

SAPS

Optimalizácia smerovania, redistribúcia, policy routing

Vytvorené v rámci projektu KEGA 026TUKE-4/2021

*Katedra počítačov a informatiky
Fakulta elektrotechniky a informatiky
Technická univerzita v Košiciach*



Redistribúcia medzi smerovacími protokolmi



Súčasné použitie viacerých smerovacích protokolov

- Dočasný stav pri migrácii smerovacích protokolov
- Niektoré protokoly podporujú špecifické funkcie
 - Neexistuje univerzálny „všeobjímajúci“ smerovací protokol
- Spojenie sietí pod rôznou administratívnou správou
 - Napríklad zákaznícke lokality a sieť ISP
- „Politické“ dôvody
 - Iné než technické dôvody, prečo musí v sieti pracovať niekoľko smerovacích protokolov
- Nekompatibilita zariadení
 - Rôzni výrobcovia – rôzne protokoly – rôzne problémy

Spolupráca smerovacích protokolov: redistribúcia

- V istých situáciách je potrebné smerovacie informácie preniesť z jedného protokolu do druhého (jednosmerne), prípadne prenášať ich navzájom (obojsmerne)
- Tento prenos smerovacej informácie sa v Cisco poňatí nazýva redistribúcia
- Pre pripomenutie
 - Každý smerovací protokol má vlastné nezávislé pracovné databázy a z týchto databáz naplňa smerovaciu tabuľku
 - Daným smerovacím protokolom sa prenáša len obsah príslušných databáz (EIGRP si distribuuje svoje siete, RIP svoje, OSPF svoje, IS-IS svoje)

Redistribúcia

- Na smerovači teda môže bežať niekoľko smerovacích protokolov, ale informácie si navzájom bez dodatočnej konfigurácie navzájom nevymenia
- Jediné spoločné miesto, v ktorom sa „zbehnú“ informácie zo všetkých smerovacích protokolov, je smerovacia tabuľka
- Redistribúcia je vlastne prevzatie smerov cudzieho smerovacieho protokolu, ktoré sú zaznamenané v smerovacej tabuľke
 - Istá sieť musí byť najprv v smerovacej tabuľke, až potom ju možno redistribuovať
 - Neprenáša sa topologický detail, len info o existencii siete

Redistribúcia

- Pretože každý smerovací protokol funguje inak a inak si počíta svoju metriku, nedá sa metrika len tak prevziať
- Riešenie:
 - Pri redistribúcii sa stanoví počiatočná metrika v cieľovom protokole (tzv. seed metric), s ktorou sa smery z iného protokolu redistribuujú
 - Táto metrika sa potom zvyšuje v danom smerovacom protokole obvyklým spôsobom
- Niektoré protokoly zvyknú dosadiť implicitnú počiatočnú metriku, iné dosadzujú nekonečno
 - Je zásadne vhodné stanoviť metriku pre redistribúciu vždy, nespoliehať sa na automaticky dosadené hodnoty

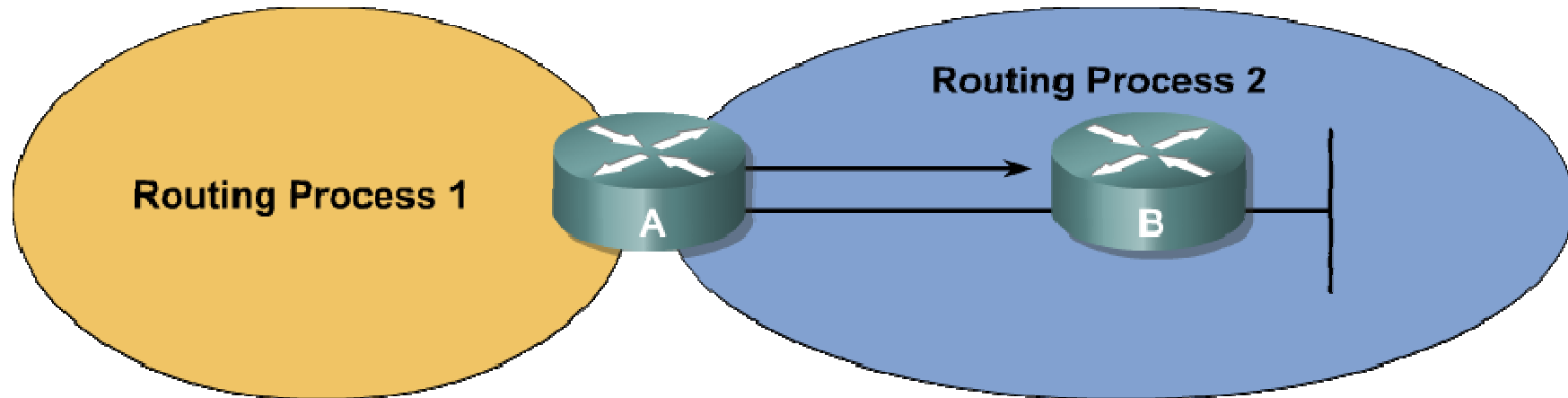
Štandardné počítačové metriky

Protokol	Štandardná počítačová metrika
RIP	Nekonečno
IGRP/EIGRP	Nekonečno
OSPF	20 pre všetky smerovacie protokoly okrem BGP, 1 pre BGP
IS-IS	0
BGP	BGP metrika sa nastaví na metriku príslušného IGP protokolu

Redistribúcia

- Redistribuovať možno z ľubovoľného smerovacieho protokolu
 - Presnejšie – redistribuovať možno z čohokoľvek, čo napĺňa smerovaciu tabuľku, čiže aj statické smery, priamo pripojené siete, default route z DHCP
- Redistribuovať sa dá do všetkých bežných smerovacích protokolov
 - Nedá sa redistribuovať do ODR
- Redistribúcia sa deje na jednom resp. viacerých smerovačoch, na ktorých beží súčasne viacero smerovacích protokolov
 - Môže byť jednosmerná alebo obojsmerná

Plánovanie redistribúcie



- Nájsť hranicu medzi rôznymi smerovacími protokolmi
- Stanoviť, ktorý smerovací protokol je hlavný (core resp. backbone process)
- Stanoviť, ktorý smerovací protokol je vedľajší (edge resp. migration process)
- Nakonfigurovať prenos informácie z vedľajšieho smerovacieho protokolu do hlavného

Redistribúcia – základné info o konfigurácii

- Uvedomenie si, ktorý smerovací protokol je cieľový a ktorý je zdrojový, je kľúčové pre správnu konfiguráciu
 - Cieľový proces je **príjemcom**, **preberateľom** smerovacej informácie
 - Zdrojový proces je **autorom**, **dodávateľom** preberanej smerovacej informácie
- Na Cisco smerovačoch sa redistribúcia konfiguruje vždy ako súčasť **cieľového** protokolu
 - Cieľový protokol si **vyberá**, odkiaľ preberie smerovacie info
 - Nie je možné z konfigurácie zdrojového protokolu vnútiť nejaké smery cieľovému smerovaciemu protokolu
- Príkaz **redistribute** hovorí o zdrojovom smerovacom protokole, z ktorého sa budú preberať smerovacie informácie

Redistribúcia do RIP

```
Router(config-router)# redistribute protocol  
[process-id] [match route-type]  
[metric metric-value] [route-map map-tag]
```

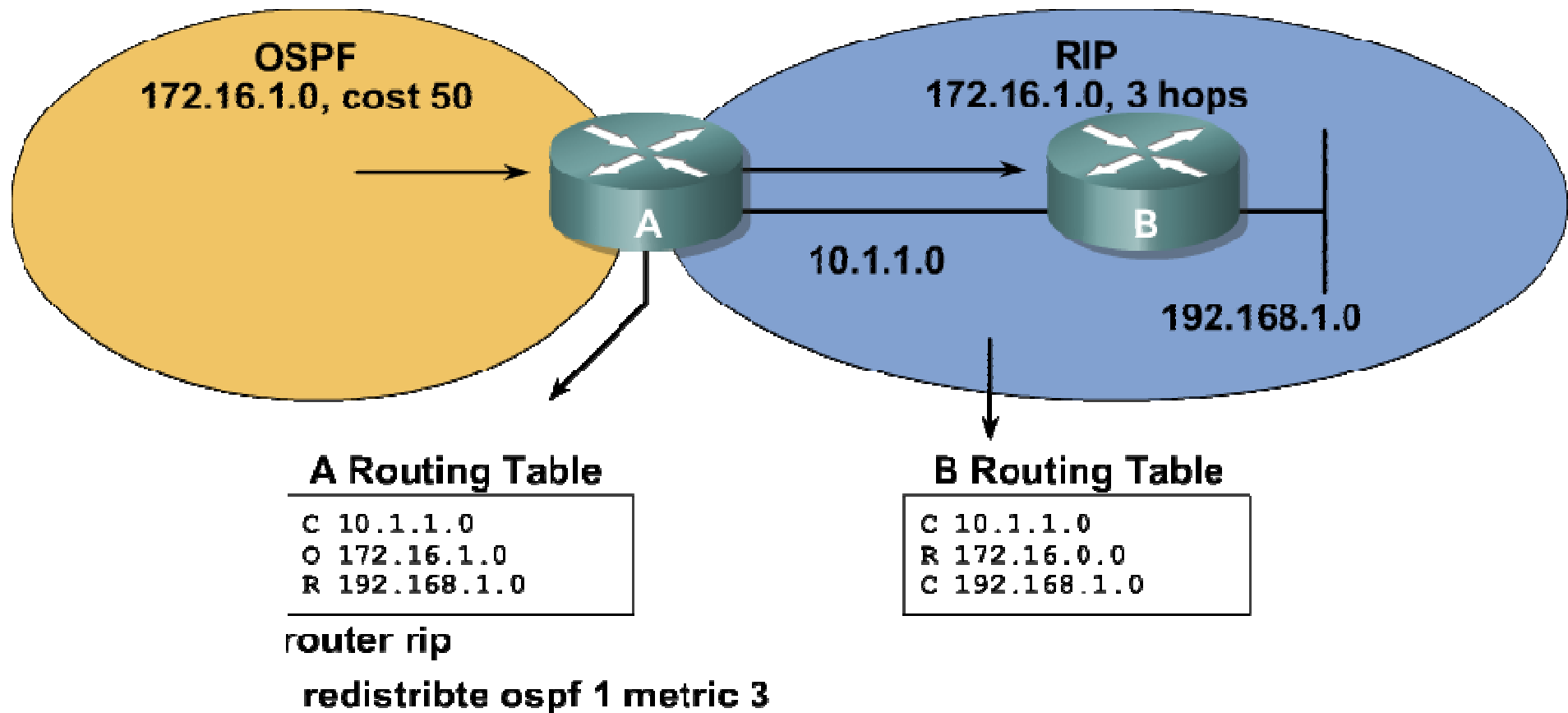
```
RtrA(config)# router rip  
RtrA(config-router)# redistribute ospf ?  
  
<1-65535> Process ID  
RtrA(config-router)# redistribute ospf 1 ?  
  
match      Redistribution of OSPF routes  
metric     Metric for redistributed routes  
route-map  Route map reference  
...  
<cr>
```

- Štandardná počiatočná metrika je nekonečno

Parametre príkazu `redistribute` pre RIP

Parameter	Popis
<code>protocol</code>	Zdrojový protokol, z ktorého sa preberú smerovacie informácie
<code>process-id</code>	Identifikátor čísla procesu (OSPF) alebo autonómneho systému (EIGRP, BGP)
match <code>route-type</code>	(Nepovinné) Test na typ OSPF ciest (interné, externé, NSSA-externé, typ 1, typ 2). Nepoužiteľné pri iných zdrojových smerovacích protokoloch.
metric <code>metric-value</code>	(Nepovinné) Definuje počiatočnú metriku. Pokiaľ nie je parameter uvedený, preberie sa hodnota z príkazu default-metric v konfigurácii RIP. Ak ani tento príkaz nie je prítomný, dosadí sa nekonečná metrika a smery sa neredistribuujú. Metrikou v protokole RIP je počet hopov.
route-map <code>map-tag</code>	(Nepovinné) Identifikátor route-map, ktorá sa použije na filtrovanie a úpravu importovaných smerov.

Redistribúcia do RIP – príklad

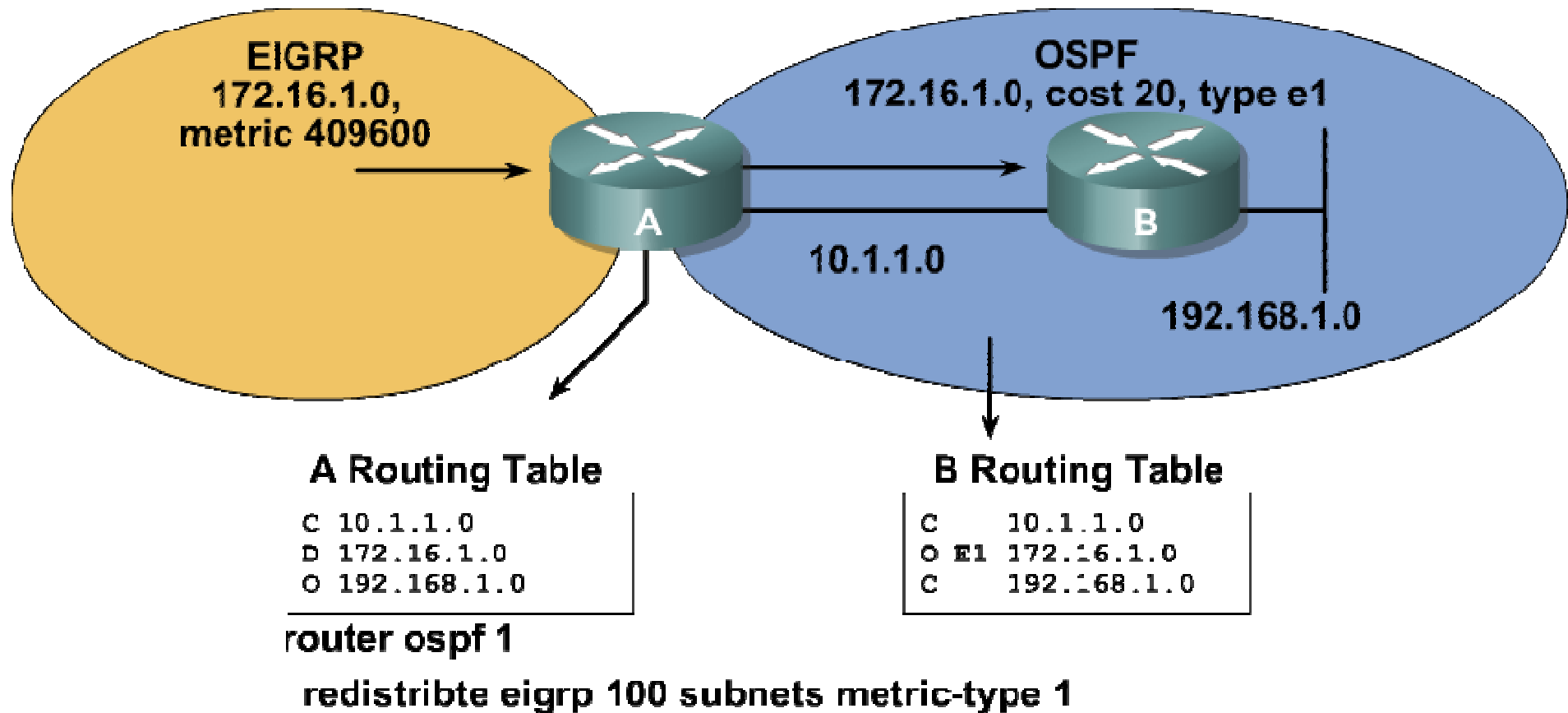


Redistribúcia do OSPF

```
Router (config-router) # redistribute protocol  
  [process-id] [metric metric-value]  
  [metric-type type-value] [route-map map-tag]  
  [subnets] [tag tag-value]
```

- Štandardná počiatočná metrika je 20 (ak BGP, potom 1)
- Redistribuované smery sa vždy prenášajú ako externé smery (LSA5 resp. LSA7), štandardne typ 2
- Podsieť sa štandardne neredistribuuje (preberú sa iba classful siete)
 - Redistribúcia podsieťovaných rozsahov sa povolí parametrom **subnets**

Redistribúcia do OSPF – príklad



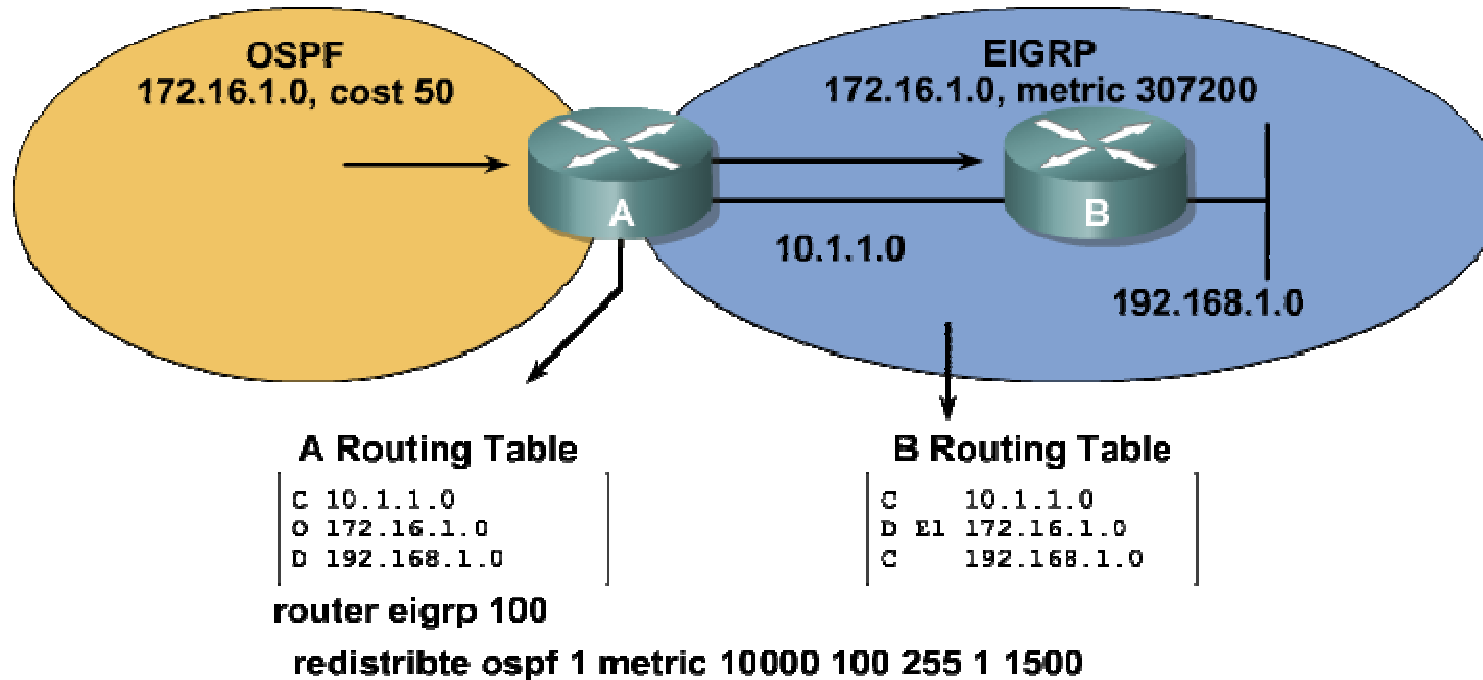
Redistribúcia do EIGRP

```
Router(config-router)# redistribute protocol  
[process-id] [match {internal | external 1 |  
external 2}] [metric metric-value] [route-map  
map-tag]
```

```
RtrA(config)# router eigrp 100  
RtrA(config-router)# redistribute ospf ?  
  
<1-65535> Process ID  
RtrA(config-router)# redistribute ospf 1 ?  
  
match          Redistribution of OSPF routes  
metric         Metric for redistributed routes  
route-map      Route map reference  
  
...  
<cr>
```

- Štandardná počiatočná metrika je nekonečno

Redistribúcia do EIGRP – príklad



- **B**andwidth v kilobitoch = 10000
- **D**elay v desiatkach mikrosekúnd = 100
- **R**eliability = 255 (maximum)
- **L**oad = 1 (minimum)
- **M**TU = 1500 bajtov

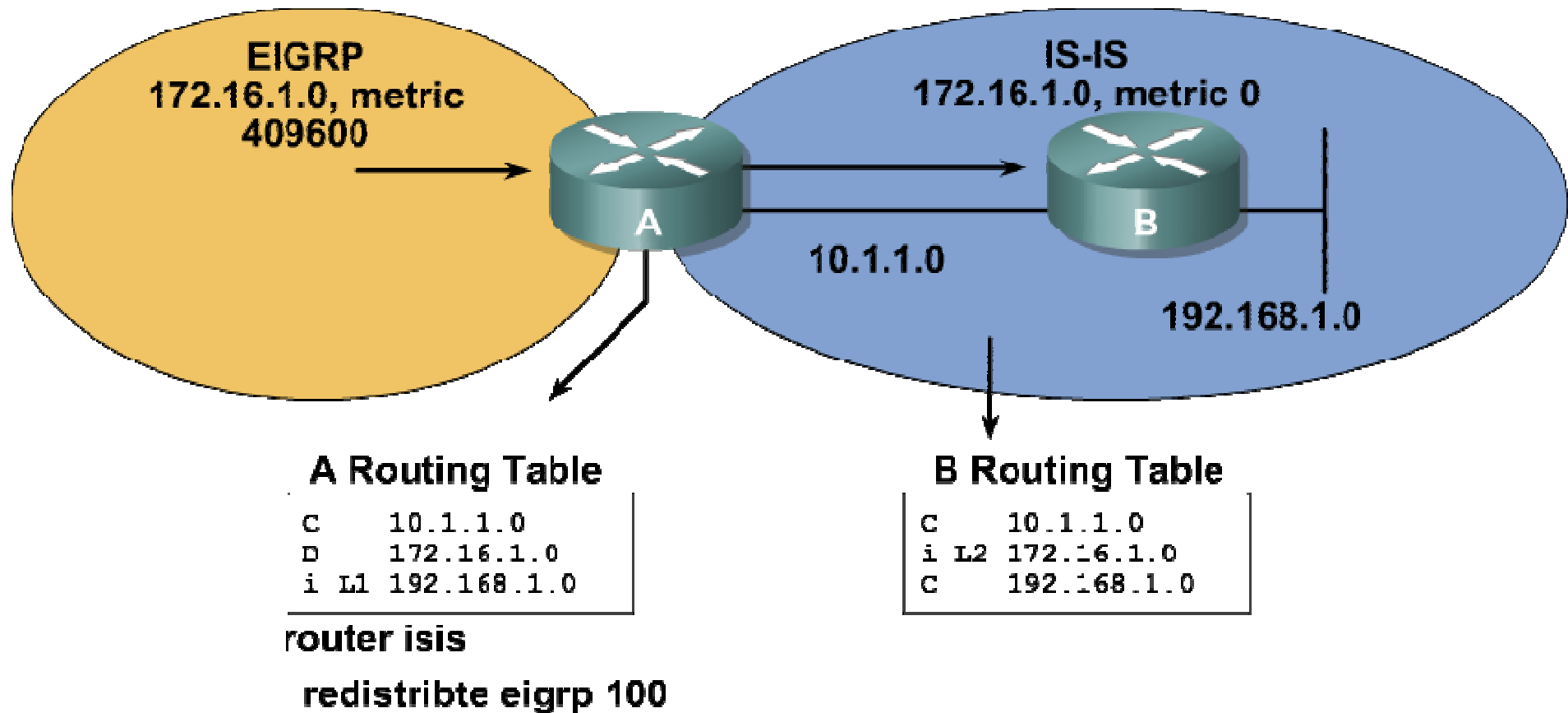
Redistribúcia do IS-IS

```
Router(config-router)# redistribute protocol  
[process-id] [level level-value]  
[metric metric-value] [metric-type type-value]  
[route-map map-tag]
```

```
RtrA(config)# router isis  
RtrA(config-router)# redistribute eigrp 100 ?  
  
level-1      IS-IS level-1 routes only  
level-1-2    IS-IS level-1 and level-2 routes  
level-2      IS-IS level-2 routes only  
metric       Metric for redistributed routes  
metric-type  OSPF/IS-IS exterior metric type for redistributed routes  
route-map    Route map reference  
..  
Output Omitted
```

- Smery sa štandardne importujú do Level2 so štartovacou metrikou 0

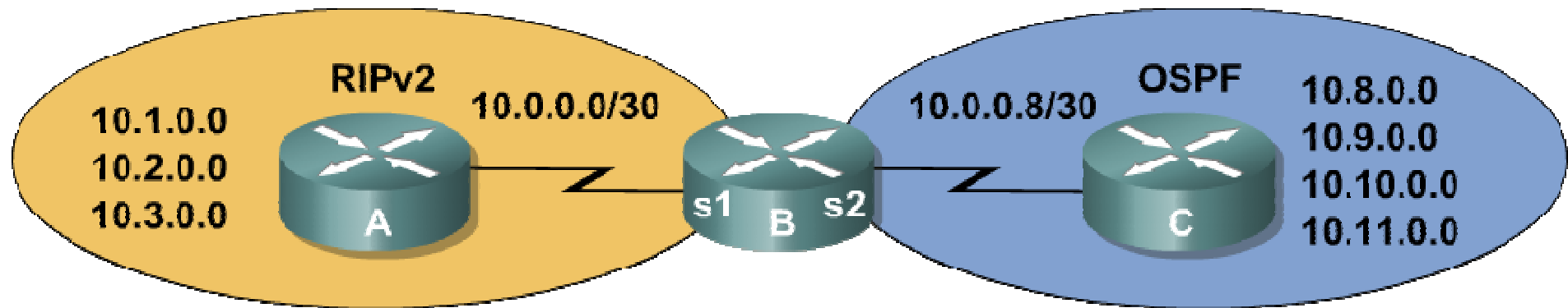
Redistribúcia do IS-IS – príklad



Príklad na redistribúciu



Pred redistribúciou

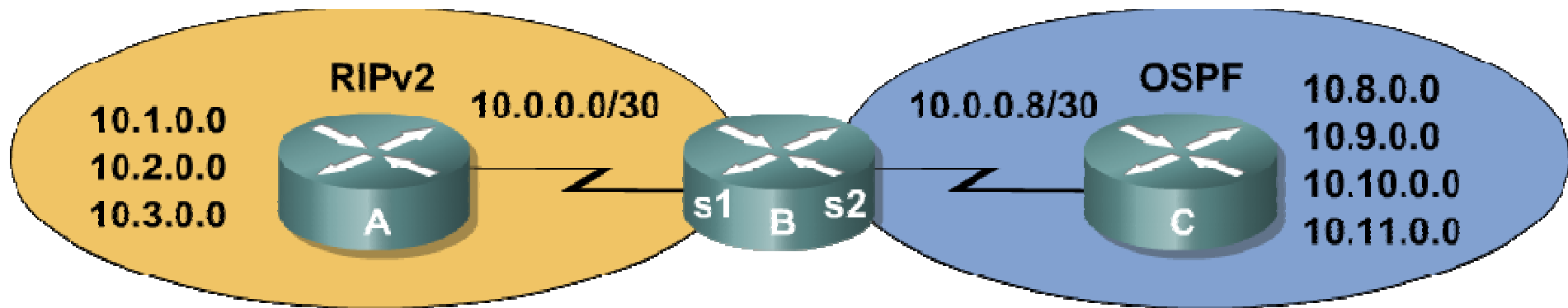


Router B Configuration

```
router ospf 1
  network 10.0.0.8 0.0.0.3 area 0

router rip
  network 10.0.0.0
  version 2
  passive-interface s2
```

Pred redistribúciou



A Routing Table

C	10.0.0.0
R	10.1.0.0
R	10.2.0.0
R	10.3.0.0

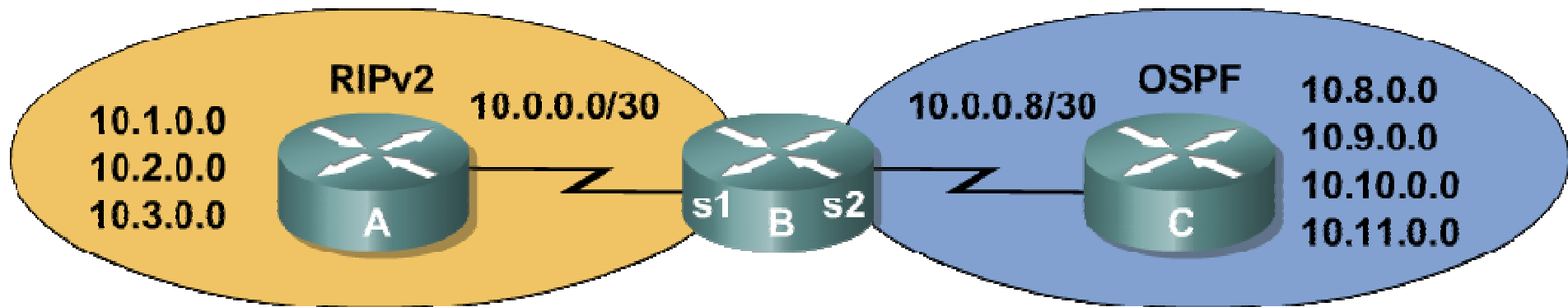
B Routing Table

C	10.0.0.0
C	10.0.0.8
R	10.1.0.0
R	10.2.0.0
R	10.3.0.0
O	10.8.0.0
O	10.9.0.0
O	10.10.0.0
O	10.11.0.0

C Routing Table

C	10.0.0.8
O	10.8.0.0
O	10.9.0.0
O	10.10.0.0
O	10.11.0.0

Redistribúcia na smerovači B

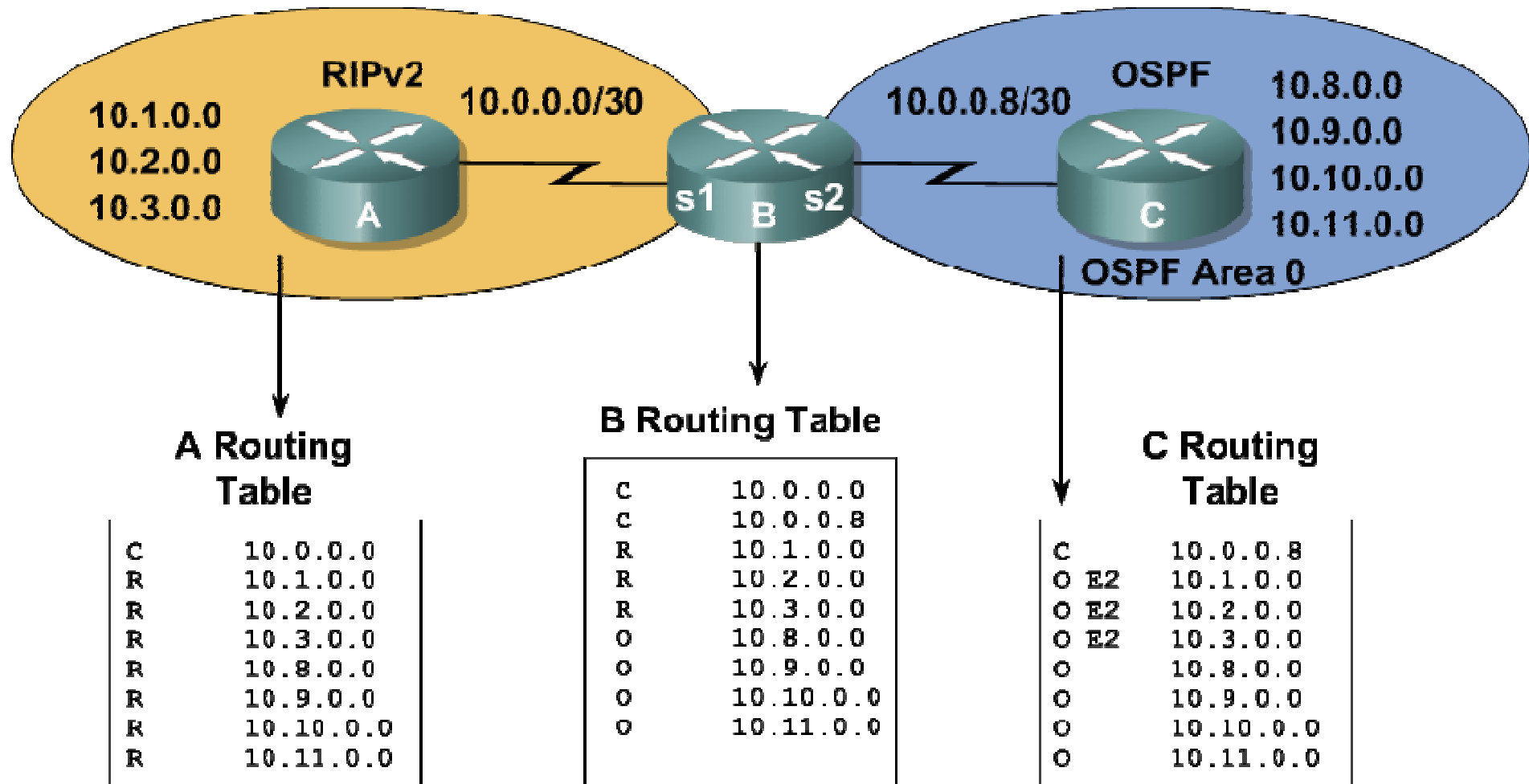


Router B Configuration

```
router ospf 1
 network 10.0.0.8 0.0.0.3 area 0
 redistribute rip subnets metric 300

router rip
 network 10.0.0.0
 version 2
 passive-interface s2
 redistribute ospf 1 metric 5
```

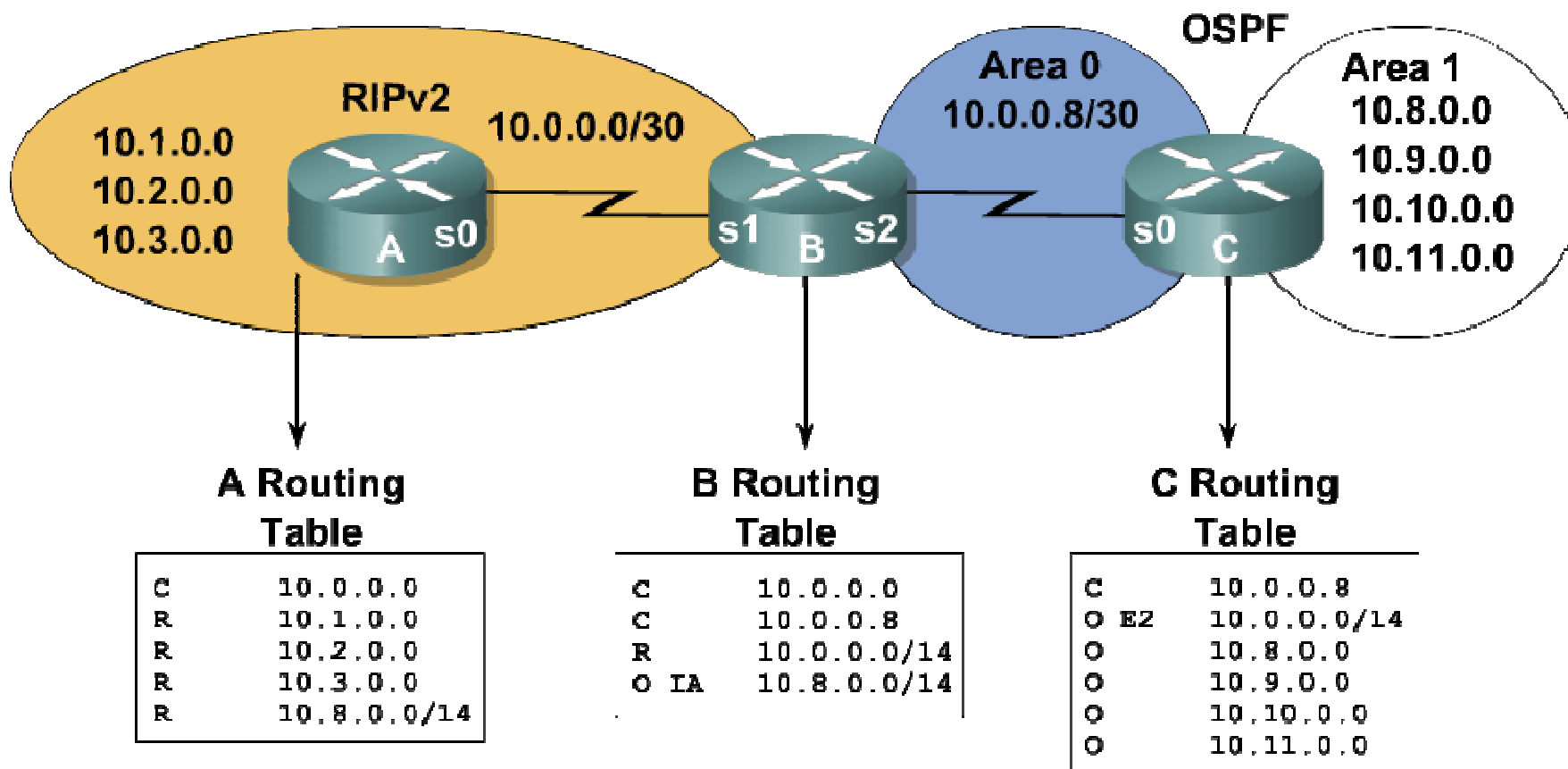
Po nakonfigurovaní redistribúcie



Konfigurácia sumarizácie, výsledné smerovacie tabuľky

```
RouterA(config) #interface s0
RouterA(config-if) #ip summary-address rip 10.0.0.0 255.252.0.0
```

```
RouterC(config) #router ospf 1
RouterC(config-router) #area 1 range 10.8.0.0 255.252.0.0
```



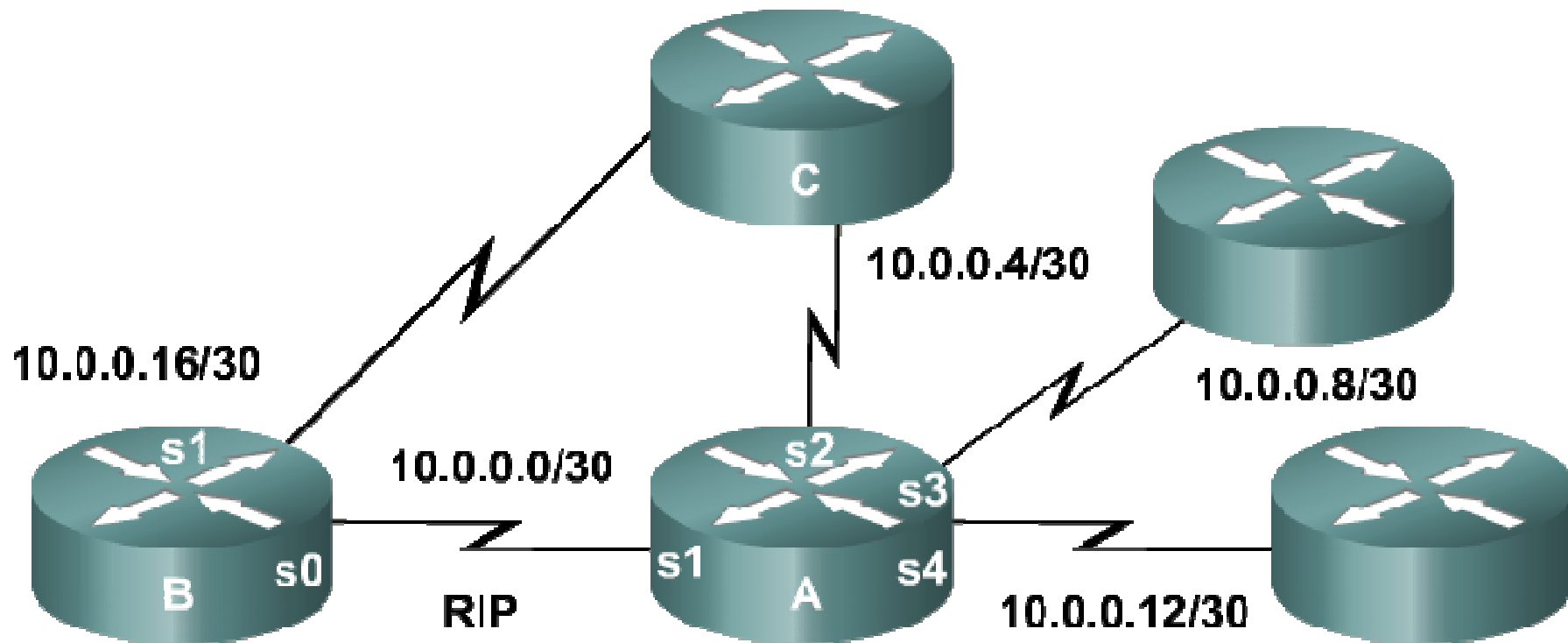
Riadenie obsahu smerovacích aktualizácií



Riadenie obsahu smerovacích aktualizácií

- Cisco smerovače majú mnohé prostriedky, ktorými dokážu riadiť obsah a rozposielanie smerovacích informácií
- Pasívne rozhrania
 - Týmito rozhraniami sa neodosielajú nijaké smerovacie informácie
- Distribučné zoznamy
 - Filtrujú obsah smerovacích informácií, ktoré v rámci daného smerovacieho protokolu posielame alebo prijímame
- Route-map
 - Filtrujú a upravujú obsah smerovacích informácií, ktoré do daného smerovacieho protokolu redistribuujeme
 - Niektoré smerovacie protokoly dovoľujú využiť route-map aj v distribučných zoznamoch alebo v príkaze neighbor

Príkaz passive-interface



Router B Configuration

```
router rip
network 10.0.0.0
passive-interface s1
```

Router A Configuration

```
router rip
network 10.0.0.0
passive-interface default
no passive-interface s1
```

Príkaz `passive-interface`

- Pasívne rozhrania v zásade neodosielajú pakety smerovacieho protokolu, čiže na danom rozhraní efektívne odfiltrujú úplne všetky siete odosielané daným protokolom
 - V protokole RIP však pasívne rozhranie akceptuje prijaté RIP pakety. Pokiaľ je to neželané, treba ich odfiltrovať pomocou ACL
 - V protokoloch používajúcich Hello pakety (EIGRP, OSPF, IS-IS) pasívne rozhranie neposiela ani neprijíma pakety príslušného smerovacieho protokolu
 - V IS-IS definovanie pasívneho rozhrania navyše spôsobí, že sa bude sieť z neho bude oznamovať, a to bez toho, aby na rozhraní bol príkaz **`ip router isis`**

Príkaz `passive-interface`

- Pasívne rozhrania možno vymenovať v konfigurácii daného smerovacieho protokolu:
 - Buď vymenovaním konkrétnych rozhraní príkazom `passive-interface IF`
 - Alebo konštruktom `passive-interface default` a následným „aktivovaním“ konkrétnych rozhraní príkazom `no passive-interface IF`

Distribučné zoznamy

- Ak je potrebné z prenášaných informácií v rámci jedného smerovacieho protokolu odfiltrovať neželané siete, je to možné realizovať tzv. distribučnými zoznamami (distribution list)
- Medzi IGP sa chovanie distribučných zoznamov líši:
 - Distance-vector protokoly (RIP, EIGRP) pomocou distribučných zoznamov dokážu odfiltrovať ľubovoľnú sieť prijatú alebo odoslanú v smerovacej správe (pakete)
 - OSPF v smere **in** filtruje len cesty, ktoré sa dostanú do smerovacej tabuľky, ale zmena v LSDB nenastane; v smere **out** filtruje iba externé smery, ak je daný router v úlohe ASBR
 - Protokol IS-IS nepodporuje distribučné zoznamy

Distribučné zoznamy

- Konfigurácia:

```
Router(config-router)# distribute-list { 1-199 |  
1300-2699 | NAME | prefix NAME | route-map NAME }  
{in | out} [interface]
```

- Parametre:

- Číslo alebo meno: identifikácia ACL
 - prefix NAME: identifikácia tzv. prefix listu
 - route-map NAME: identifikácia tzv. route-map
 - in/out: smer, v ktorom sa má filtrovanie diať
 - interface: rozhranie, na ktorom sa filtrovanie informácií má diať
- Nie všetky parametre a ich kombinácie sú povolené

Distribučné zoznamy a ACL

- V distribučných zoznamoch sa filtrujú siete
 - Sieť je dvojica [IP adresa siete, maska siete]
- Do ACL táto dvojica vstupuje ako fiktívny odosielateľ a príjemca paketu
 - IP adresa siete je „odosielateľ“
 - Maska siete je „adresát“
- Podľa typu ACL sa kontrolujú položky obsiahnuté v smerovacom pakete
 - Štandardné ACL kontroluje iba IP adresu siete, nie masku
 - Rozšírené ACL kontroluje aj IP adresu siete, aj jej masky
- Význam akcií:
 - Permit: povolí odoslanie resp. prijatie danej siete
 - Deny: zakáže odoslanie resp. prijatie danej siete

Distribučné zoznamy a prefix listy

- Použitie ACL na filtrovanie obsahu smerovacích info je síce použiteľné, ale trochu ťarbavé
- Prefix listy umožňujú to isté filtrovanie, ktoré je však zapísané efektívnejšie
- Prefix list je zoznam čísel (prefixov) sietí a rozsahov ich masiek spolu s definovanou akciou permit alebo deny

```
Router(config)# ip prefix-list MENO [seq N]  
                {permit | deny} SIEŤ/MASKA [ge D] [le H]
```

- Parametre:
 - seq N: sekvenčné číslo – poradie položky v prefix liste
 - SIEŤ/MASKA: udáva číslo (prefix) siete
 - ge D: stanovuje, že maska musí byť aspoň /D
 - le H: stanovuje, že maska musí byť najviac /H
 - Musí platiť $MASKA < D \leq H$

Distribučné zoznamy a prefix listy

- Povedané inak, riadku

```
ip prefix-list PL permit  
192.0.2.0/26 ge 28 le 31
```

bude vyhovovať každá sieť A.B.C.D/M, pre ktorú platia obe tieto podmienky:

- $A.B.C.D \& 255.255.255.192 = 192.0.2.0$
- $255.255.255.240 \leq M \leq 255.255.255.254$

Distribučné listy a prefix listy resp. ACL

- Prefix listy sú spracovávané efektívnejšie než ACL, preto by sa mali prednostne používať na implementovanie filtrovania obsahu smerovacích paketov
- Prefix listy sú určené len na použitie v súvislosti s riadením obsahu posielanej alebo prijímanej smerovacej informácie
 - Nie sú použiteľné ako náhrada ACL v iných aplikáciách, kde sa doposiaľ ACL používa
- Treba pamätať na to, že aj ACL, aj prefix listy končia implicitným pravidlom „deny any“
 - Každá sieť, ktorá nie je povolená, bude odfiltrovaná

Route Map

- Route map je sofistikovaná konštrukcia, ktorá dovoľuje vytvárať pravidlá v tvare if-then-else
- Základná idea: route-map sa skladá z jedného alebo viacerých blokov tvaru Test, Zmena, Akcia
 - Skontrolovať, či sú splnené predpísané predpoklady (**match**)
 - Ak áno, vykonať požadovanú akciu (**permit/deny**) a prípadne realizovať v obsahu paketu alebo v jeho putovaní nejaké zmeny (**set**)
- Spracovanie je podobné ako v ACL
 - Postupuje sa jednotlivými blokmi zhora nadol
 - Pri prvej zhode sa vykoná akcia spolu so zmenami a vyhodnocovanie končí
 - Na konci každej route-map je implicitné „match any / deny“
 - Bloky sú očíslované
- Route-map ako celok má meno, každý blok musí niesť to isté meno

Použitie konštrukcie route-map

- Route-map má široké využitie vo viacerých aplikáciách, preto je jej špecifikácia pomerne rozsiahla a univerzálna
 - Riadenie redistribúcie alebo filtrovanie obsahu rozosielaných smerovacích informácií
 - Rozširuje možnosť filtrovania, ako ju poznáme z distribučných zoznamov, o schopnosť čiastočných úprav obsahu
 - Policy-based routing
 - Schopnosť smerovať tok paketov nielen na základe príjemcu, ale i dodatočných kritérií
 - BGP politiky
 - Route-map je jeden z kľúčových prostriedkov efektívneho využívania protokolu BGP

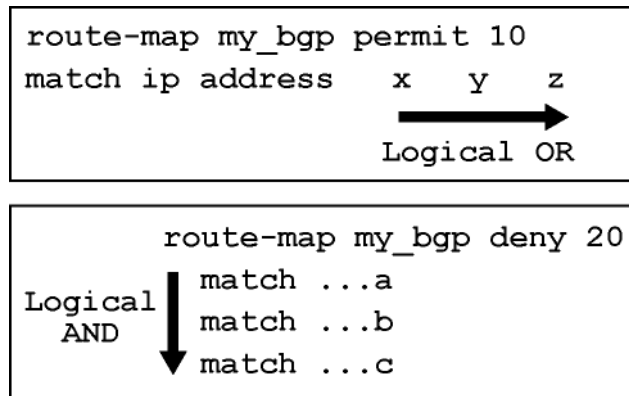
Tvar a činnosť route-map konštruktu

- Route-map pozostáva z blokov, každý blok má nepovinnú časť pre test (**match**), nepovinnú časť pre zmeny (**set**) a povinnú časť pre akciu (**permit/deny**)
- Bloky sa vyhodnocujú zhora nadol, konkrétne ich časti pre test
- Blok, v ktorom sa nájde prvá zhoda, uplatní svoje zmeny a akciu. Tým vyhodnocovanie konkrétnej route-map končí
- Poradové čísla sa využívajú pri vsúvaní alebo odstraňovaní konkrétneho bloku z konkrétnej pozície

```
route-map my_bgp permit 10
    { match statements }
    { match statements }
    { set statements }
    { set statements }
route-map my_bgp deny 20
    ::          ::  ::
    ::          ::  ::
route-map my_bgp permit 30
    ::          ::  ::
    ::          ::  ::
```

Tvar a činnosť route-map konštruktu

- Príkaz match môže obsahovať viacero argumentov v jednom riadku
 - Medzi nimi platí logické OR – stačí zhoda v jednom argumente



- Ak blok obsahuje viaceré match príkazy, musia byť splnené všetky – platí logické AND
- Ak testovaná sieť resp. paket vyhovelo všetkým riadkom match v bloku, akcia permit alebo deny v záhlaví bloku určí definitívny osud

Príkazy na tvorbu route-map

```
router(config) #
```

```
route-map map-tag [permit | deny] [sequence-number]
```

- Vytvorí blok route-map a definuje akciu

```
router(config-route-map) #
```

```
match {conditions}
```

- V bloku route-map definuje test

```
router(config-route-map) #
```

```
set {actions}
```

- V bloku route-map definuje zmenu

```
router(config-router) #
```

```
redistribute protocol [process id] route-map map-tag
```

- Príklad použitia definovanej route-map v redistribúcii

Príkazy match

- Príkaz `match` tvoria testovaciu časť bloku `route-map`
- Akcia v záhlaví tohto bloku sa vykoná na všetkých paketoch resp. sieťach, ktoré testom vyhoveľi

```
router (config-route-map) #
```

```
match {options}

  options :
  ip address ip-access-list
  ip route-source ip-access-list
  ip next-hop ip-access-list
  interface type number
  metric metric-value
  route-type [external | internal | level-1 | level-2 |local]
  ...
```

Príkazy match

Príkaz	Popis
<code>match community</code>	Testuje BGP atribút Community
<code>match interface</code>	Testuje, či next hop pre danú cieľovú sieť leží v sieti za uvedeným rozhraním
<code>match ip address</code>	Testuje tvar cieľovej siete a jej masku voči definovanému ACL alebo prefix listu
<code>match ip next-hop</code>	Testuje next-hop cieľovej siete voči definovanému ACL
<code>match ip route-source</code>	Testuje autora danej smerovacej informácie voči definovanému ACL
<code>match length</code>	Testuje L3 dĺžku paketu
<code>match metric</code>	Testuje metriku cieľovej siete
<code>match route-type</code>	Testuje typ cieľovej siete
<code>match tag</code>	Testuje návěstie (značku, tag) cieľovej siete

Príkazy set

- Príkaz **set** ovplyvňuje obsah smerovacej informácie alebo postup paketu

```
router (config-route-map) #
```

```
set {options}
  options :
  metric metric-value
  metric-type [type-1 | type-2 | internal | external]
  level [level-1 | level-2 | level-1-2 | stub-area | backbone]
  ip next-hop next-hop-address
```

Príkazy set

Príkaz	Popis
<code>set as-path</code>	Modifikuje AS cestu v BGP
<code>set automatic-tag</code>	Automaticky určí hodnotu značky v cieľovej sieti
<code>set community</code>	Nastaví hodnotu BGP atribútu Community
<code>set default interface</code>	Interfejs, ktorým preposlať pakety, pre ktoré nie je v smerovacej tabuľke explicitná cieľová sieť
<code>set interface</code>	Interfejs, ktorým preposlať pakety
<code>set ip default next-hop</code>	Brána, na ktorú preposlať pakety, pre ktoré nie je v smerovacej tabuľke explicitná cieľová sieť
<code>set ip next-hop</code>	Brána, na ktorú preposlať pakety
<code>set level</code>	Definuje, kam importovať externé smery v IS-IS alebo OSPF
<code>set local-preference</code>	Nastaví hodnotu BGP atribútu local preference
<code>set metric</code>	Nastaví metriku cieľovej siete
<code>set metric-type</code>	Nastaví typ metriky cieľovej siete
<code>set tag</code>	Nastaví značku cieľovej siete
<code>set weight</code>	Nastaví hodnotu BGP atribútu weight

Použitie route-map v redistribúcii

```
Router(config)# router ospf 10
Router(config-router)# redistribute rip route-map redis-rip
```

- Cieľové siete vyhovujúce ACL 23 alebo 29 sa redistribuujú ako OSPF E1 smery s metrikou 500
- Cieľové siete vyhovujúce ACL 37 sa neredistribuujú
- Všetky ostatné smery sa importujú ako E2 s metrikou 5000

```
Router(config)#
route-map redis-rip permit 10
match ip address 23 29
set metric 500
set metric-type type-1

route-map redis-rip deny 20
match ip address 37

route-map redis-rip permit 30
set metric 5000
set metric-type type-2
```

```
Router(config)#
access-list 23 permit 10.1.0.0 0.0.255.255
access-list 29 permit 172.16.1.0 0.0.0.255
access-list 37 permit 10.0.0.0 0.255.255.255
```

Úprava administratívnej vzdialenosti

Router (config-router) #

```
distance administrative-distance [address wildcard-mask  
[access-list-number | name]]
```

- Platí pre všetky smerovacie protokoly okrem EIGRP a BGP
 - address: adresa smerovača, ktorý oznamuje túto info
 - wildcard-mask: maska pre adresu smerovača (môže ich byť viac)
 - ACL resp. name: dodatočné ACL na výber siete, ktorej upravíme AD
- Používa sa pri zložitejších redistribúciách, avšak definitívne použitie si treba veľmi dôkladne naplánovať

Router (config-router) #

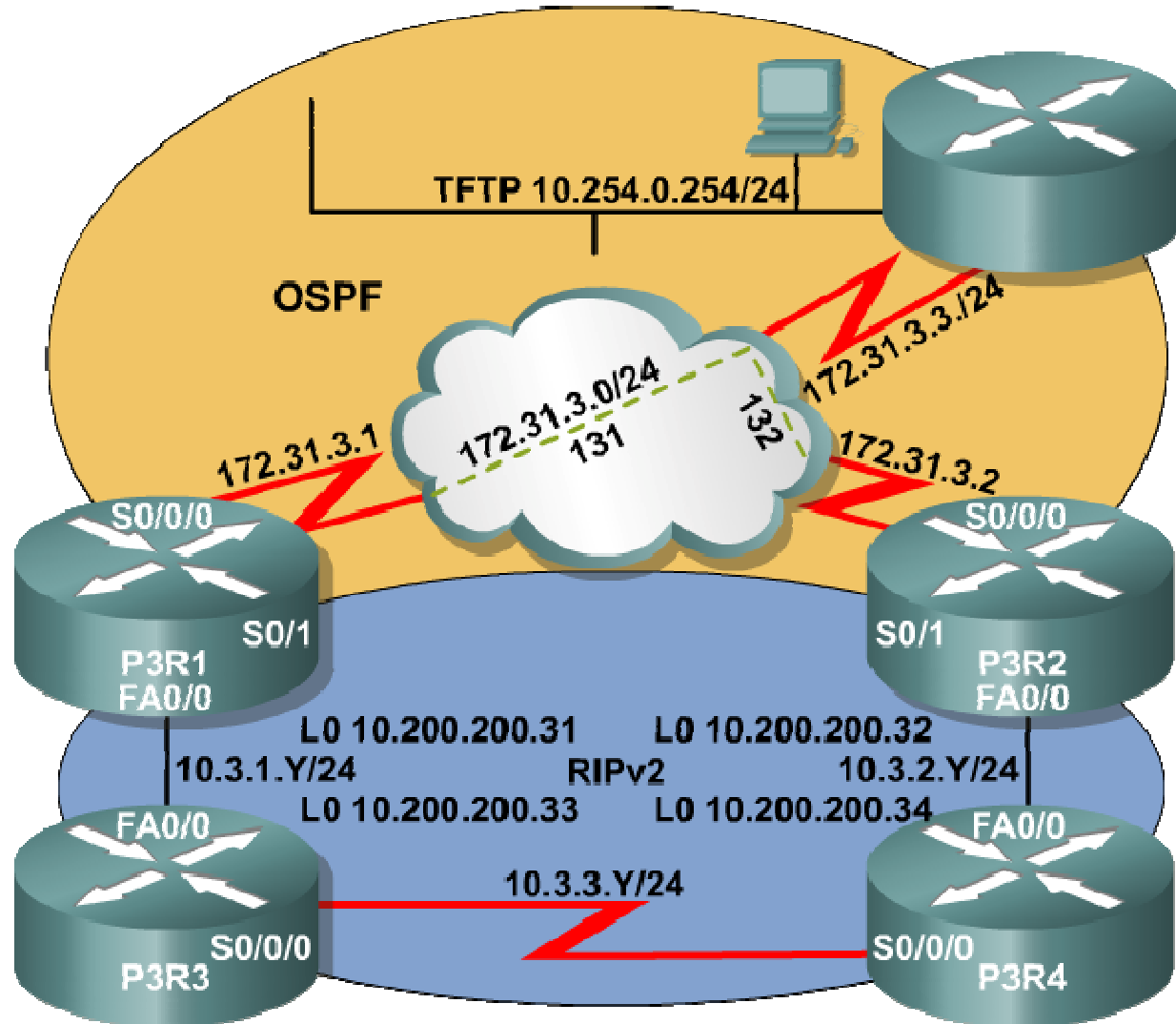
```
distance eigrp internal-distance external-distance
```

- Definovanie interných a externých AD pre EIGRP

Príklad na úpravu administratívnej vzdialenosti



Základná topológia



Konfigurácia hraničných routerov

Router P3R1

```
router ospf 1
 redistribute rip metric 10000 metric-type 1 subnets
 network 172.31.0.0 0.0.255.255 area 0
!
router rip
 version 2
 redistribute ospf 1 metric 5
 network 10.0.0.0
 no auto-summary
```

Router P3R2

```
router ospf 1
 redistribute rip metric 10000 metric-type 1 subnets
 network 172.31.3.2 0.0.0.0 area 0
!
router rip
 version 2
 redistribute ospf 1 metric 5
 network 10.0.0.0
 no auto-summary
```

Výsledok po redistribúcii na P3R2

With OSPF and RIP running:



```
P3R2#show ip route
<Output Omitted>

Gateway of last resort is not set

    172.31.0.0/24 is subnetted, 7 subnets
O       172.31.55.0 [110/2342] via 172.31.3.3, 00:09:46, Serial0/0/0
C       172.31.3.0 is directly connected, Serial0/0/0
O       172.31.2.0 [110/1562] via 172.31.3.3, 00:09:46, Serial0/0/0
10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
O E1   10.3.1.0/24 [110/10781] via 172.31.3.1, 00:09:47, Serial0/0/0
O E1   10.3.3.0/24 [110/10781] via 172.31.3.1, 00:04:51, Serial0/0/0
C      10.3.2.0/24 is directly connected, fastethernet0/0
O E1   10.200.200.31/32 [110/10781] via 172.31.3.1, 00:09:48, Serial0/0/0
O E1   10.200.200.34/32 [110/10781] via 172.31.3.1, 00:04:52, Serial0/0/0
C      10.200.200.32/32 is directly connected, Loopback0
O E1   10.200.200.33/32 [110/10781] via 172.31.3.1, 00:04:52, Serial0/0/0
O E2   10.254.0.0/24 [110/50] via 172.31.3.3, 00:09:48, Serial0/0/0
```

- P3R2 includes suboptimal paths and loops.

Úprava administratívnych vzdialeností

```
hostname P3R1
!
router ospf 1
  redistribute rip metric 10000 metric-t 1 subnets
  network 172.31.0.0 0.0.255.255 area 0
  distance 125 0.0.0.0 255.255.255.255 64
!
router rip
  version 2
  redistribute ospf 1 metric 5
  network 10.0.0.0
  no auto-summary
!
access-list 64 permit 10.3.1.0 0.0.0.255
access-list 64 permit 10.3.3.0 0.0.0.255
access-list 64 permit 10.3.2.0 0.0.0.255
access-list 64 permit 10.200.200.31
access-list 64 permit 10.200.200.34
access-list 64 permit 10.200.200.32
access-list 64 permit 10.200.200.33
```

```
hostname P3R2
!
router ospf 1
  redistribute rip metric 10000 metric-t 1 subnets
  network 172.31.3.2 0.0.0.0 area 0
  distance 125 0.0.0.0 255.255.255.255 64
!
router rip
  version 2
  redistribute ospf 1 metric 5
  network 10.0.0.0
  no auto-summary
!
access-list 64 permit 10.3.1.0 0.0.0.255
access-list 64 permit 10.3.3.0 0.0.0.255
access-list 64 permit 10.3.2.0 0.0.0.255
access-list 64 permit 10.200.200.31
access-list 64 permit 10.200.200.34
access-list 64 permit 10.200.200.32
access-list 64 permit 10.200.200.33
```

Po úprave administratívnych vzdialeností

With OSPF changing administrative distance:



```
Gateway of last resort is not set
```

```
172.31.0.0/16 is variably subnetted, 8 subnets, 2 masks
O    172.31.55.4/32 [110/781] via 172.31.33.4, 00:00:01, Serial0/0/0
C    172.31.33.0/24 is directly connected, Serial0/0/0
O    172.31.33.1/32 [110/1562] via 172.31.33.4, 00:00:01, Serial0/0/0
O    172.31.33.4/32 [110/781] via 172.31.33.4, 00:00:01, Serial0/0/0
O    172.31.44.4/32 [110/781] via 172.31.33.4, 00:00:01, Serial0/0/0
O    172.31.22.4/32 [110/781] via 172.31.33.4, 00:00:01, Serial0/0/0
O    172.31.11.4/32 [110/781] via 172.31.33.4, 00:00:03, Serial0/0/0
O    172.31.66.4/32 [110/781] via 172.31.33.4, 00:00:03, Serial0/0/0
10.0.0.0/8 is variably subnetted, 8 subnets, 2 masks
R    10.3.1.0/24 [120/2] via 10.3.2.4, 00:00:03, FastEthernet0/0
R    10.3.3.0/24 [120/1] via 10.3.2.4, 00:00:03, FastEthernet0/0
C    10.3.2.0/24 is directly connected, FastEthernet0/0
R    10.200.200.31/32 [120/3] via 10.3.2.4, 00:00:04, FastEthernet0/0
R    10.200.200.34/32 [120/1] via 10.3.2.4, 00:00:04, FastEthernet0/0
C    10.200.200.32/32 is directly connected, Loopback0
R    10.200.200.33/32 [120/2] via 10.3.2.4, 00:00:04, FastEthernet0/0
O E2 10.254.0.0/24 [110/50] via 172.31.33.4, 00:00:04, Serial0/0/0
```

- Router P3R2 prefers RIP routes.

**Ten istý príklad riešený
o niečo inteligentnejším
spôsobom – značkováním**



Označkovanie redistribuovaných smerov

```
hostname P3R1
!
router ospf 1
  redistribute rip metric 10000 metric-type 1
  route-map rip2ospf subnets
  distribute-list route-map fromrip in
  network 172.31.0.0 0.0.255.255 area 0
!
router rip
  version 2
  redistribute ospf 1 metric 5 route-map ospf2rip
  network 10.0.0.0
  no auto-summary
!
route-map rip2ospf deny 10
  match tag 110

route-map rip2ospf permit 20
  set tag 120

route-map ospf2rip deny 10
  match tag 120

route-map ospf2rip permit 20
  set tag 110

route-map fromrip deny 10
  match tag 120

route-map fromrip permit 20
```

```
hostname P3R2
!
router ospf 1
  redistribute rip metric 10000 metric-type 1
  route-map rip2ospf subnets
  distribute-list route-map fromrip in
  network 172.31.3.2 0.0.0.0 area 0
!
router rip
  version 2
  redistribute ospf 1 metric 5 route-map ospf2rip
  network 10.0.0.0
  no auto-summary
!
route-map rip2ospf deny 10
  match tag 110

route-map rip2ospf permit 20
  set tag 120

route-map ospf2rip deny 10
  match tag 120

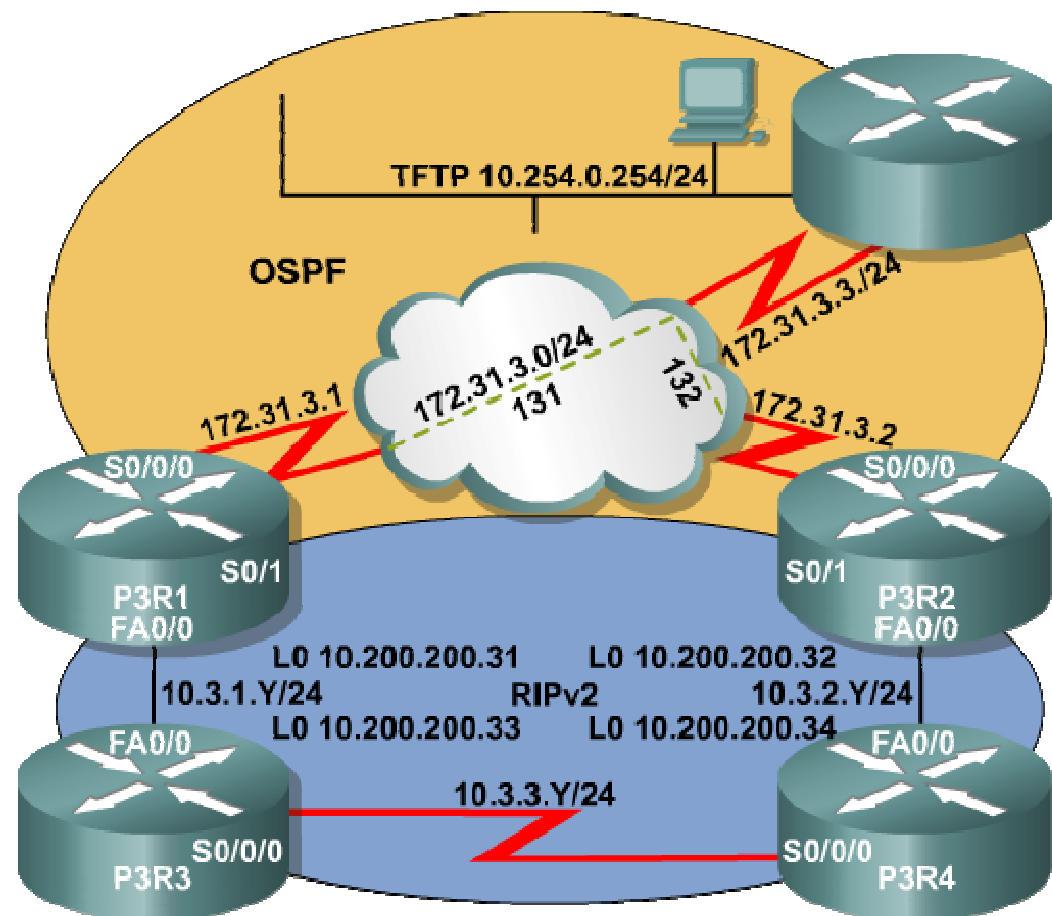
route-map ospf2rip permit 20
  set tag 110

route-map fromrip deny 10
  match tag 120

route-map fromrip permit 20
```

Know Your Network – heslo dňa ☺

- Je potrebné veľmi dôsledne poznať sieť ešte pred redistribúciou
- Problémové budú hlavne smerovače, ktoré sú na hraniciach alebo majú viac možností na výber
- Pri použití príkazu **distance** nedochádza k strate smerovacích info



IP Policy Routing

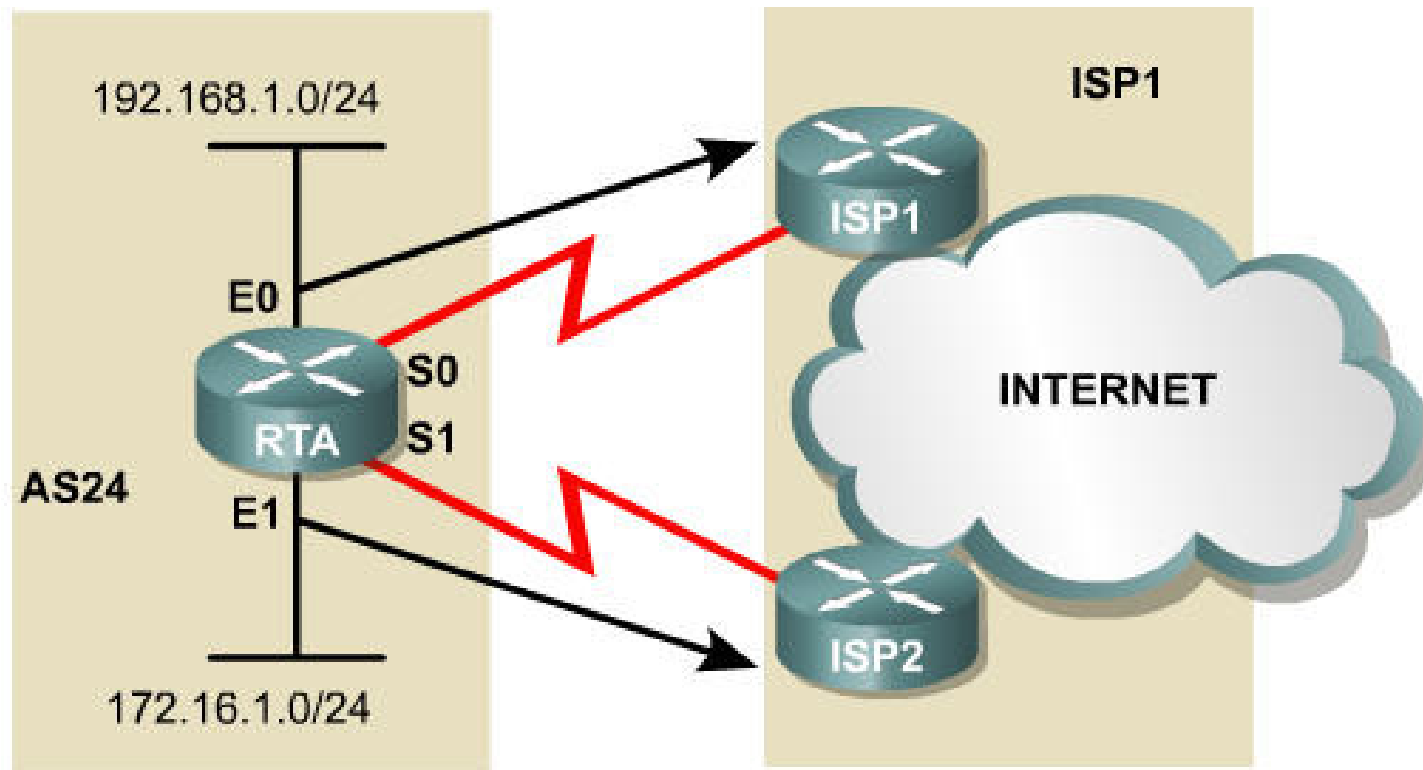


Policy routing

- Smerovanie na základe dodatočného kritéria
 - Odosielateľ
 - Veľkosť paketu
- Realizované pomocou route-map konštruktov
 - Časť **match** vyberie pakety
 - Akcia definuje, či budú pakety smerované pomocou policy routingu (**permit**) alebo podľa smerovacej tabuľky (**deny**)
 - Ak je akcia permit, časť **set** hovorí, ako sa paket prepošle
 - **set ip next-hop, set interface, set ip default next-hop, set default interface**
- Výsledný route-map sa aplikuje na vstupné rozhranie príkazom

```
Router(config-if) # ip policy route-map MENO
```

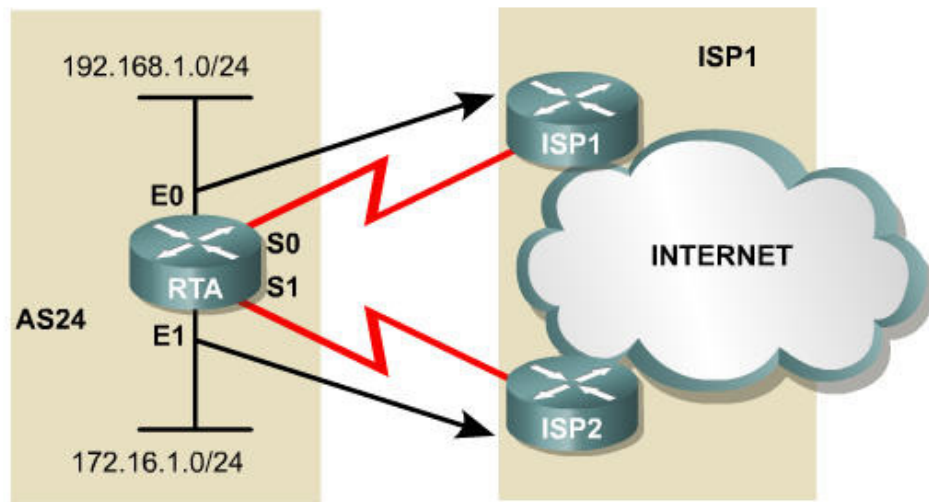
Typická aplikácia policy routingu



Policy routing on RTA can be used to route traffic based on its source network, in addition to its destination. Traffic from the 192.168.1.0 network will use the link to ISP1. Traffic from the 172.16.1.0 network will use the link to ISP2.

Riešenie úlohy

```
hostname RTA
!  
interface Ethernet 0  
  ip address 192.168.1.1 255.255.255.0  
  ip policy route-map ISP1  
  
interface Ethernet 1  
  ip address 172.16.1.1 255.255.255.0  
  ip policy route-map ISP2  
  
route-map ISP1 permit 10  
  match ip address 1  
  set interface Serial0  
  
route-map ISP2 permit 10  
  match ip address 2  
  set interface Serial1  
  
access-list 1 permit 192.168.1.0 0.0.0.255  
access-list 2 permit 172.16.1.0 0.0.0.255
```



Network Address Translation a route-map

- Route-map sa s výhodou dá využiť aj pri NAT
- Úloha na voľný čas a rozmýšľanie:
 - Máme dvoch ISP, prípadne používame aj policy routing
 - Každý ISP má vlastný adresný priestor
 - Podľa toho, cez ktorého ISP prechádzame, musíme realizovať NAT do príslušného poolu
 - Ako to urobiť?

Konfigurácia DHCP



Konfigurácia DHCP servera

```
Router(config)#ip dhcp pool [pool name]
```

- Vytvorí DHCP pool

```
Router(config-dhcp)#network [network address][subnet mask]
```

- Definuje sieť a jej masku, z ktorej pool prideliuje adresy

```
Router(config-dhcp)#default-router [host address]
```

- Definuje bránu (pridelenú klientom)

```
Router(config)#ip dhcp excluded-address low-address high-address
```

- Špecifikuje vyhradený (nepridelovaný) rozsah IP

Nepovinné príkazy v konfigurácii DHCP

```
Router (config-dhcp) #domain-name domain
```

- Definuje doménové meno

```
Router (config-dhcp) #dns-server address
```

- Definuje DNS server, ktorý má klient používať

```
Router (config-dhcp) #netbios-name-server address
```

- Definuje IP adresu WINS servera

```
Router (config-dhcp) #lease {days [hours] [minutes] | infinite}
```

- Definuje dĺžku výpožičky

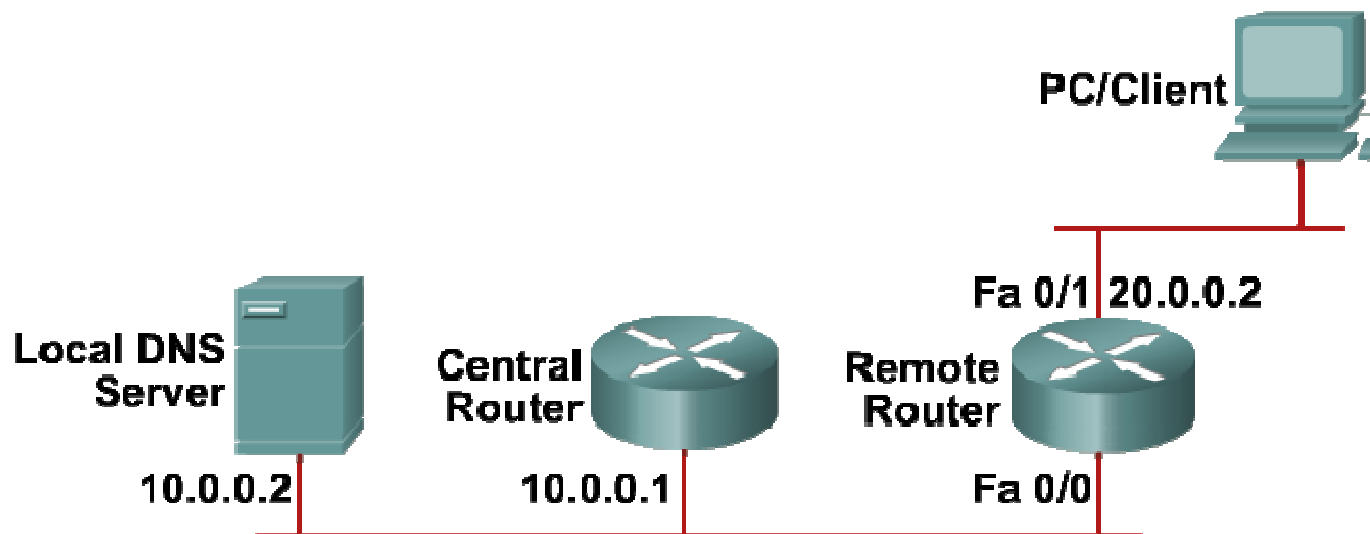
Konfigurácia databázového agenta

```
Router(config)#ip dhcp database url [timeout seconds | write-  
delay seconds]
```

- Konfiguruje databázového klienta a špecifikuje, kde si bude DHCP server ukladať databázu väzieb

```
ip dhcp database ftp://user:passwords@172.16.4.253/router-dhcp write-delay 120  
  
ip dhcp excluded-address 172.16.1.100 172.16.1.103  
  
ip dhcp excluded-address 172.16.2.100 172.16.2.103  
  
ip dhcp pool SomePool  
  
    network 172.16.0.0 /16  
  
    domain-name global.com  
  
    dns-server 172.16.1.102 172.16.2.102  
  
    netbios-name-server 172.16.2.103 172.16.2.103  
  
    default-router 172.16.1.100
```

Import a autokonfigurácia



```
Router (config-dhcp) #import all
```

- Príkaz importuje niektoré nastavenia do poolu, takže ich nie je potrebné zadávať znovu a stále

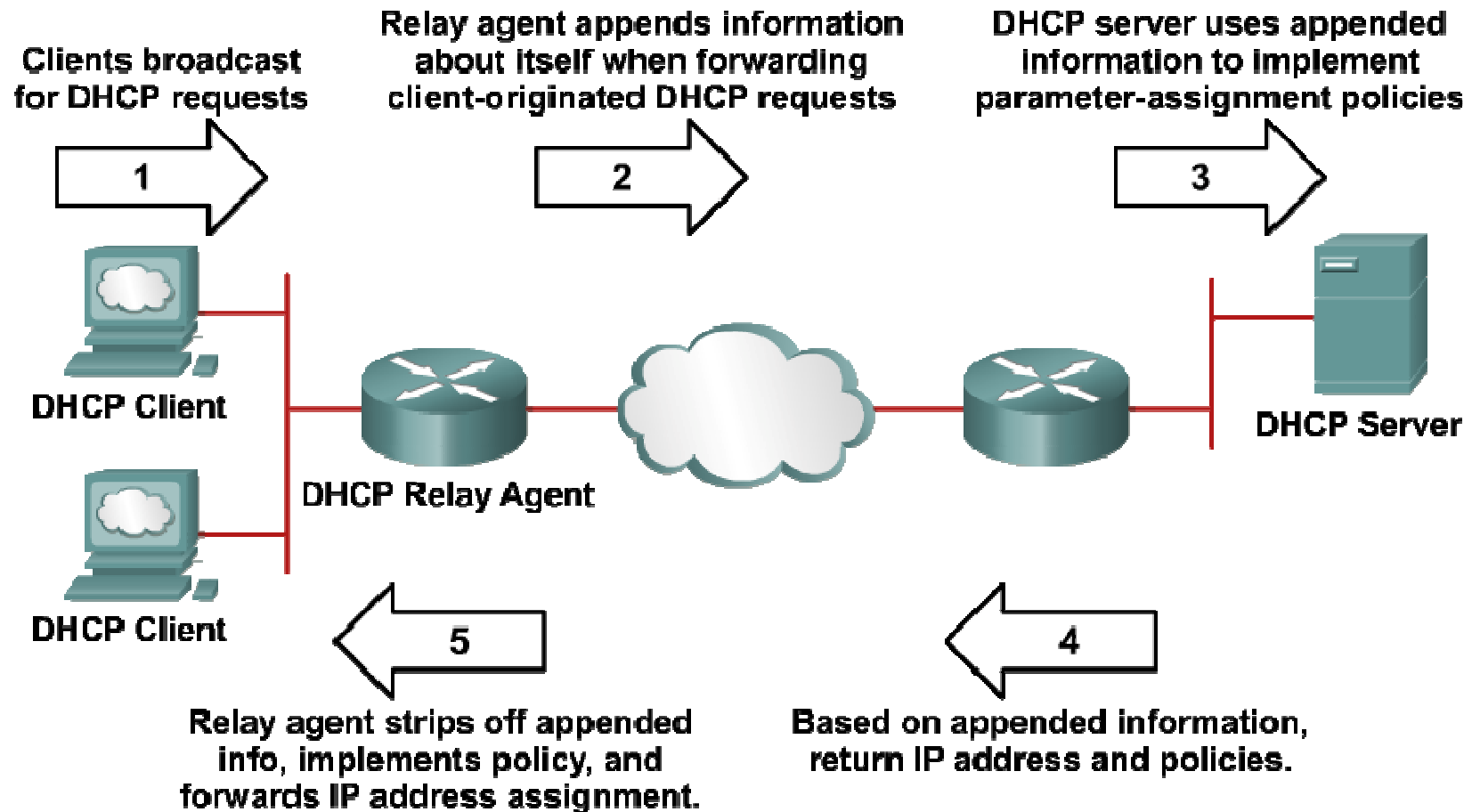
Import a autokonfigurácia

```
ip dhcp-excluded address 10.0.0.1 10.0.0.5
ip dhcp pool central
  network 10.0.0.0 255.255.255.0
  default-router 10.0.0.1 10.0.0.5
  domain name central.com
  dns-server 10.0.0.2
  netbios-name-server 10.0.0.2
interface fastethernet0/0
  ip address 10.0.0.1 255.255.255.0
```



```
ip dhcp-excluded address 20.0.0.2
ip dhcp pool client
  network 20.0.0.0 255.255.255.0
  default-router 20.0.0.2
  import all
interface fastethernet0/0
  ip address dhcp
```

Relay Agent – Option 82



DHCP Relay a Relay Agent Option 82

- DHCP Relay a DHCP Relay Agent Option 82 nie sú to isté
 - DHCP Relay je služba, ktorá umožňuje preposlať broadcastové DHCP správy na preddefinovanú unicastovú adresu, a je sprostredkovaná tzv. Relay Agentom
 - Pritom sa do DHCP správy zapíše aj IP adresa DHCP Relay Agentu, aby DHCP server vedel, kam má poslať odpoveď
 - Konfiguruje sa známym príkazom **ip helper-address A.B.C.D** na sieťovom rozhraní, kde očakávame DHCP klientov
 - DHCP Relay Agent Option 82 je rozšírenie funkčnosti DHCP Relay o ďalšiu dodatočnú informáciu – Option 82, ktorá nejakým spôsobom presnejšie identifikuje, odkiaľ požiadavka prichádza
 - Napr. na ktorom porte prepínača je klient pripojený