSKUMU A ŠPORTU SLOVENSKEJ REPUBLIKY

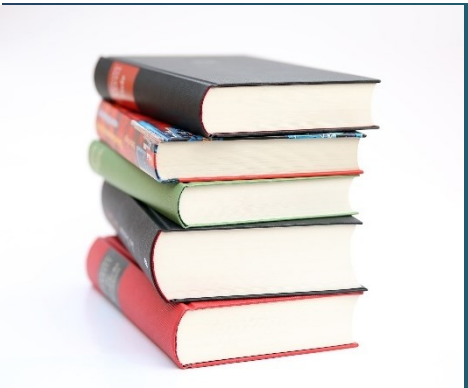
PIKS, prednáška 8

Subsieťovanie v IPv4 sieťach

Introduction to Networks v6.0 – 7.3, 8.1.3, 8.1.4, 8.1.5, 8.2

Katedra informačných sietí
Fakulta riadenia a informatiky, UNIZA

KIS FRI UNIZA – Vytvorené v rámci projektu KEGA 011STU - 4/2017.



Obsah prednášky

- 7.3 Protokol ICMPv4
- Opakovanie – adresovanie v IPv4 a úvod do subsieťovania
- 8.1.3 Subsieťovanie sietí s prefixom /16 a /8
- 8.1.4 Subsieťovanie podľa špecifických požiadaviek
- 8.1.5 VLSM
- 8.2 IPv4 adresný plán

7.3 Protokol ICMPv4

ICMPv4
ICMPv6

Internet Control Message Protocol

- Protokol ICMP je pomocný signalizačný protokol, ktorý asistuje protokolom IPv4 a IPv6 pri ich činnosti
  - Umožňuje otestovať základnú konektivitu s ďalším IP uzlom
    - Typy ICMP správ: **Ping (Echo Request, Echo Reply)**
  - Informuje o nedoručiteľnosti konkrétneho paketu a dôvode
    - Typy ICMP správ: **Destination Unreachable (mnoho podtypov), TTL Exceeded**
  - Informuje o potenciálne lepšej ceste
    - Typy ICMP správ: **Redirect**
  - V IPv6 poskytuje funkcie pre objavenie smerovača, automatickú konfiguráciu adries a nahrádza protokol ARP
    - Typy ICMP správ: **Router Discovery, Neighbor Discovery**
- Správy ICMP protokolu sa vkladajú priamo do IP paketov
  - ICMP správu môže vytvoriť ktorýkoľvek IP uzol pozdĺž cesty medzi zdrojom a cieľom (zdroj, cieľ, medziľahlý smerovač)
  - ICMP správa je spravidla určená odosielateľovi pôvodného paketu

Formát ICMP správy

- ICMP správa má svoj typ a kód:
- Typ = čoho sa daná správa týka
- Kód = bližšie špecifikuje daný typ správy

	0	1	2	3	Octet offset
ICMP hlavička (header) 8 bytes	Typ	Kód	Header checksum		0
	Ďalšie ICMP polia hlavičky (podľa konkrétneho typu a kódu správy, nemusí sa využiť)				4
ICMP telo (payload) ľubovoľnej dĺžky	ICMP dáta (voliteľné, nemusí sa využiť vôbec)				8

Typy ICM

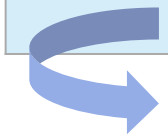
Type	Code	Status	Description
0 – Echo Reply[3]:14	0		Echo reply (used to ping)
1 and 2		unassigned	*Reserved*
3 – Destination Unreachable[3]:4	0		Destination network unreachable
	1		Destination host unreachable
	2		Destination protocol unreachable
	3		Destination port unreachable
	4		Fragmentation required, and DF flag set
	5		Source route failed
	6		Destination network unknown
	7		Destination host unknown
	8		Source host isolated
	9		Network administratively prohibited
	10		Host administratively prohibited
	11		Network unreachable for ToS
	12		Host unreachable for ToS
	13		Communication administratively prohibited
	14		Host Precedence Violation
	15		Precedence cutoff in effect

Type	Code	Status	Description
4 – Source Quench	0	deprecated	Source quench (congestion control)
5 – Redirect Message	0		Redirect Datagram for the Network
	1		Redirect Datagram for the Host
	2		Redirect Datagram for the ToS & network
	3		Redirect Datagram for the ToS & host
6		deprecated	Alternate Host Address
7		unassigned	*Reserved*
8 – Echo Request	0		Echo request (used to ping)
9 – Router Advertisement	0		Router Advertisement
10 – Router Solicitation	0		Router discovery/selection/solicitation
11 – Time Exceeded[3]:6	0		TTL expired in transit
	1		Fragment reassembly time exceeded
12 – Parameter Problem: Bad IP header	0		Pointer indicates the error
	1		Missing a required option
	2		Bad length
13 – Timestamp	0		Timestamp
14 – Timestamp Reply	0		Timestamp reply
15 – Information Request	0	deprecated	Information Request
16 – Information Reply	0	deprecated	Information Reply
17 – Address Mask Request	0	deprecated	Address Mask Request
18 – Address Mask Reply	0	deprecated	Address Mask Reply
19		reserved	*Reserved* for security
20 through 29		reserved	*Reserved* for robustness experiment
30 – Traceroute	0	deprecated	Information Request

ICMP správa typu 3: Destination Unreachable

- Tento typ správy má 15 podtypov (= kódy), niektoré z nich sú nasledovné:
  - Kód 0 – Destination **network unreachable**
    - Cieľová sieť je nedostupná, neexistuje fyzické spojenie k nej
  - Kód 1 – Destination **host unreachable**
    - Neexistuje fyzické spojenie k danému cieľovému hostovi
  - Kód 2 – Destination **protocol unreachable**
    - Daný L4 cieľový protokol nie je aktívny
  - Kód 3 – Destination **port unreachable**
    - Daný L4 cieľový port nie je aktívny
  - Kód 4 – Fragmentation required, **packet too big**
    - Paket je pre dané prenosové médium príliš veľký, vyžaduje sa fragmentácia, ale datagram má nastavený príznak DF = 'don't fragment', preto bol zahodený.
    - V tele ICMP správy sa potom pošle odosieľateľovi takéhoto veľkého IP paketu hodnota MTU, ktorá sa vyžaduje na danej linke, ktorou chcel takýto paket prejsť (tzv. Next-hop MTU)
  - Kód 13 – **Communication administratively filtered**
    - Filtrácia adminom (firewall na PC, alebo ACL na smerovači - bude sa učiť až v PS1/2)
- ICMPv6 má podobné, ale iné kódy pre tento typ správ.

Typ = **3**	Kód = **..viď vyššie...**	Header checksum
unused		**Next-hop MTU**
IP hlavička a prvých 8 bajtov pôvodného IP paketu		



- Používa sa, aby zdrojová stanica vedela ktorého IP paketu (ktorý bol zahodený) sa tento typ ICMP správy týka
  - (Pre L4 protokoly (TCP a UDP) týchto 8 bajtov zahŕňa aj čísla portov)

ICMP správa typu 11: Time Exceeded

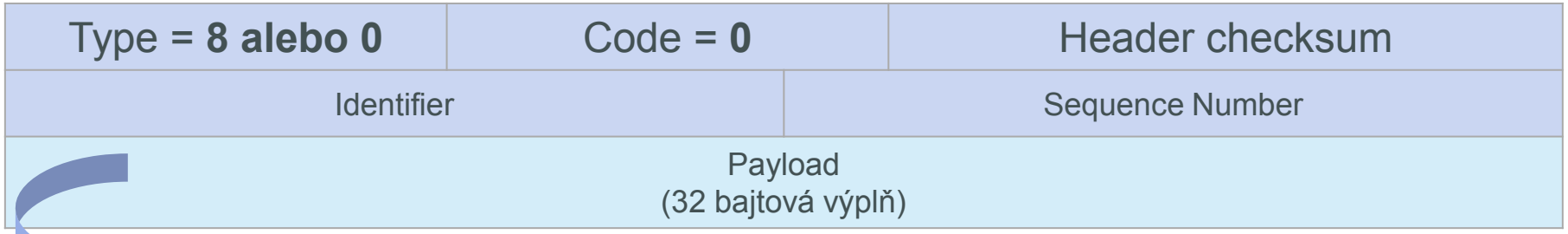
- Môže vygenerovať smerovač, keď hodnota poľa TTL v hlavičke spracovávaného paketu po dekrementácii (TTL-1) je 0
  - ICMPv6 – robí to isté, akurát pre pole Hop limit
- Pošle odosielateľovi daného paketu

Typ = **11**	Kód = **0**	Header checksum
nepoužité		
IP hlavička a prvých 8 bajtov pôvodného IP paketu		

- Používa sa, aby zdrojová stanica vedela ktorého IP paketu (ktorý bol zahodený) sa tento typ ICMP správy týka
  - (Pre L4 protokoly (TCP a UDP) týchto 8 bajtov zahŕňa aj čísla portov)

ICMP správy typu 8 a 0: Echo request a Echo reply

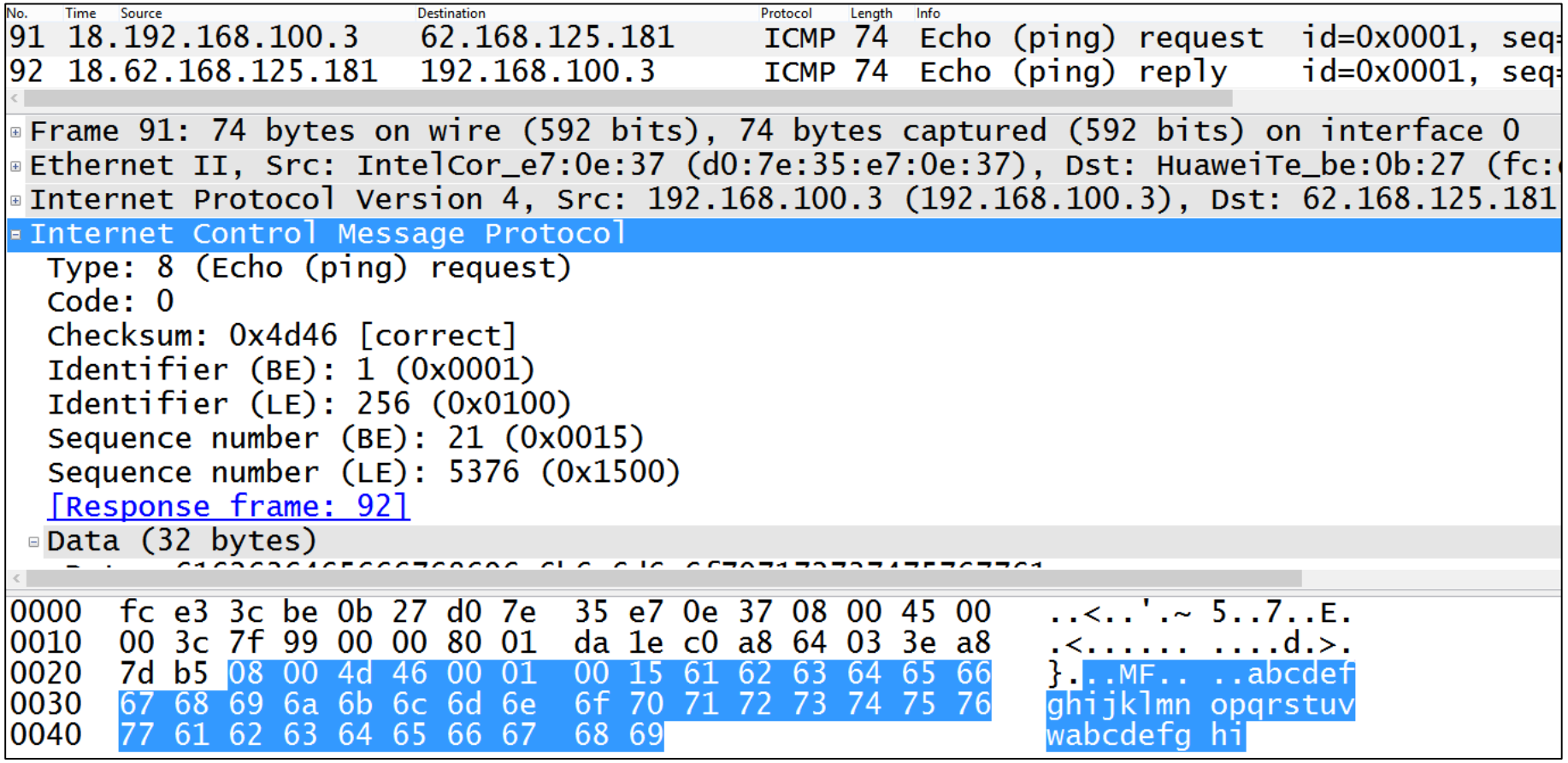
- využíva utilita **ping** na testovanie dostupnosti k nejakému hostovi v IP sieti
  - Odosiela 4 ICMP správy typu echo request a očakáva že cieľová stanica vygeneruje 4 ICMP správy typu echo reply, ako odpovede
  - Má nastavený čas čakania (timeout) na doručenie ICMP správy echo reply
    - Ak nepríde reply, indikuje to problém s konektivitou, alebo blokovanie ICMP správ niektorým smerovačom po ceste k cieľu
  - Vypočíta percentuálnu úspešnosť, a priemerný čas, za ktorý správa prišla tam a späť (tzv. round-trip time)



- Preskúmate na cvičení, čo obsahuje, aj aké hodnoty dáva Windows pre polia identifier a sequence number

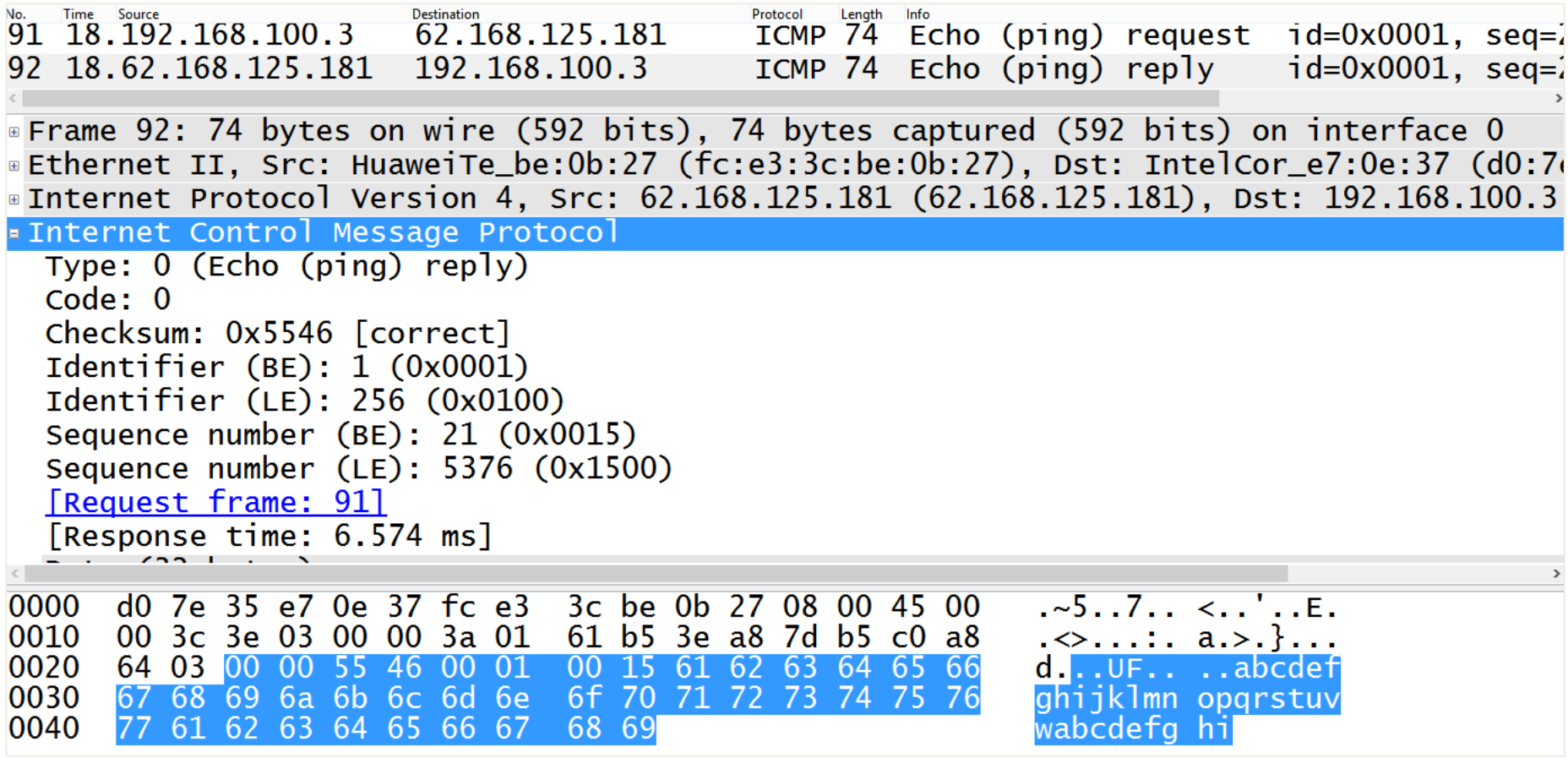
- Identifier – spoločný identifikátor rovnaký pre všetky 4 správy (reply alebo request) vrámci jedného spustenia príkazu ping (napr. 1)
- Sequence number – identifikátor konkrétneho request/reply, vrámci jedného pingu, t.j. poradové číslo requestu/reply, s číslovaním nie od nuly, ale od nejakého „n“ – toto všetko závisí od OS (napr. 22, 23, 24, 25)

ICMP echo request – z Wiresharku



- Polia Identifier aj Sequence number sú 2B t.j. 16 bitové, v odchytenej ICMP správe hodnota seq.n. v binárnom tvare: 00000000 00010101
  - Tú môžeme interpretovať dvojako - sú známe dva formáty: BE = big endian alebo LE = less endian (endian = ukončenie), Wireshark nevie na základe obsahu správy jednoznačne určiť, ktorý formát použiť, preto zobrazuje oba:
    - Sequence number vo formáte LE: 21 (00000000 00010101)
    - Sequence number vo formáte BE: 5376 (00010101 00000000)

ICMP echo reply – z Wiresharku



```
No.  Time  Source          Destination     Protocol Length Info
91   18.192.168.100.3     62.168.125.181  ICMP     74     Echo (ping) request  id=0x0001, seq=
92   18.62.168.125.181    192.168.100.3   ICMP     74     Echo (ping) reply    id=0x0001, seq=

Frame 92: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0
Ethernet II, Src: HuaweiTe_be:0b:27 (fc:e3:3c:be:0b:27), Dst: IntelCor_e7:0e:37 (d0:7
Internet Protocol Version 4, Src: 62.168.125.181 (62.168.125.181), Dst: 192.168.100.3
Internet Control Message Protocol
  Type: 0 (Echo (ping) reply)
  Code: 0
  Checksum: 0x5546 [correct]
  Identifier (BE): 1 (0x0001)
  Identifier (LE): 256 (0x0100)
  Sequence number (BE): 21 (0x0015)
  Sequence number (LE): 5376 (0x1500)
  [Request frame: 91]
  [Response time: 6.574 ms]

0000  d0 7e 35 e7 0e 37 fc e3  3c be 0b 27 08 00 45 00   .~5..7.. <..'..E.
0010  00 3c 3e 03 00 00 3a 01  61 b5 3e a8 7d b5 c0 a8   .<>...:. a.>.}...
0020  64 03 00 00 55 46 00 01  00 15 61 62 63 64 65 66   d...UF.. ..abcdef
0030  67 68 69 6a 6b 6c 6d 6e  6f 70 71 72 73 74 75 76   ghijklmn opqrstuv
0040  77 61 62 63 64 65 66 67  68 69                    wabcdefg hi
```

Testovanie konektivity utilitou ping

- ping loopback (ping 127.0.0.1)
  - Test konfigurácie IPv4, či je IP protokol správne nainštalovaný na danom zariadení
  - Netestuje sa správna konfigurácia nastavení sieťovej karty (adresa, maska, brána)
  - Netestuje sa harvér sieťovej karty, iba implementácia TCP/IP v danom operačnom systéme, či funguje správne.
- ping 192.168.1.1 (príklad)
  - Test konektivity v mojej LAN – ping na niektorého hosta, alebo na bránu
    - Ak brána neodpovedá, ale host z danej LAN áno, over nastavenia – správna IP brány?
      - ak áno, môže byť problém: s daným rozhraním danej brány, alebo brána iba blokuje ICMP správy a inak je funkčná (menej časté)
- Ping www.google.sk
  - Test konektivity do vzdialenej siete
    - Ak je neúspešný, tak skúsiť ping 8.8.8.8 (alebo iná IP v internete)
      - Ak je úspešný, tak je problém zrejme s nastavením/fungovaním DNS servera
      - Ak aj toto nejde, zrejme bude problém s pripojením brány do internetu (alebo niektorý smerovač na ceste do 8.8.8.8 blokuje ICMP správy - menej časté)

Testovanie cesty do cieľa – traceroute (Cisco)
tracert (Windows)

- Táto utilita zobrazí:
  - zoznam všetkých hopov (smerovačov) na ceste od zdroja k cieľu
  - čas za ktorý správa príde do cieľa a naspať – viem detegovať príliš veľké odozvy od smerovačov po ceste d cieľa
- Využíva:
  - pole TTL v IPv4 hlavičke (Hop Limit v IPv6 hlavičke)
  - ICMP správu typu Time exceeded
- Posiela ICMP správy postupne s TTL 1, 2, 3, ...:
  - TTL = 1
    - Odpovie prvý smerovač (na ktorom TTL=0) správou ICMP typu Time exceeded
    - Na základe tejto odpovede utilita tracert zistí IP adresu prvého hopu k cieľovému hostovi
  - TTL = 2
    - Odpovie druhý smerovač (na ktorom klesne TTL=0) ...
    - Atď až do cieľa...
  - Cyklus sa končí buď:
    - dosiahnutím cieľa – vygeneruje ICMP echo reply (alebo port unreachable)
    - alebo ak TTL dosiahne svoje maximum (závisí od OS)

Testovanie cesty k vzdelavanie.uniza.sk

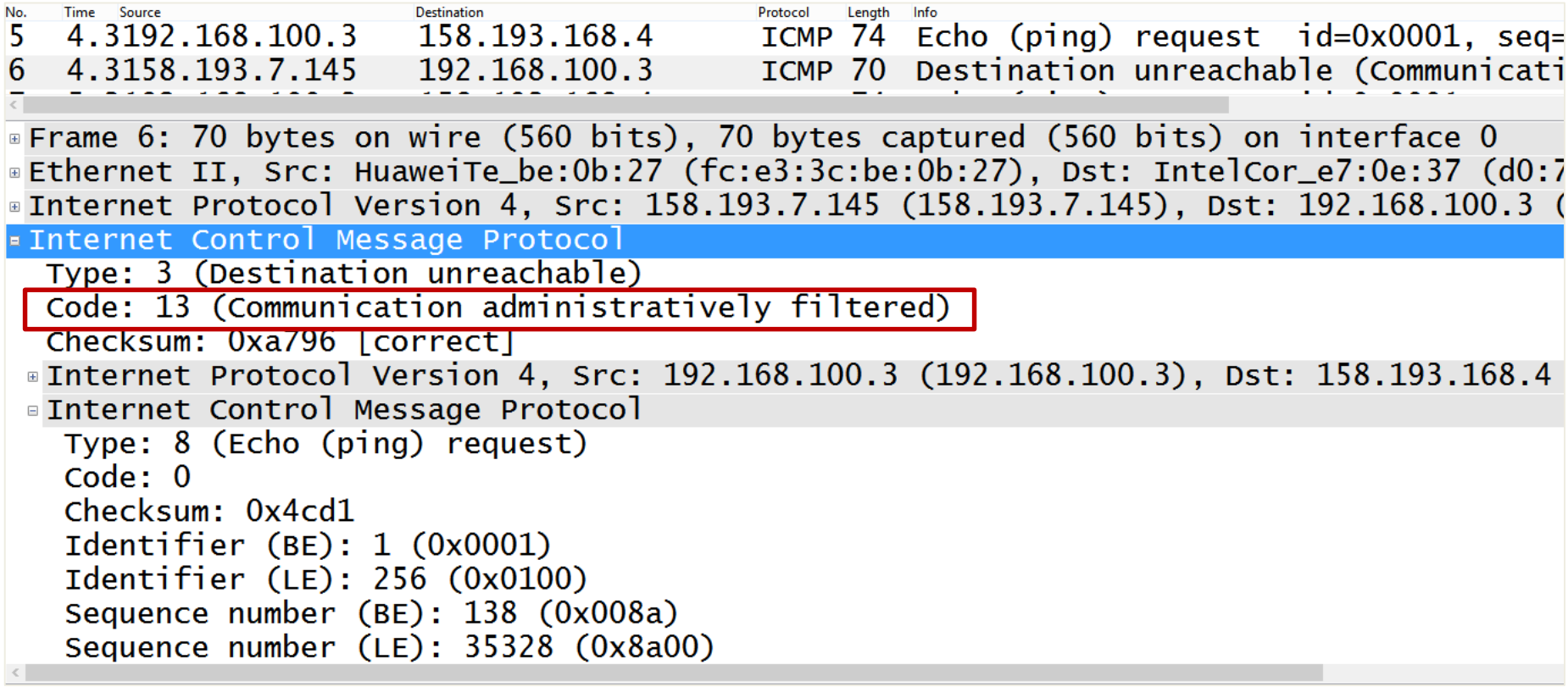
- Test cesty od hosta (192.168.100.1)
  k serveru vzdelavanie.uniza.sk (158.193.168.4)

```
C:\Program Files>tracert vzdelavanie.uniza.sk

Tracing route to vzdelavanie.uniza.sk [158.193.168.4]
over a maximum of 30 hops:

  1     *        1 ms     1 ms  192.168.100.1
  2     7 ms    10 ms     7 ms  95-105-176-1.dynamic.orange.sk [95.105.176.1]
  3     2 ms     3 ms     3 ms  192.168.102.13
  4     7 ms     8 ms     7 ms  213-151-198-190.static.orange.sk [213.151.198.190]
  5     8 ms     7 ms     6 ms  Sanet-gw.six.sk [192.108.148.10]
  6     9 ms    10 ms     9 ms  ZU-Zilina.sanet2.sk [194.160.8.197]
  7     9 ms    10 ms    10 ms  vd-ne-l3-23.net.uniza.sk [158.193.7.121]
  8    13 ms    10 ms    10 ms  vd-ne-l3-27.net.uniza.sk [158.193.7.145]
  9  vd-ne-l3-27.net.uniza.sk [158.193.7.145]  reports: Destination net unreachable.

Trace complete.
```

- Posledný smerovač na ceste, od ktorého sme dostali odpoveď je: 158.193.7.145
- Analýzou ICMP správ vo Wiresharku sa vieme dozvedieť viac

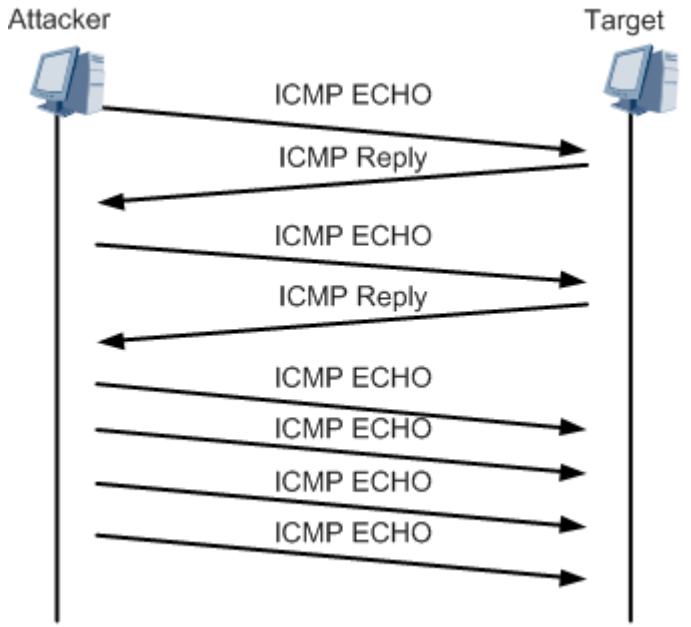
Testovanie cesty k vzdelavanie.uniza.sk

- Posledný smerovač na ceste 158.193.7.145, nám odpovedá ICMP správou typu Destination unreachable, s kódom správy Communication administratively filtered
  - čize ICMP správy sú filtrované adminom
  - t.j. cieľ môže byť dostupný (web server poskytuje obsah), iba smerovač na ceste k nemu blokuje ICMP správy



DoS útok záplavou ICMP správ (ICMP flood)

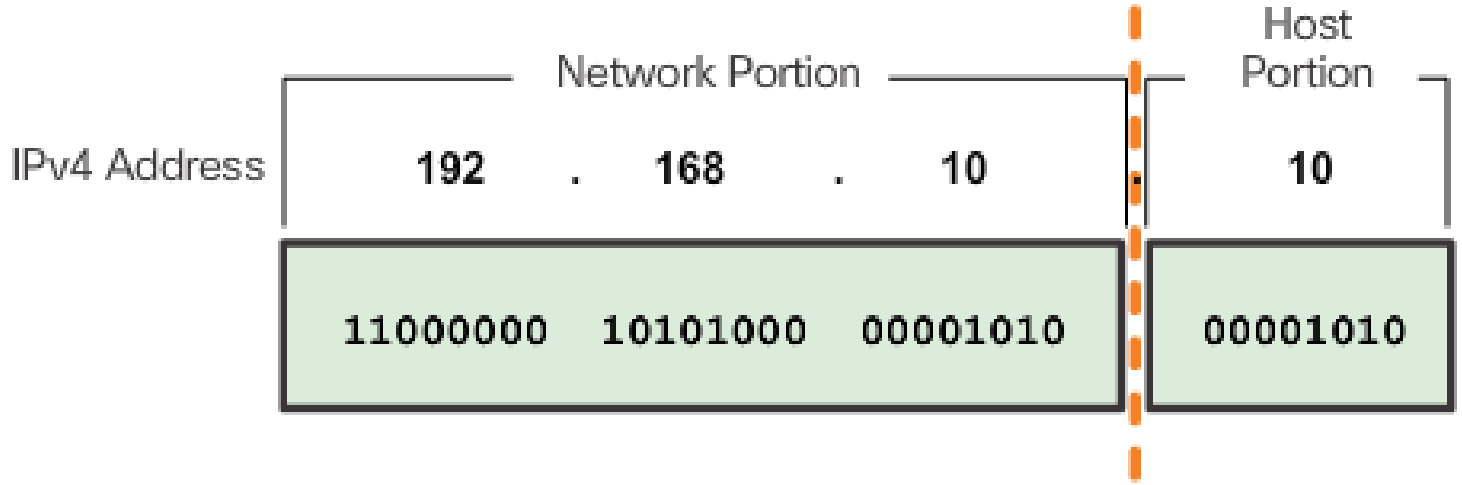
- Útočník sa pokúša zaplaviť svoju obeť ICMP správami typu echo request, ktoré posiela s veľkou intenzitou
- Preto niektoré systémy blokujú správy ICMP request, a neposielajú ICMP reply správy
- Zväčša sa však ICMP správy neblokujú, práve kvôli ich užitočnosti
  - RFC 1122 nariaďuje vždy posielať ICMP reply



Opakovanie – adresovanie v IPv4 a úvod do subsieťovania

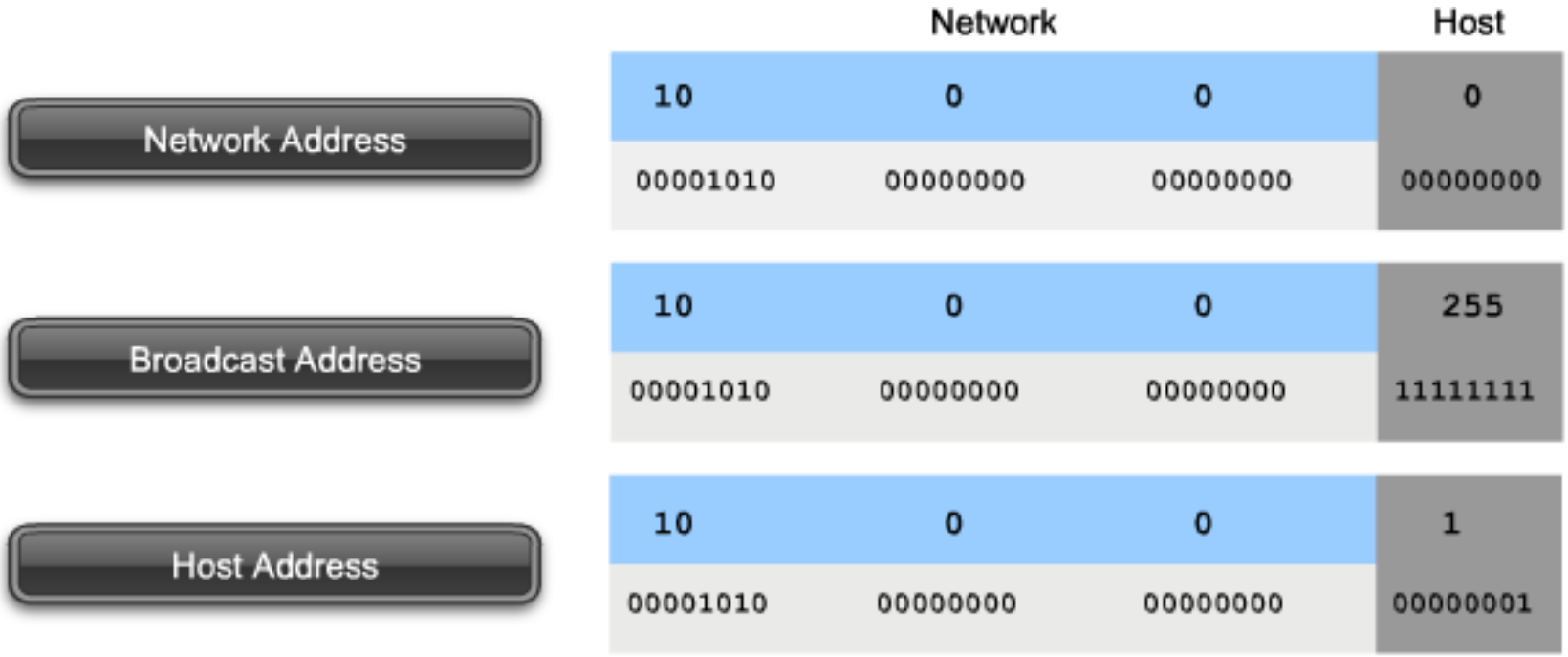
Predčíslie siete a číslo uzla

- IPv4 adresa je 4-bajtové číslo
- Toto číslo je rozdelené na dve časti
  - **Predčíslie siete** (Network Portion)
    - PSČ alebo telefónne čísla (predvoľba) sú pekným príkladom adries, ktoré vyjadrujú príslušnosť objektu do istej spoločnej skupiny príjemcov. Podobne je to s predčíslým siete.
  - **Číslo uzla** (Host Portion)
- Bajt IPv4 adresy sa zvykne nazývať aj oktet
- Hranica medzi predčíslím siete a číslom uzla je v IP adrese pohyblivá
- **Dva uzly sú v tej istej IP sieti práve vtedy, ak majú rovnaké predčíslie siete**

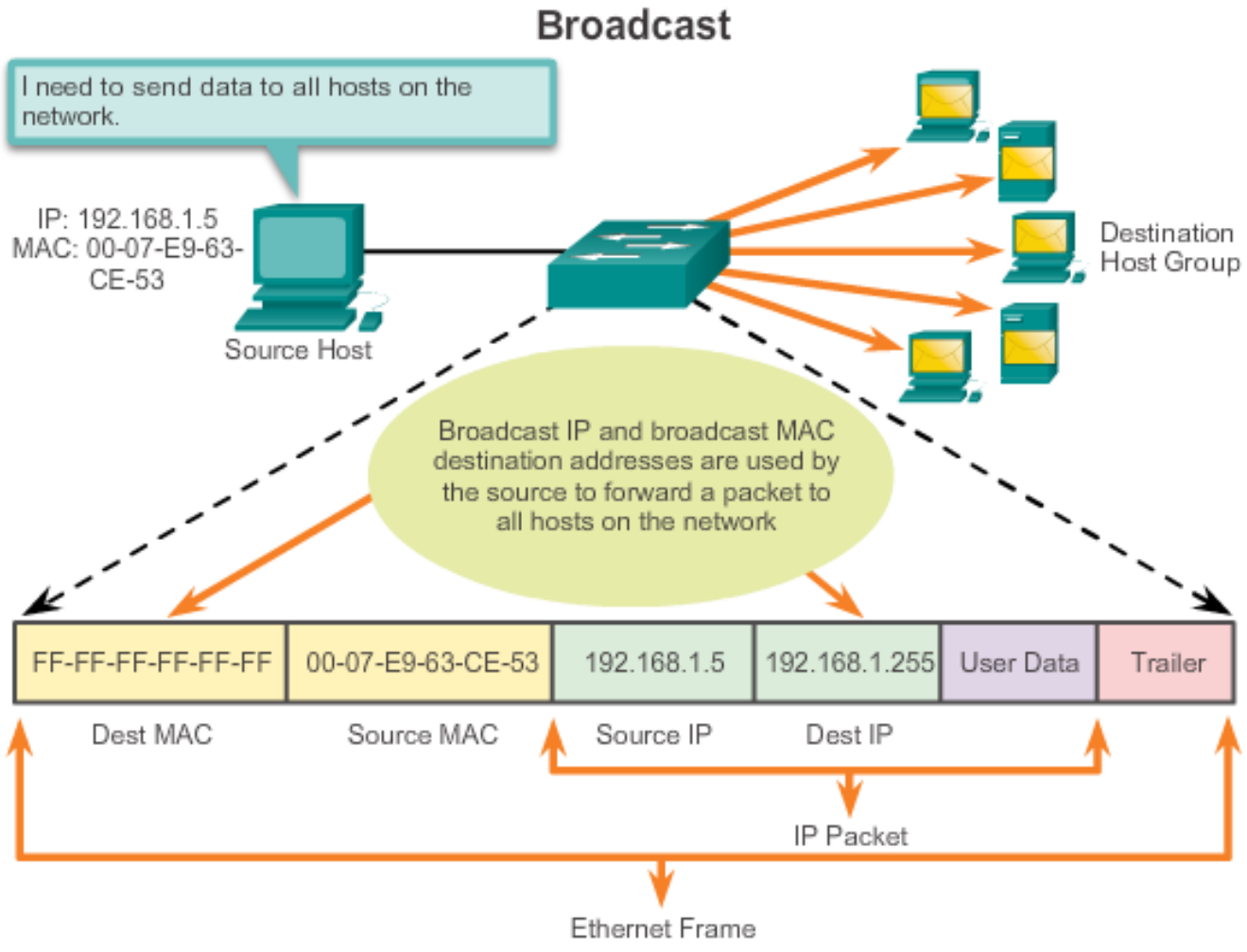


Adresa siete, broadcast, adresa uzla

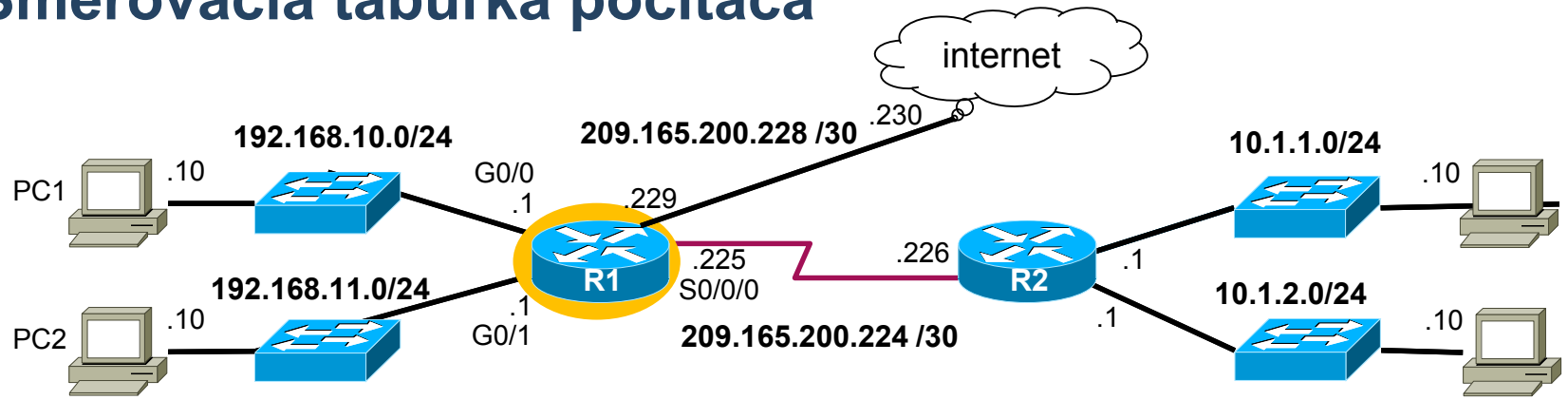
- Podľa toho, čo IP adresa označuje, rozoznávame
  - **Adresu siete**: **Najnižšia** adresa s daným predčíslím, označuje sieť ako celok (predčíslie sa doplní **nulami** do 32 bitov)
  - **Broadcastovú adresu**: **Najvyššia** adresa s daným predčíslím, počúva na nej každá stanica v danej sieti (predčíslie sa doplní **jednotkami** do 32 bitov)
  - **Adresu uzla**: **Každá iná** adresa s daným predčíslím, označuje konkrétny uzol



Broadcasting v Ethernete (L2)



Smerovacia tabuľka počítača




```
IPv4 Route Table
===========================================================================
Active Router:
Network Destinations        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0    192.168.10.1   192.168.10.10     281
        127.0.0.0        255.0.0.0         On-link       127.0.0.1     306
        127.0.0.1  255.255.255.255         On-link       127.0.0.1     306
  127.255.255.255  255.255.255.255         On-link       127.0.0.1     306
    192.168.10.0    255.255.255.0         On-link   192.168.10.10     281
   192.168.10.10  255.255.255.255         On-link   192.168.10.10     281
  192.168.10.255  255.255.255.255         On-link   192.168.10.10     281
        224.0.0.0        240.0.0.0         On-link       127.0.0.1     306
        224.0.0.0        240.0.0.0         On-link   192.168.10.10     281
  255.255.255.255  255.255.255.255         On-link       127.0.0.1     306
  255.255.255.255  255.255.255.255         On-link   192.168.10.10     281
```

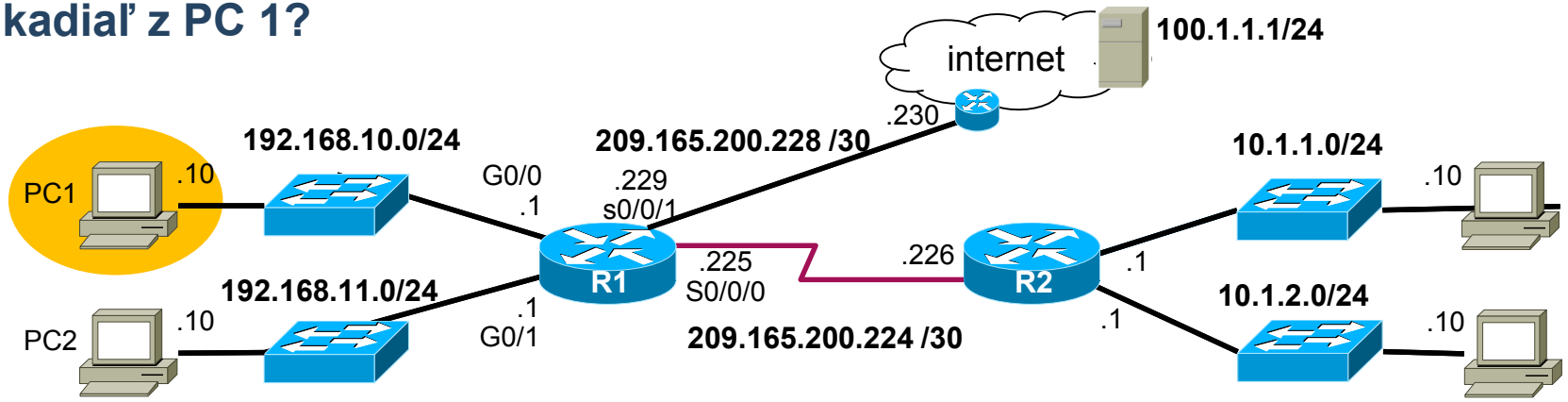
Prehľadávanie smerovacej tabuľky PC
Ako? … Longest prefix match

1. **Usporiadaj si záznamy podľa dĺžky prefixu cieľových sietí zostupne, začni prvým záznamom**
2. **Ak IP adresa cieľa & maska = cieľová sieť, použi daný next hop (via), inak ....**
3. **... ak už si prešiel celú ST (a nebol match), zahoď paket, inak choď na ďalší záznam a zopakuj krok 2.**

```
IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          127.0.0.1  255.255.255.255         On-link       127.0.0.1     306
    127.255.255.255  255.255.255.255         On-link       127.0.0.1     306
      192.168.10.10  255.255.255.255         On-link   192.168.10.10     281
     192.168.10.255  255.255.255.255         On-link   192.168.10.10     281
    255.255.255.255  255.255.255.255         On-link       127.0.0.1     306
    255.255.255.255  255.255.255.255         On-link   192.168.10.10     281
       192.168.10.0    255.255.255.0         On-link   192.168.10.10     281
          127.0.0.0        255.0.0.0         On-link       127.0.0.1     306
          224.0.0.0        240.0.0.0         On-link       127.0.0.1     306
          224.0.0.0        240.0.0.0         On-link   192.168.10.10     281
            0.0.0.0          0.0.0.0    192.168.10.1   192.168.10.10     281
```

Cesta paketu z PC1 do PC2: 10.1.1.10

…kadiaľ z PC 1?



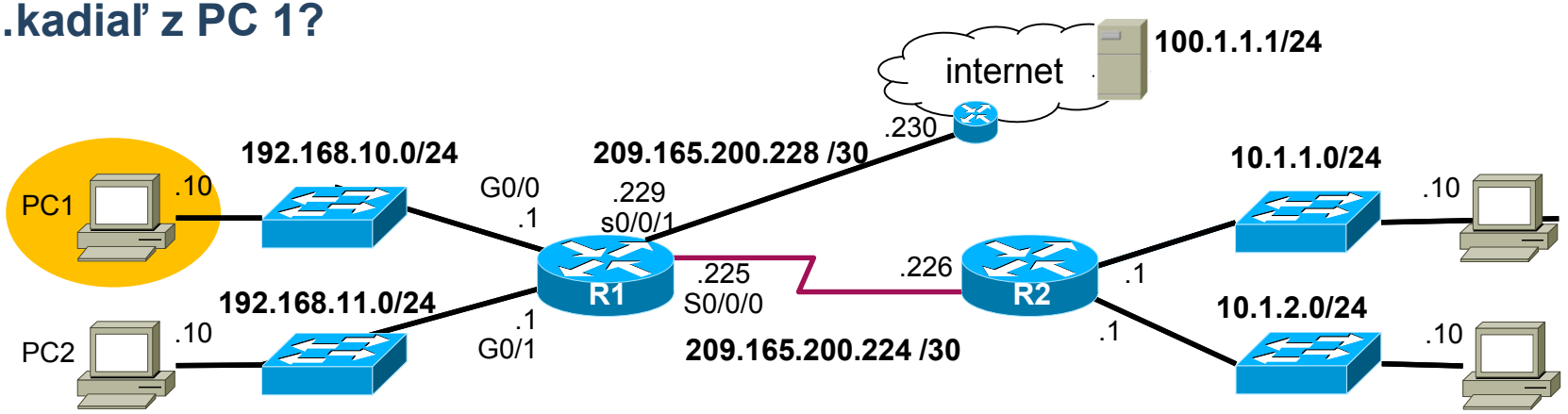
10.1.1.10 & **255.255.255.255** = 10.1.1.10

match? ☹ NIE
(rovnako aj nasledujúcich 5 riadkov)

Network Destinations	Netmask	Gateway	Interface	Metric
127.0.0.1	255.255.255.255	On-link	127.0.0.1	306
127.255.255.255	255.255.255.255	On-link	127.0.0.1	306
192.168.10.10	255.255.255.255	On-link	192.168.10.10	281
192.168.10.255	255.255.255.255	On-link	192.168.10.10	281
255.255.255.255	255.255.255.255	On-link	127.0.0.1	306
255.255.255.255	255.255.255.255	On-link	192.168.10.10	281
192.168.10.0	255.255.255.0	On-link	192.168.10.10	281
127.0.0.0	255.0.0.0	On-link	127.0.0.1	306
224.0.0.0	240.0.0.0	On-link	127.0.0.1	306
224.0.0.0	240.0.0.0	On-link	192.168.10.10	281
0.0.0.0	0.0.0.0	192.168.10.1	192.168.10.10	281

Cesta paketu z PC1 do PC2: 10.1.1.10

…kadiaľ z PC 1?



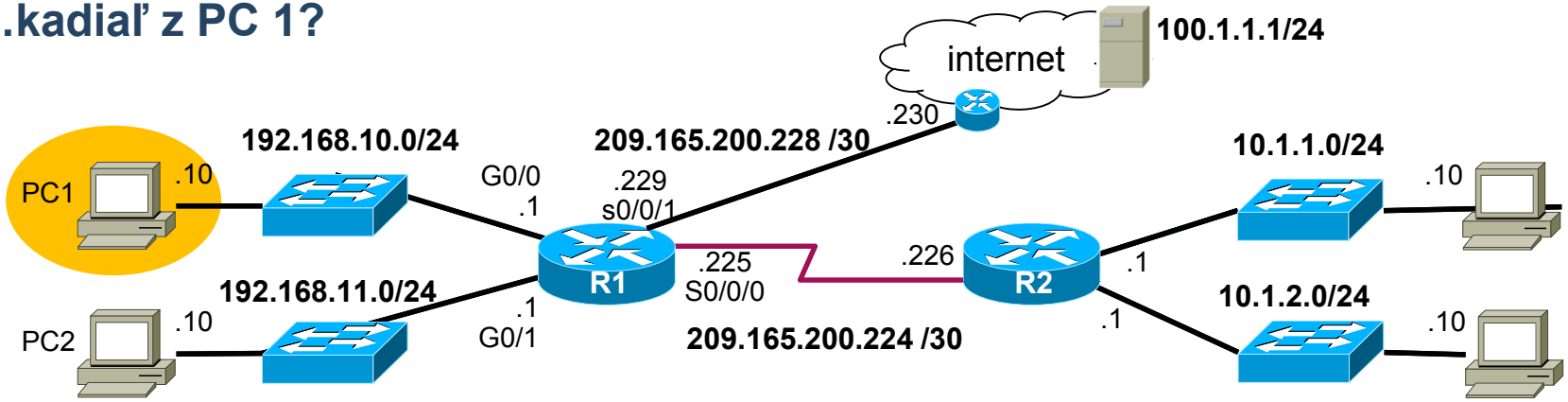
10.1.1.10 & **255.255.255.0** = 10.1.1.0

match? ☹ NIE

Network Destinations	Netmask	Gateway	Interface	Metric
127.0.0.1	255.255.255.255	On-link	127.0.0.1	306
127.255.255.255	255.255.255.255	On-link	127.0.0.1	306
192.168.10.10	255.255.255.255	On-link	192.168.10.10	281
192.168.10.255	255.255.255.255	On-link	192.168.10.10	281
255.255.255.255	255.255.255.255	On-link	127.0.0.1	306
255.255.255.255	255.255.255.255	On-link	192.168.10.10	281
192.168.10.0	255.255.255.0	On-link	192.168.10.10	281
127.0.0.0	255.0.0.0	On-link	127.0.0.1	306
224.0.0.0	240.0.0.0	On-link	127.0.0.1	306
224.0.0.0	240.0.0.0	On-link	192.168.10.10	281
0.0.0.0	0.0.0.0	192.168.10.1	192.168.10.10	281

Cesta paketu z PC1 do PC2: 10.1.1.10

…kadiaľ z PC 1?



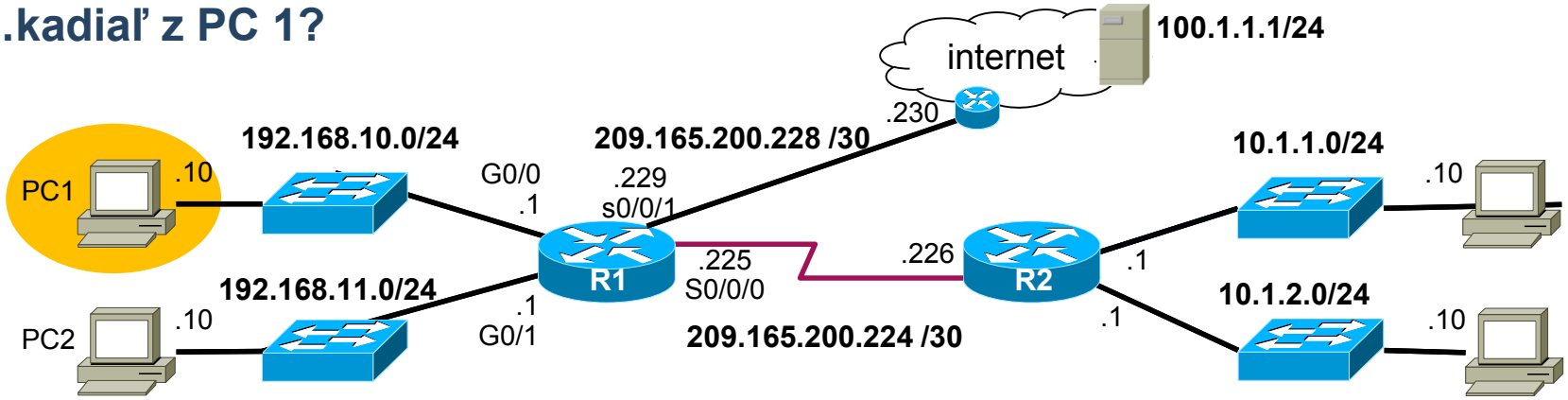
10.1.1.10 & **255.0.0.0** = ? 10.0.0.0

match? ☹ NIE

Network Destinations	Netmask	Gateway	Interface	Metric
127.0.0.1	255.255.255.255	On-link	127.0.0.1	306
127.255.255.255	255.255.255.255	On-link	127.0.0.1	306
192.168.10.10	255.255.255.255	On-link	192.168.10.10	281
192.168.10.255	255.255.255.255	On-link	192.168.10.10	281
255.255.255.255	255.255.255.255	On-link	127.0.0.1	306
255.255.255.255	255.255.255.255	On-link	192.168.10.10	281
192.168.10.0	255.255.255.0	On-link	192.168.10.10	281
127.0.0.0	255.0.0.0	On-link	127.0.0.1	306
224.0.0.0	240.0.0.0	On-link	127.0.0.1	306
224.0.0.0	240.0.0.0	On-link	192.168.10.10	281
0.0.0.0	0.0.0.0	192.168.10.1	192.168.10.10	281

Cesta paketu z PC1 do PC2: 10.1.1.10

…kadiaľ z PC 1?



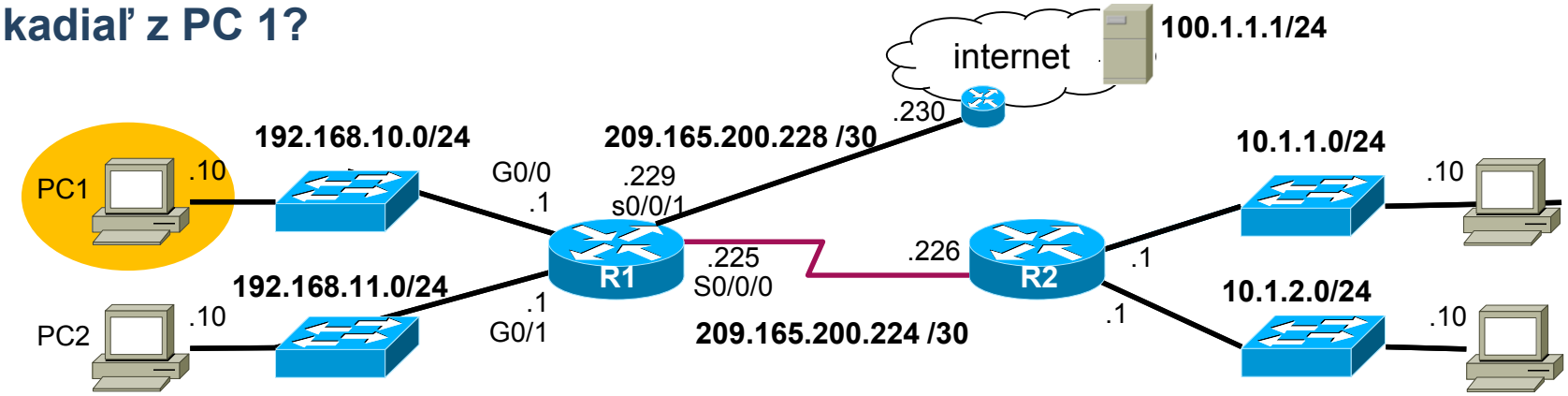
10.1.1.10 & **240.0.0.0** = 0.0.0.0 match? ☹ NIE (podobne aj ďalší riadok)

00001010.1.1.10 & **11110000.0.0.0** = 00000000.0.0.0

Network Destinations	Netmask	Gateway	Interface	Metric
127.0.0.1	255.255.255.255	On-link	127.0.0.1	306
127.255.255.255	255.255.255.255	On-link	127.0.0.1	306
192.168.10.10	255.255.255.255	On-link	192.168.10.10	281
192.168.10.255	255.255.255.255	On-link	192.168.10.10	281
255.255.255.255	255.255.255.255	On-link	127.0.0.1	306
255.255.255.255	255.255.255.255	On-link	192.168.10.10	281
192.168.10.0	255.255.255.0	On-link	192.168.10.10	281
127.0.0.0	255.0.0.0	On-link	127.0.0.1	306
224.0.0.0	240.0.0.0	On-link	127.0.0.1	306
224.0.0.0	240.0.0.0	On-link	192.168.10.10	281
0.0.0.0	0.0.0.0	192.168.10.1	192.168.10.10	281

Cesta paketu z PC1 do PC2: 10.1.1.10

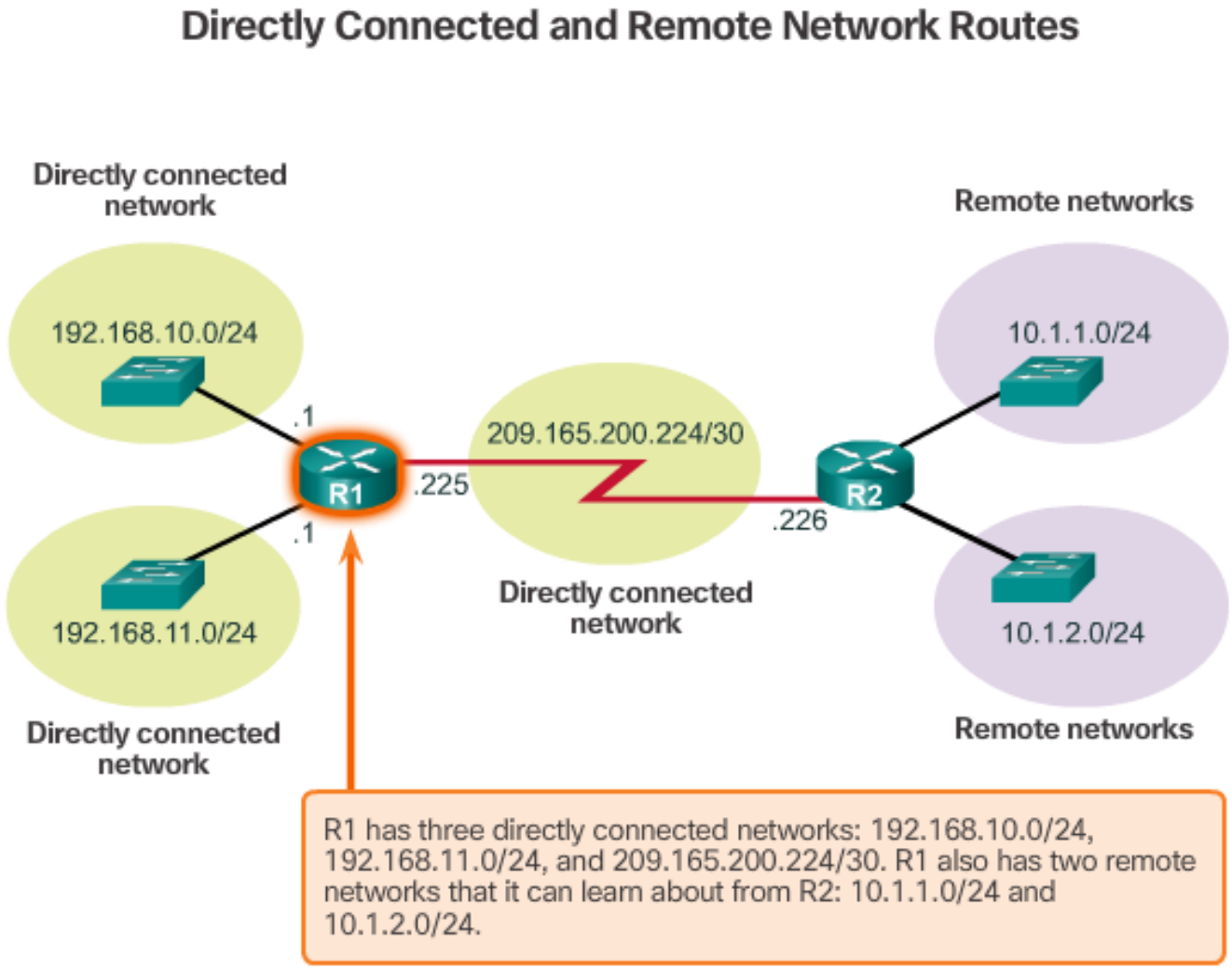
…kadiaľ z PC 1?



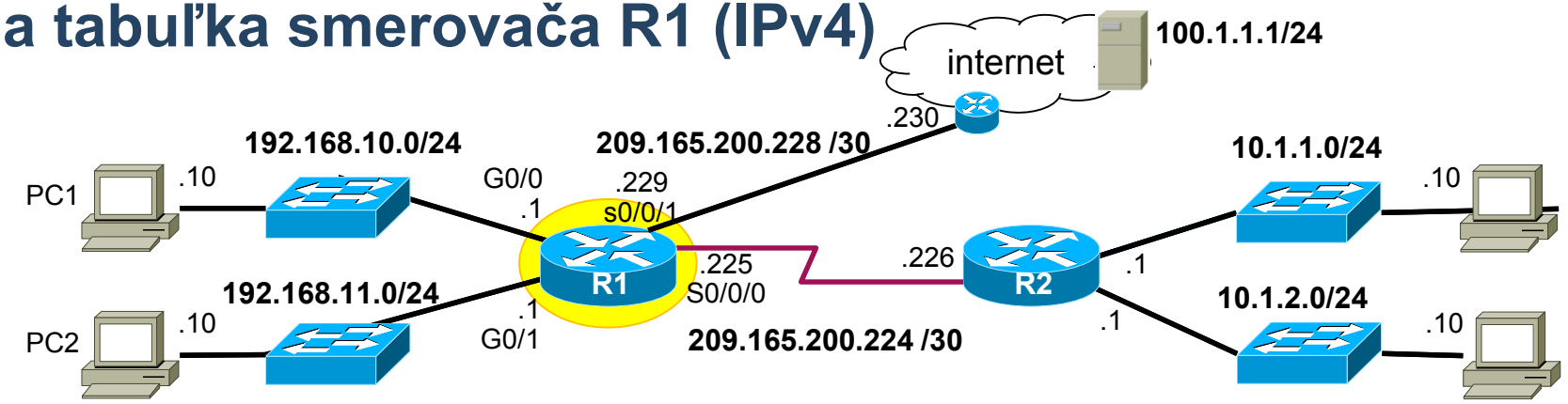
10.1.1.10 & **0.0.0.0** = 0.0.0.0 match? ☺ ÁNO next hop?

Network Destinations	Netmask	Gateway	Interface	Metric
127.0.0.1	255.255.255.255	On-link	127.0.0.1	306
127.255.255.255	255.255.255.255	On-link	127.0.0.1	306
192.168.10.10	255.255.255.255	On-link	192.168.10.10	281
192.168.10.255	255.255.255.255	On-link	192.168.10.10	281
255.255.255.255	255.255.255.255	On-link	127.0.0.1	306
255.255.255.255	255.255.255.255	On-link	192.168.10.10	281
192.168.10.0	255.255.255.0	On-link	192.168.10.10	281
127.0.0.0	255.0.0.0	On-link	127.0.0.1	306
224.0.0.0	240.0.0.0	On-link	127.0.0.1	306
224.0.0.0	240.0.0.0	On-link	192.168.10.10	281
0.0.0.0	**0.0.0.0**	**192.168.10.1**	**192.168.10.10**	**281**

Siete z pohľadu smerovača R1



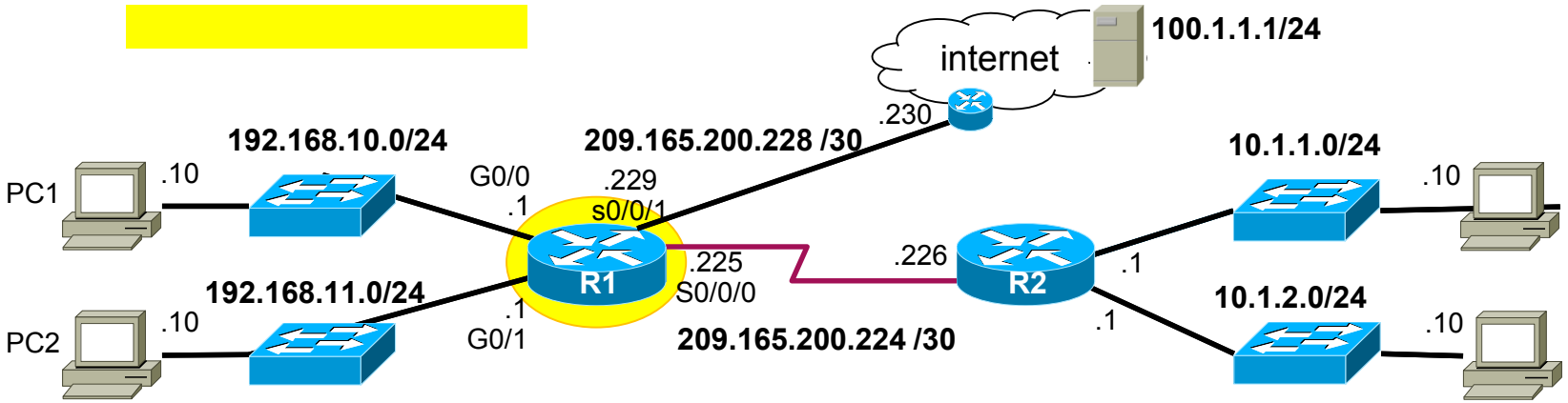
Smerovacia tabuľka smerovača R1 (IPv4)




```
R1#show ip route
[ ... Časť výpisu odstránená ... ]

     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
D       10.1.1.0/24 [90/2170112] via 209.165.200.226, 00:00:05, Se0/0/0
D       10.1.2.0/24 [90/2170112] via 209.165.200.226, 00:00:05, Se0/0/0
     192.168.10.0/24 is variably subnetted, 2 subnets, 3 masks
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0
L       192.168.10.1/32 is directly connected, GigabitEthernet0/0
     192.168.11.0/24 is variably subnetted, 2 subnets, 3 masks
C       192.168.11.0/24 is directly connected, GigabitEthernet0/1
L       192.168.11.1/32 is directly connected, GigabitEthernet0/1
     209.165.200.0/24 is variably subnetted, 2 subnets, 3 masks
C       209.165.200.224/30 is directly connected, Serial0/0/0
L       209.165.200.225/32 is directly connected, Serial0/0/0
C       209.165.200.228/30 is directly connected, Serial0/0/1
L       209.165.200.229/32 is directly connected, Serial0/0/1
S*   0.0.0.0/0 [1/0] via 200.1.1.1, Serial 0/0/1
                       is directly connected, Serial 0/0/1
```

Prehľadávanie smerovacej tabuľky (ST)

Ako? … Longest prefix match

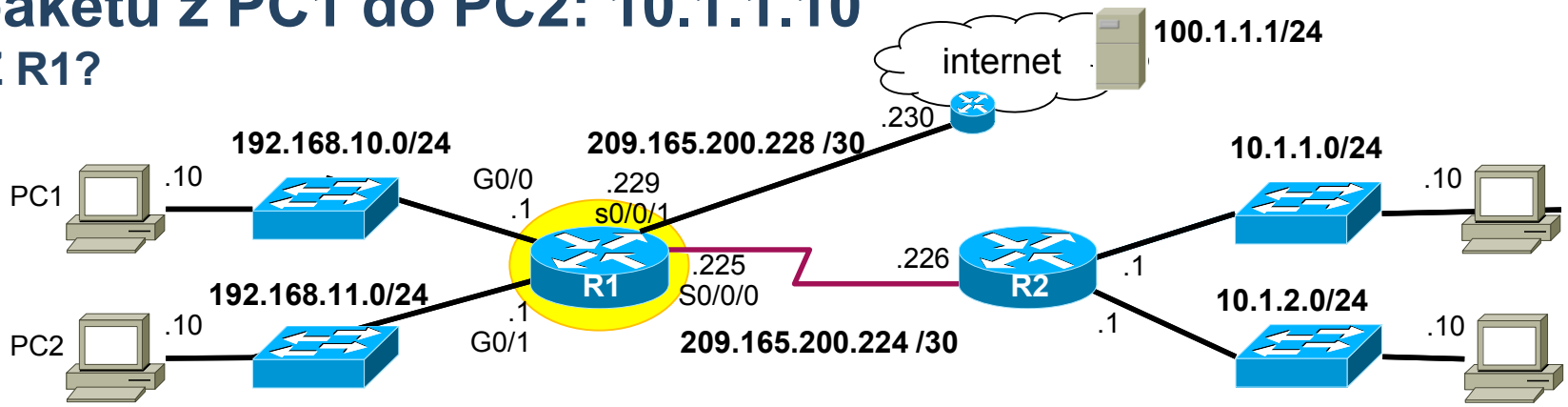


1. Usporiadaj si záznamy podľa dĺžky prefixu cieľových sietí zostupne, začni prvým záznamom
2. Ak IP adresa cieľa & maska = cieľová sieť, použi daný next hop (via), inak ....
3. ... ak už si prešiel celú ST (a nebol match), zahoď paket, inak choď na ďalší záznam a zopakuj krok 2.

```
C       209.165.200.224/30 is directly connected, Serial0/0/0
C       209.165.200.228/30 is directly connected, Serial0/0/1
D       10.1.1.0/24 [90/2170112] via 209.165.200.226, 00:00:05, Se0/0/0
D       10.1.2.0/24 [90/2170112] via 209.165.200.226, 00:00:05, Se0/0/0
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0
C       192.168.11.0/24 is directly connected, GigabitEthernet0/1
S*   0.0.0.0/0 [1/0] via 200.1.1.1, Serial 0/0/1
```

Cesta paketu z PC1 do PC2: 10.1.1.10

…kadiaľ Z R1?



Dĺžka prefixu /30 je aká maska? 11111111.11111111.11111111.11111100

255. 255. 255. 252

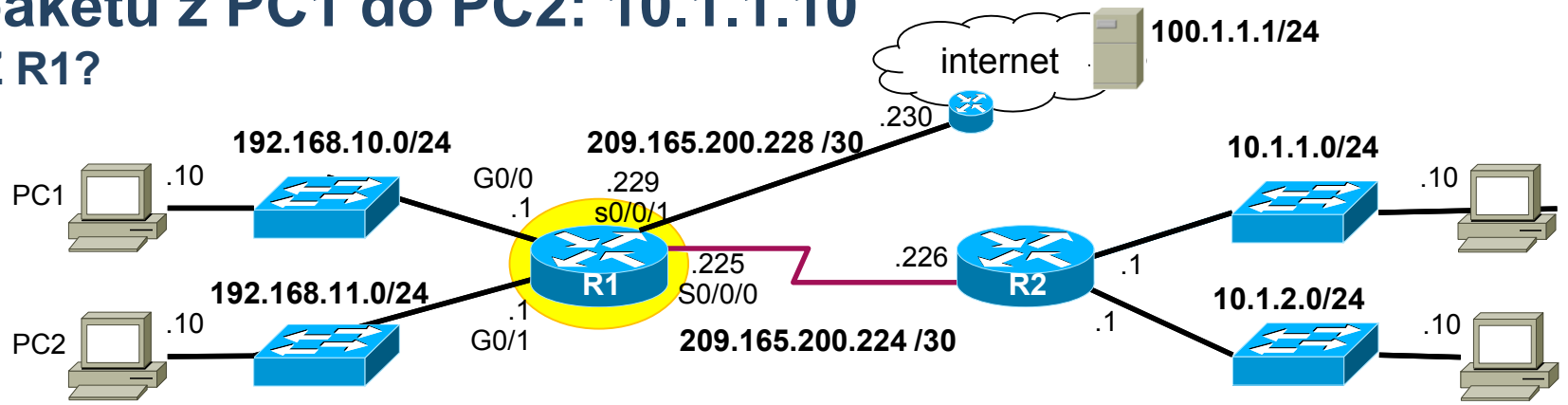
10.1.1.10 & **255.255.255.252** = 10.1.1.8 match? ☹ NIE (podobne aj ďalší záznam v ST)

10.1.1.00001010 & **255.255.255.11111100** = 10.1.1.00001000

```
C       209.165.200.224/30 is directly connected, Serial0/0/0
C       209.165.200.228/30 is directly connected, Serial0/0/1
D       10.1.1.0/24 [90/2170112] via 209.165.200.226, 00:00:05, Se0/0/0
D       10.1.2.0/24 [90/2170112] via 209.165.200.226, 00:00:05, Se0/0/0
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0
C       192.168.11.0/24 is directly connected, GigabitEthernet0/1
S*   0.0.0.0/0 [1/0] via 200.1.1.1, Serial 0/0/1
```

Cesta paketu z PC1 do PC2: 10.1.1.10

…kadiaľ Z R1?



Dĺžka prefixu /30 je aká maska? 11111111.11111111.11111111.11111100

255. 255. 255. 252

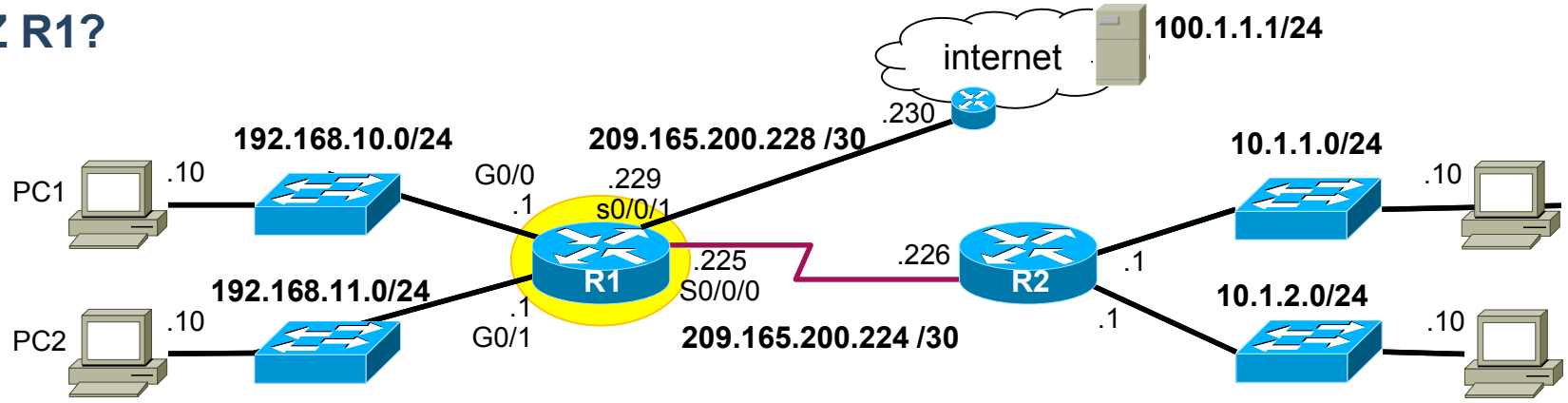
10.1.1.10 & **255.255.255.0** = 10.1.1.0 match? ☺ ÁNO

next hop?

```
C       209.165.200.224/30 is directly connected, Serial0/0/0
C       209.165.200.228/30 is directly connected, Serial0/0/1
D       10.1.1.0/24 [90/2170112] via 209.165.200.226, 00:00:05, Se0/0/0
D       10.1.2.0/24 [90/2170112] via 209.165.200.226, 00:00:05, Se0/0/0
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0
```

Cesta paketu z PC1 do PC2: 10.1.1.10

…kadiaľ Z R1?



Zrýchlime to... na ktorom riadku sa prehľadávanie zastaví?

Dĺžka prefixu /0 je aká maska? 00000000.00000000.00000000.00000000

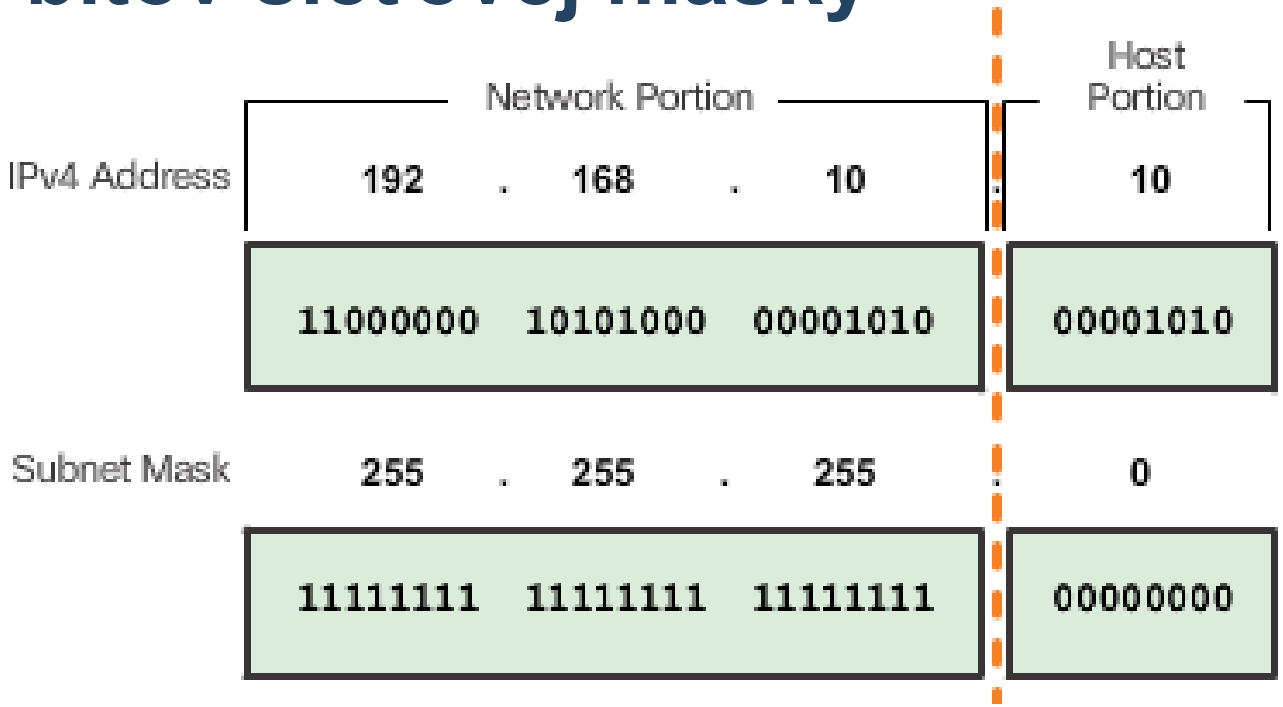
0. 0. 0. 0

100.100.100.100 & **0.0.0.0** = 0.0.0.0 match? ☺ ÁNO

next hop?

```
C        209.165.200.224/30 is directly connected, Serial0/0/0
C        209.165.200.228/30 is directly connected, Serial0/0/1
D        10.1.1.0/24 [90/2170112] via 209.165.200.226, 00:00:05, Se0/0/0
D        10.1.2.0/24 [90/2170112] via 209.165.200.226, 00:00:05, Se0/0/0
C        192.168.10.0/24 is directly connected, GigabitEthernet0/0
C        192.168.11.0/24 is directly connected, GigabitEthernet0/1
S*    0.0.0.0/0 [1/0] via 200.1.1.1, Serial 0/0/1
```

Význam bitov sieťovej masky

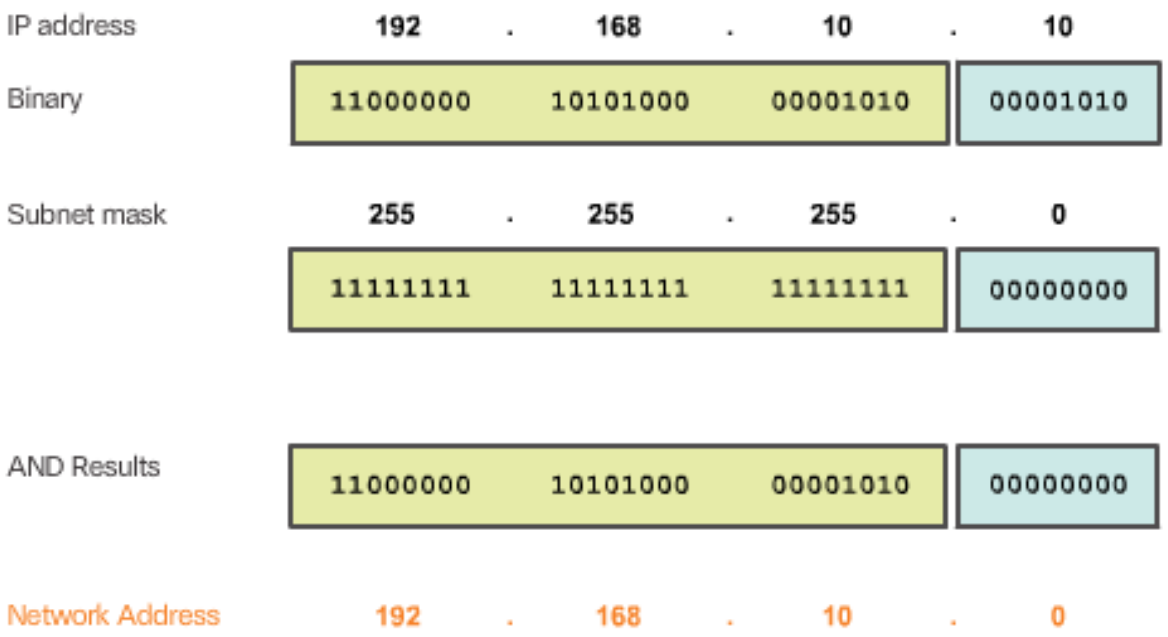


- Maska je postupnosť 32 bitov v tvare 1...10....0, t.j. súvislý blok bitov nastavených na 1 nasledovaný súvislým blokom bitov nastavených na 0
- Ak je *n*-ty bit v maske nastavený na
  - **1**: príslušný *n*-ty bit v IP adrese patrí do **predčíslia siete**
  - **0**: príslušný *n*-ty bit v IP adrese patrí do **čísla stanice**
- IP adresu rozdeľuje na predčíslie siete a číslo počítača hranica medzi blokom bitov nastavených na 1 a blokom bitov nastavených 0 v maske

Binárne AND

- Binárne AND je porovnanie dvoch bitov
- Binárnym ANDom IP adresy so sieťovou maskou získame adresu siete, do ktorej zariadenie s danou IP adresou patrí

1 AND 1 = 1
0 AND 1 = 0
0 AND 0 = 0
1 AND 0 = 0



Dĺžka prefixu

- Skrátený tvar zápisu sieťovej masky
  (tzv. CIDR zápis – Classless Interdomain Routing)
- Počet jednotiek v sieťovej maske
- Hodnota sa píše za lomítko “/”
- Príklady sieťových masiek:

Subnet Mask	32-bit Address	Prefix Length
255.0.0.0	11111111.00000000.00000000.00000000	/8
255.255.0.0	11111111.11111111.00000000.00000000	/16
255.255.255.0	11111111.11111111.11111111.00000000	/24
255.255.255.128	11111111.11111111.11111111.10000000	/25
255.255.255.192	11111111.11111111.11111111.11000000	/26
255.255.255.224	11111111.11111111.11111111.11100000	/27
255.255.255.240	11111111.11111111.11111111.11110000	/28
255.255.255.248	11111111.11111111.11111111.11111000	/29
255.255.255.252	11111111.11111111.11111111.11111100	/30

Možné hodnoty pre oktety v sieťovej maske (iné nie sú):

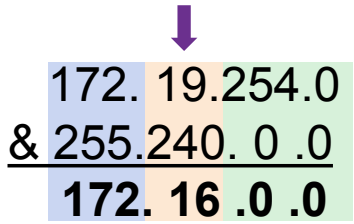
0 128 192 224 240 248 252 254 255

Ako rýchlo počítať adresy sietí?

Pre rýchly výpočet binárneho AND medzi adresou uzla a maskou si všimnime tieto fakty:

- Bajt masky (M) môže nadobúdať len hodnoty:
  0 (**00000000**)
  128 (**10000000**)
  192 (**11000000**)
  224 (**11100000**)
  240 (**11110000**)
  248 (**11111000**)
  252 (**11111100**)
  254 (**11111110**)
  255 (**11111111**)
- Maska obsahuje najviac jeden bajt, ktorý nie je ani 0, ani 255. Napr. 255.240.0.0
- Ľahko určíme AND medzi:
  - X & 255 = X Napr: 172 & 255 = 172
  - X & 0 = 0 Napr: 254 & 0 = 0

  (X je bajt z adresy uzla)



- Ak M je bajt masky, ktorého hodnota je rôzna od 0 a 255, potom X & M sa správa ako zaokrúhľovanie:
  - Isté horné bity v M sú nastavené na **1**, zvyšné na **0**. Napr. **240** (**11110000**)
    2^4=16
  - X & M prenesie z X do výsledku tie bity, ktoré sú v M nastavené na **1**, a vynuluje zvyšné bity = zaokrúhli X nadol na násobok istého rádu čísla 2, konkrétne **na násobok čísla (256-M).** Napr. 256-240=16
  - Vypočítať X & M z hlavy je teda jednoduché:
    - Zaokrúhliť X nadol na najbližší násobok čísla (256-M). Napr. 16*1<=19

Vyhradené rozsahy IP adries

- Niektoré rozsahy IP adries sú vyhradené pre špeciálne použitie (RFC 5735)
- Privátne adresy podľa RFC 1918
  - Tri rozsahy: **10.0.0.0/8**, **172.16.0.0/12**, **192.168.0.0/16**
  - Adresy, ktoré je možné ľubovoľne používať vo vlastnej sieti
  - Pri komunikácii s internetom je ich potrebné preložiť na oficiálne verejné adresy pomocou technológie NAT
- Tzv. link-local adresy podľa RFC 3927
  - Rozsah **169.254.0.0/16**
  - Rozsah používaný OS Windows pre automatickú konfiguráciu IP adresy bez DHCP
  - Adresy je možné použiť iba na komunikáciu v jednej spoločnej sieti
- Tzv. loopback network podľa RFC 1122
  - Rozsah **127.0.0.0/8**, špeciálne IP adresa 127.0.0.1
  - Interná IP adresa, ktorú má každý počítač s podporou IP
  - Pomocou tejto siete môže počítač komunikovať cez IP sám so sebou
- Tzv. Test-NET rozsahy podľa RFC 5737
  - Tri rozsahy: **192.0.2.0/24**, **198.51.100.0/24**, **203.0.113.0/24**
  - Určené pre použitie v dokumentoch, príkladoch, návodoch bez rizika konfliktu s existujúcimi skutočnými sieťami

IPv4 adresy na špeciálne využitie

- Default route
  - 0.0.0.0
- Adresa siete
  - Napr. 158.193.152.0/24 alebo 128.10.10.128/26
- Broadcast
  - Napr. 158.193.152.255 alebo 128.10.10.191

  - Môžu byt použité ako zdrojové adresy v hlavičke IP paketu?
  - Môžu byt použité ako cieľové adresy v hlavičke IP paketu?

Classfull addressing (podľa tried)

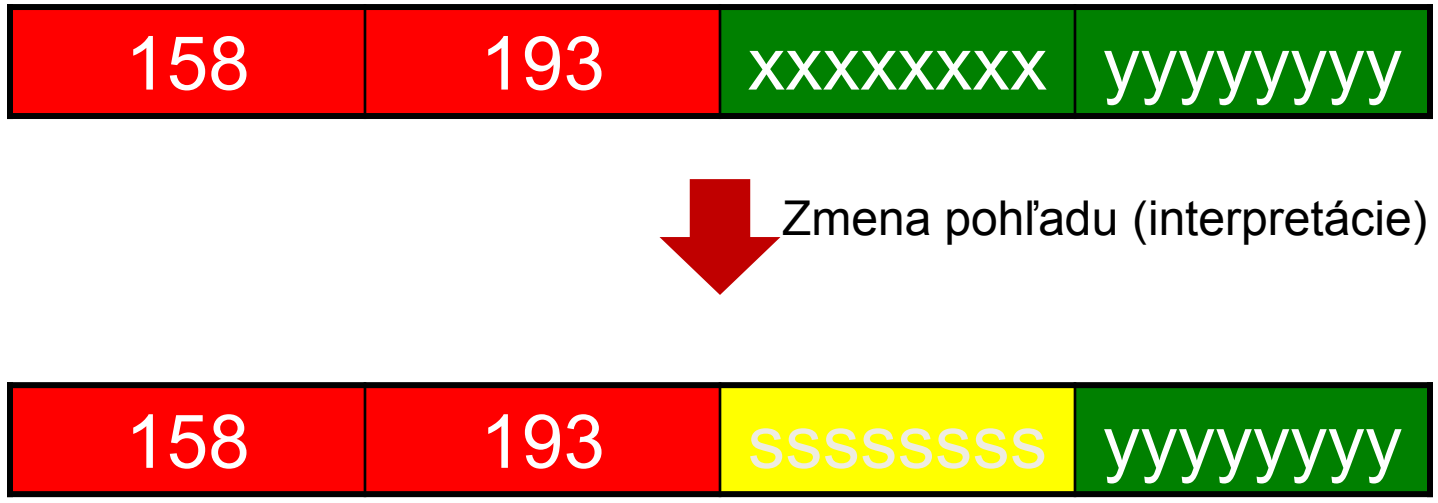
Address Class	1st octet range (decimal)	1st octet bits (green bits do not change)	Network(N) and Host(H) parts of address	Default subnet mask (decimal and binary)	Number of possible networks and hosts per network
A	1-127**	00000000-01111111	N.H.H.H	255.0.0.0	128 nets (2^7) 16,777,214 hosts per net (2^24-2)
B	128-191	10000000-10111111	N.N.H.H	255.255.0.0	16,384 nets (2^14) 65,534 hosts per net (2^16-2)
C	192-223	11000000-11011111	N.N.N.H	255.255.255.0	2,097,150 nets (2^21) 254 hosts per net (2^8-2)
D	224-239	11100000-11101111	NA (multicast)		
E	240-255	11110000-11111111	NA (experimental)		

Classless addressing

- Prax však ukázala, že aj delenie na triedy je príliš hrubé
  - Správcovia IP rozsahov prideľovali celé bloky adries podľa triedy, teda zákazník mohol dostať iba celú sieť typu A, B alebo C
  - Ak sa priestor triedy minul, nebolo možné alokovať rovnako veľkú sieť z inej triedy (napr. sieť o 65536 adresách z triedy B nebolo možné prideliť z priestoru adries v triede A, lebo každá adresa triedy A vyjadrovala príslušnosť do siete o veľkosti 16777216 adries)
- Súčasný prístup: zrušenie tried, tzv. **classless addressing**
  - Predčíslie siete v IP adrese sa už neurčuje podľa príslušnosti adresy do niektorej triedy, ale použitím pomocnej kvantity: tzv. sieťovej masky (netmask)
  - Sieťová maska je 4B hodnota podobne ako IP adresa
  - Vyčleňuje predčíslie siete z IP adresy

Tvorba podsietí na hraniciach bajtov

- Príklad:
  - Provider nám pridelil B blok **158.193.0.0**
    - Všetci príjemcovia tvaru 158.193.X.Y
    - Jedna sieť (jedno predčíslie), 2^{16} = 65 536 adries v sieti, 2 vyhradené (N, B)
  - K pôvodnému predčísliu my pričleníme ďalší octet (bajt) – z host part
    - Vznikne 2^8 = 256 podsietí, v každej 256 adries, 2 adresy v každej sieti vyhradené, prvá a posledná podsieť niekedy tiež vyhradené
    - Pričlenenie ďalšieho bajtu sa realizuje vhodnou konfiguráciou zariadení



Tvorba podsietí na hraniciach bajtov

Subnetting Networks on the Octet Boundary

Prefix Length	Subnet Mask	Subnet Mask in Binary (n = network, h = host)	# of hosts
/8	255.0.0.0	nnnnnnnn.hhhhhhhh.hhhhhhhh.hhhhhhhh 11111111.00000000.00000000.00000000	16,777,214
/16	255.255.0.0	nnnnnnnn.nnnnnnnn.hhhhhhhh.hhhhhhhh 11111111.11111111.00000000.00000000	65,534
/24	255.255.255.0	nnnnnnnn.nnnnnnnn.nnnnnnnn.hhhhhhhh 11111111.11111111.11111111.00000000	254

Tvorba podsietí na hraniciach bajtov

- Subsieťovanie pôvodnej siete 10.0.0.0/8 na druhom bajte: **10.x.0.0/16**

Subnet Address (256 Possible Subnets)	Host Range (65,534 possible hosts per subnet)	Broadcast
10.0.0.0/16	10.0.0.1 - 10.0.255.254	10.0.255.255
10.2.0.0/16	10.2.0.1 - 10.2.255.254	10.2.255.255
10.3.0.0/16	10.3.0.1 - 10.3.255.254	10.3.255.255
10.4.0.0/16	10.4.0.1 - 10.4.255.254	10.4.255.255
10.5.0.0/16	10.5.0.1 - 10.5.255.254	10.5.255.255
10.6.0.0/16	10.6.0.1 - 10.6.255.254	10.6.255.255
10.7.0.0/16	10.7.0.1 - 10.7.255.254	10.7.255.255
...	...	...
10.255.0.0/16	10.255.0.1 - 10.255.255.254	10.255.255.255

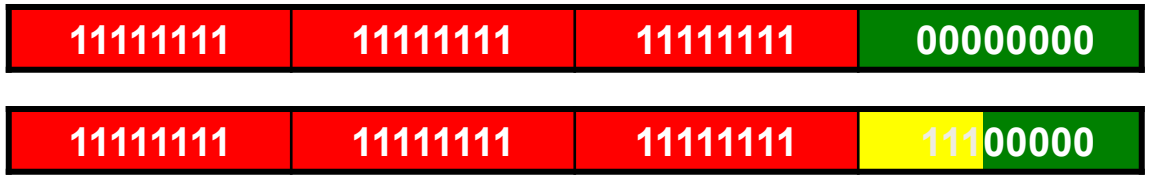
Tvorba podsietí na hraniciach bajtov

- Subsieťovanie pôvodnej siete 10.0.0.0/8 na druhom a treťom bajte: **10.x. x.0/24**

Subnet Address (65,536 Possible Subnets)	Host Range (254 possible hosts per subnet)	Broadcast
10.0.0.0/24	10.0.0.1 – 10.0.0.254	10.0.0.255
10.0.1.0/24	10.0.1.1 – 10.0.1.254	10.0.1.255
10.0.2.0/24	10.0.2.1 – 10.0.2.254	10.0.1.255
…	…	…
10.0.255.0/24	10.0.255.1 – 10.0.255.254	10.0.255.255
10.1.0.0/24	10.1.0.1 – 10.1.0.254	10.1.0.255
10.1.1.0/24	10.1.1.1 – 10.1.1.254	1.1.1.0.255
10.1.2.0/24	10.1.2.1 – 10.1.2.254	10.1.2.0.255
…	…	…
10.100.0.0/24	10.100.0.1 – 10.100.0.254	10.100.0.255
…	…	…
10.255.255.0/24	10.255.255.1 – 10.255.255.254	10.255.255.255

Tvorba podsietí na hraniciach bitov

- K predčísliu, ktoré nám bolo pridelené, pridáme pre identifikovanie našej vlastnej podsiete vhodný počet bitov z pôvodnej host part
  - Počet vzniknutých podsietí: $2^{\text{počet pridaných bitov z host part}}$
- Maska sa predĺži – zväčší sa počet bitov nastavených na 1
- Vzniknuté podsiete budú časťou pôvodnej siete
  - Veľkosť jednej podsiete: $2^{\text{počet zostávajúcich bitov v host part}}$
- Príklad
  - Pôvodná maska bola 255.255.255.0, t.j. pôvodná sieť obsahovala 256 adries
  - Nová maska predĺži predčíslie siete o 3 bity a má hodnotu 255.255.255.224
  - Vzniklo 2^3=8 nových podsietí, v každej je 2^5=32 adries -2
    (2 sú vyhradené pre: sieť a broadcast)



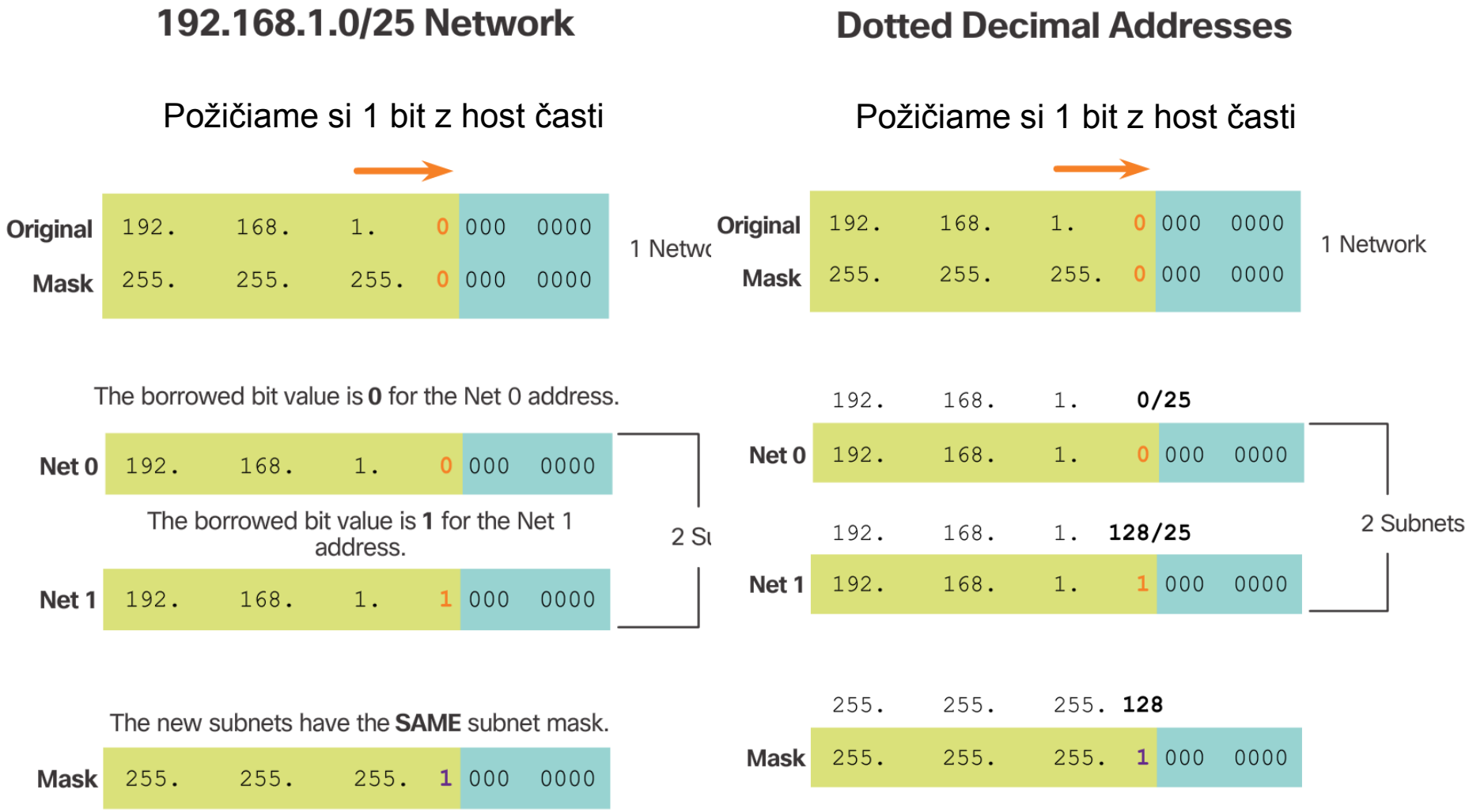
Tvorba podsietí na hraniciach bitov (kdekoľvek)

Subsieťovanie pôvodnej siete s maskou /24 ➡ môžem si požičať nejaký počet bitov zo 4. oktetu (z host part) na adresovanie podsietí (nová maska bude potom dlhšia):

- 1 bit a tak vytvorím 2^1 = 2 podsiete, v každej 2^7 -2 = 126 použiteľných adries
- 2 bity a tak vytvorím 2^2 = 4 podsiete, v každej 2^6 -2 = 62 použiteľných adries
- 3 bity a tak vytvorím 2^3 = 8 podsietí, v každej 2^5 -2 = 30 použiteľných adries
- 4 bity a tak vytvorím 2^4 = 16 podsietí, v každej 2^4 -2 = 14 použiteľných adries
- 5 bitov a tak vytvorím 2^5 = 32 podsietí, v každej 2^3 -2 = 6 použiteľných adries
- 6 bitov a tak vytvorím 2^6 = 64 podsietí, v každej 2^2 -2 = 2 použiteľné adresy

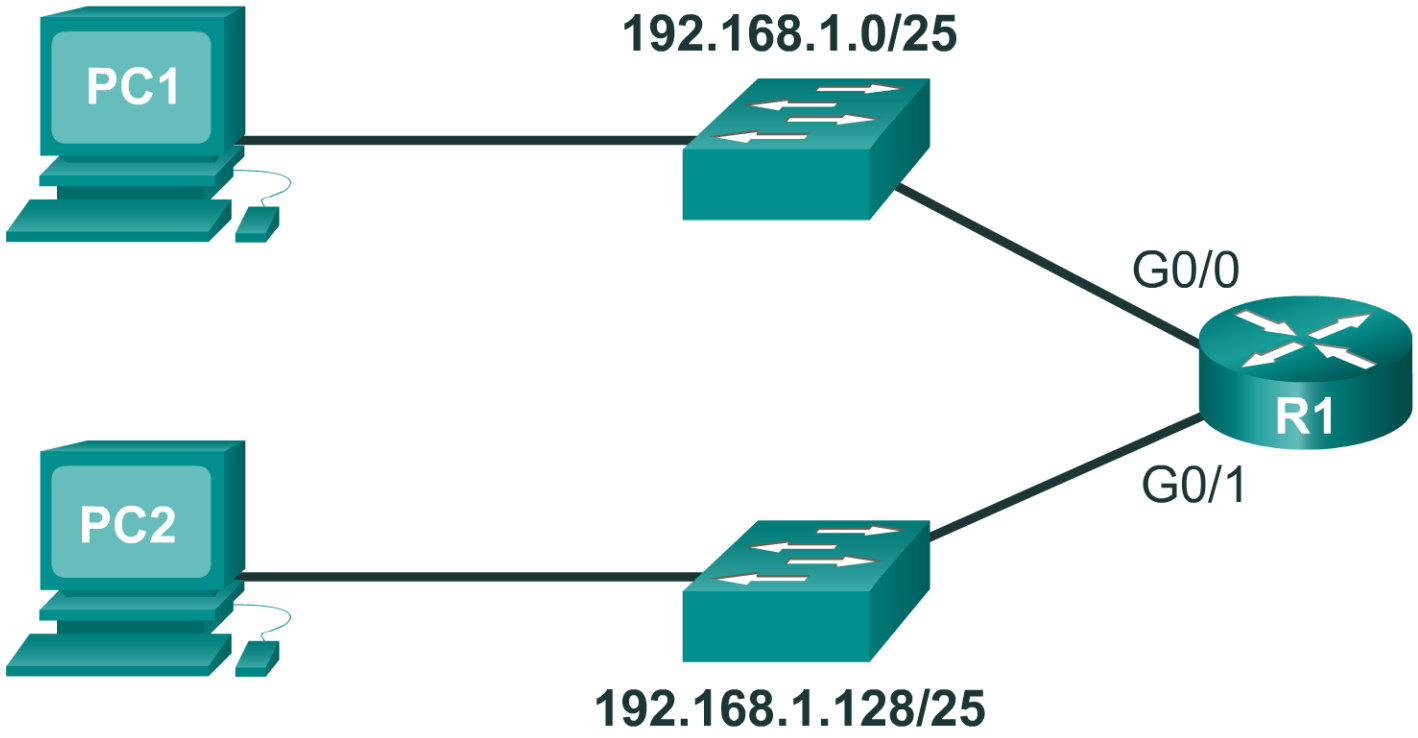
Prefix Length	Subnet Mask	Subnet Mask in Binary (n = network, h = host)	# of subnets	# of hosts
/25	255.255.255.128	nnnnnnnn.nnnnnnnn.nnnnnnnn.nhhhhhhh 11111111.11111111.11111111.10000000	2	126
/26	255.255.255.192	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnhhhhhh 11111111.11111111.11111111.11000000	4	62
/27	255.255.255.224	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnhhhhh 11111111.11111111.11111111.11100000	8	30
/28	255.255.255.240	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnhhhh 11111111.11111111.11111111.11110000	16	14
/29	255.255.255.248	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnhhh 11111111.11111111.11111111.11111000	32	6
/30	255.255.255.252	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnnhh 11111111.11111111.11111111.11111100	64	2

Rozdelenie 192.168.1.0/24 na 2 podsiete

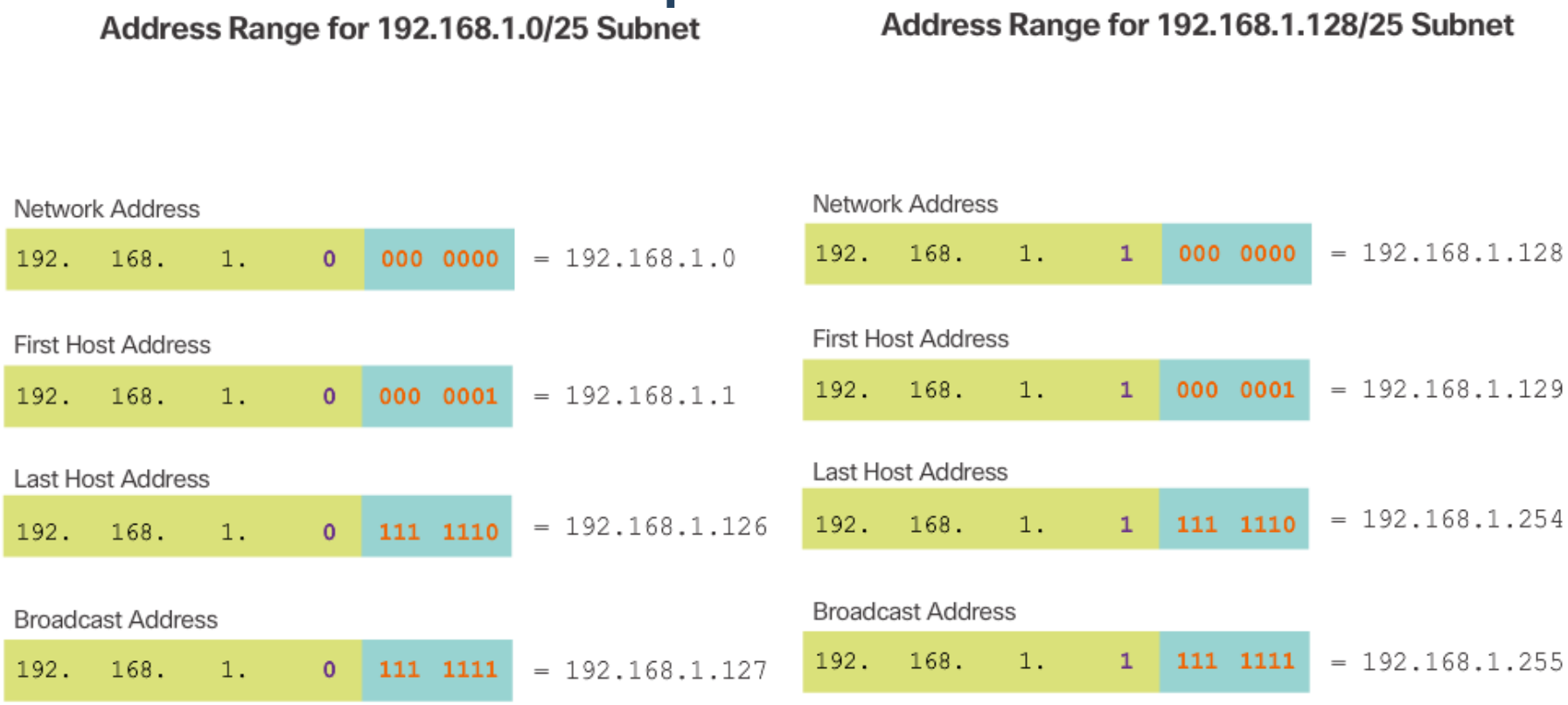


Rozdelenie 192.168.1.0/24 na 2 podsiete

- Pôvodná sieť: 192.168.1.0/24 (ako asi vyzerala topológia?)
  - ktorú sme rozdelili na 2 podsiete:



Rozdelenie 192.168.1.0/24 na 2 podsiete



- Príklad pridelenia IP adries pre sieťové zariadenia:
  - Prvá použiteľná IP adresa – pre rozhranie smerovača
  - Druhá použiteľná IP adresa – pre PC

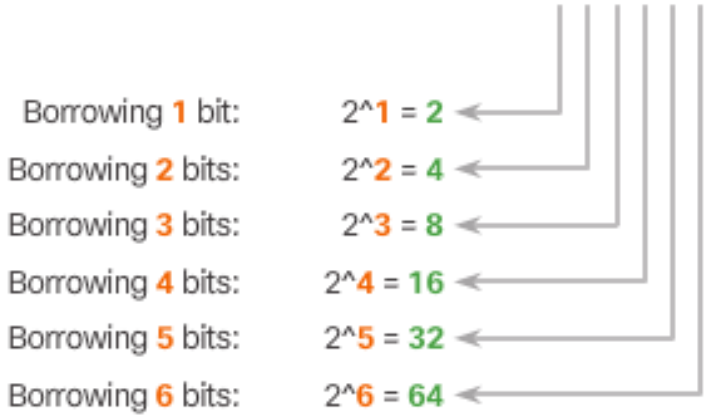
Výpočet počtu subsietí

2^n

n= bits borrowed

192 . 168 . 1 . 0

nnnnnnnn . nnnnnnnn . nnnnnnnn . hhhhhhhh



Príklady:

- Koľko subsietí viem vytvoriť, ak na subsieťovanie použijem:
  - 1 bit? → 2
  - 2 bity? → 4
  - 3 bity? → 8
  - 4 bity? → 16
- Maska pôvodnej siete bola:
  - 255.255.255.0
- Maska novej podsiete je:
  - 255.255.255.192

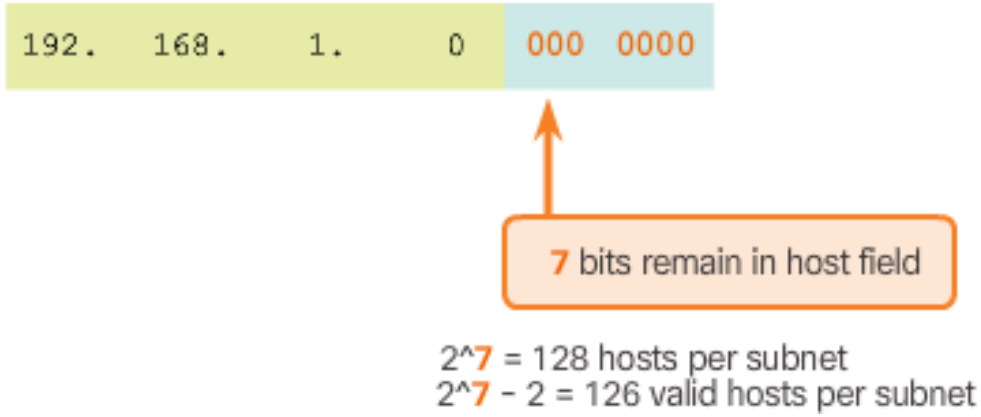
Koľko subsietí sme vytvorili takýmto subsieťovaním?

192 = 11000000
t.j. 2 bity na subsieťovanie,
a teda 4 subsiete

Výpočet počtu použiteľných IP adries

2^n-2

n= the number of bits remaining in the host field



Príklady:

- Aký počet použiteľných IP adries mám v týchto subsieťach:
  - 192.168.1.64/27 ?
  - Podsieť s maskou: 255.255.128.0 ?
  - Podsieť s dĺžkou prefixu /22 ?

- Akú masku má podsieť, o ktorej viem že má max. 30 použiteľných adries?

- Akú masku mám použiť pre subsieťovanie siete 10.0.0.0/8 ak chcem mať subsiete so 14 použiteľnými IP adresami?

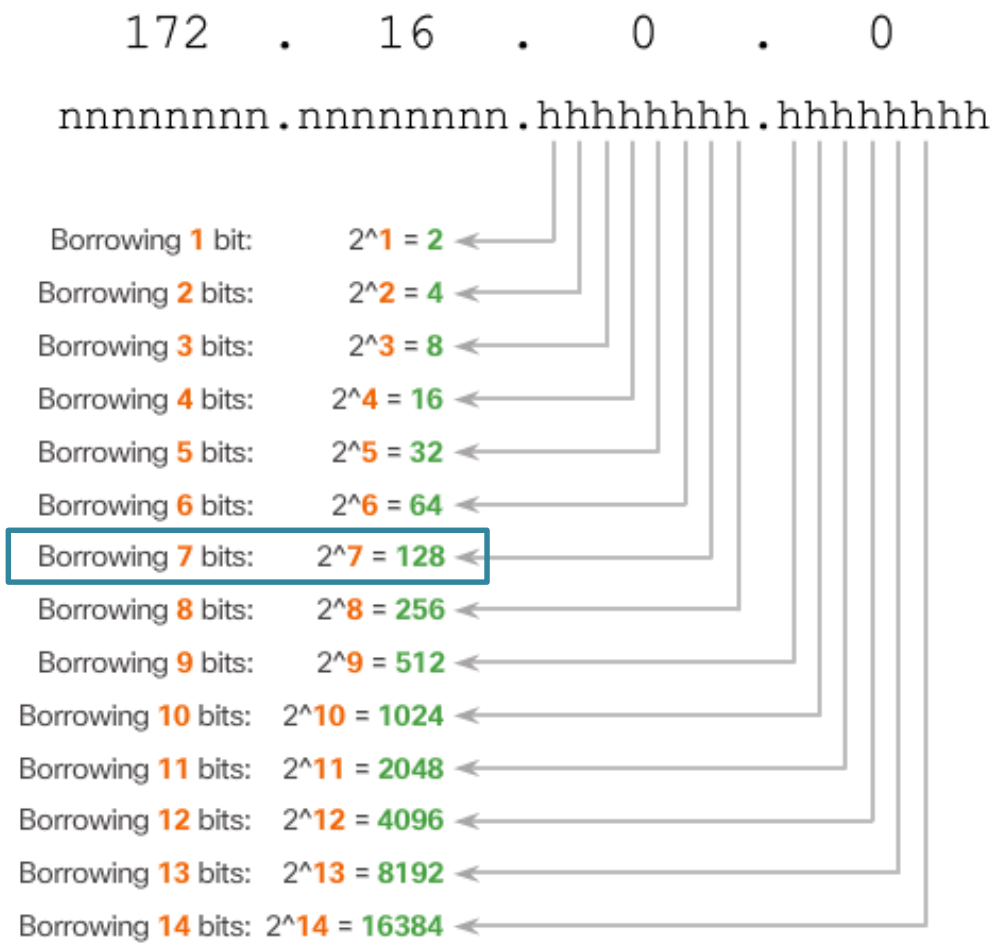
Téma 8.1.3:
Subsieťovanie s pevnou maskou s ohľadom na počet subsietí

Možné subsieťovania pôvodnej siete /16

Prefix Length	Subnet Mask	Network Address (n = network, h = host)	# of subnets	# of hosts
/17	255.255.128.0	nnnnnnnn.nnnnnnnn.nhhhhhhh.hhhhhhhh 11111111.11111111.10000000.00000000	2	32564
/18	255.255.192.0	nnnnnnnn.nnnnnnnn.nnhhhhhh.hhhhhhhh 11111111.11111111.11000000.00000000	4	16282
/19	255.255.224.0	nnnnnnnn.nnnnnnnn.nnnhhhhh.hhhhhhhh 11111111.11111111.11100000.00000000	8	8190
/20	255.255.240.0	nnnnnnnn.nnnnnnnn.nnnnhhhh.hhhhhhhh 11111111.11111111.11110000.00000000	16	4094
/21	255.255.248.0	nnnnnnnn.nnnnnnnn.nnnnnhhh.hhhhhhhh 11111111.11111111.11111000.00000000	32	2046
/22	255.255.252.0	nnnnnnnn.nnnnnnnn.nnnnnnhh.hhhhhhhh 11111111.11111111.11111100.00000000	64	1022
/23	255.255.254.0	nnnnnnnn.nnnnnnnn.nnnnnnnh.hhhhhhhh 11111111.11111111.11111110.00000000	128	510
/24	255.255.255.0	nnnnnnnn.nnnnnnnn.nnnnnnnn.hhhhhhhh 11111111.11111111.11111111.00000000	256	254
/25	255.255.255.128	nnnnnnnn.nnnnnnnn.nnnnnnnn.nhhhhhhh 11111111.11111111.11111111.10000000	512	126
/26	255.255.255.192	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnhhhhhh 11111111.11111111.11111111.11000000	1024	62
/27	255.255.255.224	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnhhhhh 11111111.11111111.11111111.11100000	2048	30

- Koľko najviac bitov môžem použiť na subsieťovanie?
  (aby mi ešte ostalo na adresovanie uzlov)

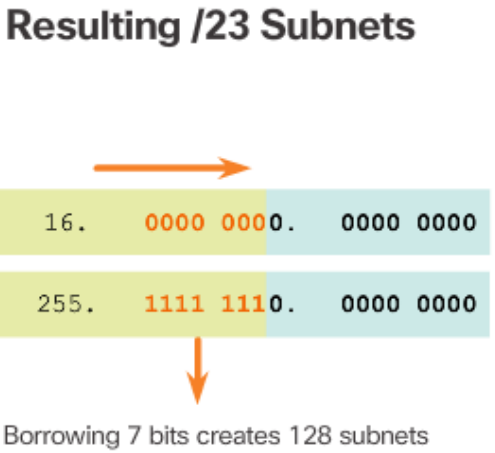
Rozdelenie 172.16.0.0/16 na 100 subsietí



- Vždy musím robiť zaokrúhľovanie na najbližšiu vyššiu mocninu dvojky
  - Prečo sa to nedá presne?
- Zoberiem teda 7 bitov na subsieťovanie
  - ostatné bity z host part (zo 16 bitov) mi ostatnú na adresáciu uzlov

Rozdelenie 172.16.0.0/16 na 100 subsietí

Posúvam hranicu medzi network part a host part až za (16+7). bit, t.j. prefix bude mať dĺžku /23



Nová maska pre podsiete:

Adresa 1. podsiete: 172. 16. 0000 0000. 0000 0000 — 172.16.0.0/23

Adresa 2. podsiete: 172. 16. 0000 0010. 0000 0000 — 172.16.2.0/23

Adresa 3. podsiete: 172. 16. 0000 0100. 0000 0000 — 172.16.4.0/23

. . to . .

Adresa 128. podsiete: 172. 16. 1111 1110. 0000 0000 — 172.16.254.0/23

Na 3. bajte budú iba párne čísla !

Výpočet počtu použiteľných IP adries

Hosts = 2^n
(where n = host bits remaining)

172. 16. 00 00 00 00. 0000 0000

9 bits remain in host field

2^9 = 512 hosts per subnet
2^9 – 2 = 510 valid hosts per subnet

Address Range for 172.16.0.0/23 Subnet

Network Address

172. 16. 00 00 00 00. 0000 0000 = 172.16.0.0/23

First Host Address

172. 16. 00 00 00 00. 0000 0001 = 172.16.0.1/23

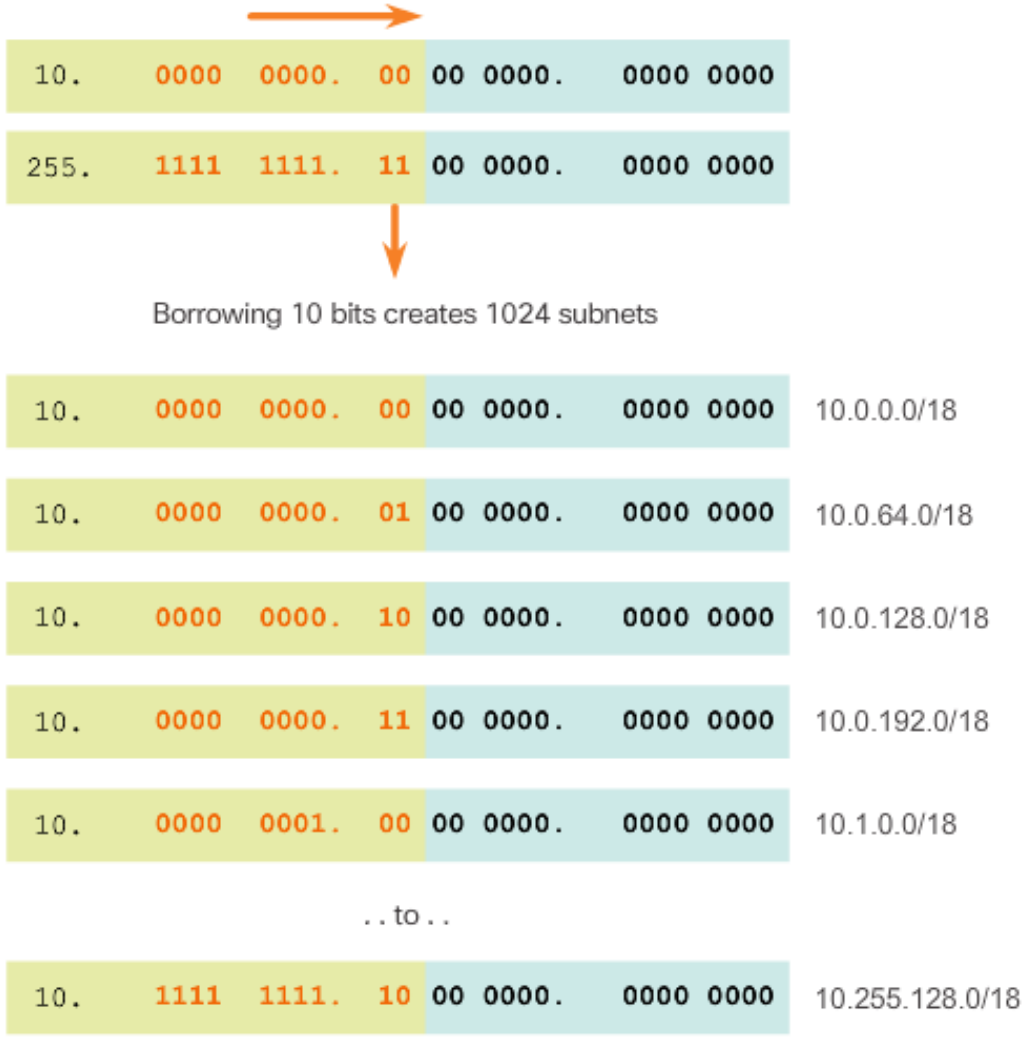
Last Host Address

172. 16. 00 00 00 01. 1111 1110 = 172.16.1.254/23

Broadcast Address

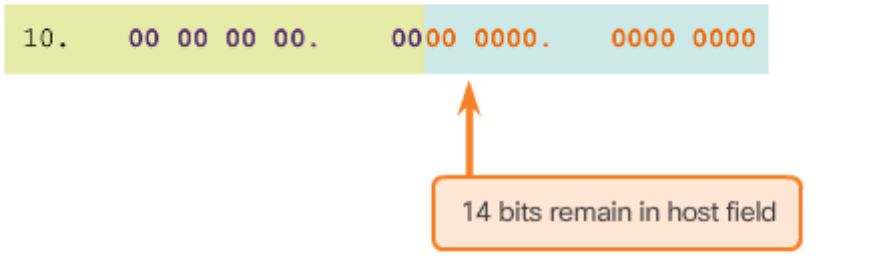
172. 16. 00 00 00 01. 1111 1111 = 172.16.1.255/23

Rozdelenie 10.0.0.0/8 na 1000 podsietí



Výpočet počtu použiteľných IP adries

Calculating Hosts



2^14 = 16384 hosts per subnet
2^14 - 2 = 16382 valid hosts per subnet

Address Range for 10.0.0.0/18 Subnet

Network Address

10. 00 00 00 00. 0000 0000. 0000 0000 = 10.0.0.0/18

First Host Address

10. 00 00 00 00. 0000 0000. 0000 0001 = 10.0.0.1/18

Last Host Address

10. 00 00 00 00. 0011 1111. 1111 1110 = 10.0.63.254/18

Broadcast Address

10. 00 00 00 00. 0011 1111. 1111 1111 = 10.0.63.255/18

Téma 8.1.4:
Subsieťovanie s pevnou maskou s ohľadom na veľkosť subsietí (t.j. s ohľadom na počet IP adries v subsieťach)

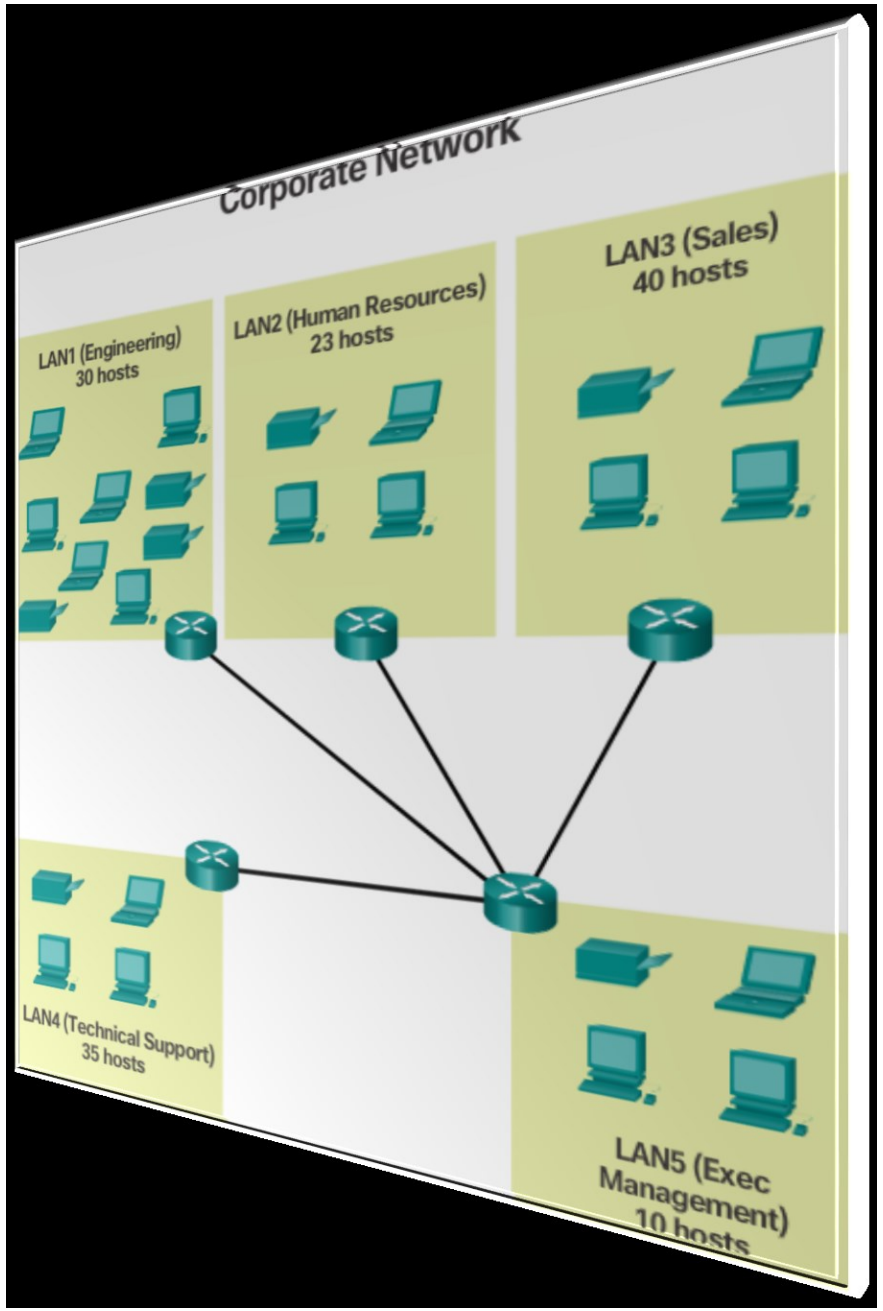
Subsieťovanie podľa požiadaviek

Vždy zvažujeme pri tvorbe adresného plánu pre podsiete:

- Počet potrebných použiteľných adries v každej podsieti.
- Počet potrebných podsietí.
- Pričom platí:
  - Čím viac bitov si požičiame z host part na tvorbu podsietí, tým menej bitov nám ostane na adresáciu hostov, a naopak

Prefix Length	Subnet Mask	Subnet Mask in Binary (n = network, h = host)	# of subnets	# of hosts
/25	255.255.255.128	nnnnnnnn.nnnnnnnn.nnnnnnnn.nhhhhhhh 11111111.11111111.11111111.10000000	2	126
/26	255.255.255.192	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnhhhhhh 11111111.11111111.11111111.11000000	4	62
/27	255.255.255.224	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnhhhhh 11111111.11111111.11111111.11100000	8	30
/28	255.255.255.240	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnhhhh 11111111.11111111.11111111.11110000	16	14
/29	255.255.255.248	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnhhh 11111111.11111111.11111111.11111000	32	6
/30	255.255.255.252	nnnnnnnn.nnnnnnnn.nnnnnnnn.nnnnnnhh 11111111.11111111.11111111.11111100	64	2

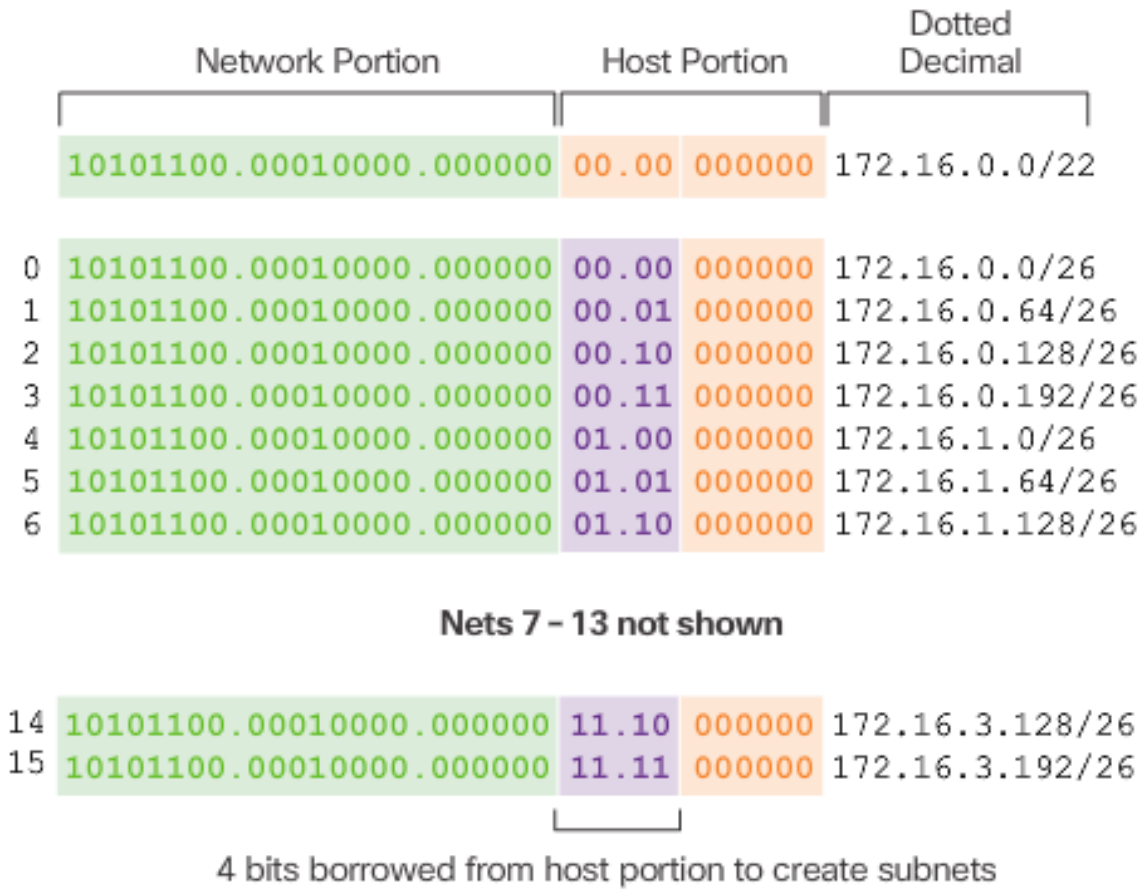
Subsieťovanie podľa požiadaviek



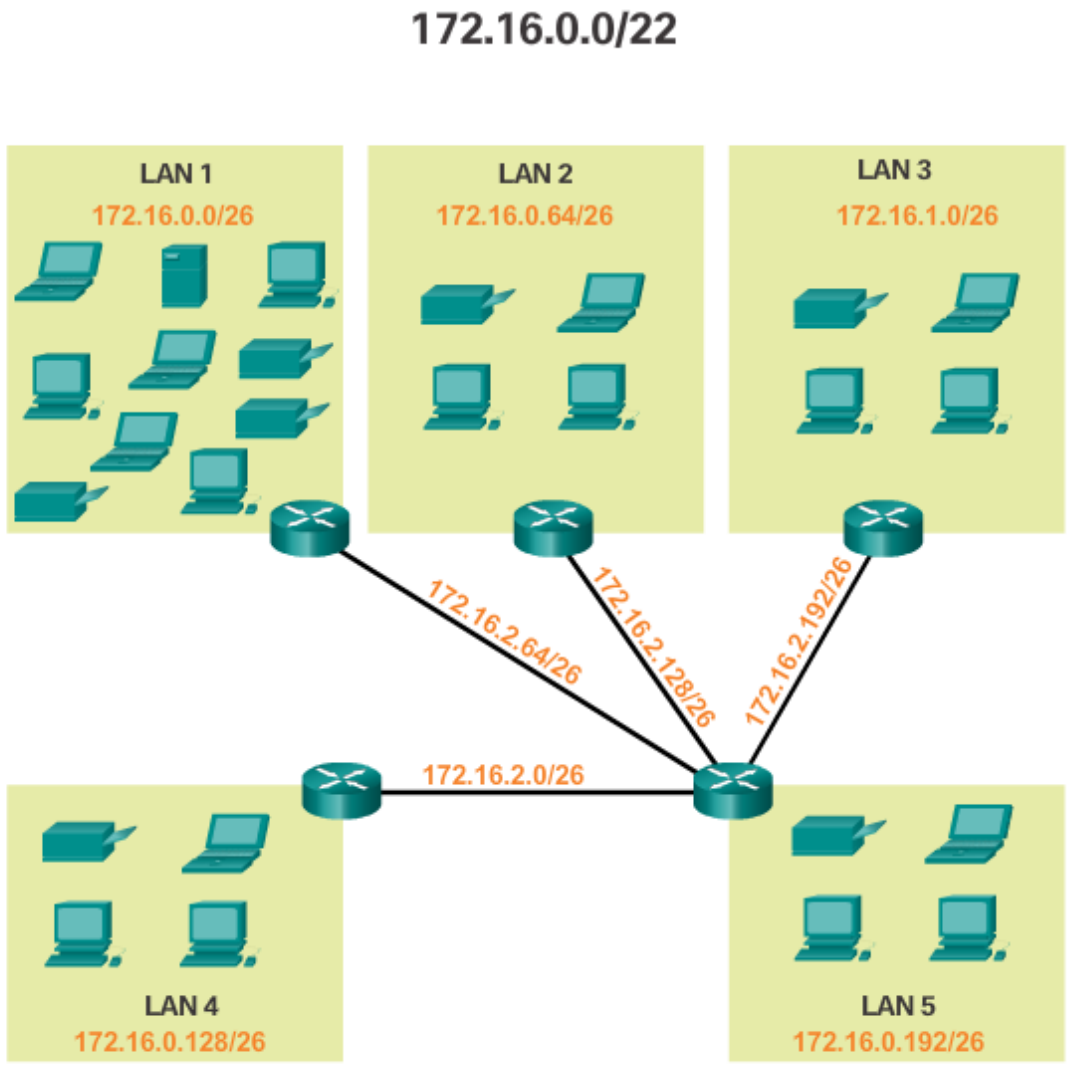
- Pridelený subnet: 172.16.0.0/22
- Požiadavky:
  - 9 podsietí (5x LAN, 4x WAN)
  - najväčšia podsieť potrebuje 40 IP adries

Subsieťovanie podľa požiadaviek (riešenie)

- 9 podsietí (5x LAN, 4x WAN)
- najväčšia podsieť potrebuje 40 IP adries



Subsieťovanie podľa požiadaviek (riešenie)

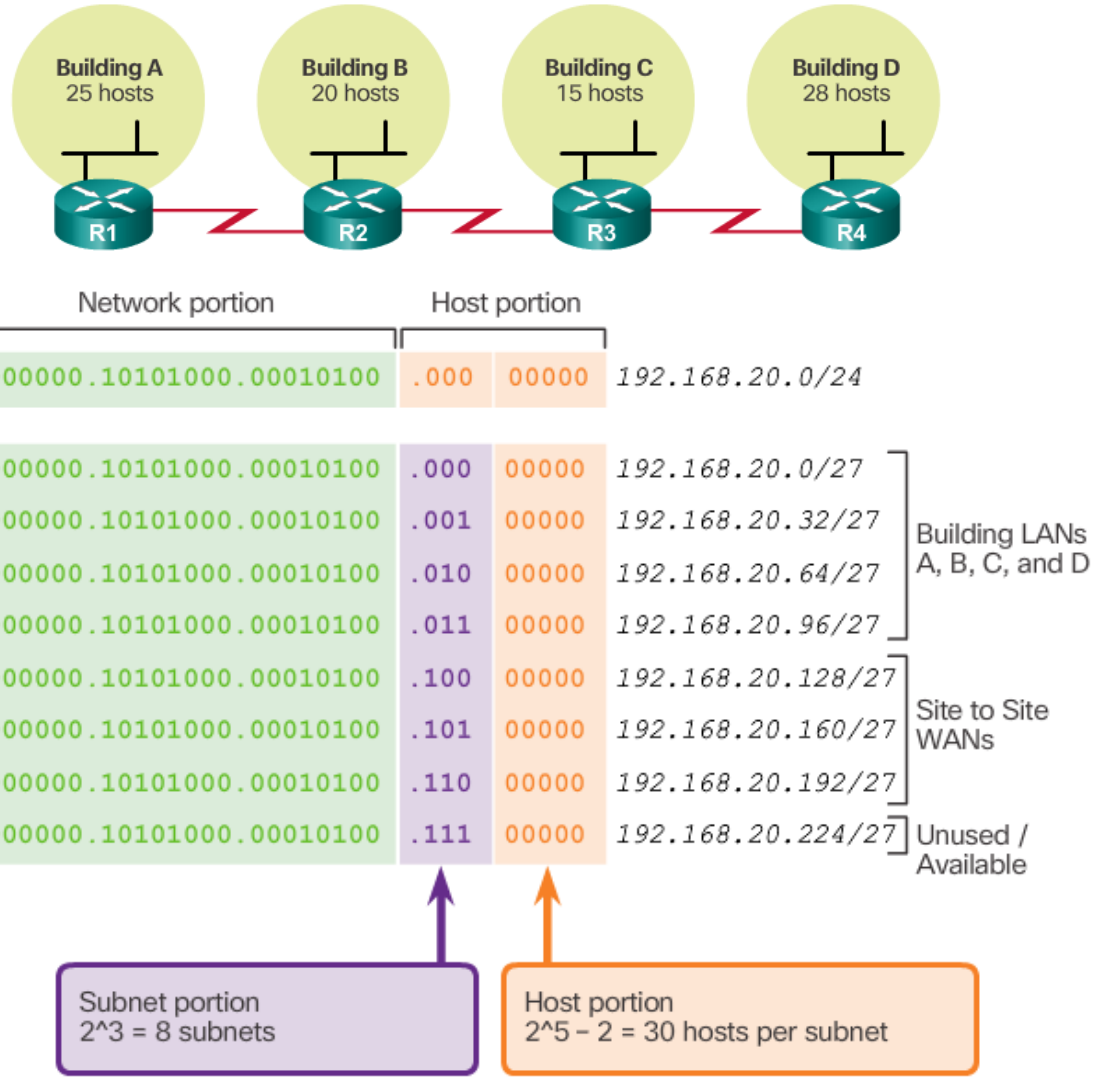


Téma 8.1.5:
Subsieťovanie s variabilnou maskou je najefektívnejšie

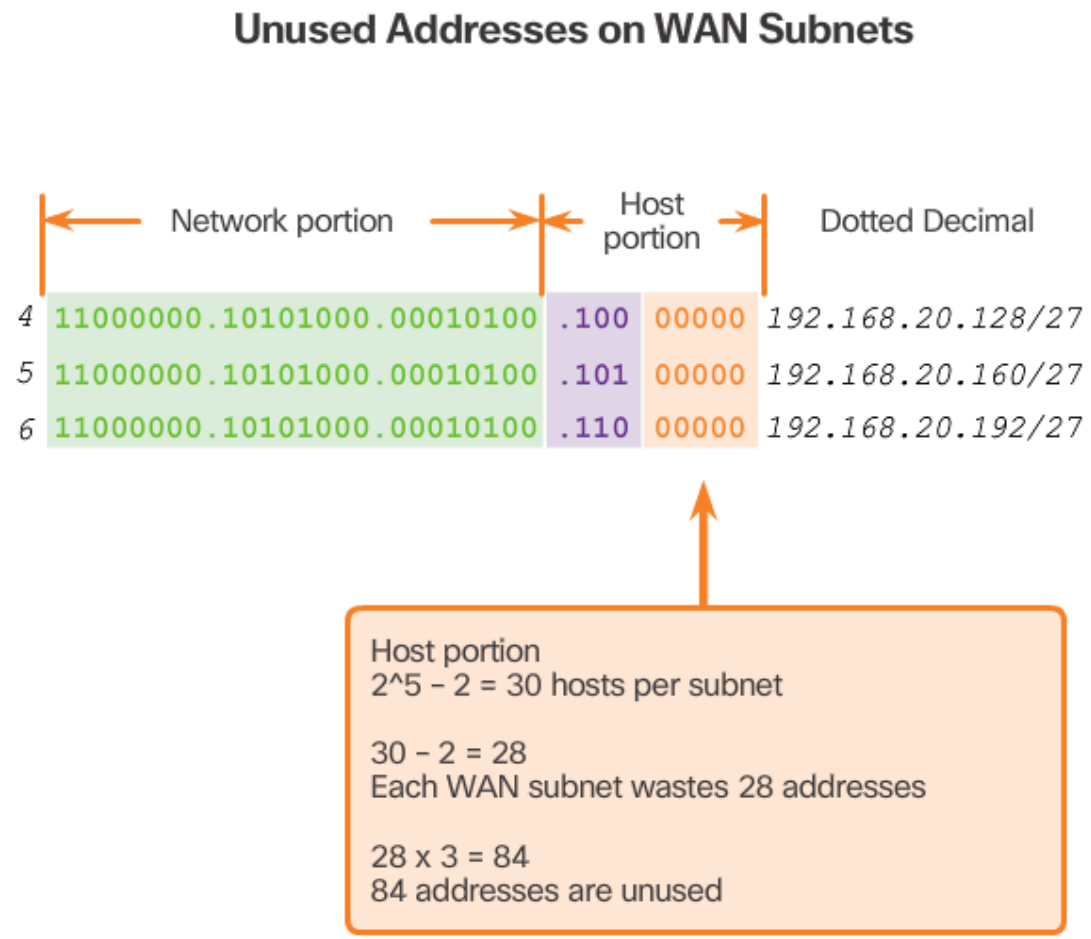
Nevýhody subsieťovania s pevnou maskou

Príklad:

- 4 LAN siete a 3 WAN siete = 7 subsietí
- Všetky subsiete rovnako veľké
- Pre WAN linky potrebujem iba **2** použiteľné adresy, ale po takomto subsieťovaní mám k dispozícii až **30** !

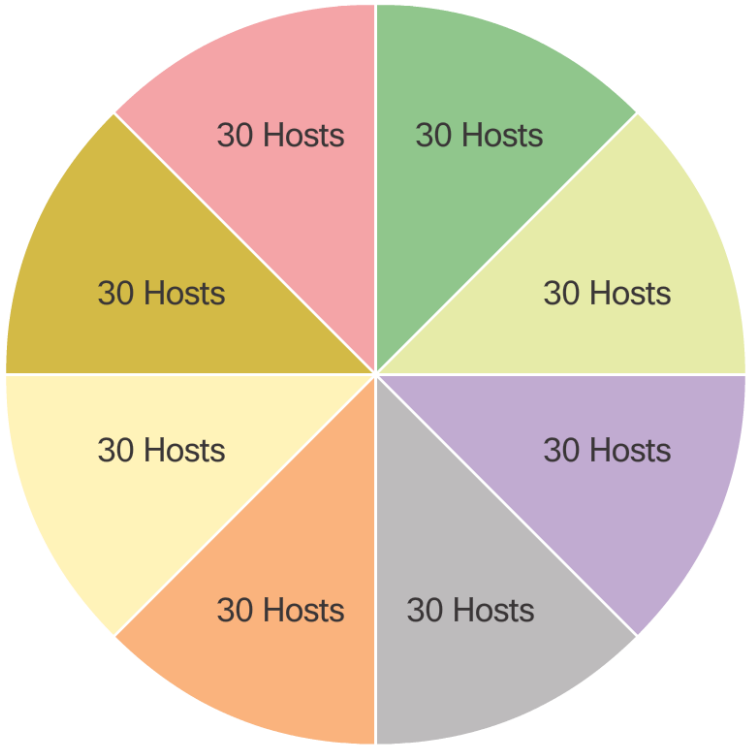


Nevýhody subsieťovania s pevnou maskou

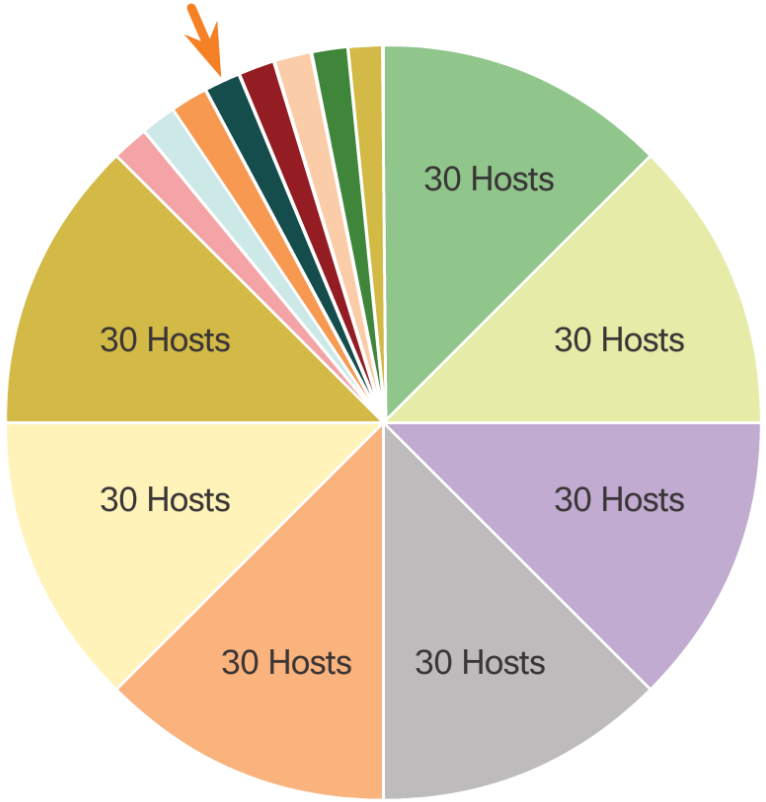


Variable Length Subnet Masks

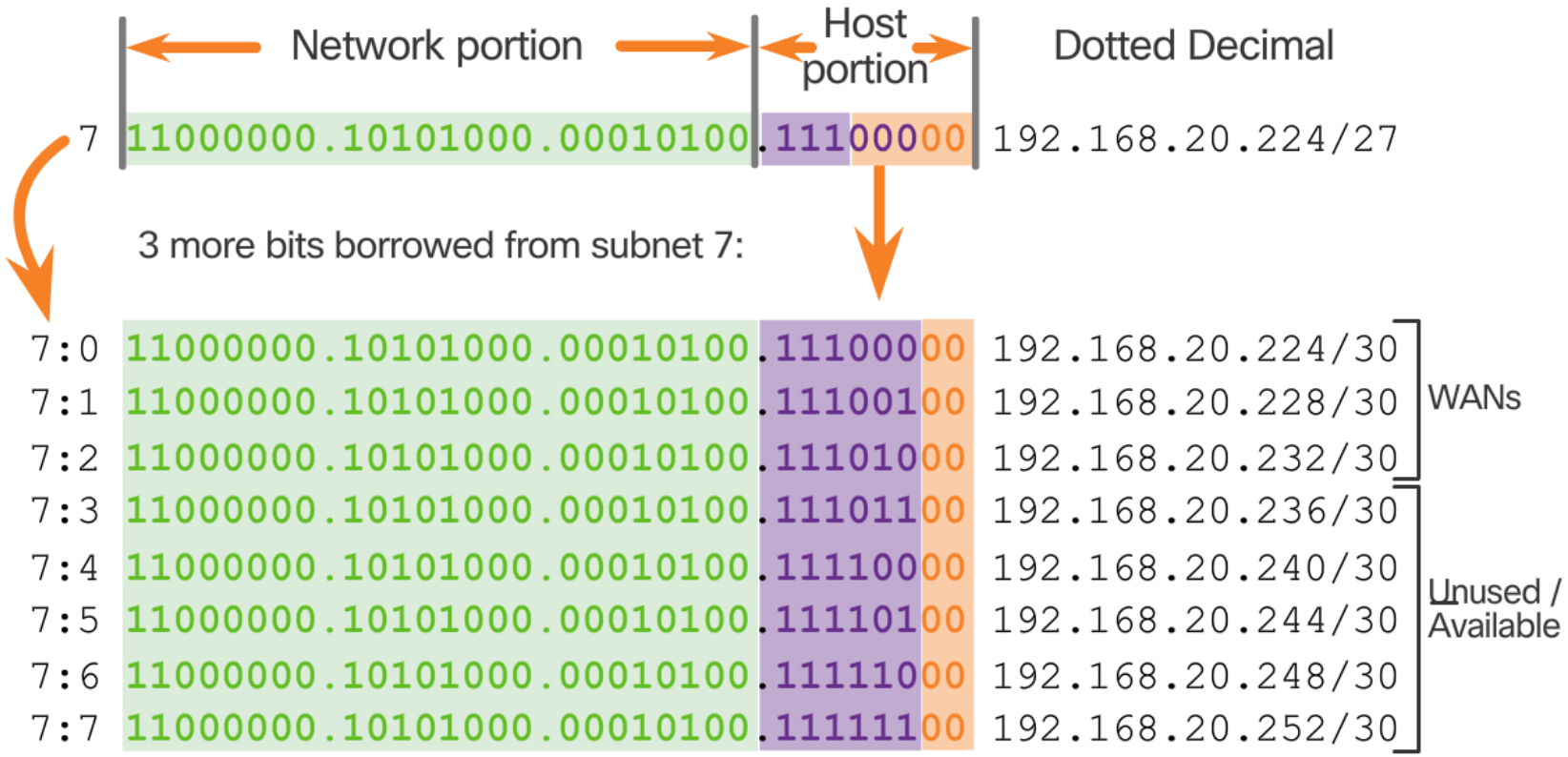
- Subsieťovanie s pevnou maskou



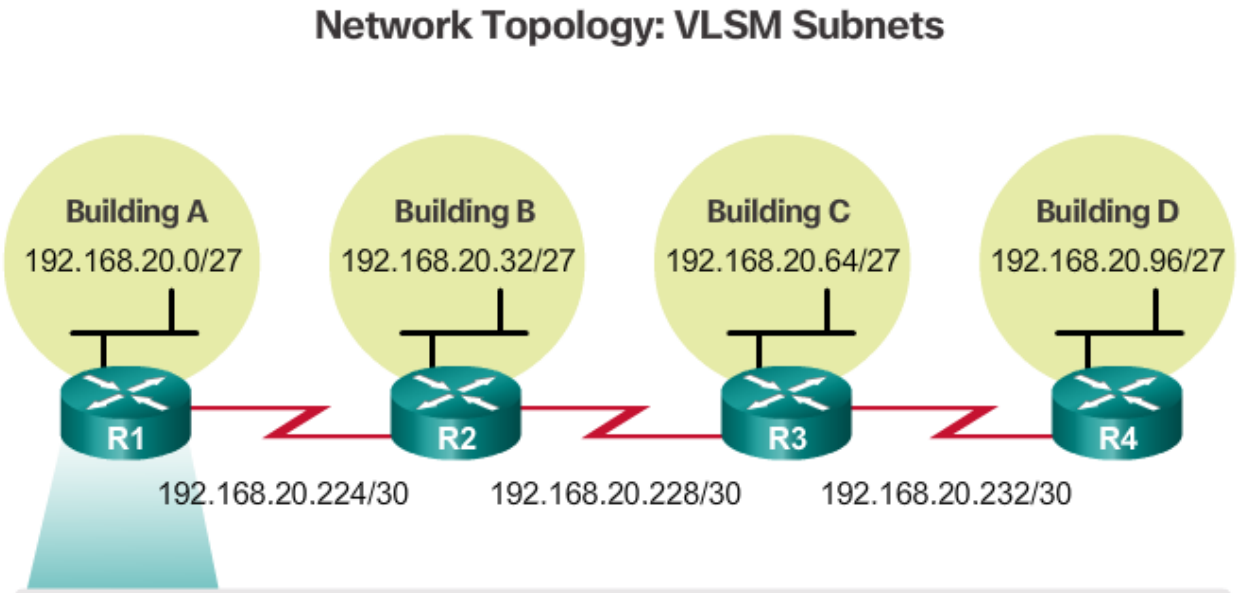
- Subsieťovanie s premenlivou maskou (VLSM)
  - Mám 4 IPv4 subsiete pre 4 LAN siete s 30 hostami (tesnejšie sa už nedá pre 25, 20, 15, 28 hostov)
  - Ďalšie 4 aktuálne nevyužívam, preto z nich:
    - Jednu (v tomto príklade poslednú, zväčša však prvú) rozdelím na menšie subsiete s 2 hostami (pre WAN linky)



VLSM

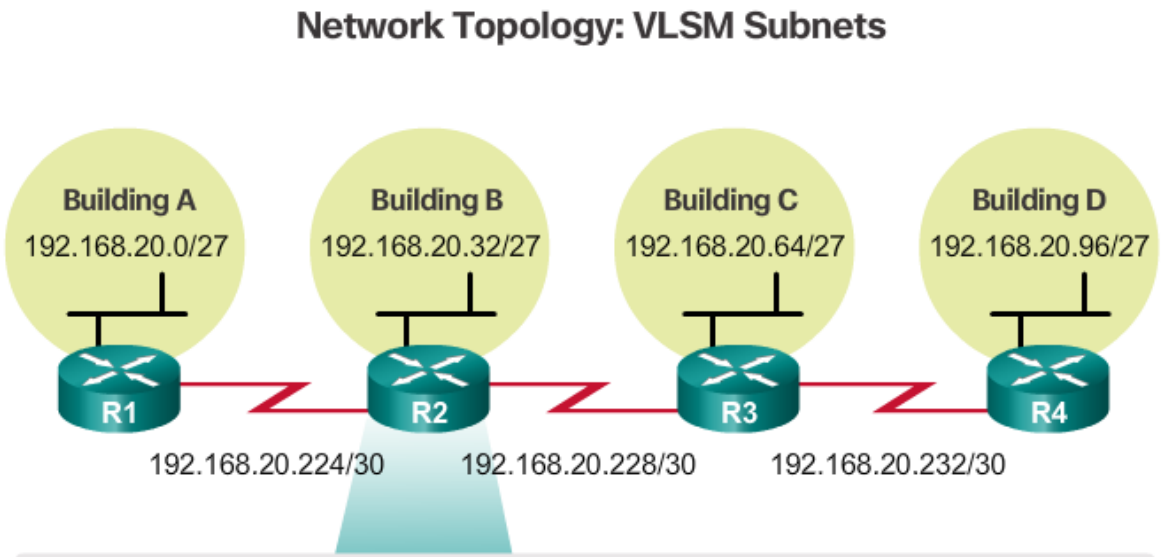

Network portion
Host portion
Dotted Decimal
7 11000000.10101000.00010100.11100000 192.168.20.224/27
3 more bits borrowed from subnet 7:
7:0 11000000.10101000.00010100.11100000 192.168.20.224/30
7:1 11000000.10101000.00010100.11100100 192.168.20.228/30
7:2 11000000.10101000.00010100.11101000 192.168.20.232/30
7:3 11000000.10101000.00010100.11101100 192.168.20.236/30
7:4 11000000.10101000.00010100.11110000 192.168.20.240/30
7:5 11000000.10101000.00010100.11110100 192.168.20.244/30
7:6 11000000.10101000.00010100.11111000 192.168.20.248/30
7:7 11000000.10101000.00010100.11111100 192.168.20.252/30
WANs
Unused / Available

Pridelenie IP adries rozhraniam R1
po VLSM subsieťovaní



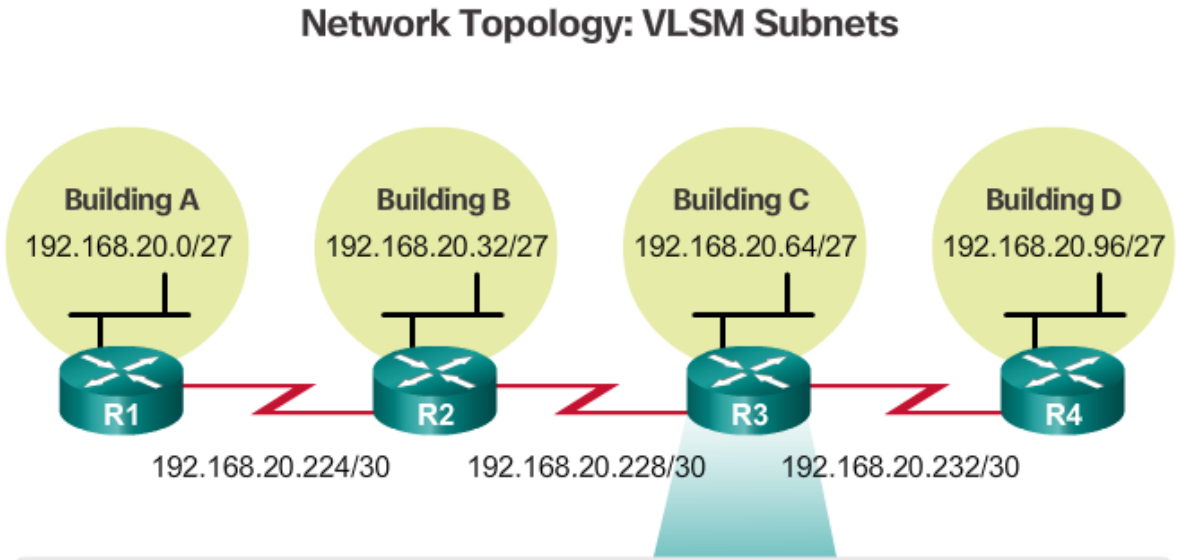

```
R1(config)# interface gigabitethernet 0/0
R1(config-if)# ip address 192.168.20.1 255.255.255.224
R1(config-if)# exit
R1(config)# interface serial 0/0/0
R1(config-if)# ip address 192.168.20.225 255.255.255.252
R1(config-if)# end
R1#
```

Pridelenie IP adries rozhraniam R2 po VLSM subsieťovaní



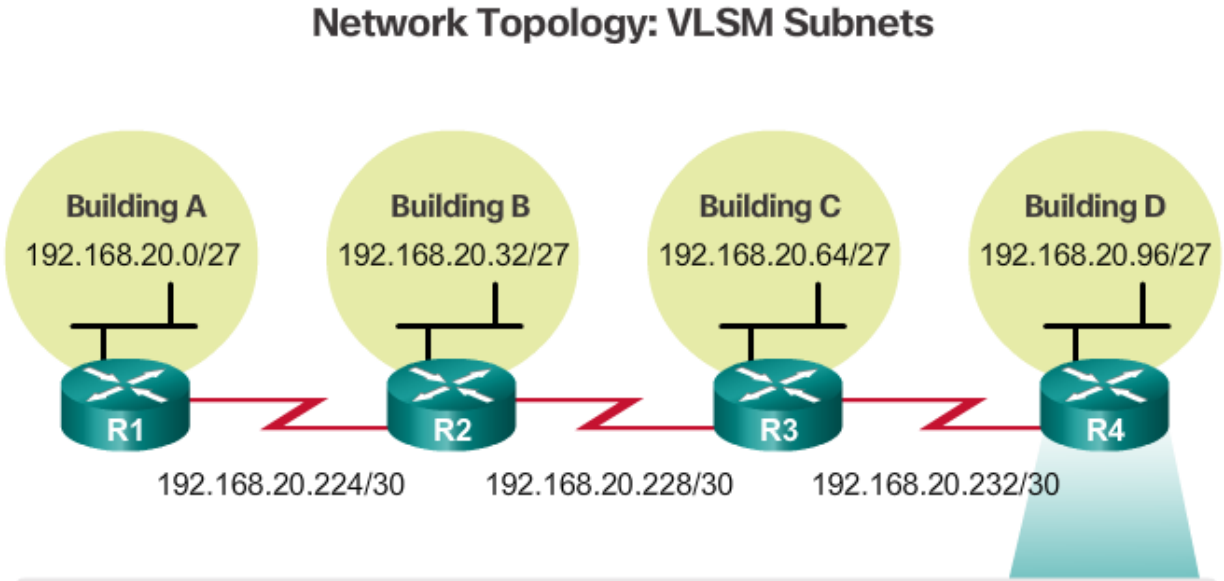

```
R2(config)# interface gigabitethernet 0/0
R2(config-if)# ip address 192.168.20.33 255.255.255.224
R2(config-if)# exit
R2(config)# interface serial 0/0/0
R2(config-if)# ip address 192.168.20.226 255.255.255.252
R2(config-if)# exit
R2(config)# interface serial 0/0/1
R2(config)# ip address 192.168.20.229 255.255.255.252
R2(config-if)# end
R2#
```

Pridelenie IP adries rozhraniam R3
po VLSM subsieťovaní




```
R3(config)# interface gigabitethernet 0/0
R3(config-if)# ip address 192.168.20.65 255.255.255.224
R3(config-if)# exit
R3(config)# interface serial 0/0/0
R3(config-if)# ip address 192.168.20.230 255.255.255.252
R3(config-if)# exit
R3(config)# interface serial 0/0/1
R3(config)# ip address 192.168.20.233 255.255.255.252
R3(config-if)# end
R3#
```

Pridelenie IP adries rozhraniam R4
po VLSM subsieťovaní




```
R4(config)# interface gigabitethernet 0/0
R4(config-if)# ip address 192.168.20.97 255.255.255.224
R4(config-if)# exit
R4(config)# interface serial 0/0/0
R4(config-if)# ip address 192.168.20.234 255.255.255.252
R4(config-if)# end
R4#
```

Rekurzívne subsieťovanie

- Zoberiem subsieťovaný adresný priestor, a subsieťujem ho ďalej, s dlhšou masku, podľa potreby

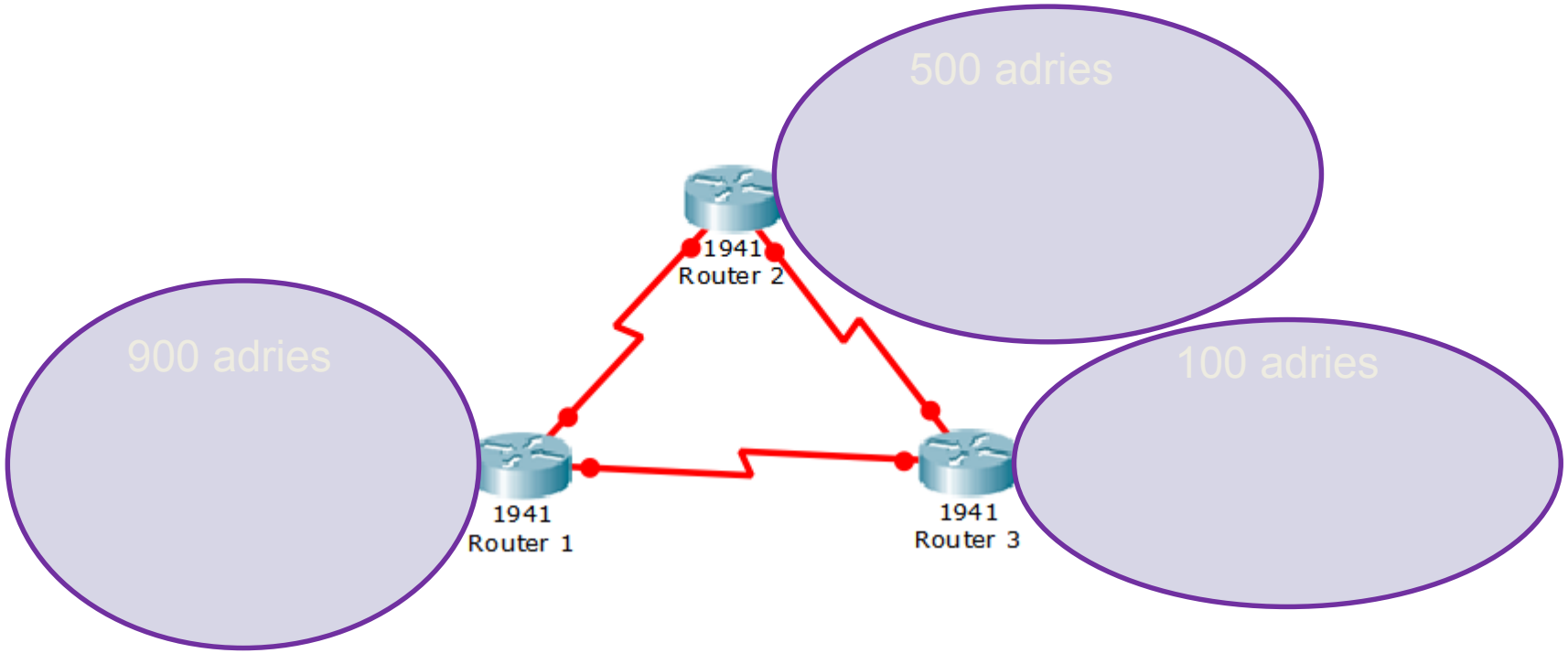
VLSM Subnetting of 192.168.20.0/24

	/27 Network	Hosts
Bldg A	.0	.1 - .30
Bldg B	.32	.33 - .62
Bldg C	.64	.65 - .94
Bldg D	.96	.97 - .126
Unused	.128	.129 - .158
Unused	.160	.161 - .190
Unused	.192	.193 - .222
	.224	.225 - .254

	/30 Network	Hosts
WAN R1-R2	.224	.225 - .226
WAN R2-R3	.228	.229 - .230
WAN R3-R4	.232	.233 - .234
Unused	.236	.237 - .238
Unused	.240	.241 - .242
Unused	.244	.245 - .246
Unused	.248	.249 - .250
Unused	.252	.253 - .254

Príklad na VLSM – zadanie

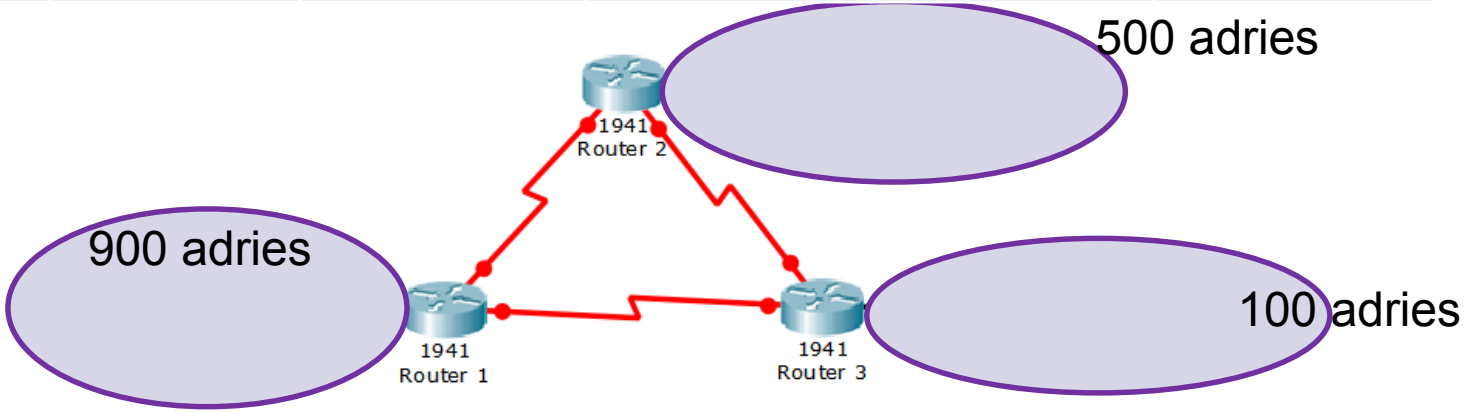
- Pridelený subnet: 192.168.0.0/20
- Požiadavky:
  - 3x LAN (veľkosti v obrázku)
  - 3x WAN (v každej 2 použiteľné IP adresy – pre rozhrania smerovačov)



Príklad na VLSM - riešenie

Pridelený subnet: 192.168.0.0/20 – ideme subsieťovať:

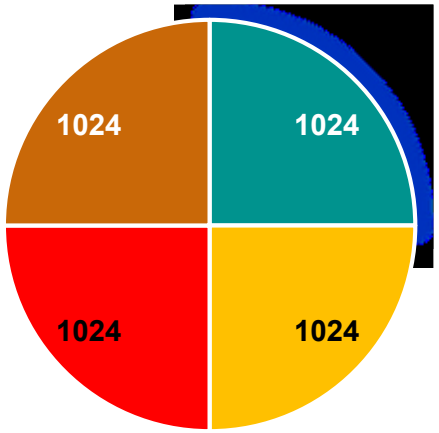
Počet požad. adries:	Veľkosť siete (2^x)	Dĺžka prefixu	Adresa subsiete	Broadcast (adresa subsiete +veľkosť siete-1)	Prvá použiteľná IP adresa (adresa siete+1)	Posledná použ. IP adresa (broadcast-1)
900	$1024=2^{10}$ =4x256	/22 (32-10)	192.168.0.0	192.168.3.255	192.168.0.1	192.168.3.254
500	$512=2^9$ =2x256	/23 (32-9)	192.168.4.0	192.168.5.255	192.168.4.1	192.168.5.254
100	$128=2^8$	/24 (32-8)	192.168.6.0	192.168.6.127	192.168.6.1	192.168.6.126
2	$4=2^2$	/30 (32-2)	192.168.6.128	192.168.6.131	192.168.6.129	192.168.6.130
2	$4=2^2$	/30 (32-2)	192.168.6.132	192.168.6.135	192.168.6.133	192.168.6.134
2	$4=2^2$	/30 (32-2)	192.168.6.136	192.168.6.139	192.168.6.137	192.168.6.138



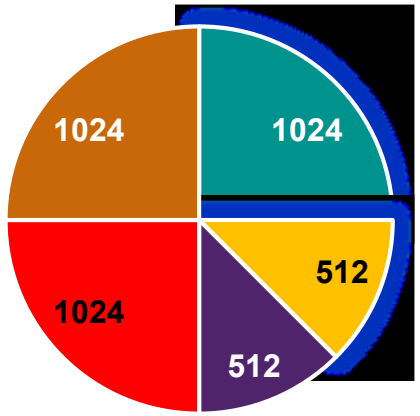
Príklad na VLSM - riešenie

Pridelený subnet: 192.168.0.0/20

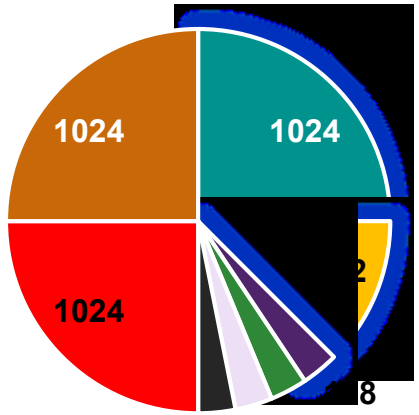
- k dispozícií sme mali 32-20 = 12 bitov, 2^{12}=4096 použiteľných IP adries
- sme rekurzívne subsieťovali
- finálne sme vytvorili 6 subsietí veľkosti 1024, 512, 128, 4, 4, 4 možných IP adries, a zvyšný priestor ostal voľný pre budúce použitie



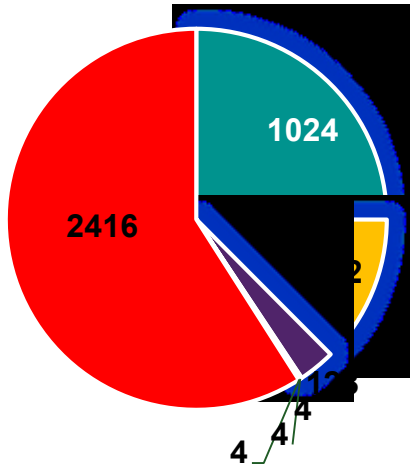
- 1.subsieť (1024)
- 2. subsieť



- 1.subsieť (1024)
- 2. subsieť (512)
- 3. subsieť
- 4. subsieť



- 1.subsieť (1024)
- 2. subsieť (512)
- 3. subsieť (128)
- 4. subsieť
- 5. subsieť
- 6. subsieť
- 7. subsieť
- 8. subsieť



- 1.subsieť (1024)
- 2. subsieť (512)
- 3. subsieť (128)
- 4. subsieť (4)
- 5. subsieť (4)
- 6. subsieť (4)
- 7. voľný priestor nerozdelený

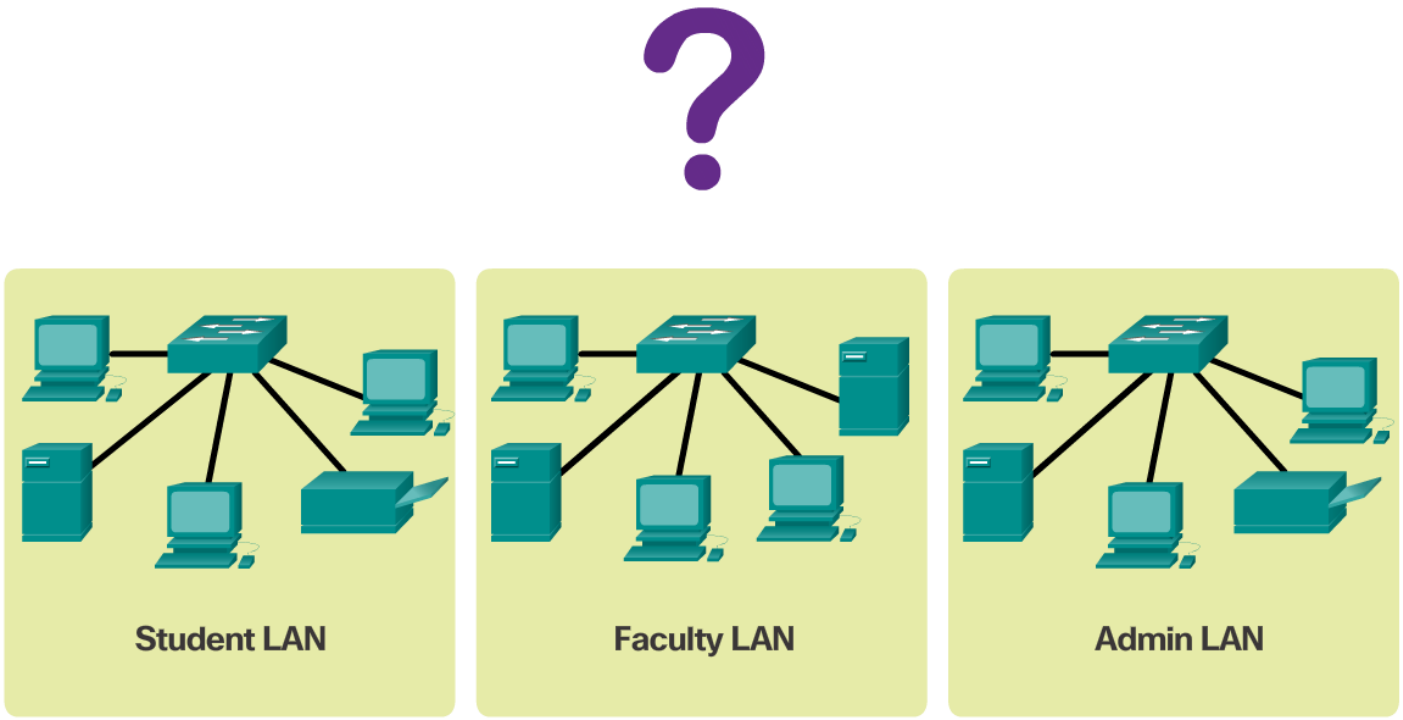
Časť 8.2: Adresné schémy

Téma 8.2.1: Štrukturovaný IP dizajn

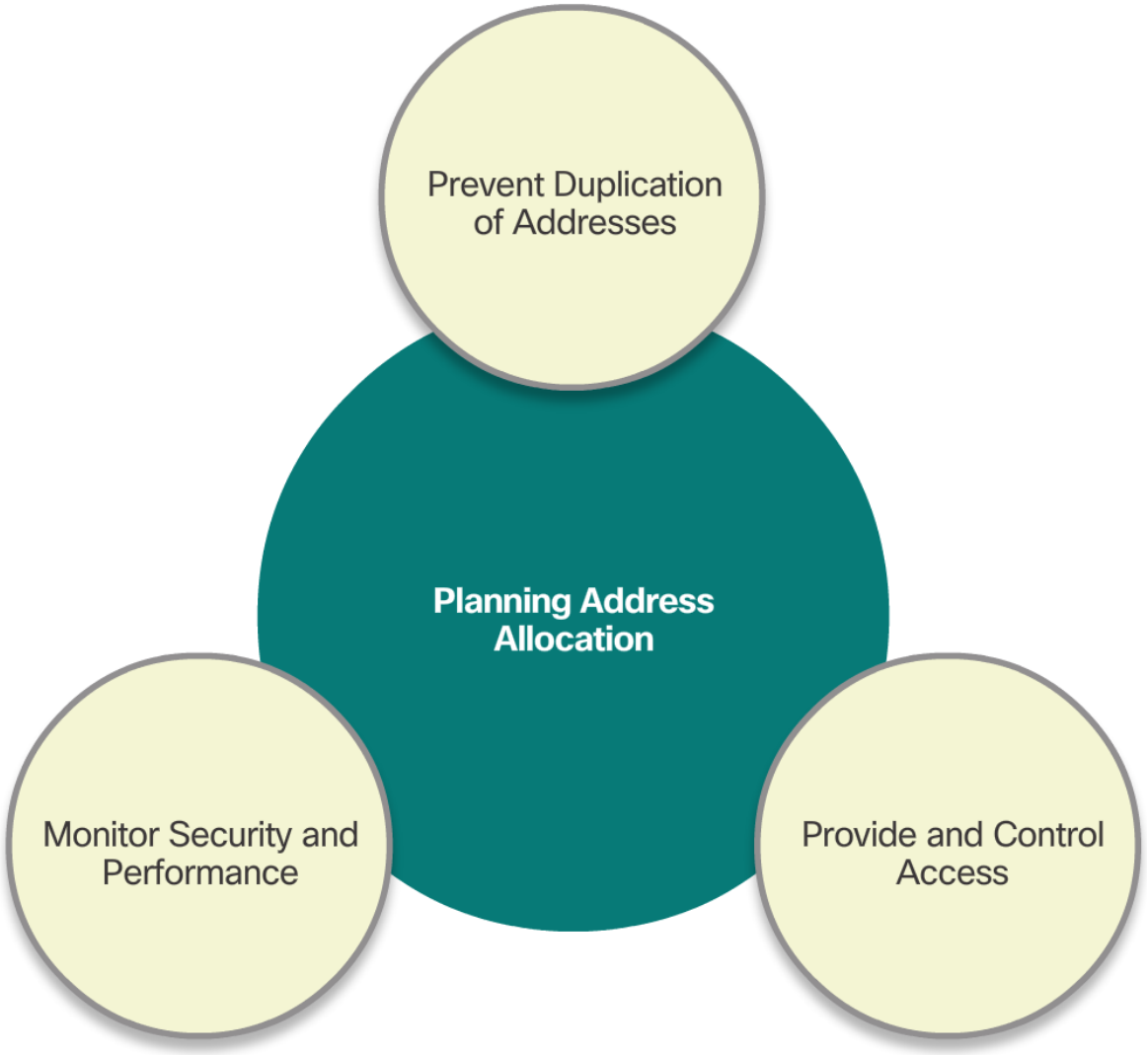
IP adresný plán

Je nutné naplánovať:

- počet podsietí
- veľkosti podsietí
- ako priradíme IP adresy (komu ktorú z danej podsiete)



Tvorba IP adresného plánu



Priradenie IP adries zariadeniam

Network: 192.168.1.0/24

Use	First	Last
Host Devices	.1	.229
Servers	.230	.239
Printers	.240	.249
Intermediary Devices	.250	.253
Gateway (router LAN interface)	.254	

MINISTERSTVO
ŠKOLSTVA, VEDY,
VÝSKUMU A ŠPORTU
SLOVENSKEJ REPUBLIKY

Ďakujem za pozornosť

Ohodnoť našu CNA na google:

- https://goo.gl/maps/BAnFvQKYCBpffcEX7