

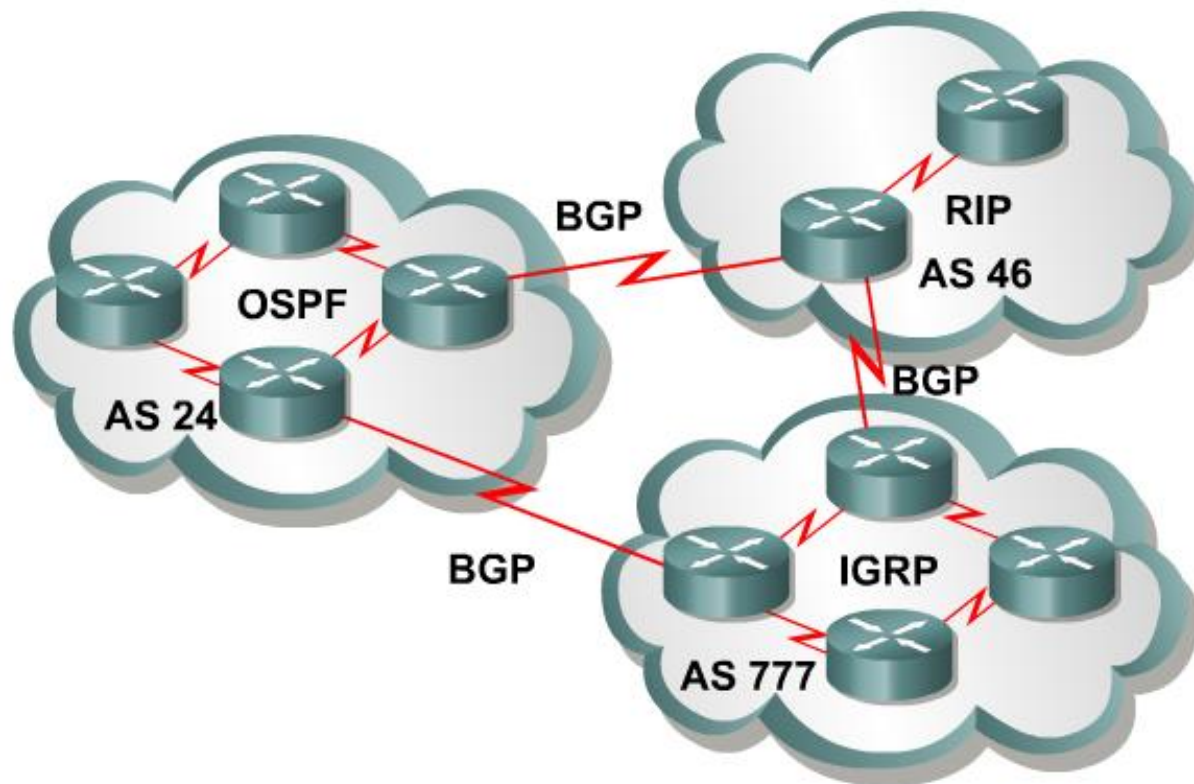


Border Gateway Protocol (BGP)



ROUTE Module 7

Logický pohľad na Internet



Internet je skupina navzájom poprepájaných
Autonómnych systémov (AS)

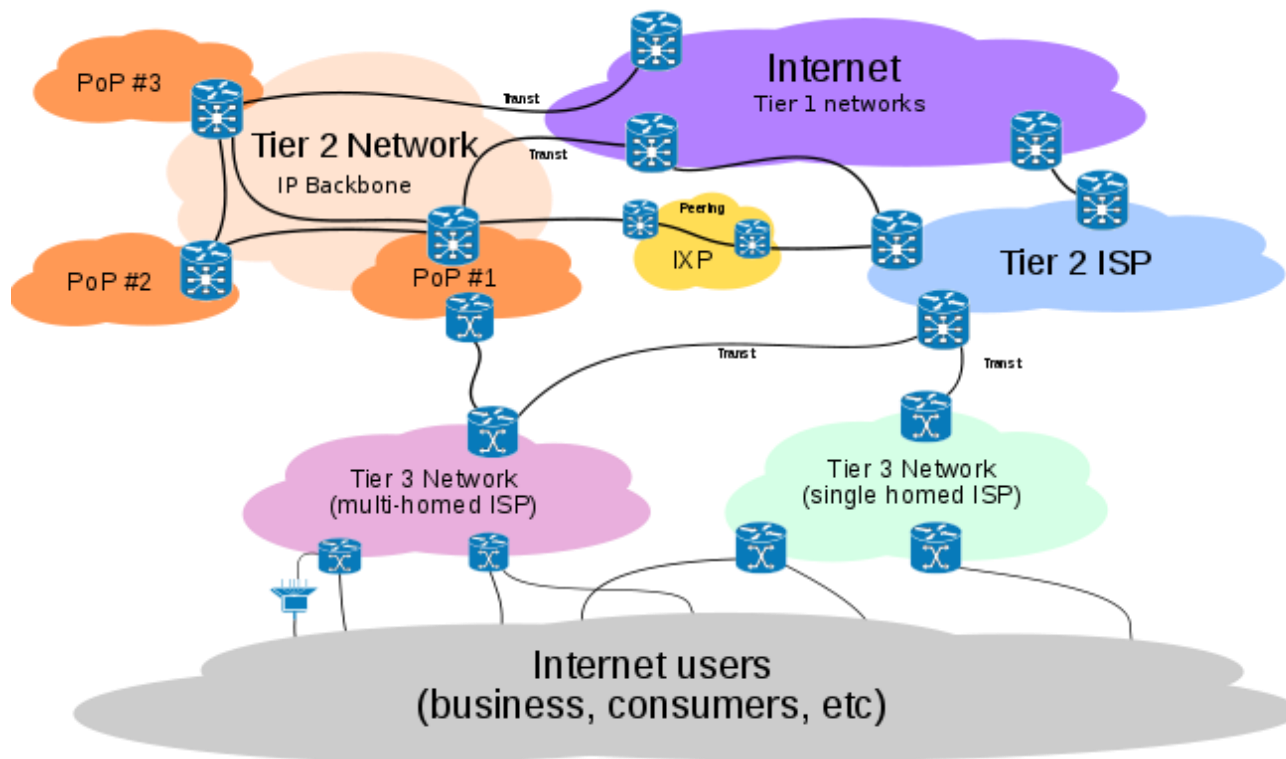
Autonómny systém

- Autonómny systém (AS) je skupina sietí a smerovačov, ktorá používajú spoločnú smerovaciu politiku a patria pod spoločnú administratívnu doménu
 - Smerovacia politika: spôsob výberu ciest do rôznych cieľov, filtrovanie smerovacích informácií, oznamovanie smerov...
 - Administratívna doména: dosah administratívnej právomoci správcu
- Vo vnútri AS môže pracovať jeden alebo niekoľko IGP
 - AS však ako celok patrí spravidla jednej organizácii
- Zvonku je AS vnímaný ako jedna nerozdelená entita
 - Všetky členské siete v AS sú v ňom z pohľadu iných AS priamo dostupné
- V prípade, že je AS pripojený na verejný Internet s použitím EGP (BGP)
 - AS musí byť pridelené IANA

Autonómny systém

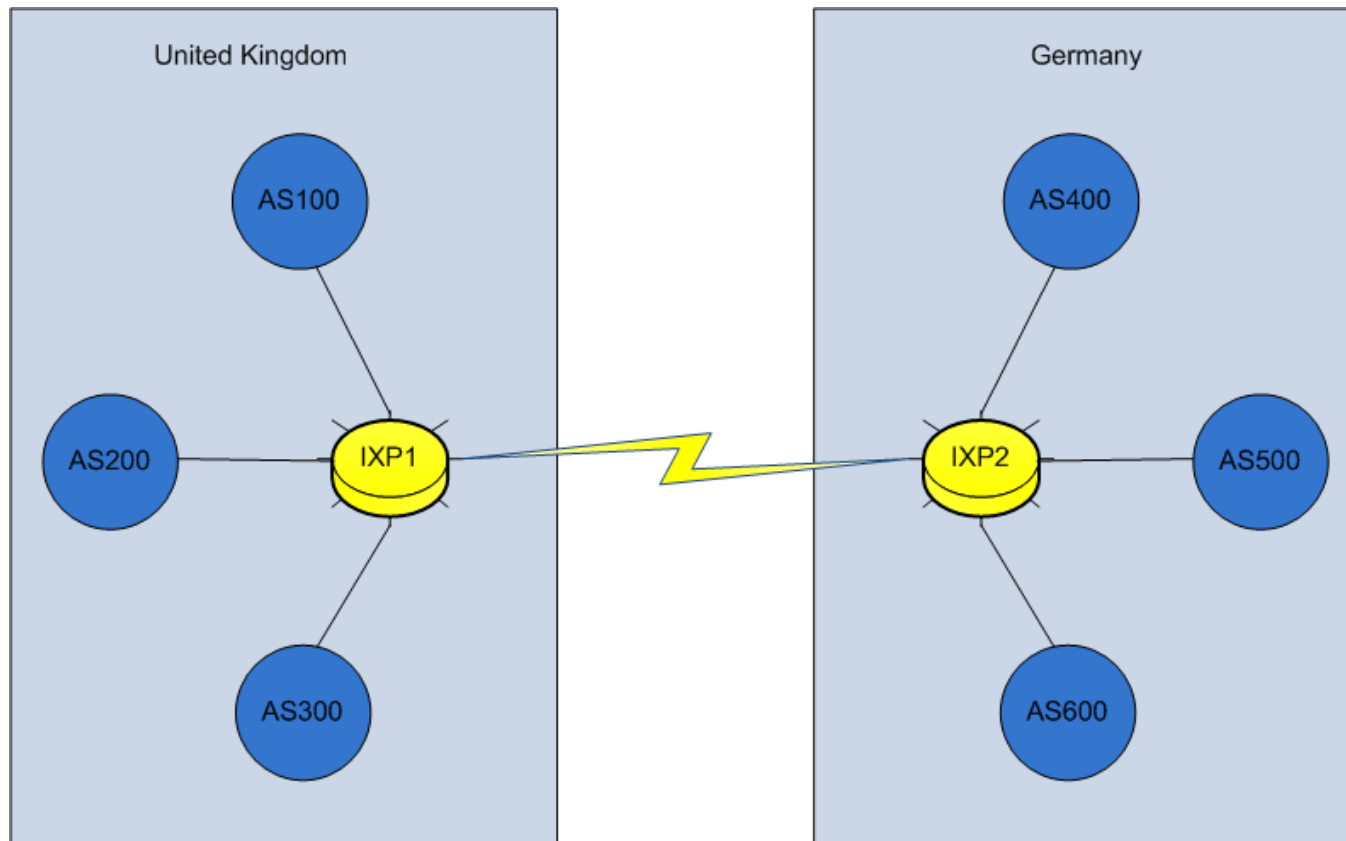
- AS sú číslované – ASN
 - Čísla AS rozdeľuje IANA na regionálne internetové registre,
 - Tí následne prideľujú AS jednotlivým žiadateľom
 - Regional Internet Registry (RIR):
 - ARIN (Severná Amerika)
 - RIPE NCC (Európa, Stredný Východ, Stredná Ázia)
 - APNIC (Ázia, Pacifická oblasť)
 - LACNIC (Latinská Amerika, Karibik)
 - AfriNIC (Afrika)
 - V súčasnosti sa používajú 2B čísla (0 – 65535)
 - RFC 4893 špecifikuje použitie 4B čísel (v dekadickom zápise 2B.2B)
 - Časť priestoru od 64512 po 65535 je vyhradená pre privátne ASN
- IANA nástojí na tom, aby organizácie, ktoré chcú mať vlastné číslo AS, avšak majú iba jediného ISP a zdieľajú jeho smerovacie politiky, zásadne používali privátne ASN
 - Privátne čísla AS sa objavujú len v sieti ISP a sú zamenené za ASN providera, keď sa prenášajú do iných AS

Štruktúra Internetu - AS infraštruktúra



- Jednotlivé AS (ISP AS) sa prepájajú cez Internet Packet Exchange (IPX) Gateways v tzv. Internet Exchange Points (IXP)
 - IXP je priamy prepoj, cez ktorý si ISP vymieňajú navzájom svoje dáta
 - A redukujú množstvo, ktoré musia posilať cez svojich *tranzitných* providerov
 - Peering:
 - dobrovoľný prepoj AS za účelom vzájomnej výmeny dát („ak prepošleš moje ja prepošlem tvoje“)

ISP AS infraštruktúra



Physical Internet Architecture

Svetové IXP

<http://www.datacentermap.com>




Európske IXP



IXP na Slovensku

- Slovensko má dva IXP body
 - BA a KE
- Slovak IXP (SIX) – www.six.sk
 - Špičková prevádzka – okolo 58Gbps
 - Priemer: okolo 28Gbps
 - 53 peerov




PEERING

DOKUMENTY


LINKY

ENGLISH VERSION




Prehľad aktívnych peeringov
Prehľad zátáže liniek
Looking glass

Stalo sa...




Centrum výpočtovej techniky STU
Nám. slobody 17, 812 43 Bratislava
tel.: 02/524 51 301, 02/529 61 573,
fax: 02/524 94 351, e-mail: six-ba@six.sk



Prehľad aktívnych peeringov
Prehľad zátáže liniek
Looking glass

Stalo sa...



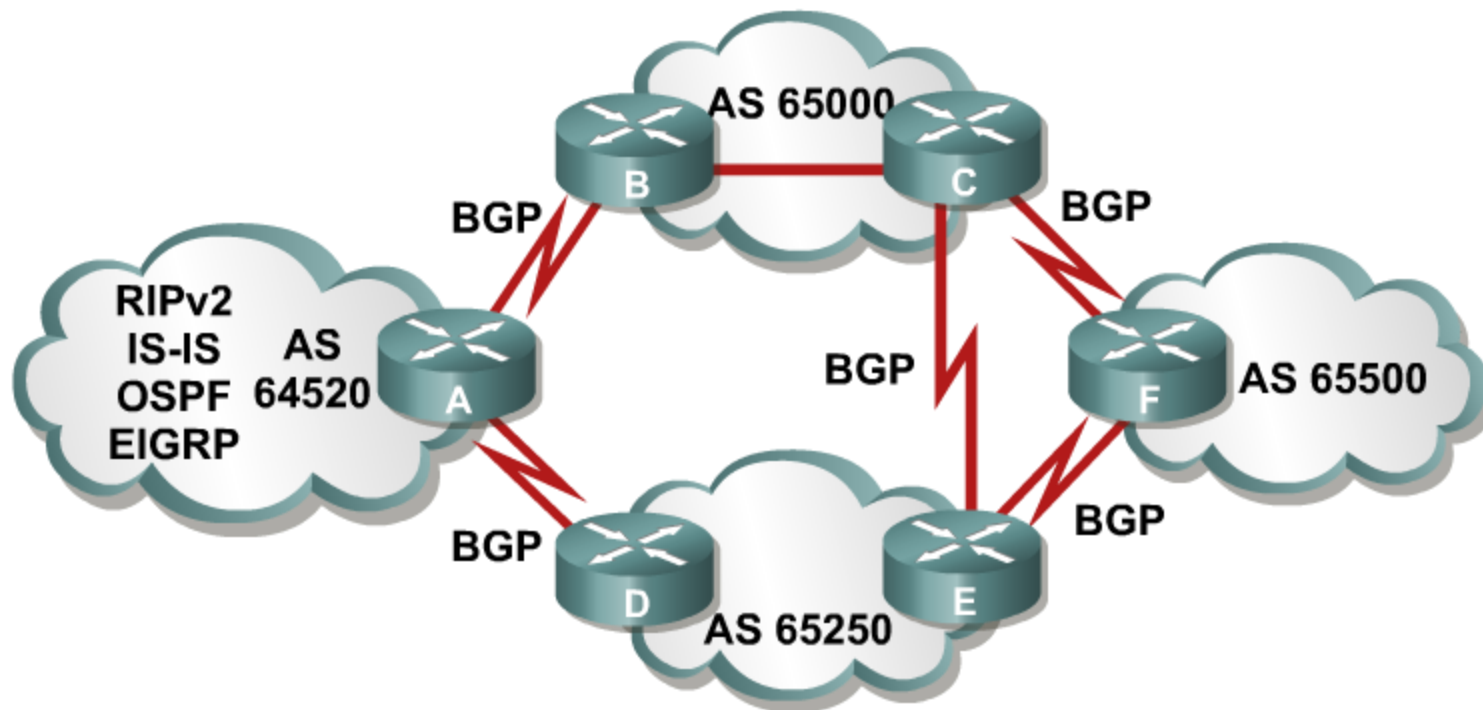
Ústav výpočtovej techniky TU Košice
B. Nemcovej 3, 042 01 Košice
tel.: 055/602 51 56, 055/602 50 00,
fax: 055/625 35 82, e-mail: six-ke@six.sk

Webstránky používajú kódovanie UTF-8

Druhy autonómnych systémov

- Autonómne systémy sa tradične rozdeľujú na 3 druhy
- Single-homed
 - AS, ktorý má jediný hraničný router do ostatného sveta
 - Single-homed AS častokrát vôbec nepotrebujú EGP routing
- Multihomed
 - AS, ktorý má viacero hraničných routerov do ostatného sveta
 - Napriek tomu, že sa pripája viacerými výstupnými bodmi, nedovoľuje, aby cez neho tiekla cudzia prevádzka
- Transit
 - AS, ktorý má viacero hraničných routerov do ostatného sveta a slúži na prenos tranzitnej prevádzky (medzi inými AS)

Smerovanie medzi AS



- Vo vnútri AS sa používajú IGP
- Medzi AS sa informácie vymieňajú pomocou BGP
- Smerovanie medzi AS
 - Cez aký ďalší AS sa dostanem k cieľu?
 - Ako sa po najkratšej ceste dostanem k tomuto susednému AS?

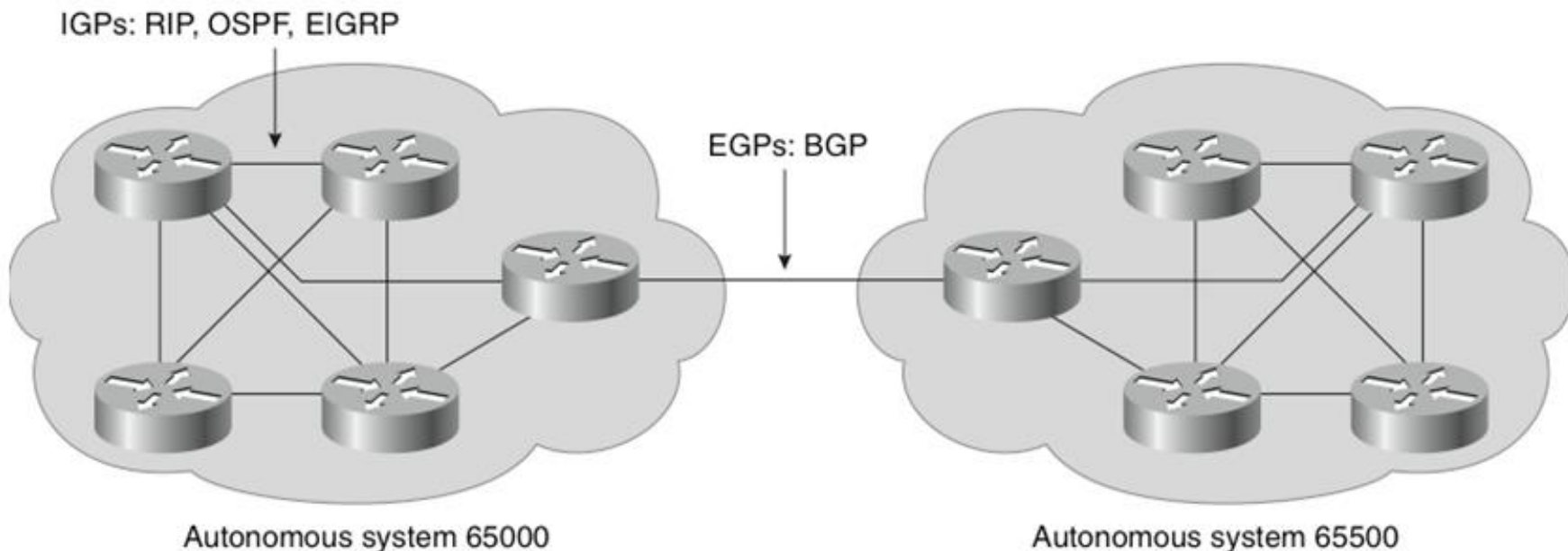
IGP versus EGP

■ Interior gateway protocol (IGP)

- Smerovací protokol pracující vo vnútri Autonomous System (AS).
 - Napr. RIP, OSPF, a EIGRP

■ Exterior gateway protocol (EGP)

- Smerovací protokol pracující medzi rôznymi AS
- BGP je tzv. interdomain routing protocol (IDRP) a je aj EGP



Smerovanie medzi vs. vnútri AS

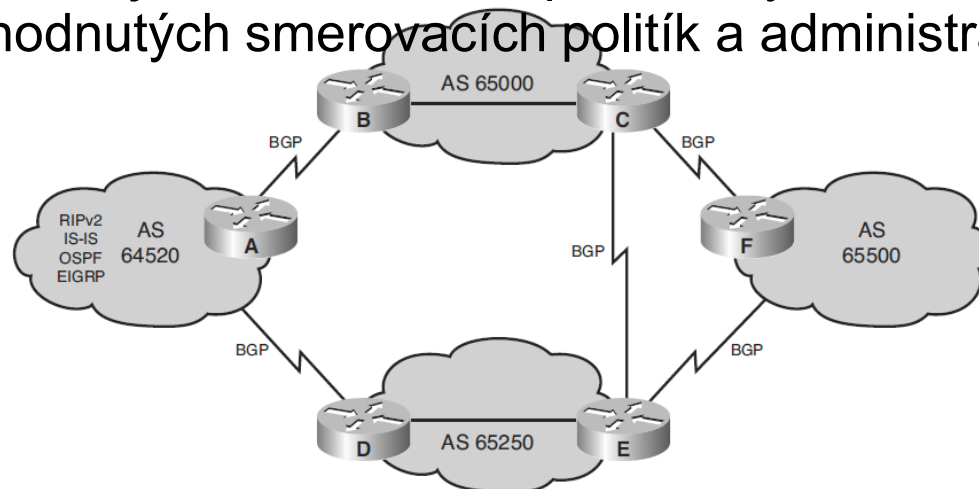
- Smerovanie medzi AS sa zásadne líši od smerovania vo vnútri AS
- IGP protokoly:
 - Susedné smerovače sa navzájom objavujú automaticky
 - Snahou IGP je vymeniť si čo najkompletnjšiu informáciu o vnútornej topológii AS a jeho členských sieťach
 - Svet za hranicami AS je „zahmlený“
 - Nahradený sumárnymi smermi alebo využitím default route, vždy bez topologickej predstavy
 - Metrika odráža výhodnosť trasy na základe počtu hopov, prenosovej rýchlosti, oneskorenia, záťaže, teda jej prenosové vlastnosti

Smerovanie medzi vs. vnútri AS

- Smerovanie medzi AS sa zásadne líši od smerovania vo vnútri AS
- EGP protokoly (BGP):
 - Susedné smerovače musia pre vzájomnú komunikáciu byť explicitne nakonfigurované
 - EGP protokoly sa nezaujímajú o vnútornú topológiu AS
 - Riešenie vnútornej dosiahnuteľnosti prenechávajú IGP
 - EGP protokoly sa zaujímajú o hraničné smerovače na okrajoch AS a o vzájomné prepojenie AS medzi sebou
 - Metrika sa skladá z parametrov, ktoré vyjadrujú pôvod siete a cestu cez tranzitné AS (*Path*), jej lokálnu preferenciu
 - neodráža nutne fyzický charakter cesty, ale jej administratívne vlastnosti;

Smerovanie medzi AS

- Smerovanie medzi AS musí byť zaručene bezslučkové
- Objem vymieňaných informácií je obrovský – desiatky až stovky megabajtov informácií obsiahnutých v smerovacích tabuľkách
 - <http://bgp.potaroo.net/>
 - Internet Core (<http://www.cidr-report.org/>)
 - => 636106 v roku 2016 (529 189 v 2014, (482.880 v 2013 (407.000 v 2012))) CIDR položiek
 - => 55542 v roku 2016 (48 913 v 2014 (45.592 v 2013 (40659 v 2012))) AS
- Výber ciest sa nerealizuje na základe ich prenosových charakteristík, ale na základe dohodnutých smerovacích politík a administratívnych rozhodnutí





Možnosti nasadenia BGP v „Enterprise“



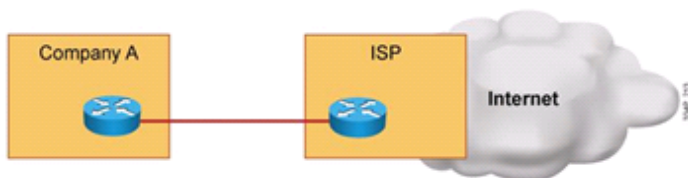
Štruktúra AS

- Môže byť hierarchická
 - Ale nie je podmienka
- Rozlišujú sa tri typy vzťahov
 - Customer - Provider
 - Zákazník (AS) platí poskytovateľovi (AS) za prístup k internetu.
 - Poskytovateľ poskytuje tranzitnú službu
 - Koncový zákazník platí ISP,
 - a ISP na nižšej úrovni ("tiers") platí ISP na vyššej úrovni
 - Peers
 - AS ktoré si umožňujú navzájom tranzit
 - Napr. ISP na rovnakej úrovni
 - Customer-Backup Provider
 - Záložný ISP

Možnosti pripojenia na ISP (medzi AS)

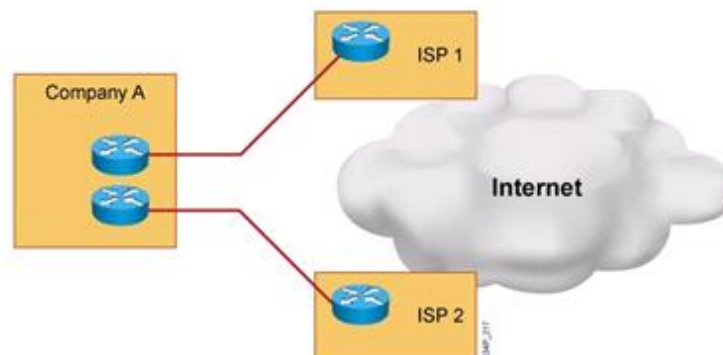
Connecting to One ISP

Single-homed

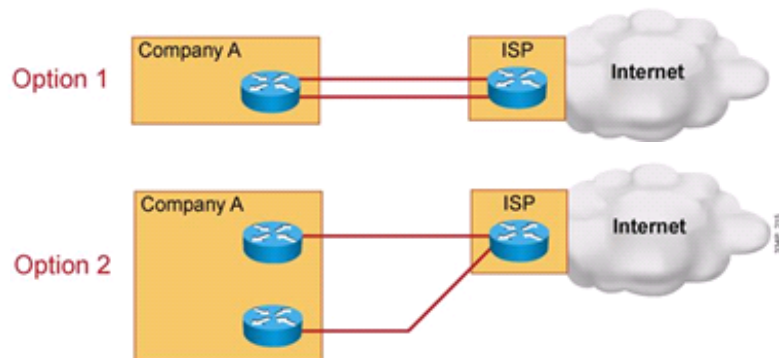


Connecting to Two or more ISPs

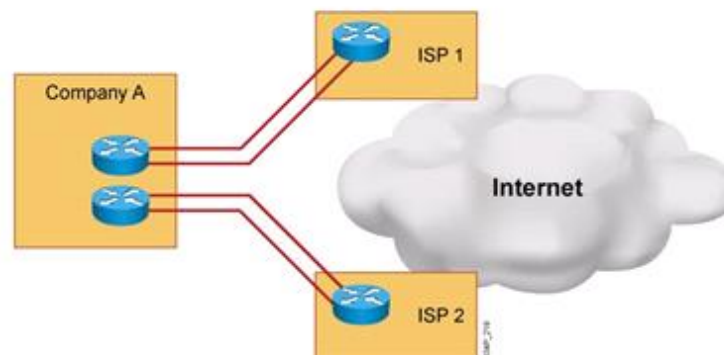
Multihomed



Dual-homed



Dual-multihomed



AS BGP politiky – multihomed prostredie

BGP multihoming

1. Multihoming je najvodnejšie prostredie pre nasadenie BGP

1. Pozor však na situáciu aby sa sieť zákazníka nestala tranzitnou AS pre dáta ISP

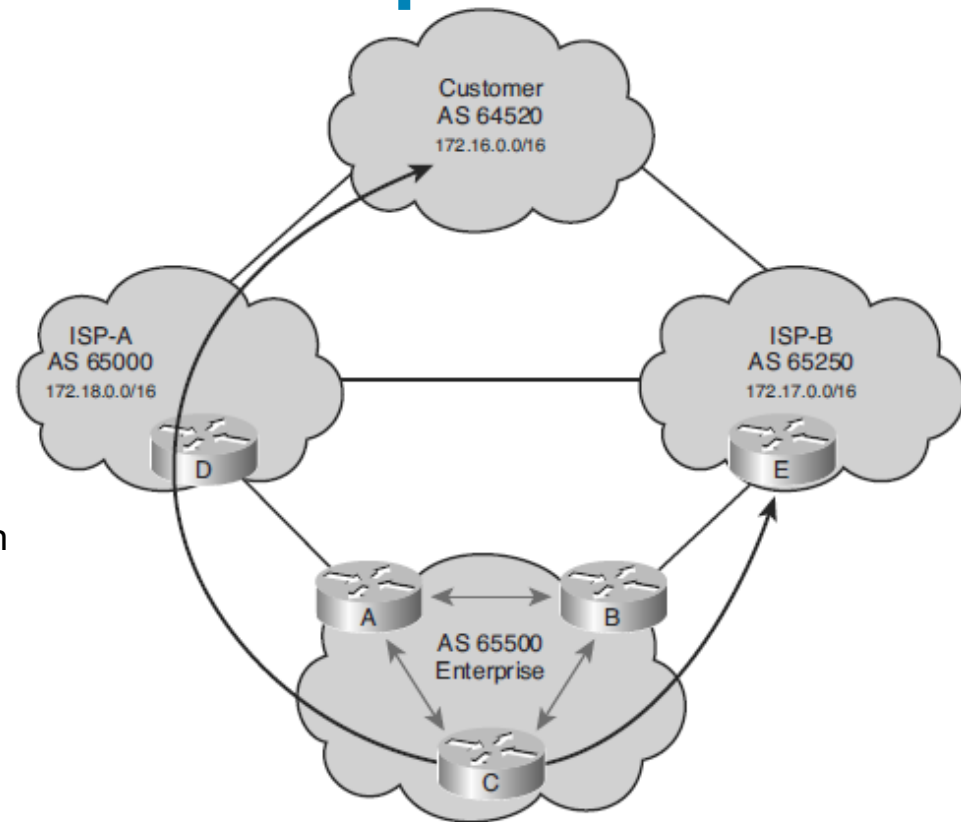
2. Riešenie prepoja ISP a zákazníka

1. Každý ISP informuje zákazníka (AS) len o default route

- Default route je posunutá zákazníkovi a ten ďalej interným smerovačom cez IGP
- Najmenšie nároky na zdroje
- Môže viesť k neoptimálnemu smerovaniu mimo AS zákazníka
- Nemáme veľa možnosti na manipuláciu so smerovacími cestami mimo AS

2. Každý ISP informuje o default route a o sieťach jeho zákazníkov

- Tieto cesty sú ďalej posunuté interným smerovačom
- Smerovanie je presnejšie ako v prvom prípade
- Def. routing môže viesť k neoptimálnemu smerovaniu mimo AS zákazníka
- Potencionálna záťaž pri redistribúcii veľa ciest z BGP do IGP

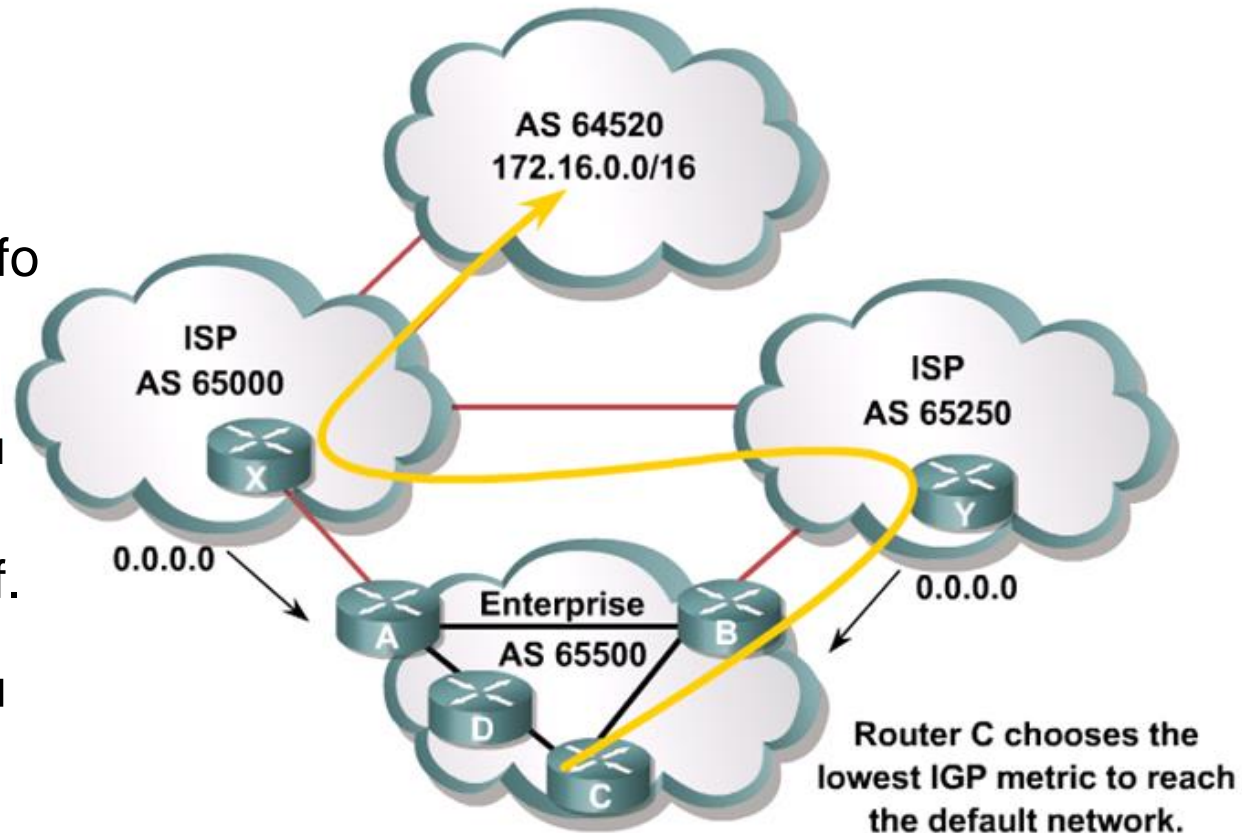


3. Každý ISP informuje o všetkých sieťach.

- Všetky smerovače v tranzit ceste musia mať BGP na výmenu updates
- Smerovanie je najpresnejšie
- Najvyššie požiadavky na zdroje zákazníka

Príklad: Default Routes from All Providers

- Zákazník distribuuje svoje siete na ISP dynamicky
- ISP na okrajové smerovače posiela informáciu o default route (BGP)
 - Tie ďalej distribuuujú cez IGP
- Ostatní si vyberú Default Route na základe metriky IGP protokolu
- Pri použití niektorých IGP protokolov (RIP), môže byť dosiahnuté neoptimálne smerovanie



Kedy použiť/nepoužiť BGP

■ Použiť BGP

- Najvhodnejšie ak je jasný prínos nasadenia BGP a existuje najmenej jedna z nasledujúcich situácií
 - „Naše“ AS má viaceré prepojenia na iné AS
 - „Naše“ AS umožňuje tranzit paketom cez seba na ceste do iných AS
 - Je potrebná manipulácia s výberom smerovacích ciest pre pakety opúšťajúce AS
 - Firma chce odlíšiť svoju prevádzku od prevádzky ISP

■ Nepoužiť BGP

- Ak existuje najmenej jedna z nasledujúcich situácií
 - Jedno pripojenie na Internet alebo AS
 - Slabé zariadenie na pozícii okrajových smerovačov
 - Malo pamäte, nízky výkon
 - „Slabé vedomosti o filtrácii ciest a činnosti BGP“ (dnes odstránime)
- V týchto prípadoch výhodné nasadenie statických ciest or default smerovania

BGP

Základné pojmy

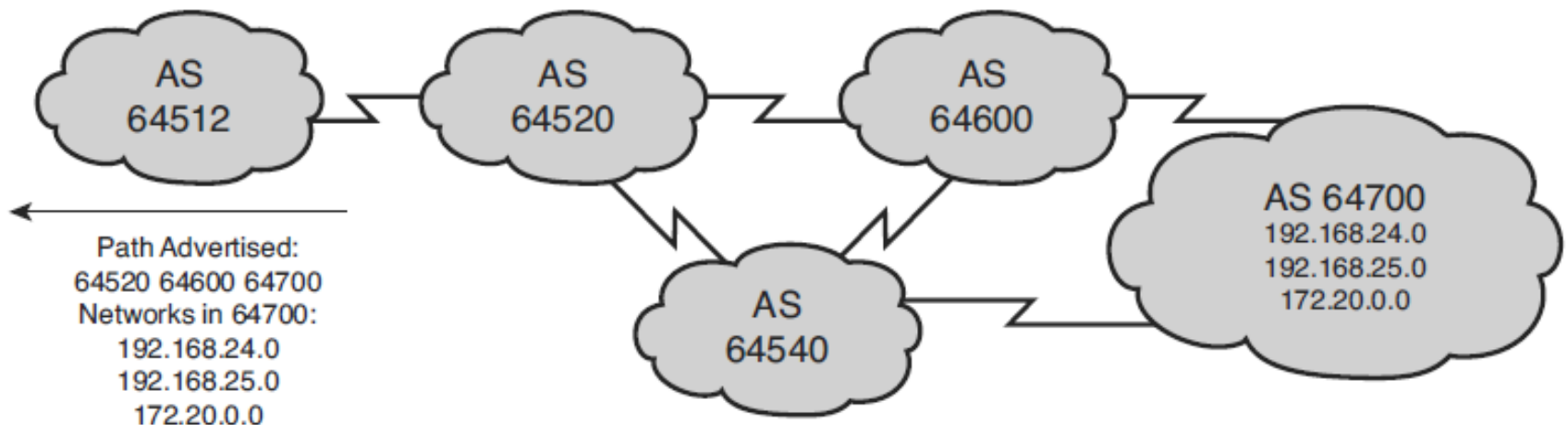


Border Gateway Protocol

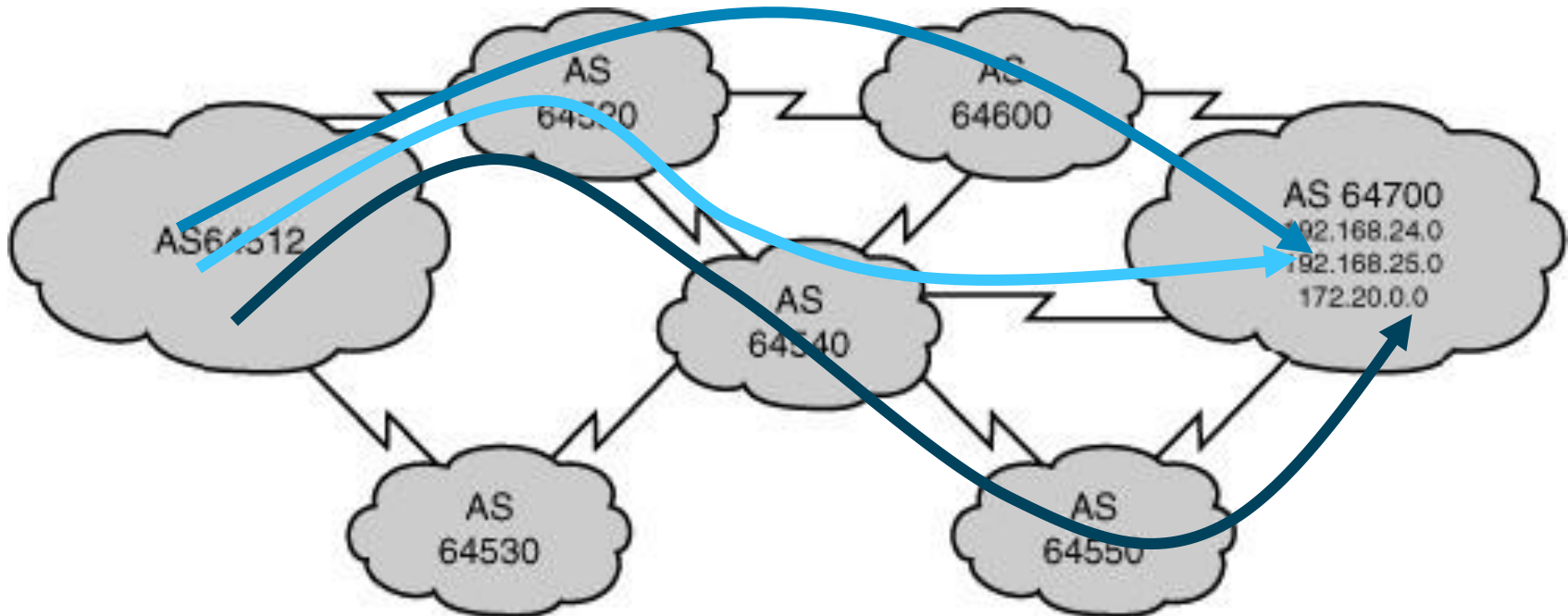
- BGP je v súčasnosti prakticky jediný používaný smerovací protokol pre inter-AS smerovanie
 - Je typu Path vector (zoznam AS a atribútov)
 - Garantuje bezslučkovú výmenu smerovacích informácií
 - Nedovoľuje akceptovať update, ktorý obsahuje jeho vlastné číslo AS
- Aktuálna verzia: BGPv4 špecifikovaná v RFC 4271
 - Početné ďalšie RFC rozširujú schopnosti BGP o smerovanie multicastov, podporu MPLS a ďalšie
- BGP beží nad TCP protokolom, cieľový port 179
 - Preto BGP nepotrebuje vlastné error recovery mechanizmy
- BGP patrí do rodiny EGP protokolov, no pojem EGP je zároveň aj meno staršieho externého smerovacieho protokolu, s ktorým BGP nesúvisí
 - Exterior Gateway Protocol je popísaný v RFC 904

BGP Path Vector

- IGP fungujú na princípe oznámenia zoznamu sietí a zoznamu parametrov (atribútov) o ceste do nich (metrika)
- BGP používa informácie o dostupnosti sietí, volaných **path vector**
 - Path vector informácie zahŕňajú napr.
 - Zoznam BGP AS čísel (hop by hop) potrebných traverzovať na dosiahnutie cieľovej siete
 - Na ktoré môžeme aplikovať Policy routing
 - A iné atribúty zahŕňajúce IP adresu ako sa dostať do ďalšieho AS (*next-hop attribute*) a ako sa dostali koncové siete do BGP (origin code attribute).

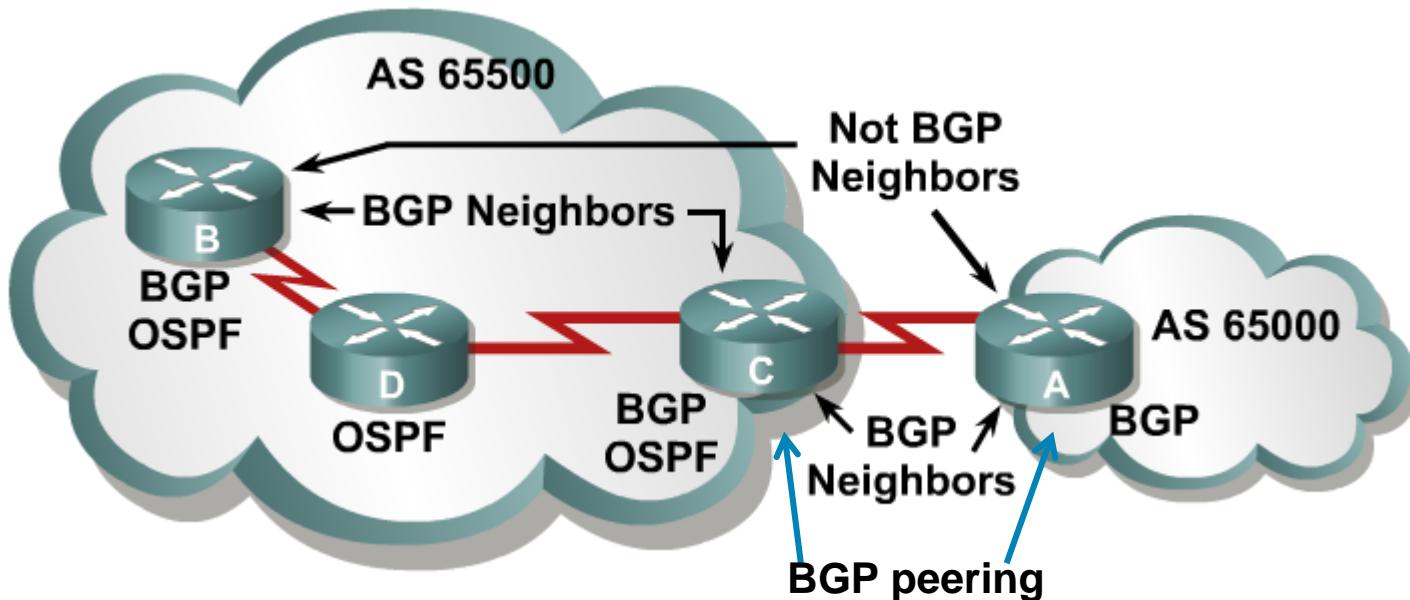


BGP – Hop by Hop správanie (best)



- Aké sú možnosti pre AS 64512 dostať sa cez 64520 do 64700?
 - 64512 => 64520 => 64600 => 64700
 - 64512 => 64520 => 64540 => 64700
 - 64512 => 64520 => 64540 => 64550 => 64700
- AS64512 sa však od 64520 dozvie len info o **najlepšej route**, napr. prvej, nie o všetkých
 - Všetky ďalšie pakety medzi AS 64512 a 64700 cez 64520 pôjdu touto cestou
 - Lebo BGP ma hop by hop AS správanie
- Alebo cez úplne iný AS, napr. 64530, podľa politik v 64512

BGP speaker, sused (peer, neighbor)



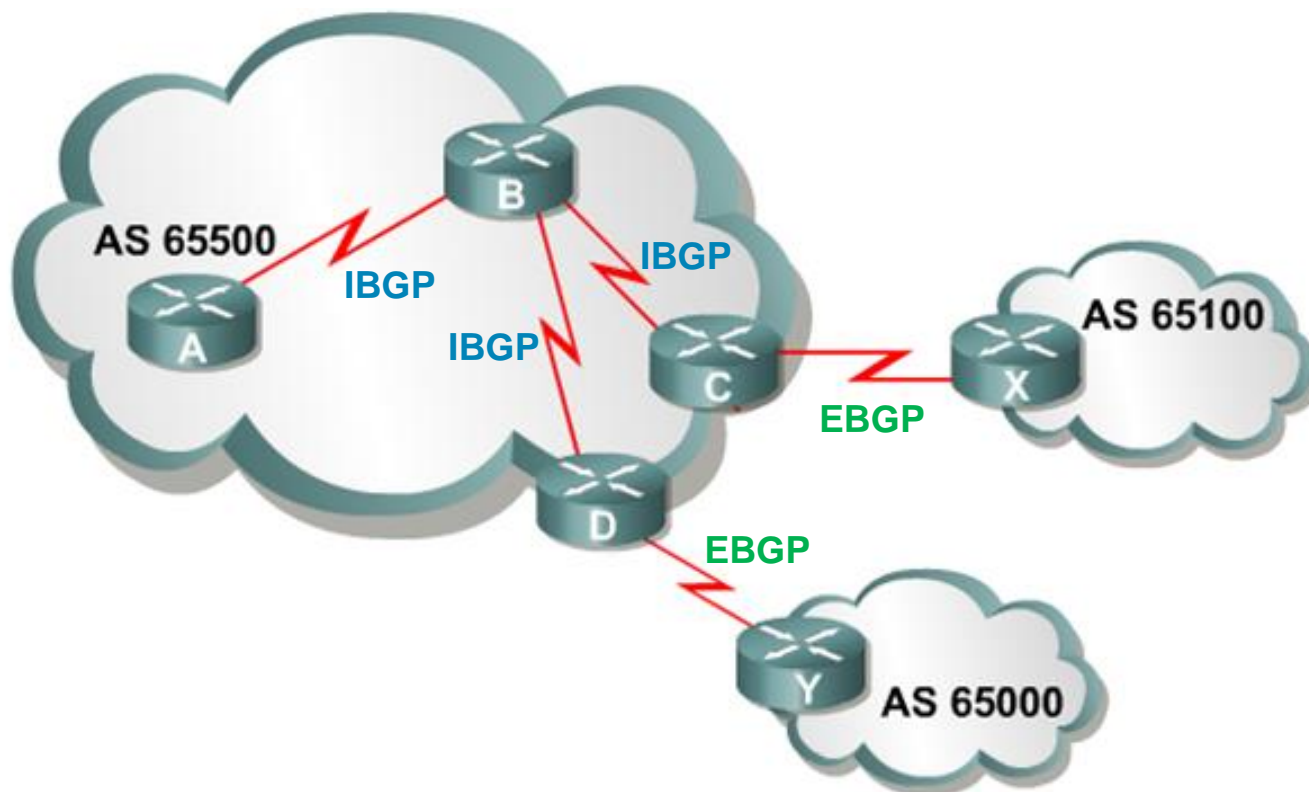
- **BGP speaker** je každý router, ktorý hovorí BGP protokolom
 - Je na ňom spustený BGP
- **BGP peers or neighbors** (susedia) je pojem, ktorý označuje dvojicu vzájomne komunikujúcich BGP speakerov
 - Všetky BGP smerovače nie sú susedia každý s každým
 - Majú založené TCP spojenie za účelom výmeny smerovacích informácií

Komunikácia v BGP

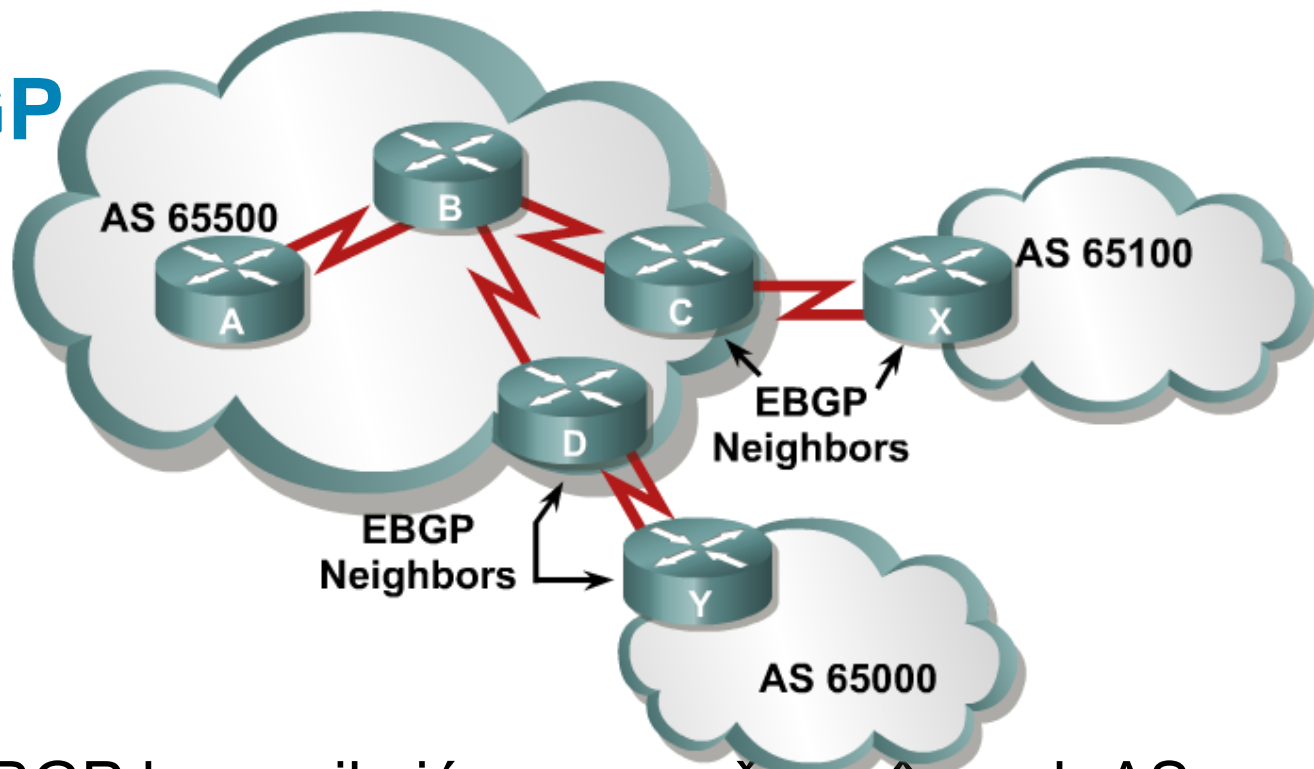
- Keď BGP susedia vytvoria spojenie, vzájomne sa synchronizujú
 - Navzájom si oznámia všetky **najlepšie** smery zo svojich BGP tabuliek
- Po úvodnej synchronizácii sa posielajú iba **inkrementálne aktualizácie** – zmeny (pridanie alebo odobranie smeru)
 - Inkrementálne aktualizácie sú efektívnejšie než prenosy úplných smerovacích tabuliek
 - Pri BGP sa jedná o obzvlášť zásadnú záležitosť, keďže veľkosť smerovacích tabuliek na chrbticových smerovačoch dosahuje rádovo desiatky až stovky MB
- Udržiavanie susedstva
 - Keepalive správy
 - Niečo ako Hello v OSPF a EIGRP

Nasadenie BGP v AS

- Z pohľadu nasadenia sa rozlišujú dva typy BGP
 - Internal BGP (iBGP or IBGP)
 - External BGP (eBGP or EBGP)

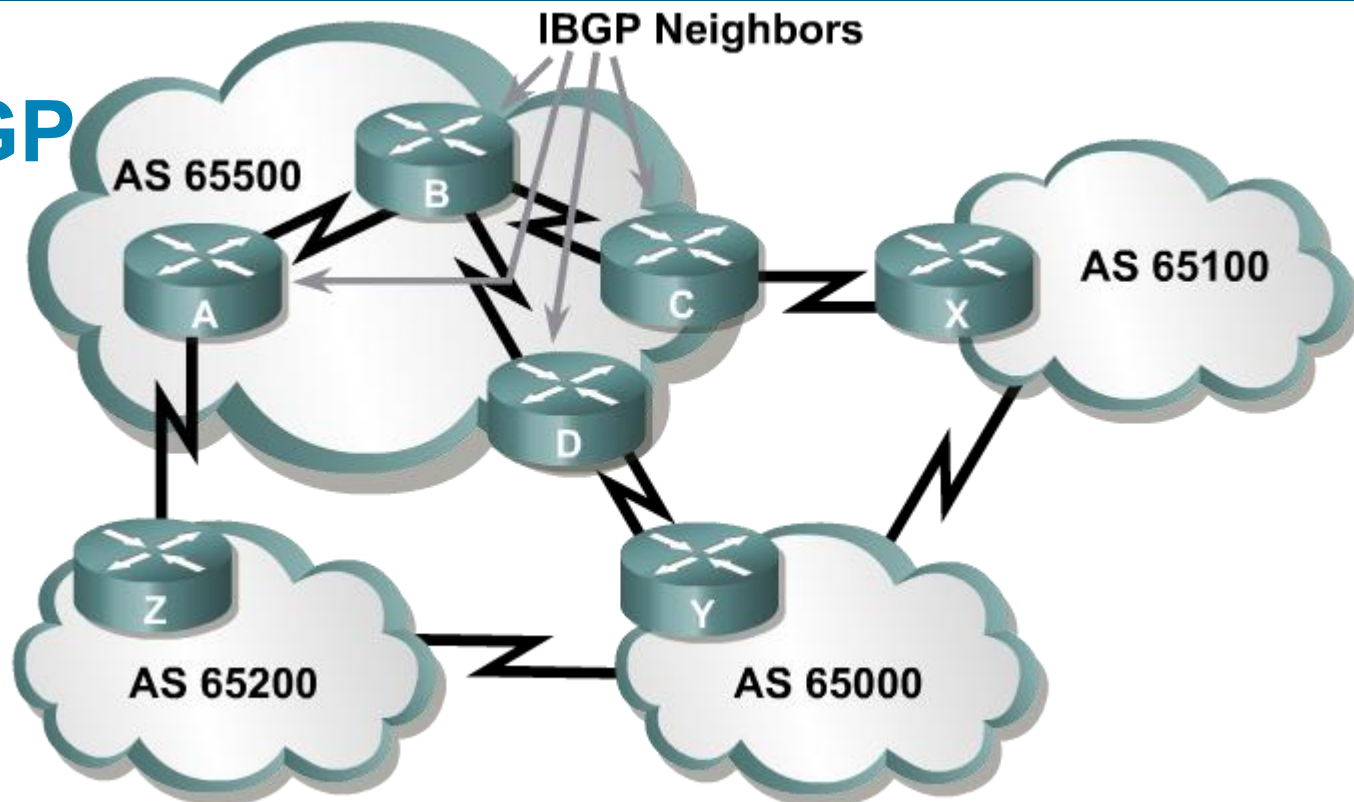


Externé BGP



- Ak pomocou BGP komunikujú smerovače v rôznych AS, hovoríme o **eBGP** (external BGP)
- eBGP susedia
 - Musia byť za normálnych okolností priamo prepojení linkou
 - Zvyčajne dosiahnuteľný bez pomoci IGP protokolu
 - Musia byť zadefinovaný (nakonfigurovaný príkazom neighbor)
 - TCP spojenie musí byť založené skôr ako si vymenia BGP updates
 - Musia mať **odlišné** čísla AS

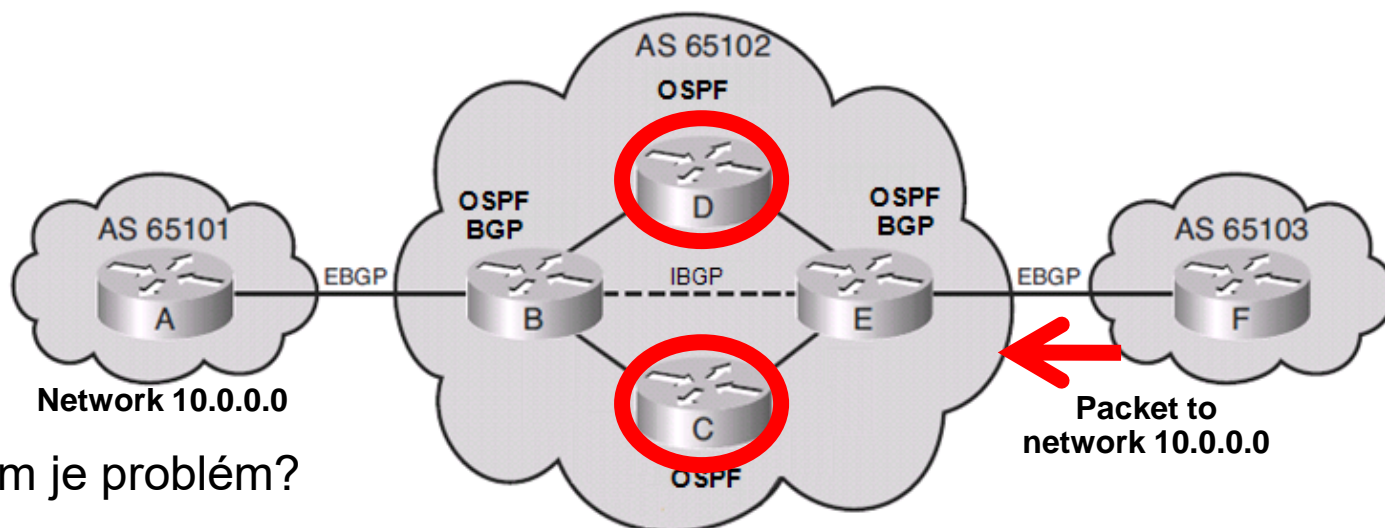
Interné BGP



- Ak pomocou BGP komunikujú smerovače v tom istom AS, hovoríme o **iBGP** (internal BGP)
- iBGP susedia
 - Nemusia byť priamo spojené
 - Konektivita na základe **IGP** or static routes
 - Za týmto účelom sa často používa loopback int
 - Musia byť zadefinovaný (nakonfigurovaný príkazom neighbor)
 - TCP spojenie musí byť založené skôr ako sa vymenia BGP updates
 - Musia mať **rovnaké** čísla AS

Špecifiká - Nasadenie BGP v Tranzitných AS

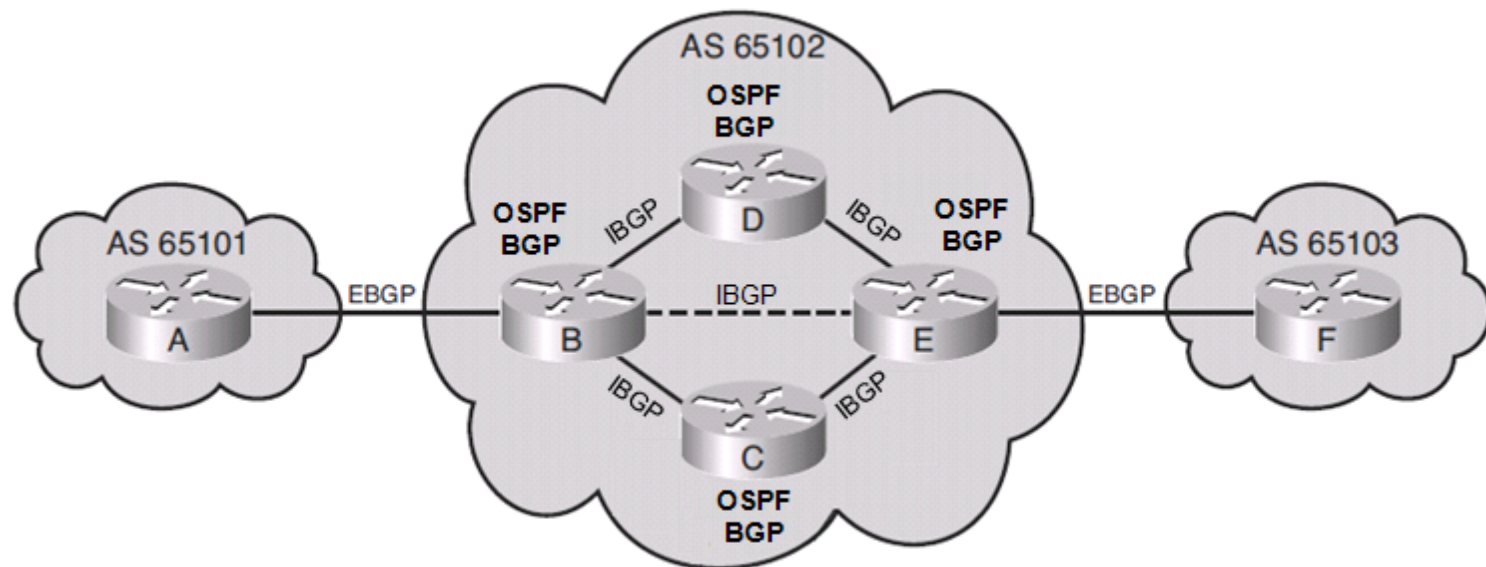
- Tranzitná AS je AS ktorá umožňuje smerovanie premávky z jednej externej AS do inej externej AS cez seba
- Príklad: AS 65102 je sieť ISP
 - Máme dva okrajové smerovače (router B a E) s BGP
 - Medzi sebou majú založený IBGP vzťah
 - Dosiahnuteľnosť iBGP peerov je cez OSPF



- V čom je problém?
 - Všetky smerovače tranzitnej AS musia mať informácie o všetkých externých cestách
 - „celom internete“
- Riešenie: redistribúcia BGP do OSPF
 - problém č.2: ?
 - OSPF nie je stavané na redistribúciu tak veľkých smerovacích tabuliek

Nasadenie BGP v Tranzitných AS - riešenie

- Riešením je nasadenie BGP na všetkých smerovačoch.
 - iBGP na interných smerovačoch s konfiguráciou *každý s každým*
 - iBGP majú kompletnú smerovaciu tabuľku o všetkých externých cestách
 - Vyhneme sa komplikovanej redistribúcii
- Poznámka:
 - BGP musí byť spustené na všetkých smerovačoch „po tranzitnej ceste“



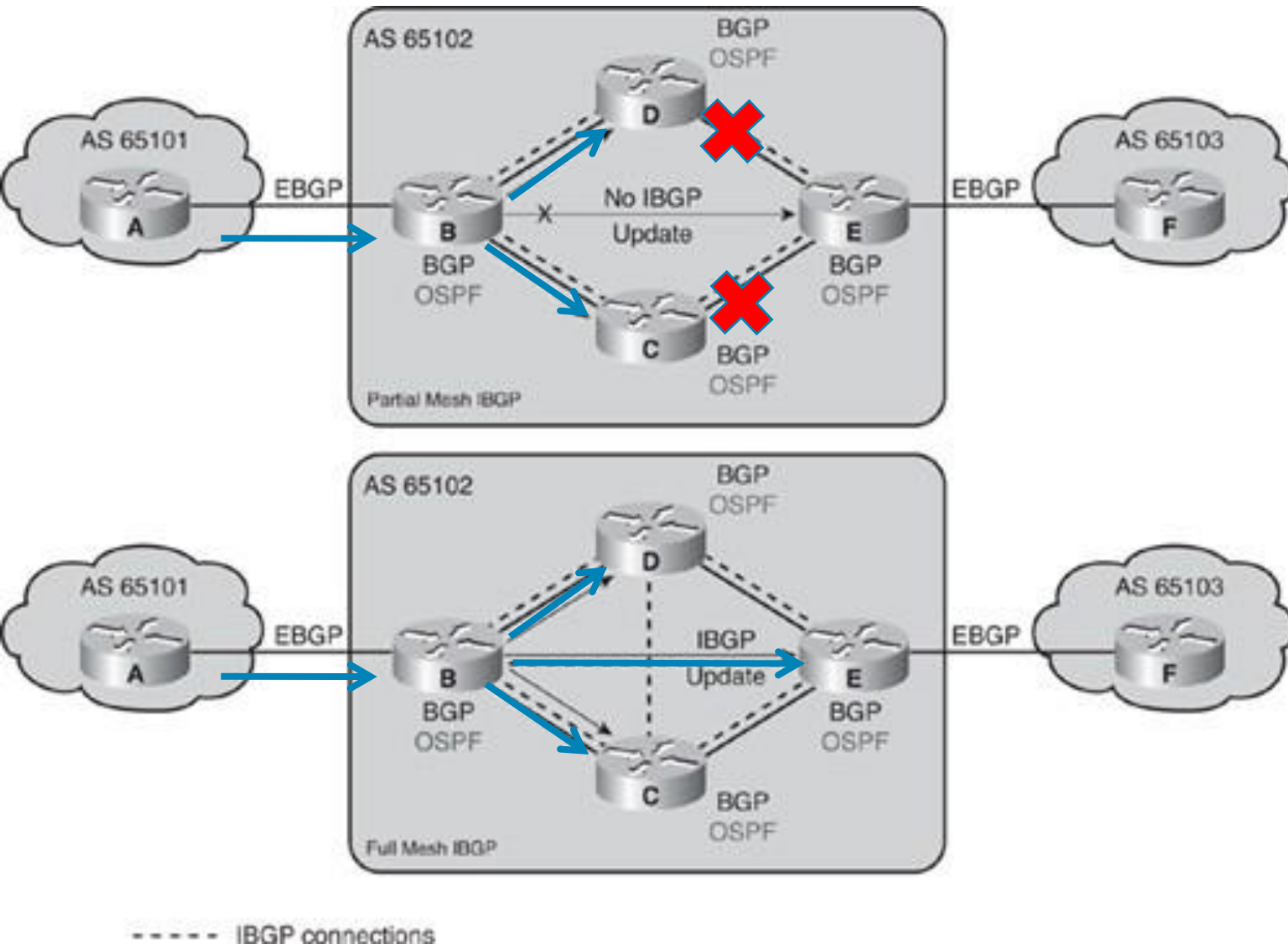
Rozdiel medzi iBGP a eBGP?

- Správanie BGP sa líši v závislosti od toho, či komunikácia prebieha na úrovni iBGP alebo eBGP
- Zásadný rozdiel:
 - **eBGP susedia** si navzájom odovzdávajú BGP smery obvyklým spôsobom
 - čo som sa cez eBGP naučil, to cez ľubovoľné BGP odovzdám,
 - a obrátene
 - **iBGP susedia** si odovzdávajú informáciu len priamo, nikdy nie sprostredkované
 - Ak sa router o nejakej ceste dozvie cez iBGP, neodovzdá túto informáciu nijakému ďalšiemu susedovi cez iBGP
 - čo som sa cez iBGP naučil, to si v iBGP nechám len pre seba, smiem to však povedať eBGP susedom (synchronization off)
 - Je to prísna, ale účinná ochrana pred vznikom smerovacích slučiek

Prečo sa rozlišuje iBGP a eBGP?

- Nasadenie iBGP prináša komplikáciu v tom, že jeho nasadenie na všetkých smerovačoch v AS nie je povinné/nevyhnutné
 - Non Tranzit AS
 - Je potrebné zabezpečiť aby každý iBGP mal rovnakú smerovaciu informáciu bez možnosti vzniku smerovacích slučiek vo vnútri AS
 - Preto existuje dané spomínané pravidlo, kde
 - „iBGP susedia si odovzdávajú informáciu len priamo, nikdy nie sprostredkovane“
- Toto správanie si vynucuje závažný architekturný rys pri implementácii BGP vo vnútri AS
 - Všetky iBGP smerovače v danom AS musia byť navzájom BGP **peermi** (konfiguračne, nie fyzicky)
 - zapojenie iBGP smerovačov v topológii každý s každým, tzv. „**Full Mesh**“
 - Nedodržanie tohto predpokladu vedie k nekonzistencii smerovacích tabuliek a potencionálnemu vzniku smerovacích slučiek

Dopad správania iBGP v AS



- Partial mesh, B a E nie sú iBGP susedia

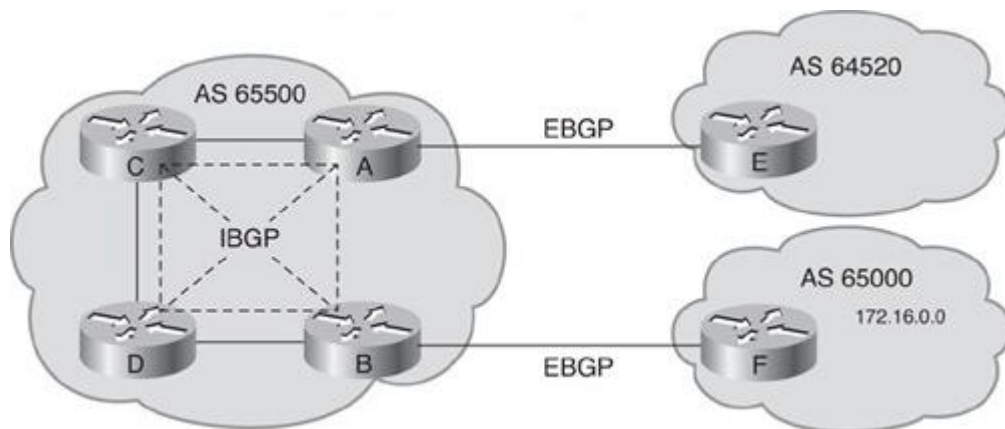
- B prijme info o zmene od A cez eBGP a pošle iBGP update na D a C
- D a C predpokladajú spojenia full mesh, preto E o zmene neinformujú,
- E nemá info o dostupnosti siete cez B

- Full mesh, B je iBGP sused s C, D, E

- B prijme info o zmene od A cez eBGP a pošle iBGP update na C, D a E

BGP Synchronizácia

- BGP synchronizácia definuje, že:
 - *“A BGP router should not use, or advertise to an external neighbor, a route learned by IBGP, unless that route is local or is learned from the IGP.”*
 - Ak je synchronizácia **zapnutá**,
 - Smerovač nesmie propagovať cestu naučenú cez iBGP na eBGP router až kým sa o nej sám nedozvie cez IGP
 - Ak je synchronizácia **vypnutá**
 - BGP môže informovať externého BGP suseda aj o cestách naučených cez iBGP a ktoré nie sú v smerovacej tabuľke
 - *BGP synchronizácie je defaultne vypnutá v Cisco IOS Software Release 12.2(8)T a novších*



■ Príklad

- Na A a B nie je redistribúcia eBGP od IGP
- A, B, C, a D sú iBGP susedia
- Čo sa stane v sieti AS 65500 ak F proklamuje sieť 172.16.0.0 ak
 - Synchro Off
 - Synchro On

Tabuľky v BGP

- Tabuľka susedov – Neighbor table

- Obsahuje zoznam a stav BGP susedov

- BGP tabuľka

- (forwarding database, topology database)
 - Obsahuje zoznam všetkých sietí získaných od každého suseda
 - K jednému cieľu môže obsahovať niekoľko záznamov
 - Ku každej ceste si eviduje jej BGP atribúty
 - Z viacerých ciest do toho istého cieľa je vždy len jedna **BEST**

- Smerovacia tabuľka – IP routing table

- Zoznam najlepších ciest do cieľových sietí
 - Z BGP tabuľky sú vybraté len najlepšie cesty
 - Pozor: klasický postup cez AD, prefix, metrika

Typy správ a stavy komunikácie v BGP



BGP Správy

Open Message

Octets	16	2	1	1	2	2	4	1	7
	Marker	Length	Type	Version	AS	Hold Time	BGP ID	Optional Length	Optional

Update Message

Octets	16	2	1	2	Variable	2	Variable	Variable
	Marker	Length	Type	Unfeasible (withdrawn) Routes length	Withdrawn Routes	Attribute Length	Attributes	NLRI

Notification Message

Octets	16	2	1	1	1	Variable
	Marker	Length	Type	Error Code	Error Sub-code	Diagnostic Data

Keepalive Message

Octets	16	2	1
	Marker	Length	Type

Každá zo správ obsahuje prvé tri polia hlavičky

BGP správy

▪ Open

- Obsahuje číslo verzie, ASN, holdtime, BGP router ID
- Posiela sa pri otváraní BGP spojenia medzi susedmi ako prvá správa
 - Ak druhá strana s Open súhlasí, pošle Keepalive správu
- Ak je príjem potvrdený môžu byť vymenené ďalšie typy správ

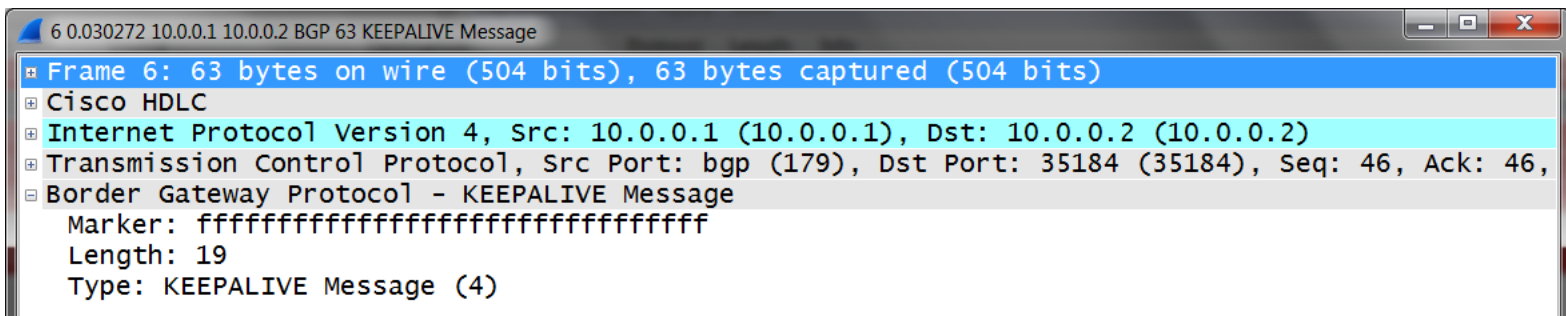
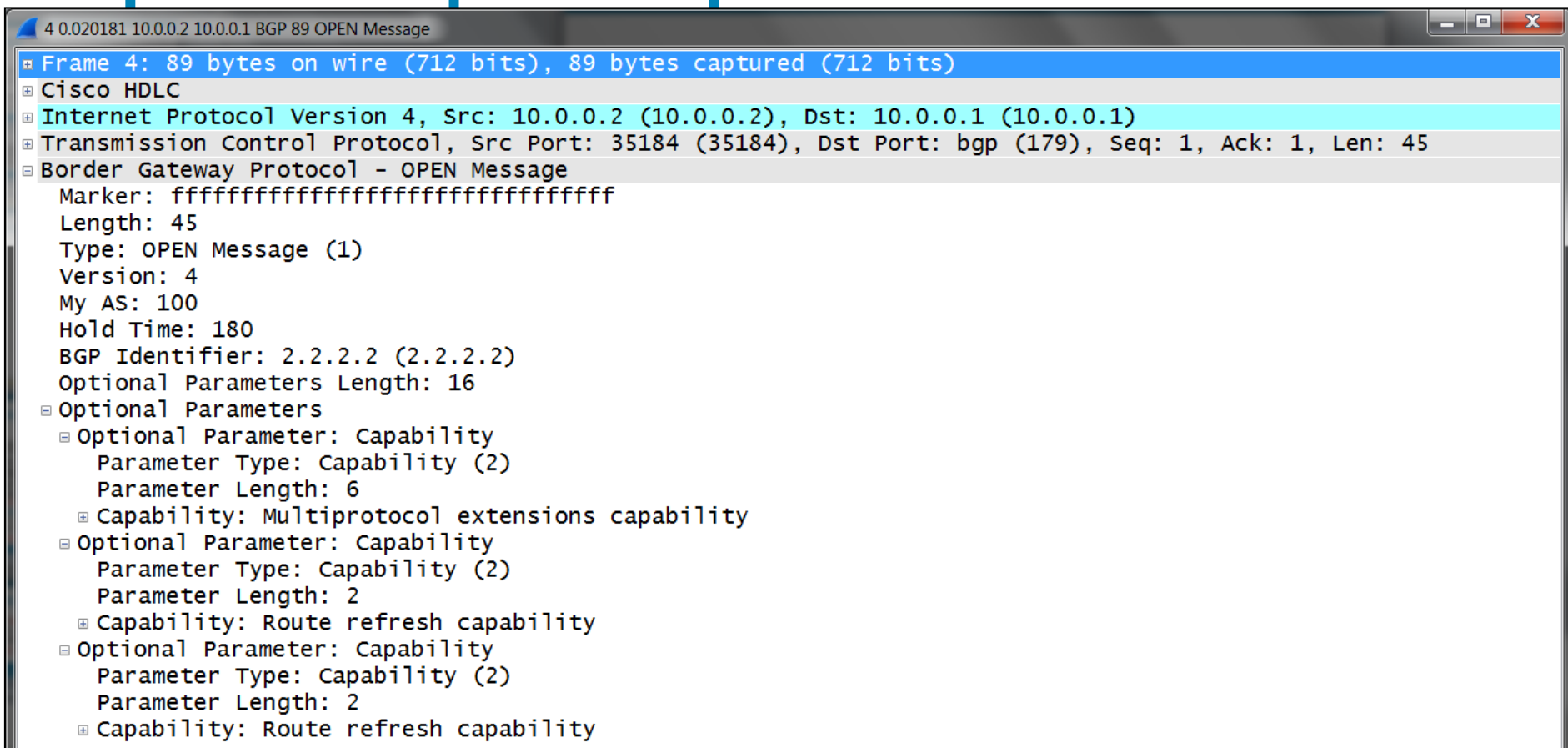
Octets	16	2	1	1	2	2	4	1	7
	Marker	Length	Type	Version	AS	Hold Time	BGP ID	Optional Length	Optional

▪ Keepalive

- Posiela sa periodicky na overenie, či sused žije, zodpovedajúco podľa dohodnutého holdtime
 - Default 60sekúnd, hold time je 3x keepalive

Octets	16	2	1
	Marker	Length	Type

Open a keepalive správa - wireshark



BGP správy

▪ Update

- Prenáša všetky informácie potrebné BGP na získanie bezslučkového obrazu siete
- **Jedna správa** prenáša informáciu len **o jednej ceste**
 - Cestou sa rozumie postupnosť AS, táto cesta môže viesť k rozličným cieľovým sieťam
 - Viac ciest = viac správ
- Správa obsahuje:
 - Informácia o nedostupných sieťach
 - Atribúty cesty
 - Network Layer Reachability Information (NLRI)
 - Zoznam sietí (IP prefixy a netmask) dostupných touto cestou

			Unreachable Routes Information		Path Attributes Information		NLRI Information	
Octets	16	2	1	2	Variable	2	Variable	Variable
	Marker	Length	Type	Unfeasible Routes Length	Withdrawn Routes	Attribute Length	Path Attributes	NLRI

Formát NLRI

- NLRI je list <**length**, **prefix**> n-tíc.
 - Jedna n-tica pre každý dosiahnuteľný cieľ.
 - **Prefix** reprezentuje dosiahnuteľný cieľ
 - Prefix **length** reprezentuje počet bitov subsieťovej masky

IP Address Subnet Mask	NLRI
10.1.1.0 255.255.255.0	24 , 10.1.1.0
192.24.160.0 255.255.224.0	19 , 192.24.160.0

Update – nová cesta

The image shows a Wireshark 1.10.6 window displaying a packet capture of a BGP UPDATE message. The packet list shows a single packet (No. 1) at time 0.000000, source 10.0.0.1, destination 10.0.0.2, protocol BGP, length 96, and info 96 UPDATE Message. The packet details pane shows the following structure:

- Frame 1: 96 bytes on wire (768 bits), 96 bytes captured (768 bits)
- Cisco HDLC
- Internet Protocol Version 4, Src: 10.0.0.1 (10.0.0.1), Dst: 10.0.0.2 (10.0.0.2)
- Transmission Control Protocol, Src Port: bgp (179), Dst Port: 35184 (35184), Seq: 1, Ack: 1, Len: 52
- Border Gateway Protocol - UPDATE Message
 - Marker: ffffffffffffffffffffffffffffffffff
 - Length: 52
 - Type: UPDATE Message (2)
 - Unfeasible routes length: 0 bytes
 - Total path attribute length: 25 bytes
 - Path attributes
 - ORIGIN: IGP (4 bytes)
 - Flags: 0x40 (Well-known, Transitive, Complete)
 - Type code: ORIGIN (1)
 - Length: 1 byte
 - Origin: IGP (0)
 - AS_PATH: 200 (7 bytes)
 - Flags: 0x40 (Well-known, Transitive, Complete)
 - Type code: AS_PATH (2)
 - Length: 4 bytes
 - AS path: 200
 - NEXT_HOP: 10.0.0.1 (7 bytes)
 - Flags: 0x40 (Well-known, Transitive, Complete)
 - Type code: NEXT_HOP (3)
 - Length: 4 bytes
 - Next hop: 10.0.0.1 (10.0.0.1)
 - MULTI_EXIT_DISC: 0 (7 bytes)
 - Flags: 0x80 (Optional, Non-transitive, Complete)
 - Type code: MULTI_EXIT_DISC (4)
 - Length: 4 bytes
 - Multiple exit discriminator: 0
 - Network layer reachability information: 4 bytes
 - 10.1.1.0/24
 - NLRI prefix length: 24
 - NLRI prefix: 10.1.1.0 (10.1.1.0)

The status bar at the bottom indicates: Frame (frame), 96 bytes | Pa... | Profile: Default

BGP správy

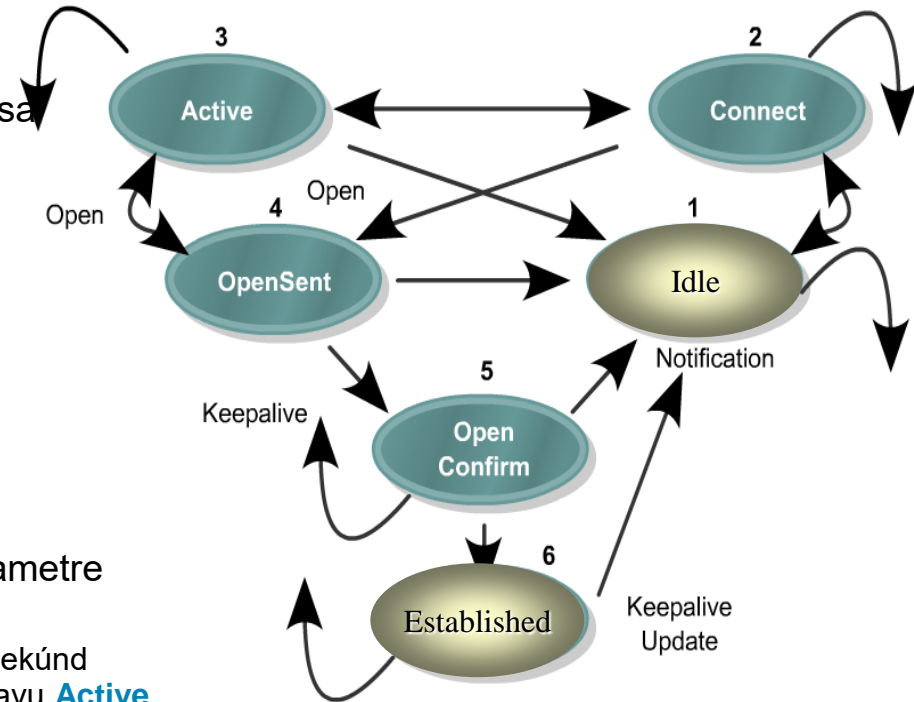
■ Notification

- Posiela sa v prípade chyby a obsahuje kód chyby a jej popis
- BGP spojenie sa po odoslaní Notification ukončí

Error Code	Error Subcode
1--Message Header Error	1--Connection Not Synchronized 2--Bad Message Length 3--Bad Message Type
2--OPEN Message Error	1--Unsupported Version Number 2--Bad Peer AS 3--Bad BGP Identifier 4--Unsupported Optional Parameter 5--Authentication Failure 6--Unacceptable Hold Time
3--UPDATE Message Error	1--Malformed Attribute List 2--Unrecognized Well-Known Attribute 3--Missing Well-Known Attribute 4--Attribute Flags Error 5--Attribute Length Error 6--Invalid Origin Attribute 7--AS Routing Loop 8--Invalid NEXT_HOP Attribute 9--Optional Attribute Error 10--Invalid Network Field 11--Malformed AS_path
4--Hold Timer Expired	NOT applicable
5--Finite State Machine Error (for errors detected by the FSM)	NOT applicable
6--Cease (for fatal errors besides the ones already listed)	NOT applicable

BGP stavy pri komunikácii so susedom

- BGP sa riadi stavovým strojom a stavmi, ktorými proces prechádza
- **Idle**
 - Štartovací stav. Sused je definovaný, ale zatiaľ sme sa nepokúsili kontaktovať ho (neighbor príkaz)
 - Mali by sme mať info o ceste na suseda
- **Connect**
 - So susedom sme začali nadväzovať TCP spojenie
 - TCP SYN odoslané, čakáme SYN/ACK
- **Active**
 - TCP založené, zatiaľ žiadne BGP správy odosl./prij.
- **Open sent**
 - Susedovi sme poslali správu OPEN obsahujúcu parametre spoločnej relácie, čakáme na Open z druhej strany
 - Ak na odoslanú OPEN správu v stave Open Sent do 5 sekúnd nepríde potvrdenie ani zamietnutie, presúvame sa do stavu **Active**
- **Open confirm**
 - Od suseda sme prijali správu Open, v ktorej sused vyjadruje súhlas pre peering s nami, čakáme na Keepalive or Notification
- **Established**
 - Sme úspešní susedia, môže začať výmena smerovacích informácií
 - Môžeme vymeniť keepalive, notification a update správy



Debug ip bgp all

```
*Mar 1 01:29:30.215: BGP: 10.255.255.3 passive open to 10.255.255.1
*Mar 1 01:29:30.215: BGP: 10.255.255.3 went from Idle to Connect
*Mar 1 01:29:30.223: BGP: 10.255.255.3 rcv message type 1, length (excl. header) 26
*Mar 1 01:29:30.223: BGP: 10.255.255.3 rcv OPEN, version 4, holdtime 180 seconds
*Mar 1 01:29:30.227: BGP: 10.255.255.3 went from Connect to OpenSent
*Mar 1 01:29:30.227: BGP: 10.255.255.3 sending OPEN, version 4, my as: 110, holdtime 180
seconds
*Mar 1 01:29:30.227: BGP: 10.255.255.3 rcv OPEN w/ OPTION parameter len: 16
*Mar 1 01:29:30.227: BGP: 10.255.255.3 rcvd OPEN w/ optional parameter type 2 (Capability)
len 6
*Mar 1 01:29:30.231: BGP: 10.255.255.3 OPEN has CAPABILITY code: 1, length 4
*Mar 1 01:29:30.231: BGP: 10.255.255.3 OPEN has MP_EXT CAP for afi/safi: 1/128
*Mar 1 01:29:30.231: BGP: 10.255.255.3 rcvd OPEN w/ optional parameter type 2 (Capability)
len 2
*Mar 1 01:29:30.231: BGP: 10.255.255.3 OPEN has CAPABILITY code: 128, length 0
*Mar 1 01:29:30.235: BGP: 10.255.255.3 OPEN has ROUTE-REFRESH capability(old) for all
address-families
*Mar 1 01:29:30.235: BGP: 10.255.255.3 rcvd OPEN w/ optional parameter type 2 (Capability)
len 2
*Mar 1 01:29:30.235: BGP: 10.255.255.3 OPEN has CAPABILITY code: 2, length 0
*Mar 1 01:29:30.235: BGP: 10.255.255.3 OPEN has ROUTE-REFRESH capability(new) for all
address-families
BGP: 10.255.255.3 rcvd OPEN w/ remote AS 110
*Mar 1 01:29:30.239: BGP: 10.255.255.3 went from OpenSent to OpenConfirm
*Mar 1 01:29:30.239: BGP: 10.255.255.3 send message type 1, length (incl. header) 45
*Mar 1 01:29:30.395: BGP: 10.255.255.3 went from OpenConfirm to Established
*Mar 1 01:29:30.395: %BGP-5-ADJCHANGE: neighbor 10.255.255.3 Up
R1#
```

Riešenie BGP stavov Active a Idle

- **Idle:** Ak sused zostáva v stave Idle, z nejakého dôvodu nie je možné vytvoriť s ním TCP spojenie
 - Existuje v našej smerovacej tabuľke cesta k tomuto susedovi? Táto cesta nesmie byť default route!
 - Nie je v IP adrese suseda preklep?
- **Active:** Router poslal OPEN správu susedovi a čaká (zatiaľ neúspešne) na odpoveď, alebo čaká na vytvorenie TCP spojenia z druhej strany
- Stav môže oscilovať medzi Active a Idle.
- Táto situácia naznačuje na problém vo vzájomnej komunikácii medzi susedmi. Niektoré možné príčiny:
 - Sused nemá cestu nazad k nám alebo my k nemu
 - Nesprávne adresy susedov v konfigurácii BGP
 - Sused nás nemá nakonfigurovaných ako svojho suseda
 - Nezhoda v číslach AS
 - Firewall blokuje komunikáciu medzi nami a susedom

Konfigurácia BGP



Spustenie BGP

Router(config) #

```
router bgp AUTONOMOUS-SYSTEM
```

- Príkaz definuje, v akom AS sa router nachádza, a otvorí konfiguračný kontext pre protokol BGP
- Na routeri môže bežať najviac jedna inštancia protokolu BGP
- Číslo AS v záhlaví príkazu sa porovná s číslami AS definovanými pri jednotlivých susedoch
 - Tak sa zistí, či je sused v tom istom alebo v inom AS, než sme my.
 - Podľa toho sa so susedom vytvorí iBGP alebo eBGP peering
- BGP má svoje *RouterID*
 - Vyberá sa rovnakým algoritmom ako pri OSPF, resp. EIGRP

Konfigurácia BGP suseda

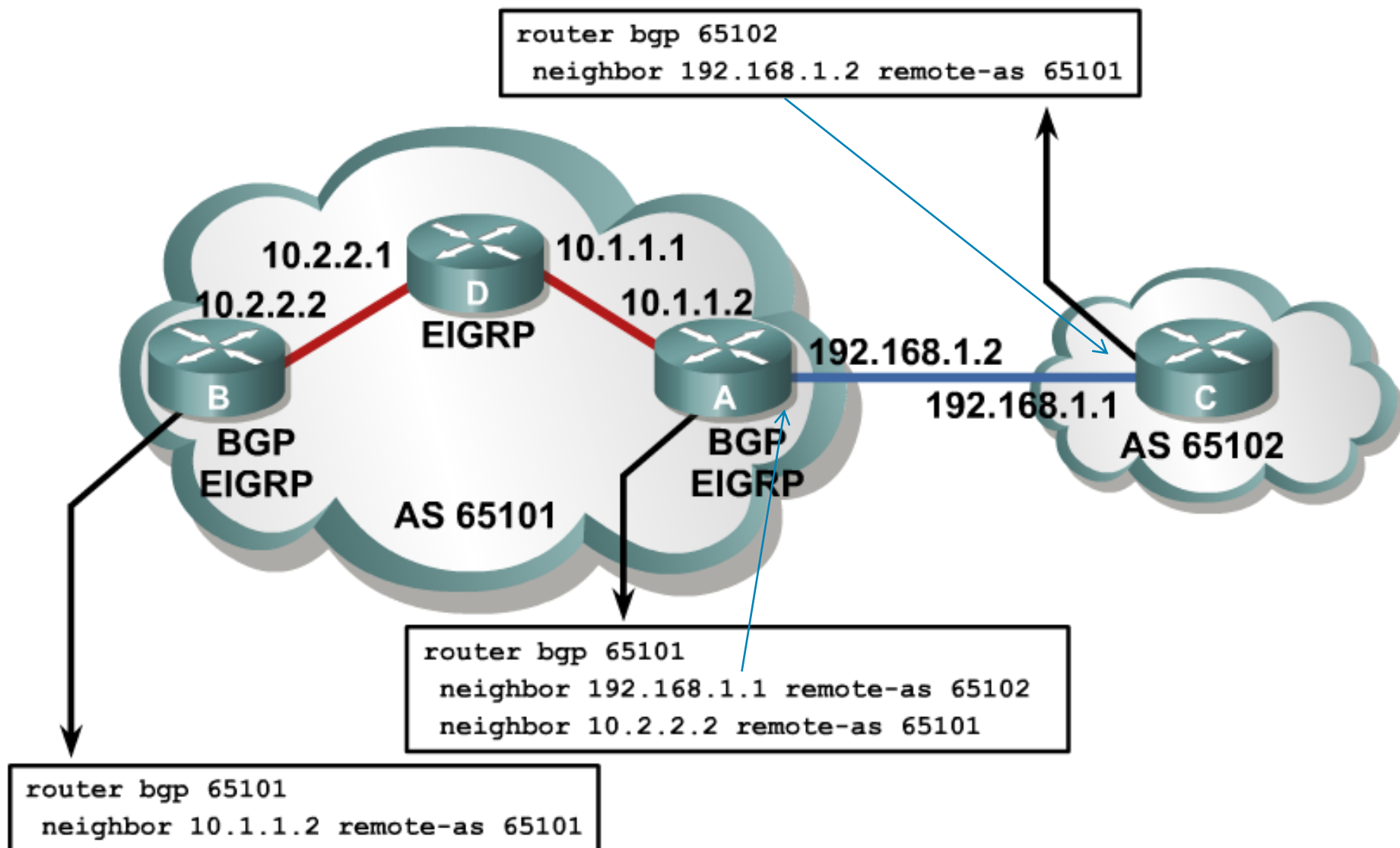
- príkaz `neighbor remote-as`

Router (config-router) #

```
neighbor {ip-address | peer-group-name}  
    remote-as autonomous-system
```

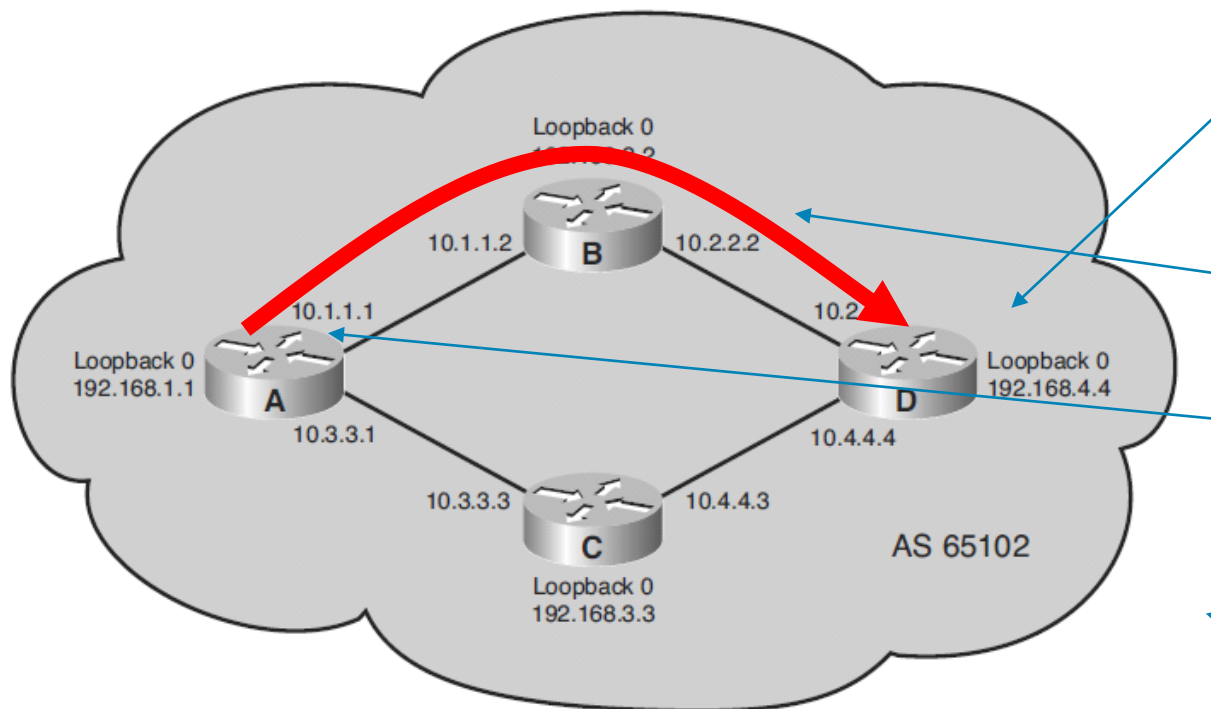
- Príkaz **neighbor** definuje suseda a aktivuje s ním peering
- **Ip-address** špecifikuje cieľovú adresu, na ktorú sa budú posielať BGP pakety pre tohto suseda
 - K danej IP adrese musí existovať v našej smerovacej tabuľke nejaká cesta
 - Pozor – default route sa na dosiahnutie suseda nikdy nepoužije!
- Argument **remote-as** hovorí, v akom AS sa nachádza príslušný sused
 - Podľa toho sa zakladá iBGP alebo eBGP relácia
 - iBGP, adresa nemusí byť z priamo pripojených
 - eBGP, adresa musí byť z priamo pripojených
 - Týmto príkazom sa definujú všetci susedia – externí aj interní
- Medzi dvojicou BGP susedov musia IP adresy uvedené v príkaze **neighbor** vzájomne korešpondovať
 - Zdrojová IP adresa BGP paketov od jedného suseda musí zodpovedať IP adrese v príkaze **neighbor** u druhého suseda, a obrátene

Príklad: BGP príkaz neighbor



Problém so zdrojovou adresou pri BGP

- BGP neprijíma nevyžiadané updates
 - Musí mať jasne definovaných susedov (príkaz neighbor)
 - BGP paket **musí** prísť z danej IP adresy suseda
 - Nie je problém pri eBGP (musí byť priamo pripojený), problém je pri iBGP
- Problém pri viacerých rozhraniach iBGP peerov



Router D pre router A:
neighbor 10.3.3.1
remote-as 65102

Ale IGP dostupnosť D z pohľadu A je cez B:

Cisco smerovače defaultne ako source IP dávajú IP adresu outgoing rozhrania, tu 10.1.1.1

BGP relácia medzi D a A sa nezaloží z dôvodu odlišnosti adries

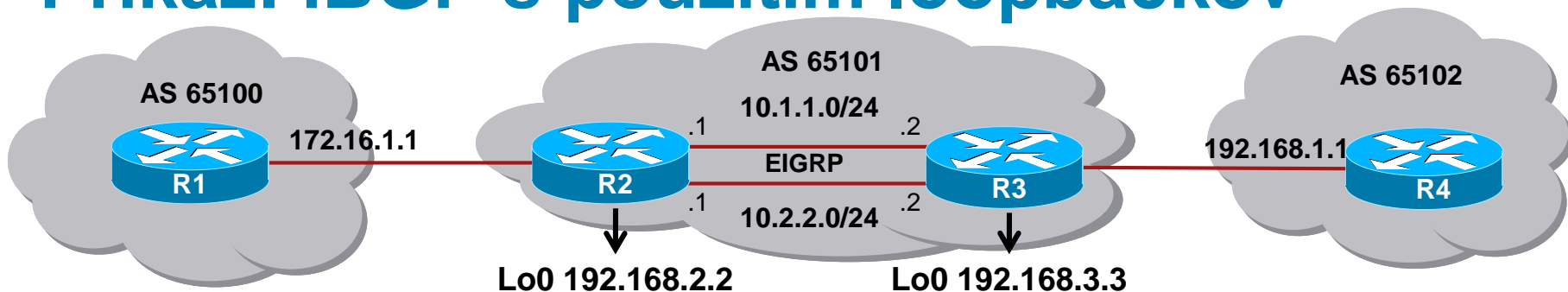
Riešenie - príkaz neighbor update-source

Router(config-router) #

```
neighbor {ip-address | peer-group-name} update-source  
interface-type interface-number
```

- Zdrojová IP adresa BGP paketov odosielaných danému susedovi bude nastavená na IP adresu uvedeného rozhrania
- Najvhodnejšie je použiť loopback
 - ktorý je samozrejme potrebné ohlásiť v IGP, aby sused vedel odpovedať
- IP adresa v príkaze **neighbor** u *suseda* bude cieľovou adresou jeho BGP paketov, a teda musí byť nastavená na IP adresu *nášho* loopbacku

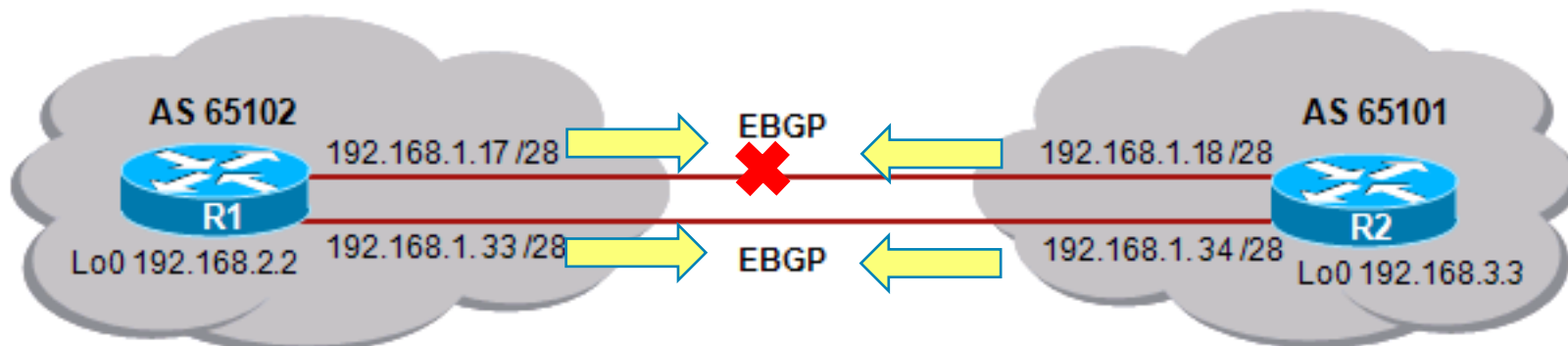
Príkaz: iBGP s použitím loopbackov



```
R2(config)# router bgp 65101
R2(config-router)# neighbor 172.16.1.1 remote-as 65100
R2(config-router)# neighbor 192.168.3.3 remote-as 65101
R2(config-router)# neighbor 192.168.3.3 update-source loopback0
R2(config-router)# exit
R2(config)# router eigrp 1
R2(config-router)# network 10.0.0.0
R2(config-router)# network 192.168.2.0
R2(config-router)#
```

```
R3(config)# router bgp 65101
R3(config-router)# neighbor 192.168.1.1 remote-as 65102
R3(config-router)# neighbor 192.168.2.2 remote-as 65101
R3(config-router)# neighbor 192.168.2.2 update-source loopback0
R3(config-router)# exit
R3(config)# router eigrp 1
R3(config-router)# network 10.0.0.0
R3(config-router)# network 192.168.3.0
R3(config-router)#
```

eBGP: Dual-Homed Problem



- Pri eBGP peeringu
 - IGP sa nepoužíva
 - Jediná adresa dosiahnuteľná bez ďalšej konfigurácie je IP adresa suseda na priamo pripojenej sieti, a loopback nie je priamo pripojené rozhranie
- Ak máme navyše eBGP peerov v dual home zapojení nastáva problém:
 - Ak na oboch smerovačoch použijeme len jeden príkaz neighbor
 - Napr. použitie linky 192.168.1.16/28
 - A ak linka zlyhá, BGP relácia je stratená, a pakety nebudú doručované z AS do AS
 - Bez ohľadu na existenciu druhej linky
 - Ak na oboch použijeme dva príkazy neighbor
 - Napr. použitie linky 192.168.1.16/28 a linky 192.168.1.32/28
 - Máme redundanciu, ale aj zdvojenie BGP komunikácie

EBGP: Dual-Homed riešenie



■ Riešenie:

- Použitie loopback adries ako pri iBGP
- Konfigurácia príkazom **neighbor ebgp-multihop** aby BGP proces vedel, že sused je ďalej ako jeden hop
 - Defaultne nastavenie je, že eBGP sused je priamo pripojený (def. TTL 1)
- Konfigurácia **statických ciest** aby boli loopback adresy vzájomne dosiahnuteľné
- Prínos
 - Použitie rozhraní s redundaciou, load balancing

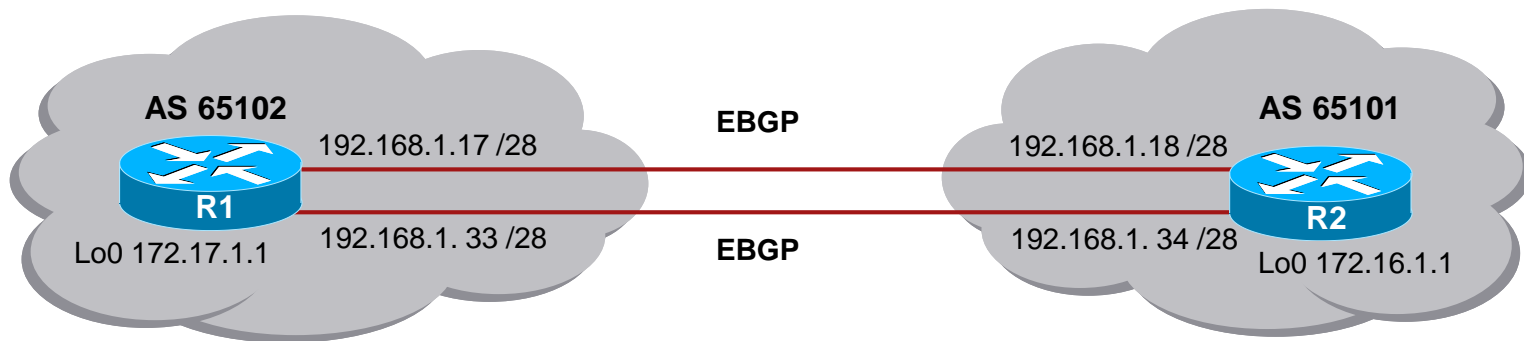
BGP príkaz `neighbor ebgp-multihop`

Router(config-router) #

```
neighbor {ip-address | peer-group-name} ebgp-multihop [ttl]
```

- Tento príkaz umožňuje zvýšiť počet hopov medzi nami a eBGP peerom
- Počet hopov sa rieši elegantne využitím hodnoty TTL v IP paketoch
- Ak sa `ttl` neuvedie, použije sa hodnota 255

Príklad na Multihop EBGP



```
R1(config)# router bgp 65102
R1(config-router)# neighbor 172.16.1.1 remote-as 65101
R1(config-router)# neighbor 172.16.1.1 update-source loopback0
R1(config-router)# neighbor 172.16.1.1 ebgp-multihop 2
R1(config-router)# exit
R1(config)# ip route 172.16.1.1 255.255.255.255 192.168.1.18
R1(config)# ip route 172.16.1.1 255.255.255.255 192.168.1.34
R1(config)#
```

```
R2(config)# router bgp 65101
R2(config-router)# neighbor 172.17.1.1 remote-as 65102
R2(config-router)# neighbor 172.17.1.1 update-source loopback0
R2(config-router)# neighbor 172.17.1.1 ebgp-multihop 2
R2(config-router)# exit
R2(config)# ip route 172.17.1.1 255.255.255.255 192.168.1.17
R2(config)# ip route 172.17.1.1 255.255.255.255 192.168.1.33
R2(config)#
```

BGP príkaz `neighbor shutdown`

Router (config-router) #

```
neighbor {ip-address | peer-group-name} shutdown
```

- Administratívne deaktivuje vybraného suseda
 - Príkaz nielen ukončí BGP reláciu ale aj odstráni všetky týkajúce sa smerovacie informácie
- Využíva sa pri údržbe konfigurácie a zmenách smerovacích politík

Router (config-router) #

```
no neighbor {ip-address | peer-group-name} shutdown
```

- Opätovne aktivuje suseda, ktorý bol deaktivovaný
 - Pozor – príkaz **neighbor activate** slúži na mierne iný účel: aktivácia suseda pre konkrétny typ sieťových adries (nebudeme sa učiť)

Nastavenie sietí v BGP - príkaz network

- Sú dve možnosti na šírenie sietí v BGP
 - Príkaz network
 - Redistribúcia z IGP

Router (config-router) #

```
network NETWORK-NUMBER [mask NETMASK] [route-map MAP-TAG]
```

- Správanie príkazu je zásadne odlišné od jeho významu v IGP protokoloch
 - IGP prehľadá rozhrania – ak nejaké rozhranie má IP adresu z rozsahu adries daného týmto príkazom, potom do IGP bude zaradená celá sieť tohto rozhrania
 - BGP prehľadá smerovaciu tabuľku – musí v nej nájsť sieť so zhodným číslom a maskou (nezávisí na pôvode informácie)
- BGP zaradí sieť s presne danou adresou siete a presnou maskou do zoznamu sietí, ktoré oznámi svojim susedom
- Ak je zadaný príkaz network s classfull sieťou *bez masky*, a BGP má aspoň jednu subnet z rozsahu v smerovacej tabuľke
 - Ak je zapnuté auto-summary - Pošle info o classfull sieti (nie subnete)
 - Ak je vypnuté (no) auto-summary – Neoznami nič (čaká net mask syntax)

BGP cesta musí byť v IP smerovacej tabuľke

- Je dôležité zdôrazniť, že ak má byť daná sieť šírená cez BGP *musí daná* sieť s danou maskou existovať v smerovacej tabuľke
 - Ak nie je definovaná maska, použije sa classful default maska
- Napríklad sumarizácia viac sietí do CIDR bloku 192.168.0.0/16:

```
network 192.168.0.0 mask 255.255.0.0  
ip route 192.168.0.0 255.255.0.0 null0
```
- Až teraz BGP nájde presnú zhodu pre sieť a masku v smerovacej tabuľke a informuje o sieti 192.168.0.0/16 jeho susedov

Sumarizácia – agregácia v BGP

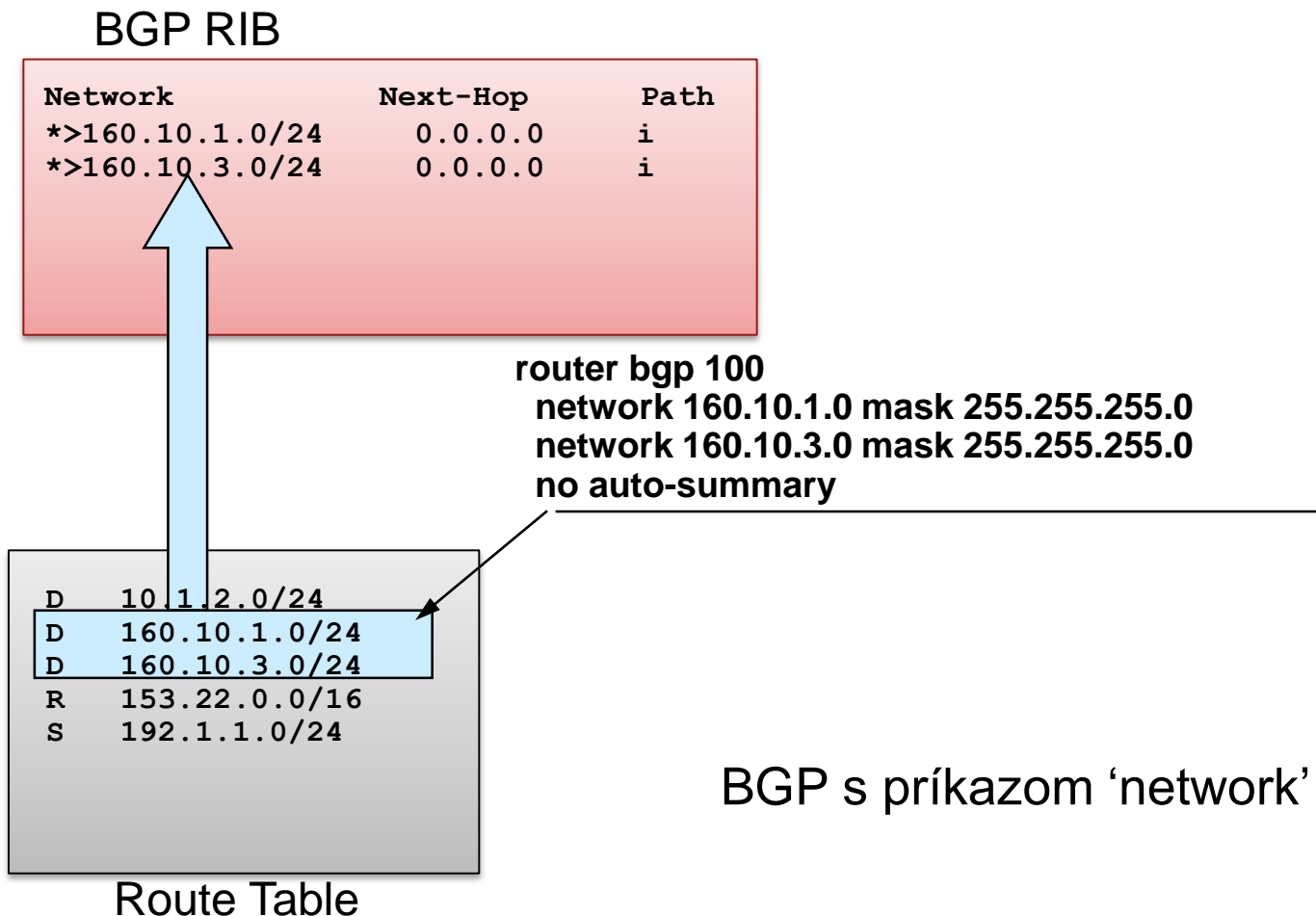
- Do BGP sa siete vnášajú spravidla pomocou redistribúcie
- Sumarizácia (v BGP terminológii sa tento proces nazýva agregácia) sa realizuje príkazom

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

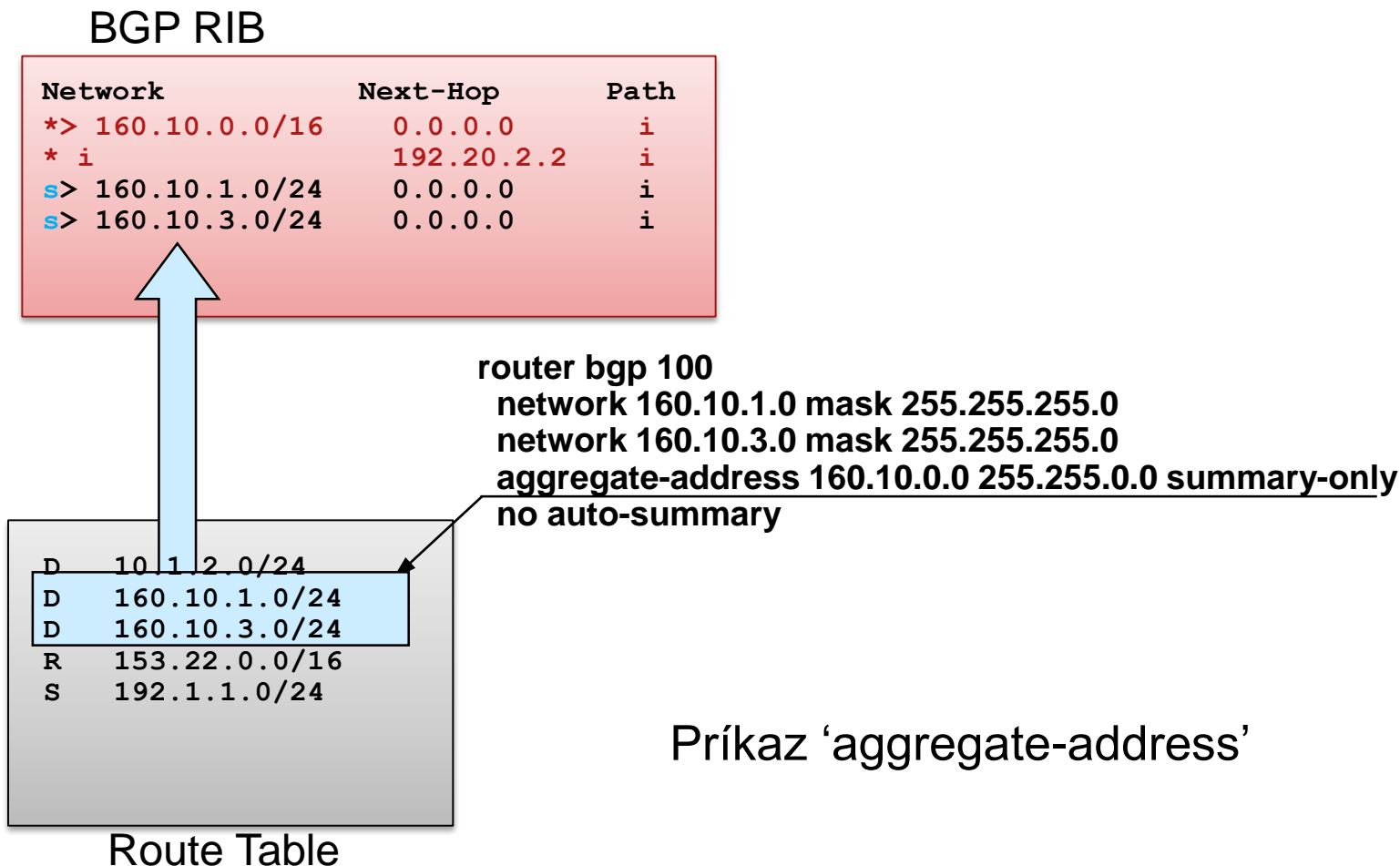
```
aggregate-address NETWORK MASK [summary-only]
```

- Parameter **summary-only** zabezpečí, že sa rozpošle iba agregovaná sieť, nie aj jej komponenty
 - Potrebné v prípadoch, že niekomu chceme poslať agregát, inému zasa špecifickejšie komponenty
- Príkaz aggregate vyžaduje aby agregované siete boli v smerovacej tabuľke (aj zadané príkazom network)

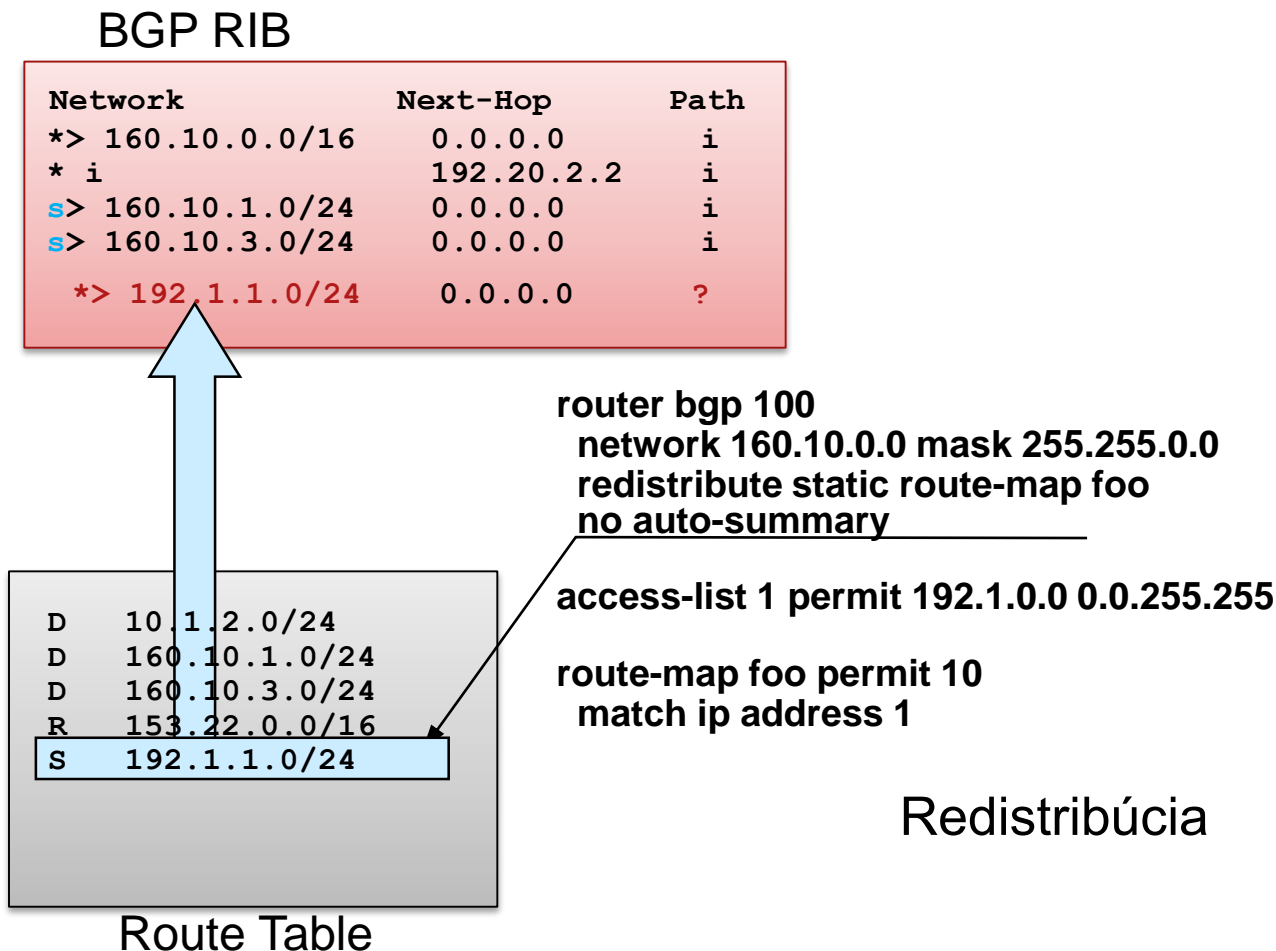
Budovanie BGP Routing Information Base



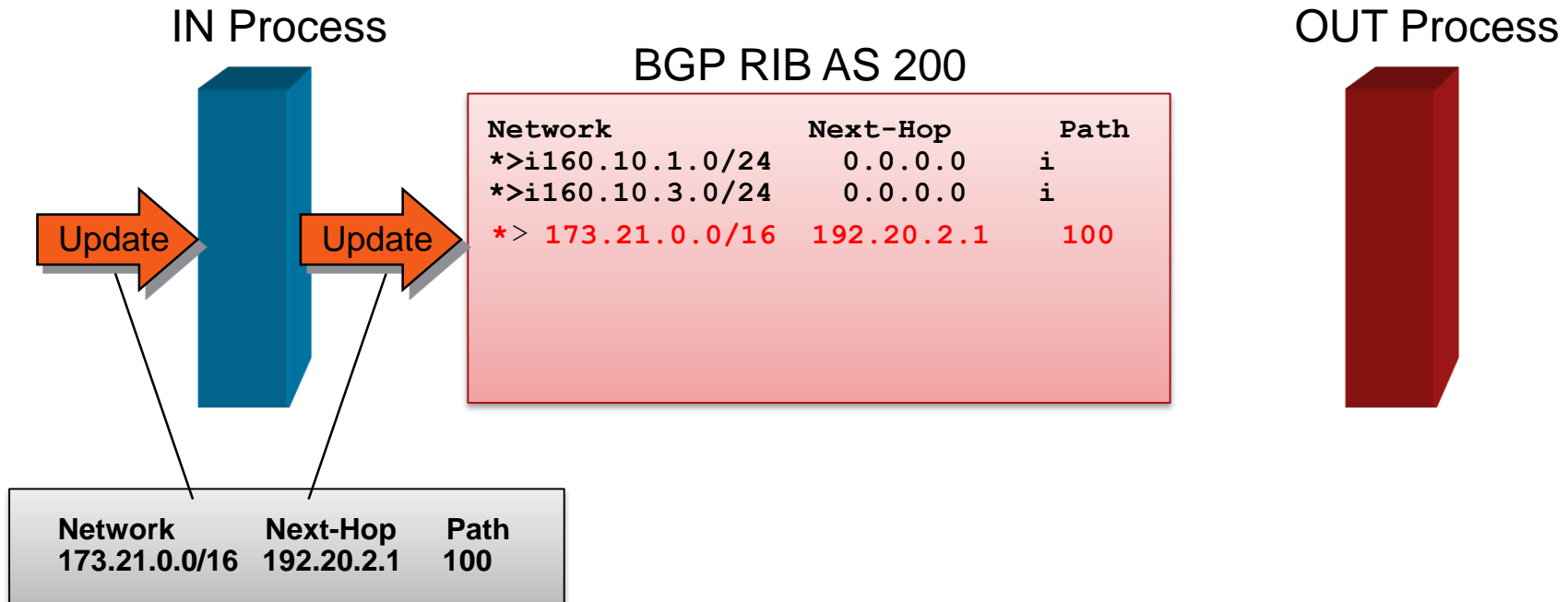
Budovanie BGP Routing Information Base



Budovanie BGP Routing Information Base



Budovanie BGP Routing Information Base



- BGP “in” proces
 - Príjem info o ceste od suseda (peer)
 - Umiestnenie BGP cesty do BGP tabuľky
 - “najlepšia cesta” je označená (denoted by “>”)

BGP Routing Information Base

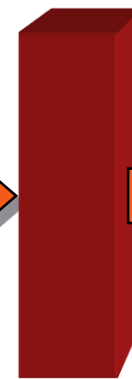
IN Process



BGP RIB AS 200

Network	Next-Hop	Path
*>i160.10.1.0/24	0.0.0.0	i
*>i160.10.3.0/24	0.0.0.0	i
*> 173.21.0.0/16	192.20.2.1	100

OUT Process

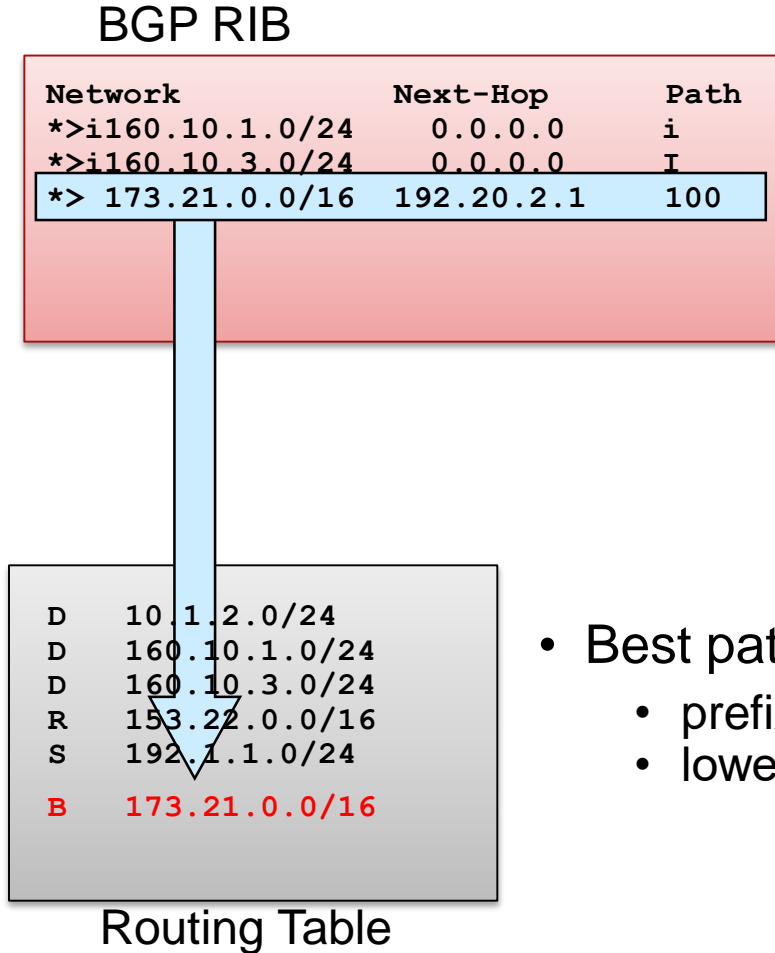


Network	Next-Hop	Path
160.10.1.0/24	192.20.2.2	200
160.10.3.0/24	192.20.2.2	200
173.21.0.0/16	192.20.2.2	200 100

Zmena Next-Hop

- BGP “out” proces
 - Update na základe info v RIB
 - Môže byť modifikovaný na základe politik
 - Poslanie update peerovi

BGP Routing Information Base



- Best paths installed in routing table if:
 - prefix and prefix length are unique
 - lowest “protocol distance”

Posielanie default route v BGP

- V BGP sa dá poslať default route vybranému susedovi príkazom

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

```
neighbor {ip-address | peer-group-name} default-originate
```

- Default route na aktuálnom BGP routeri nemusí existovať (na chrbtici internetu neexistuje default route)

BGP Synchronizácia

- *Disabled by default in Cisco IOS Software Release 12.2(8)T and later*

- Povolenie

```
Router(config-router) # synchronization
```

- Zakázanie

```
Router(config-router) # no synchronization
```



Autentifikácia v BGP

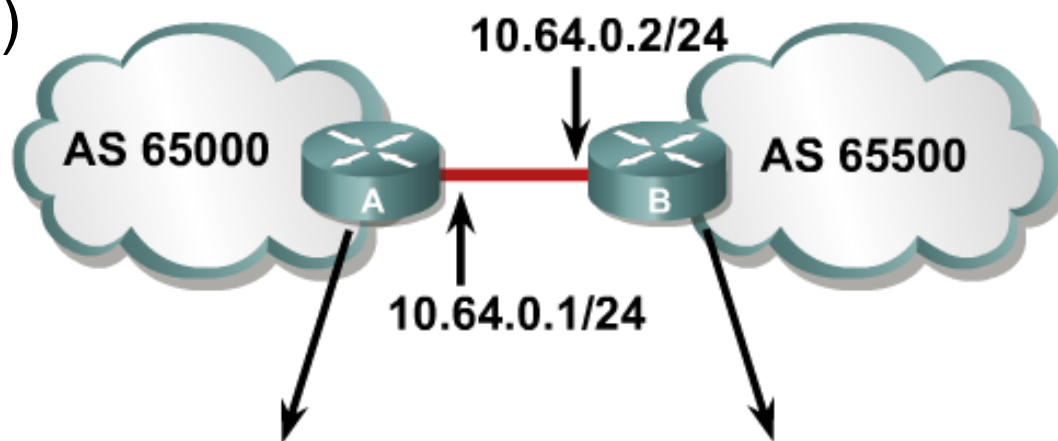


Autentifikácia v BGP

Router (config-router) #

```
neighbor {ip-address | peer-group-name} password string
```

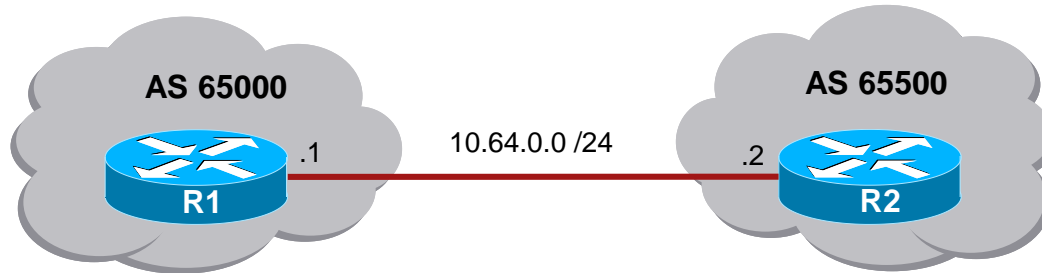
- BGP používa MD5 autentifikáciu
 - Hash sa počíta z hesla (key) a správy
- Pre každého suseda sa môže definovať nezávislý kľúč (heslo)



```
router bgp 65500
neighbor 10.64.0.2 remote-as 65500
neighbor 10.64.0.2 password v6lne0qkel33&
```

```
router bgp 65500
neighbor 10.64.0.1 remote-as 65000
neighbor 10.64.0.1 password v6lne0qkel33&
```

Príklad konfigurácie MD5 autentifikácie



```
R1(config)# router bgp 65000
R1(config-router)# neighbor 10.64.0.2 remote-as 65500
R1(config-router)# neighbor 10.64.0.2 password BGP-Pa55w0rd
R1(config-router)#
```

```
R2(config)# router bgp 65500
R2(config-router)# neighbor 10.64.0.1 remote-as 65000
R2(config-router)# neighbor 10.64.0.1 password BGP-Pa55w0rd
R2(config-router)#
```

Diagnostika:

Ak jeden smerovač má heslo pre suseda a druhý nemá:

```
%TCP-6-BADAUTH: No MD5 digest from 10.1.0.2(179) to 10.1.0.1(20236)
```

Ak oba smerovače majú heslá, ale nesprávne:

```
%TCP-6-BADAUTH: Invalid MD5 digest from 10.1.0.1(12293) to
10.1.0.2(179)
```



Diagnostika BGP



Príkazy na overenie a diagnostiku BGP

Command	Description
<code>show ip bgp</code>	Zobrazí položky v BGP tabuľke Môžeme špecifikovať adresu siete na získanie bližších informácií o danej sieti.
<code>show ip bgp neighbors</code>	Zobrazí detailné informácie o TCP a BGP spojeniach na susedov (BGP table).
<code>show ip bgp summary</code>	Zobrazí stav všetkých BGP spojení.
<code>show ip bgp neighbors {address} advertised-routes</code>	Zobrazí všetky cesty (smery) ktoré sú zasielané na susedov.
<code>show ip bgp rib-failure</code>	Zobrazí BGP cesty (smery) ktoré neboli inštalované do routing information base (RIB), a dôvod z akého sa tak stalo.
<code>debug ip bgp [dampening events keepalives updates]</code>	
<code>Show tcp brief inc 179</code>	Zobrazí, či je otvorená TCP relácia na BGP peera

Overenie BGP: show ip bgp

Zobrazí BGP topologickú databázu (BGP tabuľku).

Prvý stĺpec: STATUS CODE

* znamená, že next hop adresa v piatom stĺpci je platná

- r znamená RIB chybu, a cesta (smer, route) nebola inštalovaná do RIB

- môžu byť ešte ďalšie kódy

Druhý stĺpec: Znak „>“ označuje, ktorú cestu BGP vybralo ako najlepšiu

Táto route bude umiestnená do smerovacej tabuľky.

Tretí stĺpec: Je buď prázdny alebo obsahuje znak „i”

- Ak obsahuje znak i, smerovač sa dozvedel o tejto sieti cez iBGP suseda

- Ak je prázdny, BGP sa naučilo o sieti od externého peera

```
R1# show ip bgp
```

```
BGP table version is 14, local router ID is 172.31.11.1
```

```
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal, r RIB-failure, S Stale
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete
```

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>	10.1.0.0/24	0.0.0.0	0		32768	i
* i		10.1.0.2	0	100	0	i
*>	10.1.1.0/24	0.0.0.0	0		32768	i
*>i	10.1.2.0/24	10.1.0.2	0	100	0	i
*>	10.97.97.0/24	172.31.1.3			0	64998 64997 i
*		172.31.11.4			0	64999 64997 i
* i		172.31.11.4	0	100	0	64999 64997 i
*>	10.254.0.0/24	172.31.1.3	0		0	64998 i
*		172.31.11.4			0	64999 64998 i
* i		172.31.1.3	0	100	0	64998 i
r>	172.31.1.0/24	172.31.1.3	0		0	64998 i
r		172.31.11.4			0	64999 64998 i
r i		172.31.1.3	0	100	0	64998 i
*>	172.31.2.0/24	172.31.1.3	0		0	64998 i

Obsahuje zoznam troch BGP atribútov cesty: metric (MED), local preference, a weight.

Časť Path obsahuje zoznam AS ciest. Posledné AS číslo je originating AS.

Ak je prázdne, sieť je z daného AS

Posledný stĺpec zobrazuje atribút ORIGIN.

- i znamená, že prvý smerovač pravdepodobne použil príkaz **network** na distribúciu info cez BGP.

- ? znamená, že sieť bola pravdepodobne redistribuovaná z IGP do BGP

Overenie BGP: `show ip bgp rib-failure`

- Zobrazí BGP cesty (smery) ktoré neboli inštalované do routing information base (RIB), a dôvod z akého sa tak stalo
- Príklad
 - V Routing tabuľke už existuje daná cesta s lepším AD.

```
R1# show ip bgp rib-failure
```

Network	Next Hop	RIB-failure	RIB-NH Matches
172.31.1.0/24	172.31.1.3	Higher admin distance	n/a
172.31.11.0/24	172.31.11.4	Higher admin distance	n/a

Overenie BGP: show ip bgp *PREFIX*

```
ISP2# sh ip bgp 192.168.0.0/24
```

```
BGP routing table entry for 192.168.0.0/24, version 3
```

```
Paths: (1 available, best #1, table Default-IP-Routing-Table)
```

```
Not advertised to any peer
```

```
100
```

```
172.16.0.2 from 172.16.0.2 (192.168.1.1)
```

```
Origin IGP, metric 0, localpref 100, valid, external, best
```

Overenie BGP: show ip bgp summary

Overenie BGP susedov a stavu.

```
RouterA# show ip bgp summary
BGP router identifier 10.1.1.1, local AS number 65001
BGP table version is 124, main routing table version 124
9 network entries using 1053 bytes of memory
22 path entries using 1144 bytes of memory
12/5 BGP path/bestpath attribute entries using 1488 bytes of memory
6 BGP AS-PATH entries using 144 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 3829 total bytes of memory
BGP activity 58/49 prefixes, 72/50 paths, scan interval 60 secs
```

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.1.0.2	4	65001	11	11	124	0	0	00:02:28	8
172.31.1.3	4	64998	21	18	124	0	0	00:01:13	6
172.31.11.4	4	64999	11	10	124	0	0	00:01:11	6

RID
suseda

Verzia
BGP u
suseda

Susedovo
AS

Počet
správ
prijatých
od suseda

Počet
správ
odoslaných
na suseda

Posledná
verzia
tabuľky
zaslaná
susedovi

Počet
správ v In
a Out
fronte

Ako dlho
je sused
UP

Koľko Prefixov
som prijal od
suseda (State je
logicky
established)

Overenie BGP: debug ip bgp updates

Overenie BGP susedstva po resetovaní BGP procesu

```
R1# debug ip bgp updates
Mobile router debugging is on for address family: IPv4 Unicast
R1# clear ip bgp 10.1.0.2
<output omitted>
*May 24 11:06:41.309: %BGP-5-ADJCHANGE: neighbor 10.1.0.2 Up
*May 24 11:06:41.309: BGP(0): 10.1.0.2 send UPDATE (format) 10.1.1.0/24, next 10.1.0.1, metric 0,
path Local
*May 24 11:06:41.309: BGP(0): 10.1.0.2 send UPDATE (prepend, chgflags: 0x0) 10.1.0.0/24, next
10.1.0.1, metric 0, path Local
*May 24 11:06:41.309: BGP(0): 10.1.0.2 NEXT_HOP part 1 net 10.97.97.0/24, next 172.31.11.4
*May 24 11:06:41.309: BGP(0): 10.1.0.2 send UPDATE (format) 10.97.97.0/24, next 172.31.11.4, metric
0, path 64999 64997
*May 24 11:06:41.309: BGP(0): 10.1.0.2 NEXT_HOP part 1 net 172.31.22.0/24, next 172.31.11.4
*May 24 11:06:41.309: BGP(0): 10.1.0.2 send UPDATE (format) 172.31.22.0/24, next 172.31.11.4,
metric 0, path 64999
<output omitted>
*May 24 11:06:41.349: BGP(0): 10.1.0.2 rcvd UPDATE w/ attr: nexthop 10.1.0.2, origin i, localpref
100, metric 0
*May 24 11:06:41.349: BGP(0): 10.1.0.2 rcvd 10.1.2.0/24
*May 24 11:06:41.349: BGP(0): 10.1.0.2 rcvd 10.1.0.0/24
```

Overenie BGP: show ip bgp neighbors

Overenie BGP stavu so susedom.

```
R1# show ip bgp neighbors
```

```
BGP neighbor is 172.31.1.3, remote AS 64998, external link
```

```
BGP version 4, remote router ID 172.31.2.3
```

```
BGP state = Established, up for 00:19:10
```

```
Last read 00:00:10, last write 00:00:10, hold time is 180, keepalive  
interval is 60 seconds
```

```
Neighbor capabilities:
```

```
Route refresh: advertised and received(old & new)
```

```
Address family IPv4 Unicast: advertised and received
```

```
Message statistics:
```

```
InQ depth is 0
```

```
OutQ depth is 0
```

	Sent	Rcvd
Opens:	7	7
Notifications:	0	0
Updates:	13	38

```
<output omitted>
```

Overenie: sh tcp brief

```
R2# sh tcp brief | include 179
```

6A7B27A0	172.16.0.2.55333	172.16.0.1.179	ESTAB
6A723F68	10.0.0.2.179	10.0.0.1.62492	ESTAB



Reset BGP spojení



Prestávka, Lab 7-1

Reštart BGP spojenia

- Po zmene politík (access-listy pre filtrovanie, zmena hodnôt atribútov) sa zmeny prejavia len na **nových** prijatých alebo **nových** odoslaných prefixoch
- Zmena na doposiaľ odoslaných a prijatých prefixoch sa sama od seba neuskutoční
 - Administrátor musí ručne vyvolať akciu, ktorá spôsobí aplikovanie nových politík
- Spôsoby na vynútenie aktualizácie prefixov:
 - Hard reset
 - Soft reset
 - Route refresh

Hard Reset BGP spojení

- Reset je technika na informovanie susedov o nastaní zmeny politik
 - Ak nastane Reset relácie, všetky informácie spojené s reláciou sú zneplatnené a vymazané

Router#

```
clear ip bgp *
```

- Zruší BGP spojenia so **všetkými** susedmi
- Celá BGP tabuľka sa zahodí (flush daných ciest v RT)
- Všetky spojenia prejdú do stavu Idle (zatvorí sa tcp) a informácie sa kompletne musia preniesť od susedov nanovo

Router#

```
clear ip bgp neighbor-address
```

- Zruší BGP spojenie s **daným** susedom (flush daných ciest v RT)
- Spojenie s týmto susedom prejde do stavu Idle (zatvorí sa tcp) a je potrebné nanovo sa s ním synchronizovať
- Menej drastické než **clear ip bgp ***

Outbound Soft Reset

Router#

```
clear ip bgp { * | neighbor-address } [ soft out | out]
```

- Cesty od daného suseda sa ponechajú v našich tabuľkách
- Prepošleme však susedovi všetky naše BGP smery nanovo bez resetu spojenia
 - Spojenie so susedom zostáva v stave Established
- Táto voľba je ideálna pre situácie, keď sa mení **outbound** policy
 - Príkaz **soft out** nemá efekt, ak sa mení inbound policy

Inbound Soft Reset

Router(config-router) #

```
neighbor [ip-address] soft-reconfiguration inbound
```

- Idea Inbound Soft Reset je pamätať si *všetky* informácie od BGP suseda *nefiltrované* v osobitnej databáze
 - Pri zmene inbound policy sa využije obsah tejto databázy namiesto opätovného stiahnutia dát od suseda
- Táto funkcionálna je pamäťovo náročná
- Výhoda: Nevyžaduje sa akcia druhej strany (resend)

Router#

```
clear ip bgp { * | neighbor-address } soft in
```

- Príkaz spôsobí zabudnutie naučených informácií od daného suseda a aplikuje aktuálnu inbound policy na údaje z hore vytvorenej osobitnej databázy

Route Refresh: Dynamický Inbound Soft Reset

Router#

```
clear ip bgp { * | neighbor-address } [soft in | in]
```

- Táto funkcionálnosť umožňuje nanovo si vyžiadať od suseda odoslanie informácií, pričom budú podrobené aktuálnej inbound policy
 - Pôvodná špecifikácia BGP to neumožňovala, pri Route Refresh sa jedná o dodatočné RFC 2918
 - Zavedená nová BGP správa **ROUTE-REFRESH**
- Smery odoslané danému susedovi nie sú dotknuté
- Nevyžaduje sa kópia informácií v osobitnej databáze
- Spojenie zostáva v stave Established
- Podporované od verzie IOS 12.0(2)S and 12.0(6)T
- Ak všetci susedia podporujú Route Refresh, netreba písať **soft**

Overenie či sused podporuje soft refresh

```
ISP2# sh ip bgp neighbors
```

```
BGP neighbor is 172.16.0.2, remote AS 100, external link  
  BGP version 4, remote router ID 192.168.1.1  
  BGP state = Established, up for 03:12:30  
  Last read 00:00:30, last write 00:00:30, hold time is  
  180, keepalive interval is 60 seconds
```

```
  Neighbor capabilities:
```

```
    Route refresh: advertised and received(old & new)
```

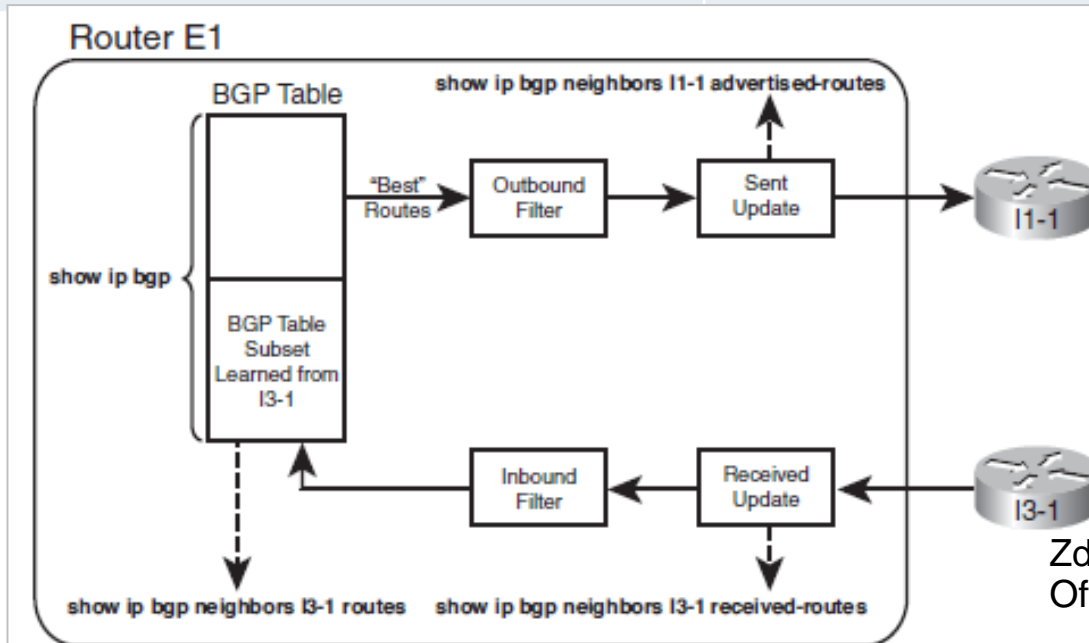
```
    Address family IPv4 Unicast: advertised and received
```

```
...
```

```
...
```

Monitoring přijatých BGP Routes

Command	Description
<code>show ip bgp neighbors {address} received-routes</code>	Displays all received routes (both accepted and rejected) from the specified neighbor.. Have to be neighbor 172.16.0.1 <i>soft-reconfiguration inbound</i> Turned on
<code>show ip bgp neighbors {address} routes</code>	Displays all routes that are received and accepted from the specified neighbor. This output is a subset of the output displayed by the <code>received-routes</code> keyword.
<code>show ip bgp</code>	Displays entries in the BGP table.
<code>show ip bgp neighbors {address} advertised-routes</code>	Displays all BGP routes that have been advertised to neighbors.



Zdroj: Wendell Odom: CCNP Route
Official certification guide

Atribúty v BGP



BGP updates a atribúty cesty

- Smerovač posiela BGP update s informáciou o jednej ceste
 - Správa obsahuje
 - Network Layer Reachability Information (NLRI)
 - Informácia o sieťach dosiahnuteľných touto cestou
 - Atribúty cesty (Path attributes)
- Atribúty cesty sú BGP metriky popisujúce cestu do týchto sietí/siete
 - BGP ich používa na výber najlepšej cesty
 - Môžu byť použité na policy routing
 - Umožňujú adminom lepšiu kontrolu nad výberom cesty
 - Filtrovanie, preferovanie ciest, kustomizácia BGP správaní apod.
 - Type, Length & Value (TLV)

BGP Attribute Type

- Type code 1 ORIGIN
- Type code 2 AS_PATH
- Type code 3 NEXT_HOP
- Type code 4 MULTI_EXIT_DISC
- Type code 5 LOCAL_PREF
- Type code 6 ATOMIC_AGGREGATE
- Type code 7 AGGREGATOR
- Type code 8 Community (Cisco-defined)
- Type code 9 Originator-ID (Cisco-defined)
- Type code 10 Cluster list (Cisco-defined)

Path Attributes Information

Update Message						Information		
Octets	16	2	1	2	Variable	2	Variable	Variable
	Marker	Length	Type	Unfeasible Routes Length	Withdrawn Routes	Attribute Length	Path Attributes	NLRI

BGP atribúty

- Atribút je vlastnosť smeru, ktorá charakterizuje nejakú jeho vlastnosť, a na základe ktorej si BGP vyberá najvhodnejšiu cestu
- BGP pozná mnoho atribútov, ktoré sú 4 základných druhov:
 - **Well-known mandatory:**
 - atribút, ktorý musí povinne podporovať každá implementácia BGP (**well-known**) a ktorý musí byť prítomný pri každom popise nejakej cesty (**mandatory**)
 - **Well-known discretionary:**
 - atribút, ktorý musí povinne podporovať každá implementácia BGP (**well-known**), ale ktorý nemusí byť prítomný v popise cesty (**discretionary**)
 - **Optional transitive:**
 - atribút, ktorý nemusí podporovať každá implementácia BGP (**optional**), avšak musí ho preposlať susedom napriek tomu, že mu nerozumie (**transitive**)
 - **Optional nontransitive:**
 - atribút, ktorý nemusí podporovať každá implementácia BGP (**optional**), a ak mu nerozumie, nesmie ho preposlať susedom (**nontransitive**)
- Všetky well-known atribúty majú tranzitívnu povahu

BGP atribúty

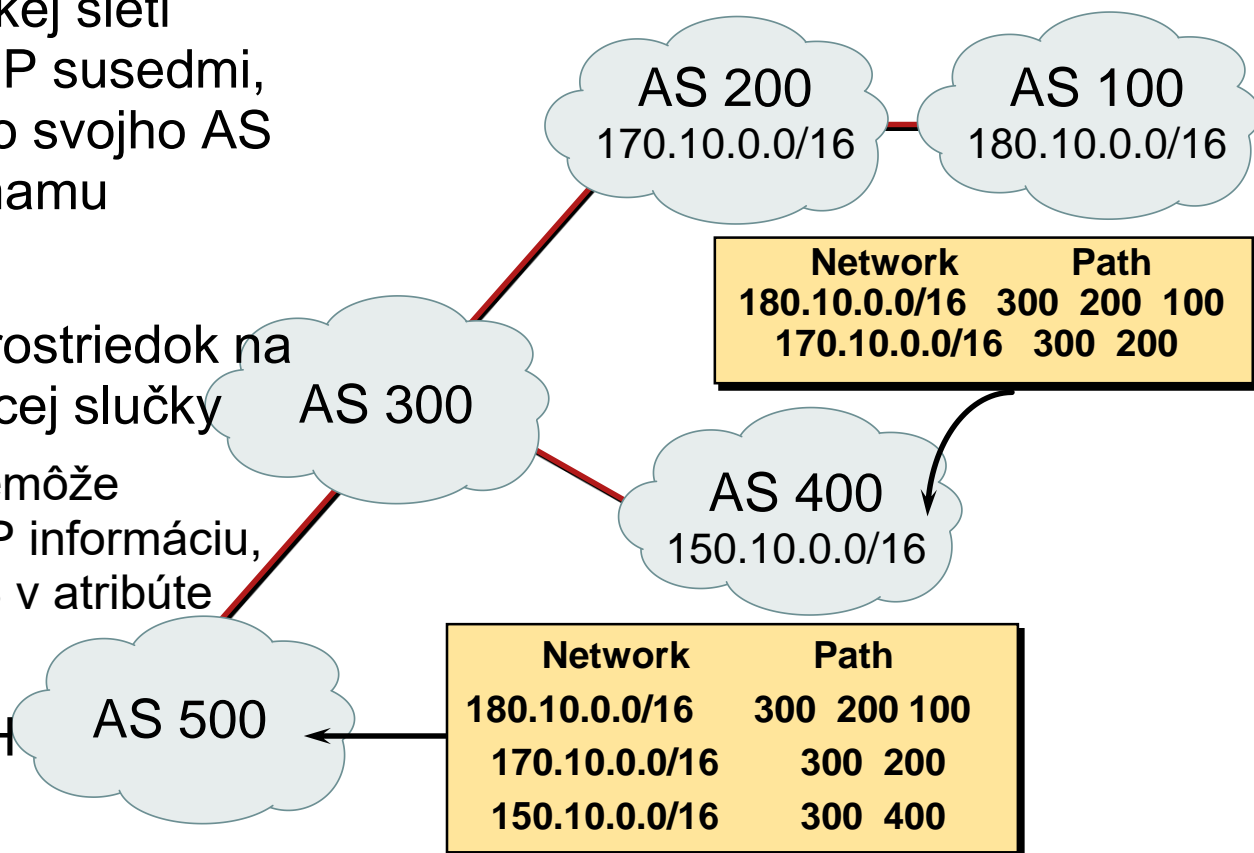
- Well-known **m**andatory: (pomôcka: **MONA**)
 - **O**RIGIN: pôvod smeru (iBGP, eBGP, neznámy)
 - **N**EXT_HOP: next hop pre daný smer
 - **A**S_PATH: zoznam AS na ceste k danému smeru
- Well-known **d**iscretionary: (pomôcka: **DALA**)
 - **A**TOMIC_**A**GGREGATE: info o nerozdeliteľnej sieti
 - **L**OCAL_PREF: vyjadrenie lokálnej preferencie
- Optional Transitive:
 - AGGREGATOR: zdroj agregovanej informácie
 - COMMUNITY:
- Optional Nontransitive:
 - MULTI_EXIT_DISC: odporúčanie pre výstup z jedného AS do druhého

BGP atribúty - Weight

- Okrem toho Cisco zavádza vlastný atribút **Weight**
 - Lokálny pre daný router, nepreposiela sa nikomu
- BGP atribúty možno nastavovať alebo kontrolovať pomocou route-map konštruktov
- Každá kontrola, nastavovanie a filtrovanie sa deje pomocou príkazu neighbor pre každého suseda alebo peer group nezávisle
 - Na rozdiel od IGP, kde sa filtrovanie robí spravidla hromadne, nezávisle od odosielateľa informácie,
 - v BGP sa zasa spravidla realizuje filtrovanie pre každého suseda zvlášť
- BGP má stanovené poradie, v akom vyhodnocuje jednotlivé atribúty pre výber najlepšej cesty

Well-known mandatory atribút: AS_PATH

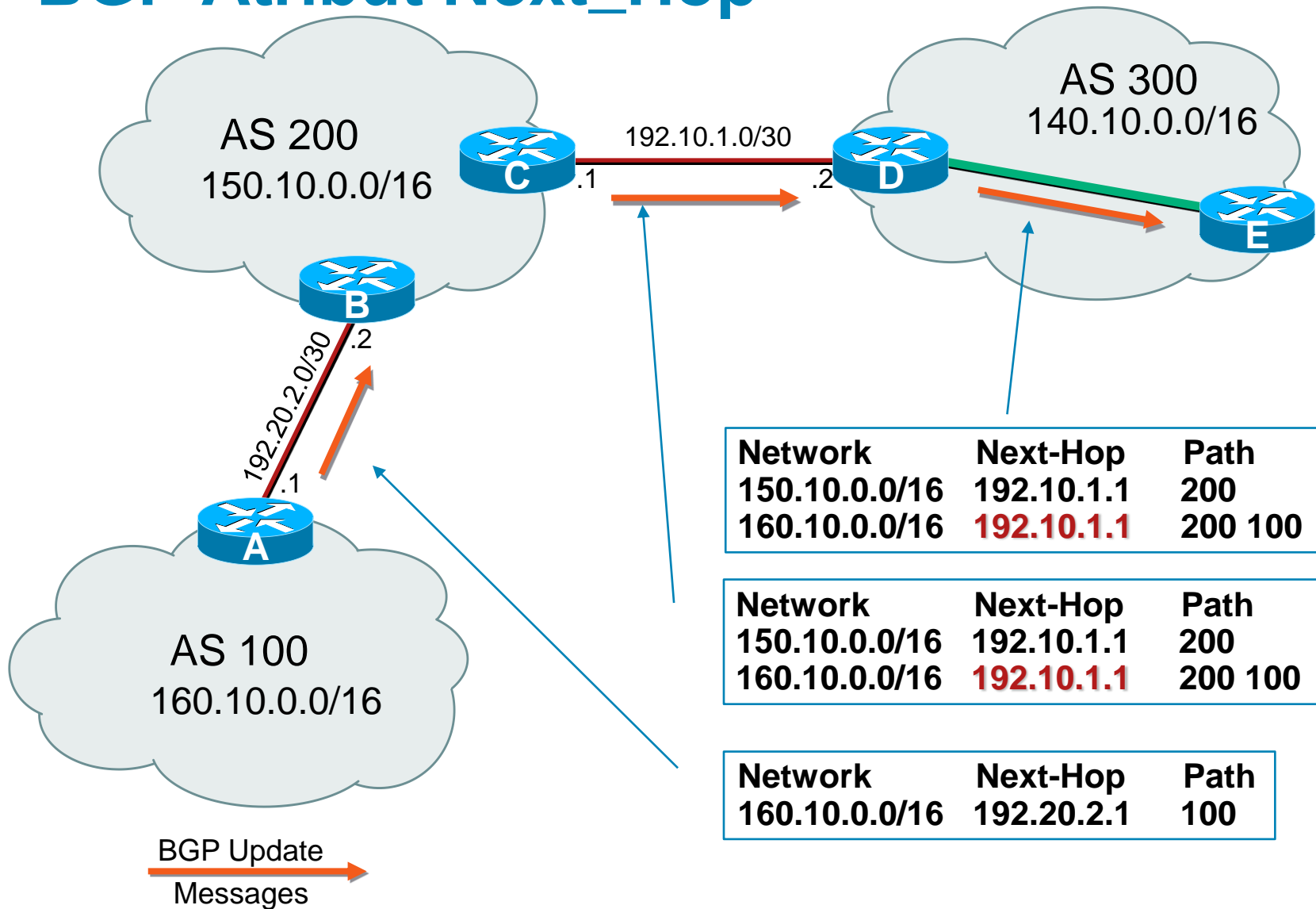
- Atribút AS_PATH predstavuje zoznam AS, cez ktoré treba prejsť, než sa dostaneme do cieľovej siete
- Keď informácia o nejakej sieti prechádza medzi eBGP susedmi, **odosielateľ** pripojí číslo svojho AS na začiatok tohto zoznamu (prepending)
- AS_PATH slúži ako prostriedok na eliminovanie smerovacej slučky
 - Router v istom AS nemôže akceptovať cez eBGP informáciu, ktorá už toto číslo AS v atribúte AS_PATH obsahuje
- iBGP AS do AS_PATH nepridáva



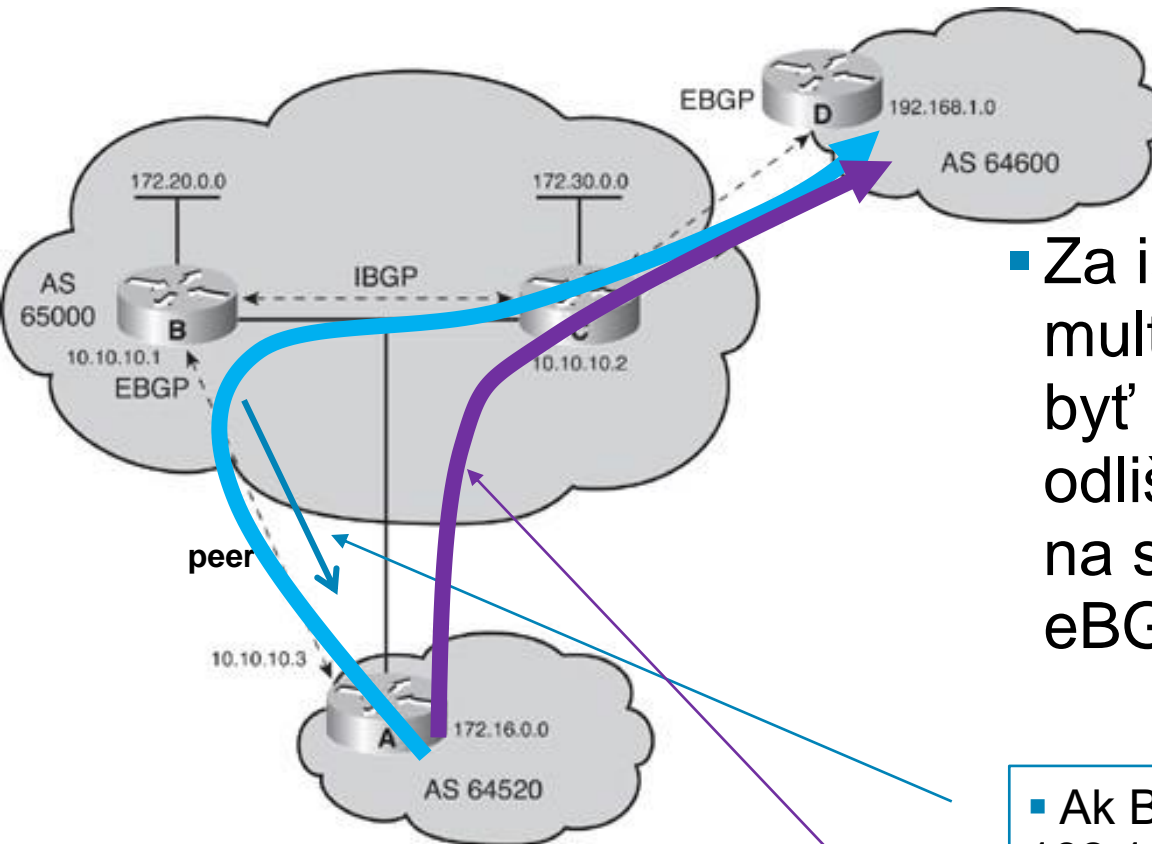
Well-known mandatory atribút NEXT_HOP

- Atribút NEXT_HOP je IP adresa nasledujúceho hopu
 - BGP vidí cestu ako poradie AS systémov, nie ako poradie routerov
 - NEXT_HOP vyjadruje IP adresu hraničného routera **v nasledujúcom AS**
- Z toho logicky vyplýva správanie sa atribútu NEXT_HOP
 - Pri eBGP (medzi AS) ho nastaví **odosielateľ** na svoju IP adresu (podľa update-source)
 - Prechodom cez iBGP (vo vnútri AS) sa jeho hodnota nebude meniť
- To má závažný dôsledok
 - Daný iBGP v AS musí poznať cestu k príslušnému hraničnému routeru v inom AS (Next-HOP)
 - Inak nebude možné vložiť tento BGP smer do smerovacej tabuľky
 - Alebo budú pakety pre daný smer dropnuté

BGP Atribút Next_Hop



BGP third party next hop

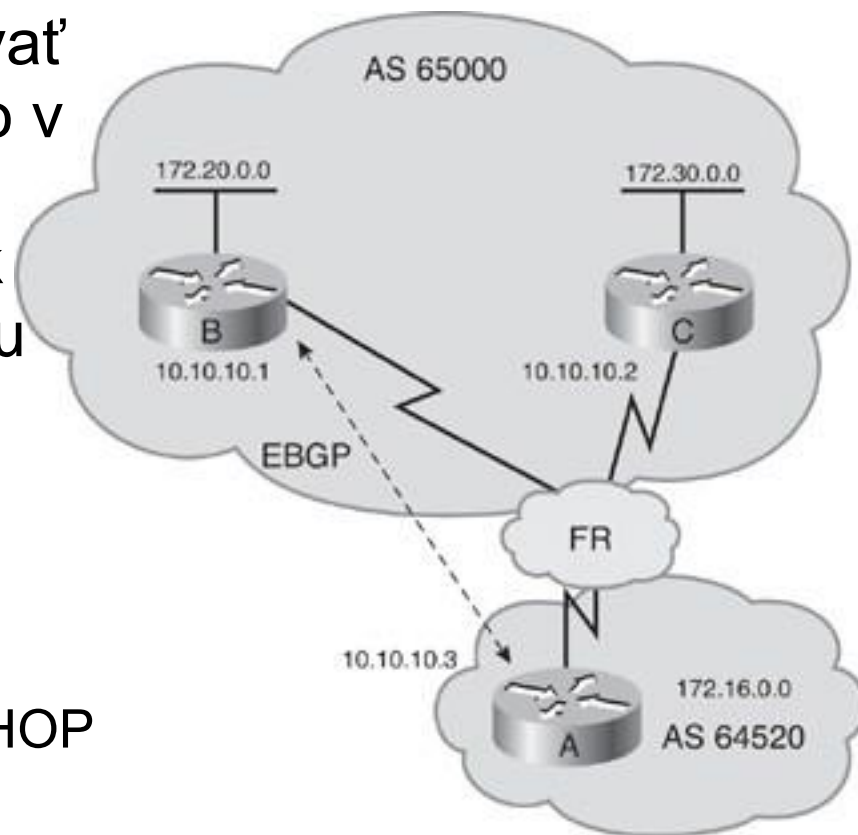


- Za istých okolností na multiaccess sieťach môže byť hodnota NEXT_HOP odlišná, vždy však bude na spoločnej IP sieti s eBGP peerom

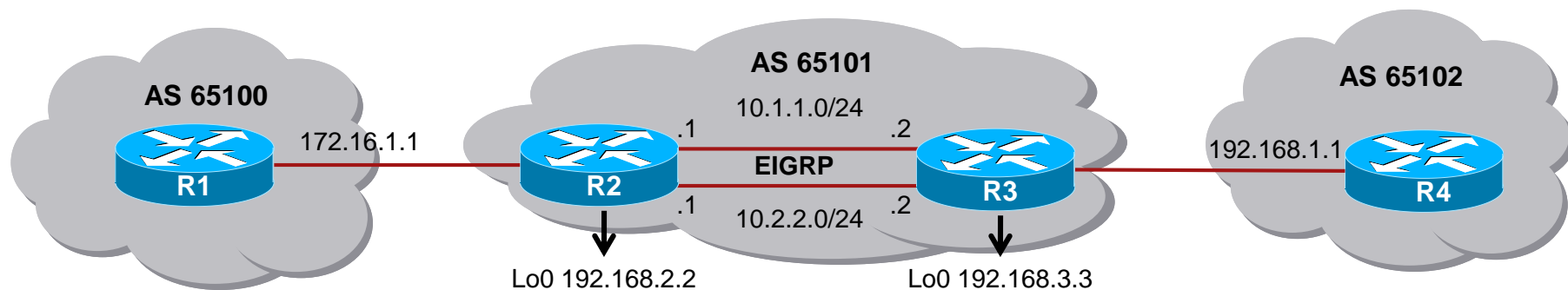
- Ak B pošle A update o sieti 192.168.1.0 uvedie ako next hop nie seba ale 10.10.10.2
- Optimálnejšie z pohľadu smerovania do cieľa

BGP atribút NEXT_HOP

- Tieto vlastnosti môžu komplikovať situáciu pri NBMA sieťach alebo v topológiách, kde nie je všetkým BGP speakerom známa cesta k príslušnému hraničnému routeru
- Riešenie: **neighbor A.B.C.D next-hop-self**
 - V smeroch získaných z eBGP a **preposielaných** na daného iBGP suseda A.B.C.D zmeníme NEXT_HOP na seba
 - Príkaz neplatí pri route reflectoroch – v takom prípade sa dá NEXT_HOP modifikovať pomocou route-map



Príklad Next Hop Self



```
R2(config)# router bgp 65101
R2(config-router)# neighbor 172.16.1.1 remote-as 65100
R2(config-router)# neighbor 192.168.3.3 remote-as 65101
R2(config-router)# neighbor 192.168.3.3 update-source loopback0
R2(config-router)# neighbor 192.168.3.3 next-hop-self
R2(config-router)# exit
R2(config)# router eigrp 1
R2(config-router)# network 10.0.0.0
R2(config-router)# network 192.168.2.0
R2(config-router)#
```

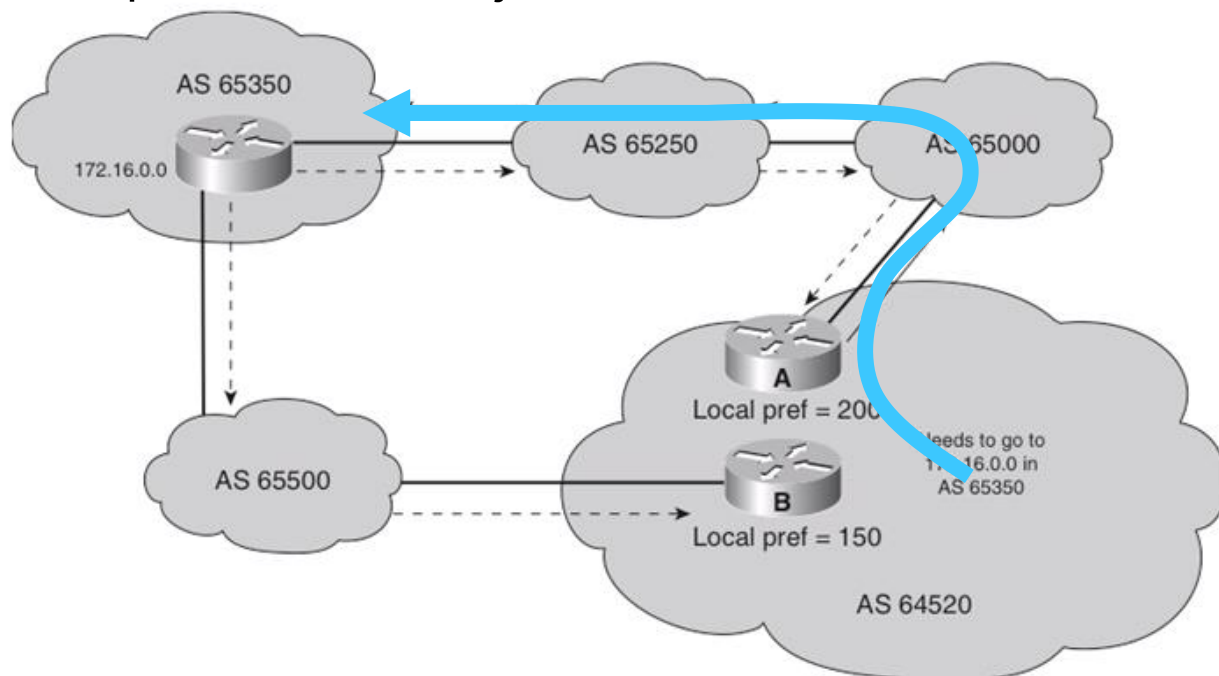
Elegantné riešenie ako zabezpečiť aby iBGP peer nemusel riešiť dostupnosť IP adresy externého eBGP

Well-known mandatory atribút ORIGIN

- Atribút ORIGIN vyjadruje pôvod informácie o ceste
 - „i“ – sieť má pôvod v súčasnom AS
 - Sieť je interná pre AS (internal)
 - vnesená do BGP príkazom network
 - Je v RT napr. cez IGP
 - „e“ – sieť je redistribuovaná z historického EGP protokolu
 - EGP už nie je podporované
 - „?“ – pôvod siete nie je známy (Incomplete)
 - Alebo naučený nejakým iným spôsobom (redistribúcia)
- Pri výbere cesty sa berie do úvahy poradie
 - **IGP (i) < EGP (e) < INCOMPLETE (?)**
 - Nižší je lepší

Well-Known Discretionary atribút LOCAL_PREF

- Atribút LOCAL_PREF označuje preferenciu výstupnej cesty z AS
 - Atribút LOCAL_PREF sa odovzdáva len cez iBGP
 - Neprechádza cez eBGP
 - Je uzatvorený vo vnútri AS (local)
 - Čím vyššia hodnota, tým lepšie
 - Implicitná hodnota je 100



■ Príklad

- Dve cesty z AS 64520 do siete 172.16.0.0 v AS 65350
- Ktorú vybrať?

Zmena štandardnej LOCAL_PREF hodnoty

Router(config-router) #

```
bgp default local-preference value
```

- Zmení hodnotu LOCAL_PREF zo štandardnej hodnoty 100 na definovanú hodnotu
- Všetky smery ohlásené iBGP susedom budú mať nastavenú danú hodnotu LOCAL_PREF

Well-Known Discretionary ATOMIC_AGGREGATE

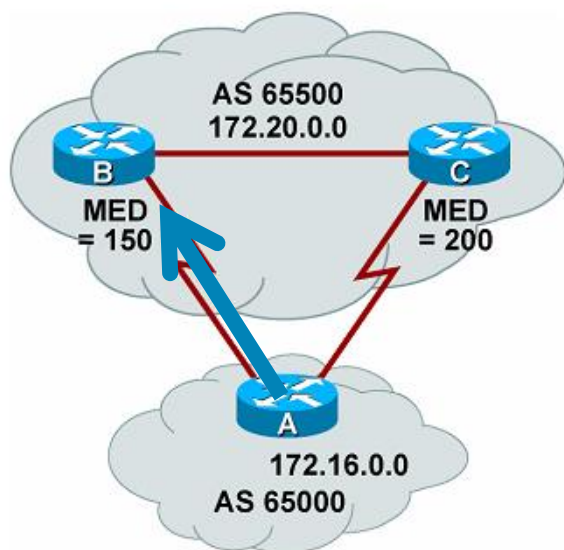
- Atribút Atomic Aggregate indikuje, že cesta bola sumarizovaná
 - Atribút varuje, že získaná informácia nemusí byť úplná
- Atribút je nastavený buď na
 - True or False
 - True = Výstraha že viaceré siete boli zlúčené do jedného update
 - Update zahŕňa router ID a AS číslo spolu s číslom supernet siete
 - Za účelom zjednodušenia identifikácie smerovača zodpovedného za agregáciu

Optional Transitive atribút Community

- Atribút Community sa používa na filtrovanie vstupujúcich a vystupujúcich ciest
 - Pomocou tohto atribútu môžu BGP smerovače značkovať (tagovať) cesty značkou (indicator/tag)
 - Na základe značky je možné vykonávať filtrovanie alebo akcie manipulácie s update
- Smerovač, ktorý rozumie atribútu ho musí propagovať ďalej
 - Ináč drop atribútu

Optional non-Transitive atribút MED

- MED je indikácia pre susedný (externý) AS, ktorú z viacerých možných ciest **do nášho** AS má použiť
 - MED sa iniciálne prenesie z nášho AS cez eBGP do susedného AS a v ňom sa rozšíri, avšak z neho už nevychádza ďalej
 - z neho sa ďalej preposiela s hodnotou 0
 - MED sa zvykne nazývať aj „metrika“ a v tomto zmysle platí
 - **čím menšia, tým lepšie**
 - Štandardne sa MED porovnáva medzi rôznymi cestami do toho istého cieľa len vtedy, ak tieto cesty prišli z rovnakého susedného AS
 - Prepisuje príkaz `always-compare-med`



▪ Príklad

- Smerovače B a C distribuujú MED atribút za účelom informovania smerovača A o preferencii vstupu do AS 65500
- A si vyberie vstup cez B
 - Nižšia hodnota MED

Zmena štandardnej hodnoty MED

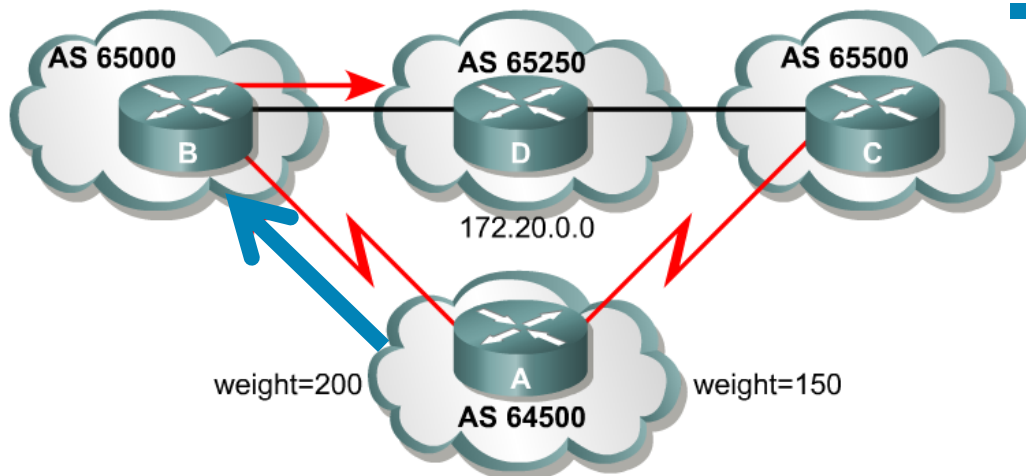
Router (config-router) #

default-metric *number*

- Zmení hodnotu MED zo štandardnej hodnoty 0 na definovanú hodnotu
- Všetky smery ohlásené eBGP susedom budú mať nastavenú hodnotu MED (metriku) na definovanú hodnotu

Cisco BGP atribút WEIGHT

- Je to Cisco proprietárny atribút
- Využiteľný ak daný smerovač má viaceré výstupné linky
- Atribút WEIGHT vyjadruje „mikrolokálnu“ preferenciu smeru
 - Atribút lokálny len pre daný router, nepreposiela sa ďalej žiadnemu susedovi
 - Čím vyššia hodnota, tým viac preferovaný smer
 - Možné hodnoty od $<0,65535>$
 - Smery, ktoré do BGP vnášame my, majú default WEIGHT 32768
 - Smery, ktoré sme sa cez BGP naučili, majú default WEIGHT 0



▪ Príklad

- Smerovač A má dve cesty na dosiahnutie siete 172.20.0.0 v AS 65250
 - Cez B a cez C
- A na vstupujúce updates aplikuje lokálnu váhu
- A si vyberie výstup cez B
 - Vyššia hodnota WEIGHT

Zmena prednastavenej váhy

- Zmeň default váhu pre všetky cesty od daného suseda

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

```
neighbor {ip-address | peer-group-name} weight number
```

- *number*: pridelené váha <0, 65535>
- **Poznámka:**
 - Váhy pridelené v route map cez **set weight route-map** prepisujú váhu danú príkazom **neighbor weight**.

Poradie vyhodnocovania atribútov

Uvažujú sa len (synchronizované) smery bez AS slučiek a s dosiahnuteľným next hop routerom:

1. Cesta s najvyšším atribútom WEIGHT (lokálny pre router)
2. Cesta s najvyšším atribútom LOCAL_PREF (globálny v rámci AS)
3. Cesta, ktorú sme do BGP my sami vniesli (locally originated)
4. Cesta s najmenším počtom AS v zozname AS_PATH
5. Cesta s najnižším kódom pôvodu (IGP < EGP < incomplete)
6. Cesta s najnižším MED
7. Preferujú sa cesty naučené cez eBGP voči cestám naučeným cez iBGP
8. Preferuje sa cesta s najnižšou IGP metrikou k BGP next-hopu
9. Pre eBGP sa preferujú najstaršie (prvé naučené) cesty
10. Preferujú sa cesty od peera s najnižším BGP router ID
11. Preferujú sa cesty od peera s najnižšou IP adresou

Z pohľadu cert skúšky CCNP ROUTE treba vedieť

BGP Peer Groups, Router Reflectors



Používanie Peer Group

Router(config-router) #

```
neighbor PEER-GROUP-NAME peer-group
```

- Týmto príkazom sa vytvorí peer group

Router(config-router) #

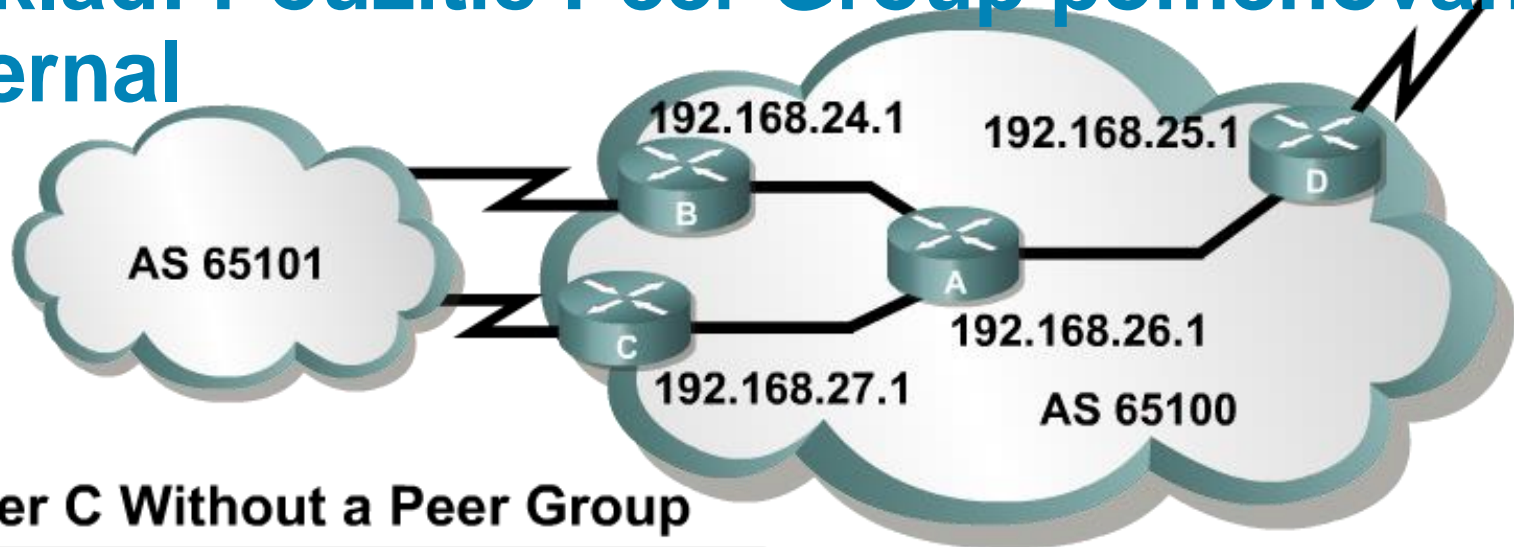
```
neighbor IP-ADDRESS peer-group PEER-GROUP-NAME
```

- Tento príkaz zaradí suseda do vybranej peer group
- Peer groups je možné s výhodou použiť, ak máme skupinu susedov, ktorí majú spoločnú „outbound policy“ – politiku, ktorá filtruje prenos informácie od nás k nim
- Členovia peer group môžu mať rozličné „inbound policies“ – vstupné politiky, ktoré filtrujú prenos informácie od nich k nám

Používanie Peer Group

- Peer Groups sú veľmi výhodné, pretože
 - Aktualizácie sú vygenerované pre celú grupu iba raz
 - Zjednodušuje sa konfigurácia
 - Všetky filtre, route-mapy a podobné konštrukcie sa aplikujú na grupu, netreba na jednotlivých členov
 - Šetrí sa procesorový čas a pamäť, pretože smerovacia tabuľka sa pre peer group kontroluje len raz, takisto aktualizácie sa generujú len raz a replikujú sa
- Peer Group zjednodušujú konfiguráciu, avšak stále je zachovaná požiadavka na full-mesh peerov

Príklad: Použitie Peer Group pomenovanej internal



Router C Without a Peer Group

```
router bgp 65100
neighbor 192.168.24.1 remote-as 65100
neighbor 192.168.24.1 update-source Loopback 0
neighbor 192.168.24.1 next-hop-self
neighbor 192.168.24.1 distribute-list 20 out
neighbor 192.168.25.1 remote-as 65100
neighbor 192.168.25.1 update-source Loopback 0
neighbor 192.168.25.1 next-hop-self
neighbor 192.168.25.1 distribute-list 20 out
neighbor 192.168.26.1 remote-as 65100
neighbor 192.168.26.1 update-source Loopback 0
neighbor 192.168.26.1 next-hop-self
neighbor 192.168.26.1 distribute-list 20 out
```

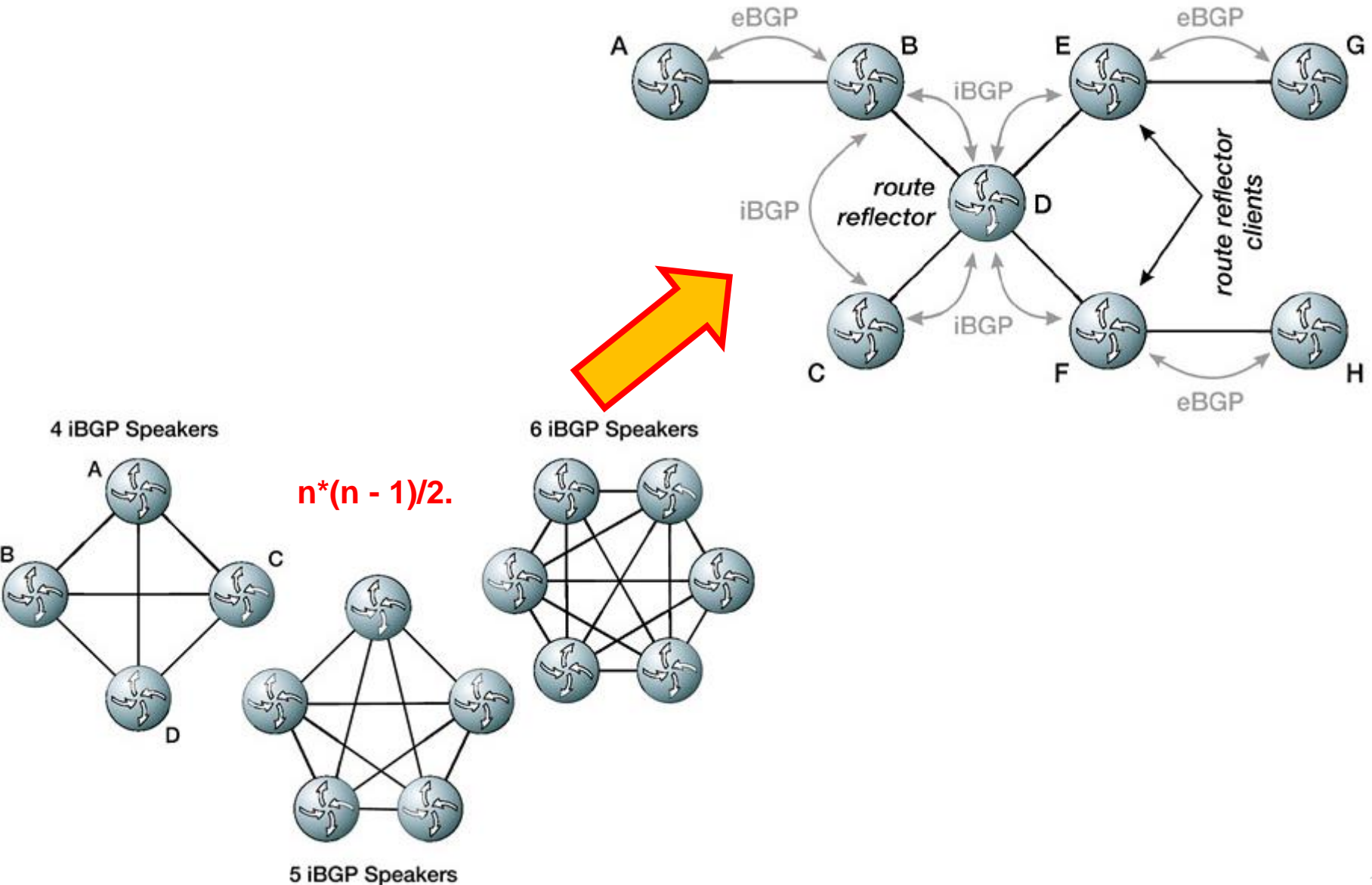
Router C Using a Peer Group

```
router bgp 65100
neighbor internal peer-group
neighbor internal remote-as 65100
neighbor internal update-source Loopback 0
neighbor internal next-hop-self
neighbor internal distribute-list 20 out
neighbor 192.168.24.1 peer-group internal
neighbor 192.168.25.1 peer-group internal
neighbor 192.168.26.1 peer-group internal
```

```
access-list 20 deny 10.0.0.0 0.255.255.255
access-list 20 deny 172.16.0.0 0.31.255.255
access-list 20 deny 192.168.0.0 0.0.255.255
access-list 20 permit any
```

Distribučný list sa používa ako v IGP, ale filtruje len updates na susedov

iBGP peering – problém



Route reflector

- Route reflector (RR) je BGP router, ktorý obchádza pravidlo, že cez iBGP sa nesmie odovzdať informácia, ktorá bola práve cez iBGP naučená
- Pomocou RR je možné zásadne zjednodušiť konfiguráciu BGP speakerov v AS
 - Z full meshed logickej topológie vznikne logická hub-and-spoke
 - RR si nakonfiguruje všetky ostatné routery ako svojich susedov a pridá si ku každému z týchto susedov riadok

`Router (config-router) #`

```
neighbor {ip-address | pg-name} route-reflector-client
```

- Všetky ostatné BGP routery si nakonfigurujú RR router ako svojho suseda úplne bežným spôsobom



Manipulácia s výberom cesty v BGP s použitím route maps



Route mapy v BGP

- Route mapy môžu byť v BGP použité na pridelenie alebo modifikáciu BGP atribútov cesty
- Implementujú sa príkazom

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

```
neighbor {ip-address | peer-group-name} route-map map-name  
{in | out}
```

- Postup
 - Definuj a pomenuj route map (ako popísané v kapitole 4)
 - Definuj `match` kritéria
 - Definuj `set` zmeny
 - Urči, ktorý atribút a na ktorého suseda sa bude meniť
 - Over výsledky
 - Flushni BGP tabuľku (hard/soft)

match příkazy použitelné v BGP

Command	Description
<code>match as-path</code>	Matches the AS_PATH attribute
<code>match ip address</code>	Matches any routes that have a destination network number address that is permitted by a standard or extended ACL
<code>match metric</code>	Matches routes with the metric specified
<code>match community</code>	Matches a BGP community
<code>match interface</code>	Matches any routes that have the next hop out of one of the interfaces specified
<code>match ip next-hop</code>	Matches any routes that have a next-hop router address that is passed by one of the ACLs specified
<code>match ip route-source</code>	Matches routes that have been advertised by routers and access servers at the address that is specified by the ACLs
<code>match route-type</code>	Matches routes of the specified type
<code>match tag</code>	Matches tag of a route

** Partial list*

Príkaz `match as-path`

- Kontrola voči BGP AS ACL listu

Router (config-route-map) #

```
match as-path path-list-number
```

- *path-list-number* identifikuje AS ACL <1, 199>

Konfigurácia Autonomous System ACL

- Konfigurácia AS-path filtra

Router (config-router) #

```
ip as-path access-list acl-number {permit | deny} regex
```

- Konfigurujú sa podobne ako IP ACL
- *acl-number*: hodnota v rozsahu <1, 500> špecifikujúca AS-path ACL
- *regex* regulárny výraz ktorý definuje podmienku AS-path filtra

Syntax regulárnych výrazov

- **Atom:** Jeden znak.

- . Vyhovuje ľubovoľný jeden znak
- ^ musí obsahovať začiatok reťazca.
- \$ musí obsahovať koniec reťazca
- \ musí odpovedať znak.

- **Quantifiers (kvantifikátory):**

- * vyhovuje výraz so **žiadnym** alebo **viac** výskytmi (max neobmedzené)
- + vyhovuje výraz aspoň s **jedným** alebo **viac** výskytmi (max neobmedzené)
- ? daný znak (atom) so žiadnym alebo max 1 výskytom
- {n} práve n-krát, {m,n} max. m-krát a min. n-krát

- **Range:** Sekvencia znakov v hranatých zátvorkách.

- Príklad [abcd], [1-5], [a-e]

Príklady regulárnych výrazov

Regular Expression	Resulting Expression
a*	Expression indicates any occurrence of the letter "a", which includes none
a+	indicates that at least one occurrence of the letter "a" must be present
ab?a	Expression matches "aa" or "aba".
100	Expression means via AS100.
_100\$	Expression indicates an origin of AS100.
^100.*	Expression indicates transmission from AS100
^\$	Expression indicates origination from this AS

Príkazy set použiteľné v BGP

Command	Description
<code>set weight <i>NUMBER</i></code>	Sets the BGP weight value. <0,65535>
<code>set local-preference <i>NUMBER</i></code>	Sets the LOCAL-PREF attribute value. Integer from 0 to 4294967295. Default 100.
<code>set as-path</code>	Modifies an AS path for BGP routes
<code>set origin</code>	Sets the ORIGIN attribute value
<code>set metric <i>VALUE</i></code>	Sets the Multi-Exit_Disc (MED) value. Integer from 0 to 294967295
<code>set community</code>	Sets the BGP communities attribute
<code>set automatic-tag</code>	Computes automatically the tag value
<code>set ip next-hop</code>	Indicates which IP address to output packets
<code>set interface</code>	Indicates which interface to output packets
<code>set ip default next-hop</code>	Indicates which default IP address to use to output packets
<code>set default interface</code>	Indicates which default interface to use to output packets <i>* Partial list</i>

Príkaz set as-path

- Modifikuje AS cestu pre BGP routes.

Router (config-route-map) #

```
set as-path {tag | prepend as-path-string}
```

Parameter	Description
tag	Converts the tag of a route into an autonomous system path. Applies only when redistributing routes into BGP.
prepend	Pripoj číslo ktoré nasleduje za prepend k atribútu AS_PATH pre route, ktorá sa zhoduje s route map. Aplikovateľné na inbound a outbound BGP route maps.
<i>as-path-string</i>	AČíslo AS pridané do atribútu AS_PATH. Rozsah je od 1 do 65535. Max 10 AS čísel môže byť pripnutých.

Route mapy v BGP

- NEXT_HOP

- match ip next-hop {1-99 | 1300-1999 | MENO | prefix-list MENO}
- set ip next-hop A.B.C.D

- ORIGIN

- match route-type local
- set origin {igp | egp | incomplete}

- AS_PATH

- match as-path 1-500 (číslo označuje tzv. AS path list)
- set as-path prepend N N N ...

Route mapy v BGP

- LOCAL_PREF:
 - match local-preference N
 - set local-preference N
- MULTI_EXIT_DISC:
 - match metric N
 - set metric N
- WEIGHT:
 - match neexistuje (atribút sa neposiela)
 - set weight N

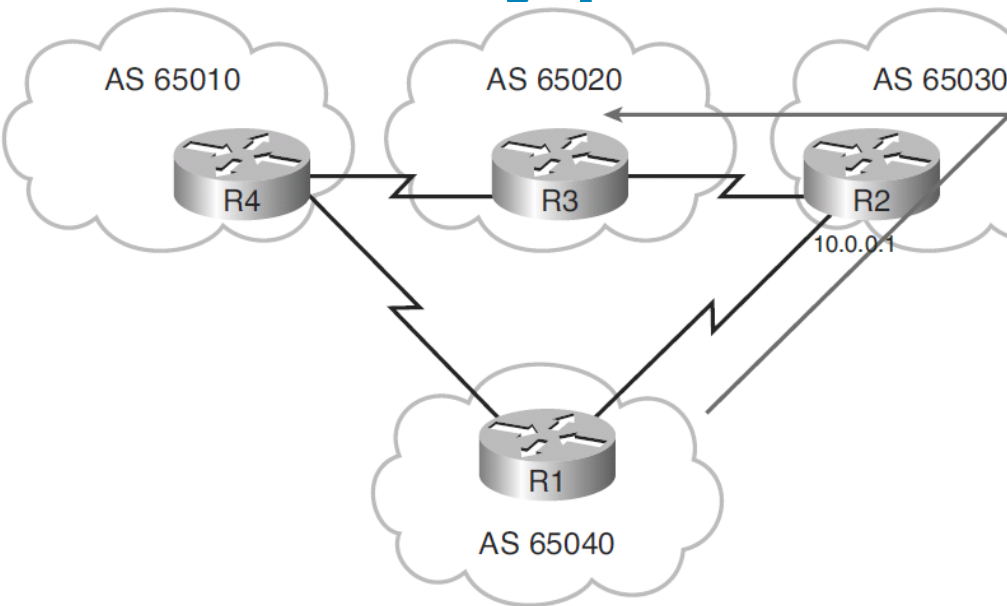
Zmena atribútu Weight (Váha)

- Použitie ak je smerovač s viac rozhraniami (multihomed).
 - Váha ovplyvní len lokálny smerovač
 - Cesty s vyššou váhou sú preferované
- Existujú dve možnosti ako ovplyvniť váhu:
 - Zmena váhy pre všetky cesty od suseda
 - Aplikuj ako súčasť neighbor príkazu (**weight**)
 - Zmena váhy pre špecifickú sieť/cestu – route map

BGP Route Selection Process

1. Prefer highest Weight
2. Prefer highest LOCAL_PREF
3. Prefer locally generated routes
4. Prefer shortest AS_PATH
5. Prefer lowest ORIGIN (IGP < EGP < incomplete)
6. Prefer lowest MED
7. Prefer EBGP over IBGP
8. Prefer routes through closest IGP neighbor
9. Prefer routes with lowest BGP router ID
10. Prefer routes with lowest neighbor IP address

Zmena váhy pomocou Route Map



Príklad:

- Smerovacie politiky pre AS 65040 vyžadujú, aby sa pre siete z AS 65020 použila cesta cez AS 65030

Ako to dosiahneme?

- AS path ACL
- Spravíme route map pre AS65020
- Zvýšime váhu pre updates od suseda z 65030 (10.0.0.1), ktoré nesú info o sieťach z AS 65020

```
R1 (config) # route-map SET-WEIGHT permit 10
R1 (config-route-map) # match as-path 10
R1 (config-route-map) # set weight 150
R1 (config-route-map) #
R1 (config-route-map) # route-map SET-WEIGHT permit 20
R1 (config-route-map) # set weight 0
R1 (config-route-map) # exit
R1 (config) # ip as-path access-list 10 permit _65020$
R1 (config) #
R1 (config) # router bgp 65040
R1 (config-router) # neighbor 10.0.0.1 remote-as 65030
R1 (config-router) # neighbor 10.0.0.1 route-map SET-WEIGHT in
```

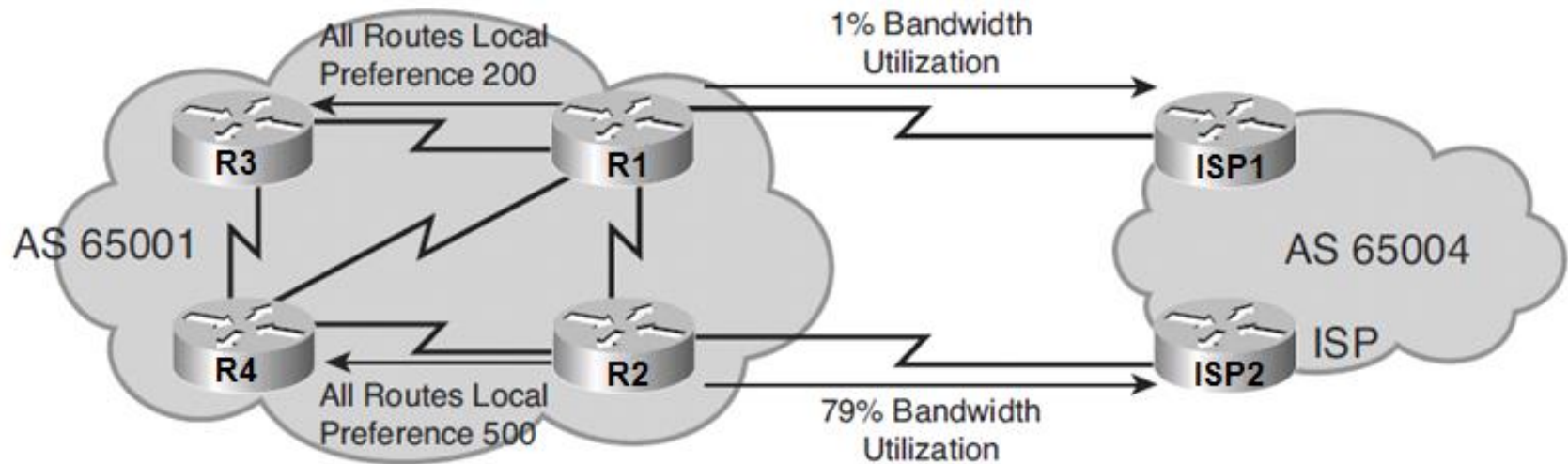
Zmena Local Preference

- Použité len v AS (medzi iBGP speakrami)
 - Špecifikuje naj cestu ako opustiť AS
 - Vyššia hodnota je preferovaná
 - Default local preference je 100.
- Dve možnosti modifikácie:
 - Pre všetky cesty odosielané smerovačom
 - `bgp default local-preference value`
 - Aplikácia na špecifické cesty s route mapami

BGP Route Selection Process

1. Prefer highest Weight
2. Prefer highest LOCAL_PREF
3. Prefer locally generated routes
4. Prefer shortest AS_PATH
5. Prefer lowest ORIGIN (IGP < EGP < incomplete)
6. Prefer lowest MED
7. Prefer EBGP over IBGP
8. Prefer routes through closest IGP neighbor
9. Prefer routes with lowest BGP router ID
10. Prefer routes with lowest neighbor IP address

Príklad local preference – všetky cesty

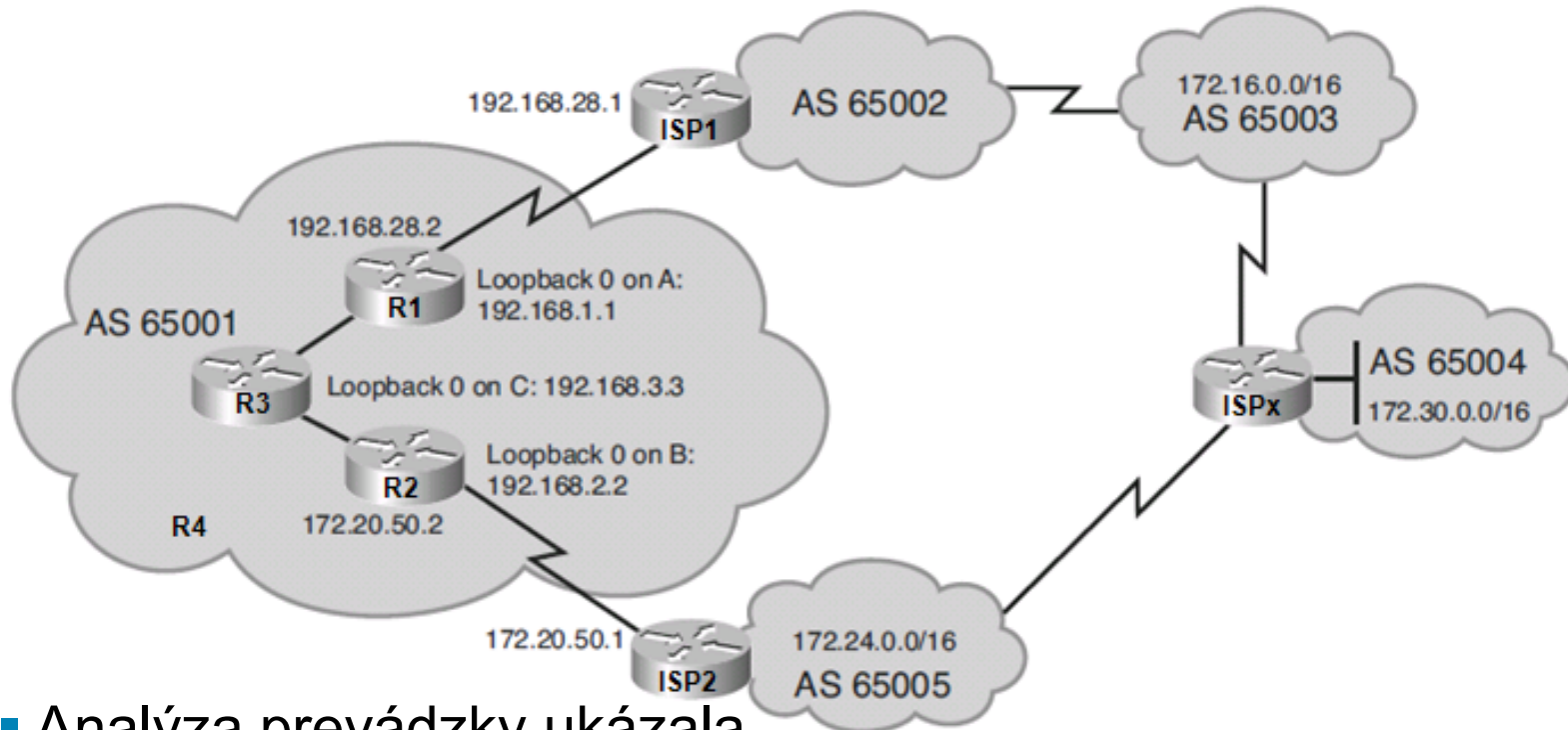


- BGP smerovacie politiky vyžadujú použitie R2 na výstup z AS

```
R1(config)# router bgp 65001
R1(config-router)# bgp default local-preference 200
R1(config-router)#
```

```
R2(config)# router bgp 65001
R2(config-router)# bgp default local-preference 500
R2(config-router)#
```

Príklad Local Preference a Route Map



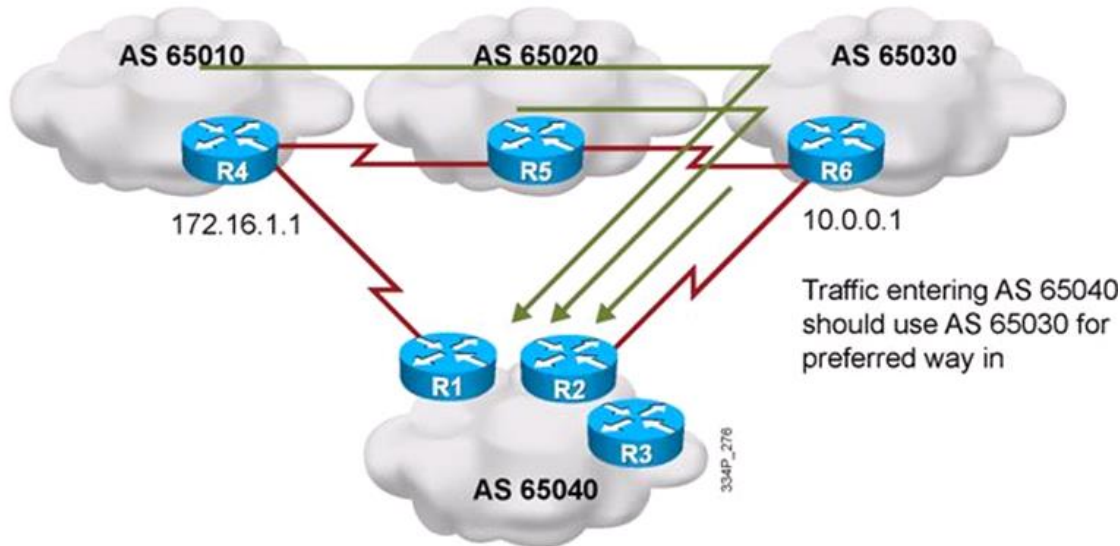
- Analýza prevádzky ukázala
 - 10% prevádzky ide cez R1 na ISP1 do siete 172.16.0.0
 - 50% prevádzky ide cez R2 na ISP2 do sietí 172.24.0.0 a 172.30.0.0.
 - Zvyšných 40 percent ide do iných cieľov
- Bola zadefinovaná smerovacia politika
 - Presmerovať prevádzku do siete 172.30.0.0 cez R1

Použitie LOCAL_PREF: Route Map na smerovači R1

```
access-list 65 permit 172.30.0.0 0.0.255.255
!
route-map LOKALNA_PREF permit 10
  match ip address 65
  set local-preference 400
!
route-map LOKALNA_PREF permit 20
!
router bgp 65001
neighbor 192.168.2.2 remote-as 65001
neighbor 192.168.2.2 update-source loopback0
neighbor 192.168.3.3 remote-as 65001
neighbor 192.168.3.3 update-source loopback0
neighbor 192.168.28.1 remote-as 65002
neighbor 192.168.28.1 route-map LOKALNA_PREF in
```

```
R3# show ip bgp
BGP table version is 7, local router ID is 192.168.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal, r
RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop          Metric LocPrf Weight Path
* i172.16.0.0        172.20.50.1          100         0 65005 65004 65003 i
*>i                  192.168.28.1          100         0 65002 65003 i
*>i172.24.0.0        172.20.50.1          100         0 65005 i
* i                  192.168.28.1          100         0 65002 65003 65004 65005 i
* i172.30.0.0        172.20.50.1          100         0 65005 65004 i
*>i                  192.168.28.1          400         0 65002 65003 65004i
```

Zmena AS Path



BGP politika požaduje aby prevádzka do AS 65040 vstupovala cez R6 v AS 65030

- A nie cez R4 v AS 65010.

Jeden zo spôsobov je zhoršiť AS path atribút (zvýšiť počet AS) a tým znevýhodniť cestu

```
R1(config)# route-map SET-AS-PATH permit 10
R1(config-route-map)# set as-path prepend 65040 65040 65040
R1(config-route-map)# exit
R1(config)# router bgp 65040
R1(config-router)# neighbor 172.16.1.1 remote-as 65010
R1(config-router)# neighbor 172.16.1.1 route-map SET-AS-PATH out
R1(config-router)# exit
R1(config)#
```

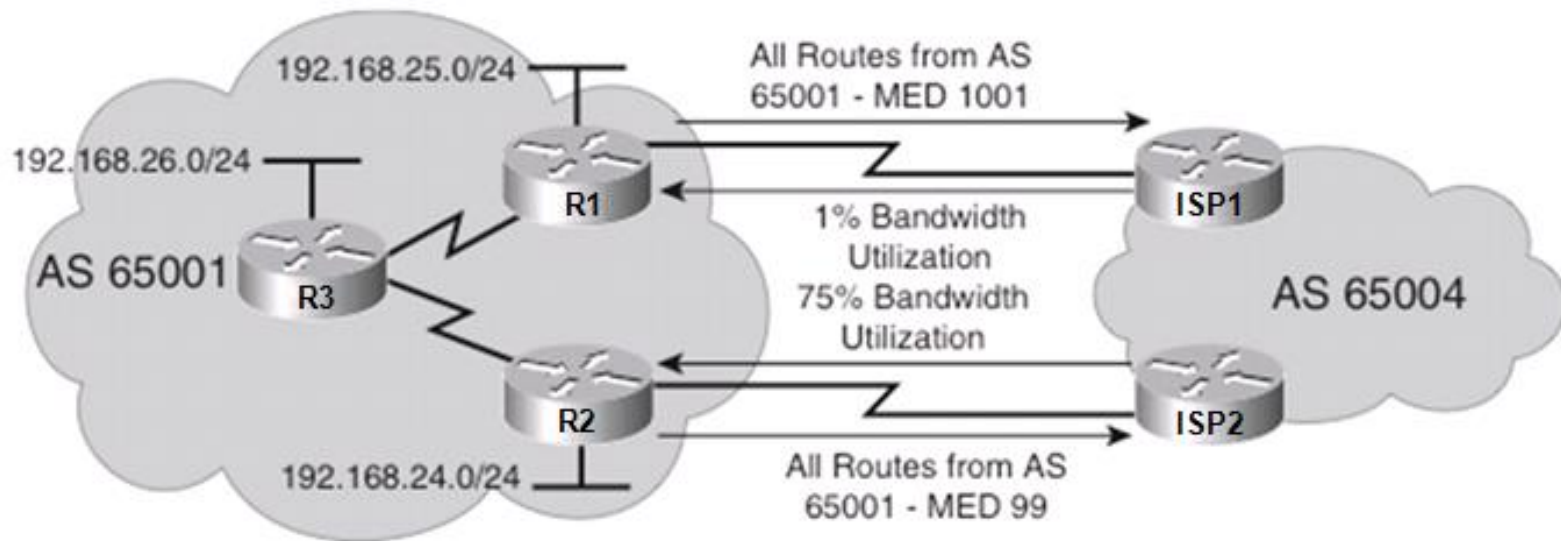

Zmena MED

- MED sa používa na ovplyvnenie vstupu do AS
 - Preferovaná je nižšia hodnota MED
 - Default je 0
- Pozor:
 - MED je vyhodnocovaný BGP až ako 6 parameter
- Sú dve možnosti
 - Pre všetky cesty **default-metric**
 - Na špecifické cesty s route map

BGP Route Selection Process

1. Prefer highest Weight
2. Prefer highest LOCAL_PREF
3. Prefer locally generated routes
4. Prefer shortest AS_PATH
5. Prefer lowest ORIGIN (IGP < EGP < incomplete)
6. **Prefer lowest MED**
7. Prefer EBGP over IBGP
8. Prefer routes through closest IGP neighbor
9. Prefer routes with lowest BGP router ID
10. Prefer routes with lowest neighbor IP address

Zmena MED – všetky cesty



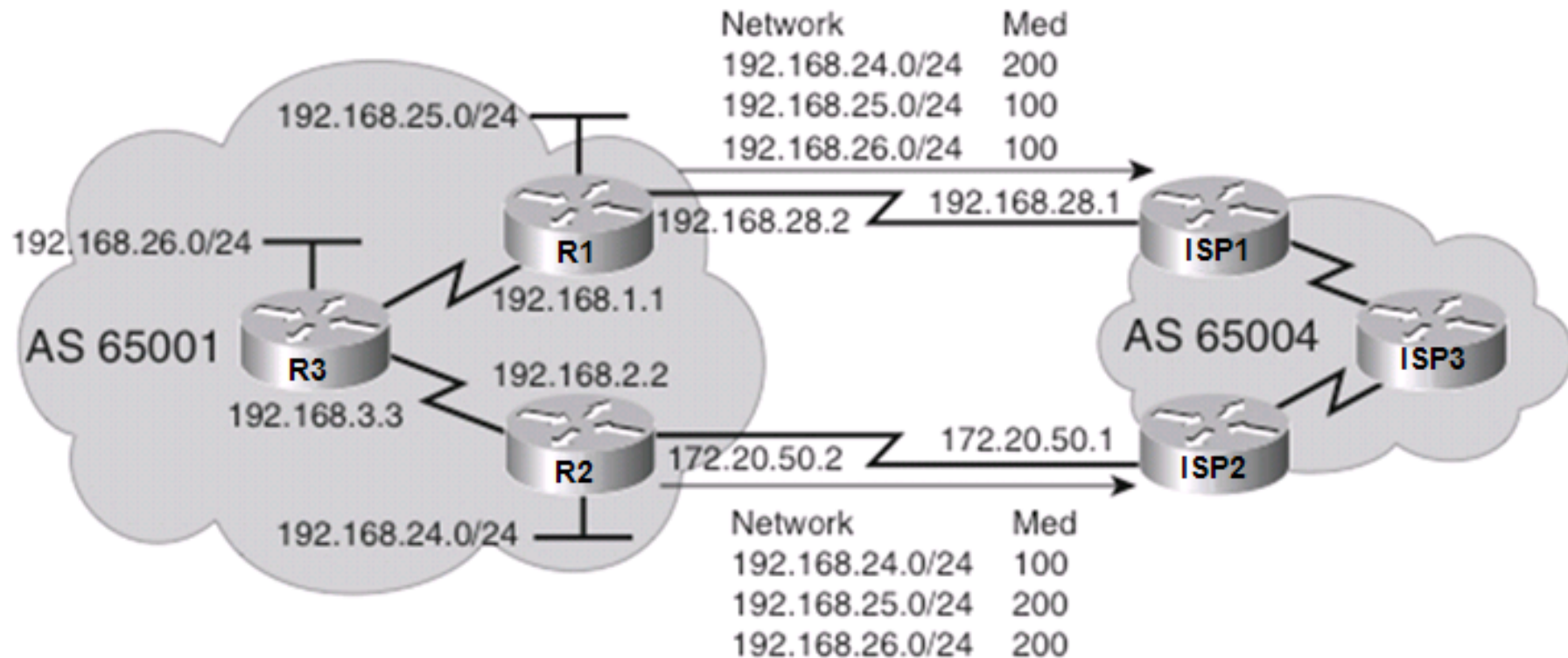
```
R1(config)# router bgp 65001
R1(config-router)# default metric 1001
R1(config-router)#
```

```
R2(config)# router bgp 65001
R2(config-router)# default metric 99
R2(config-router)#
```

■ Výsledok

- Preferovane sa použije len R2 na vstup

Zmena MED s Route Mapou



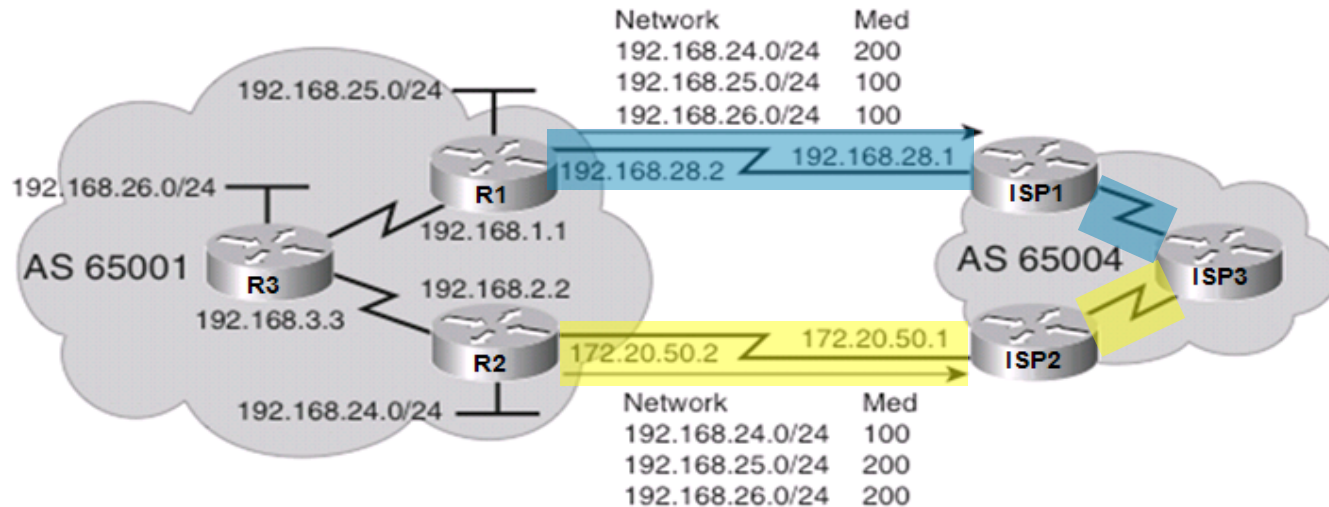
Konfigurácia smerovača R1

```
R1(config)# access-list 66 permit 192.168.25.0 0.0.0.255
R1(config)# access-list 66 permit 192.168.26.0 0.0.0.255
R1(config)#
R1(config)# route-map MED-65004 permit 10
R1(config-route-map)# match ip address 66
R1(config-route-map)# set metric 100
R1(config-route-map)#
R1(config-route-map)# route-map MED-65004 permit 100
R1(config-route-map)# set metric 200
R1(config-route-map)# exit
R1(config)# router bgp 65001
R1(config-router)# neighbor 192.168.2.2 remote-as 65001
R1(config-router)# neighbor 192.168.2.2 update-source loopback0
R1(config-router)# neighbor 192.168.3.3 remote-as 65001
R1(config-router)# neighbor 192.168.3.3 update-source loopback0
R1(config-router)# neighbor 192.168.28.1 remote-as 65004
R1(config-router)# neighbor 192.168.28.1 route-map MED-65004 out
R1(config-router)#exit
```

Konfigurácia smerovača R2

```
R2 (config)# access-list 66 permit 192.168.24.0 0.0.0.255  
R2 (config)#  
R2 (config)# route-map MED-65004 permit 10  
R2 (config-route-map)# match ip address 66  
R2 (config-route-map)# set metric 100  
R2 (config-route-map)#  
R2 (config-route-map)# route-map MED-65004 permit 100  
R2 (config-route-map)# set metric 200  
R2 (config-route-map)# exit  
R2 (config)# router bgp 65001  
R2 (config-router)# neighbor 192.168.1.1 remote-as 65001  
R2 (config-router)# neighbor 192.168.1.1 update-source loopback0  
R2 (config-router)# neighbor 192.168.3.3 remote-as 65001  
R2 (config-router)# neighbor 192.168.3.3 update-source loopback0  
R2 (config-router)# neighbor 172.20.50.1 remote-as 65004  
R2 (config-router)# neighbor 172.20.50.1 route-map MED-65004 out  
R2 (config-router)# exit
```

Ako to vidí ISP smerovač



```
ISP3# show ip bgp
```

```
BGP table version is 7, local router ID is 192.168.1.1
```

```
Status codes: s suppressed, d damped, h history, * valid, > best, i -  
internal, r RIB-failure, S Stale
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Next Hop	Metric	LocPrf	Weight	Path
*>i192.168.24.0	172.20.50.2	100	100	0	65001 i
* i	192.168.28.2	200	100	0	65001 i
* i192.168.25.0	172.20.50.2	200	100	0	65001 i
*>i	192.168.28.2	100	100	0	65001 i
* i192.168.26.0	172.20.50.2	200	100	0	65001 i
*>i	192.168.28.2	100	100	0	65001 i

Filtering BGP Routing Updates

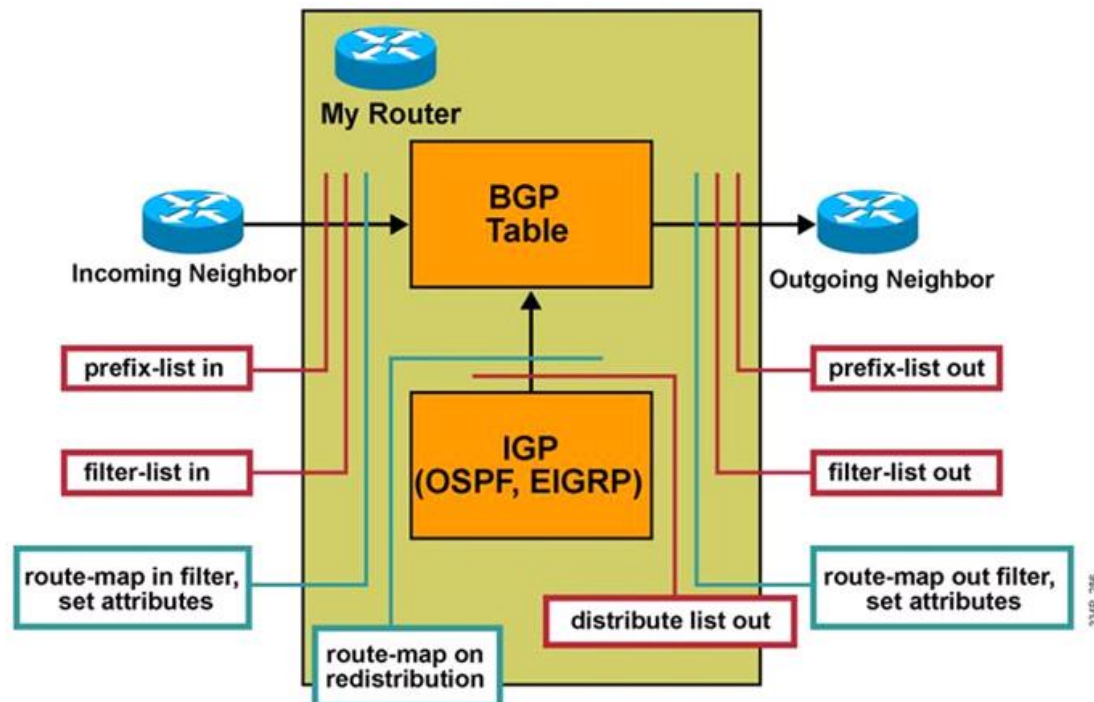


Filtrovanie BGP smerovacích updates

- BGP môže dostávať veľké množstvo smerovacích informácií
 - Je vhodné optimalizovať jeho činnosť nasadením filtrov
- Dostupné techniky:
 - Filter lists
 - Prefix lists
 - Route maps

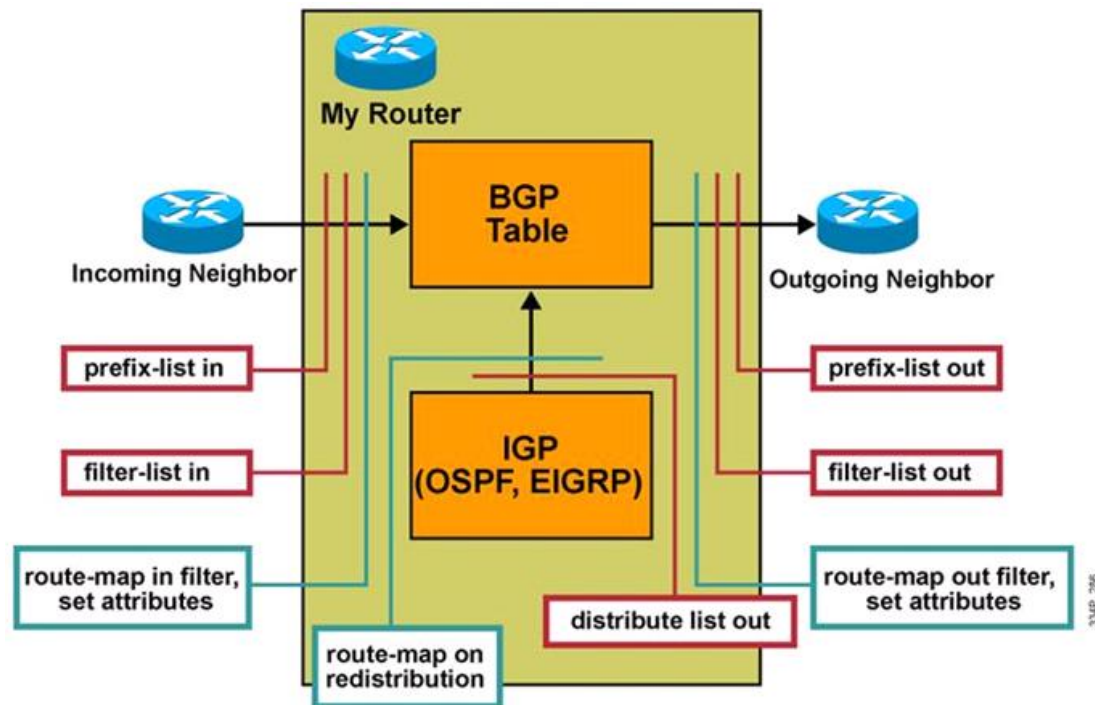
Filtrovanie vstupných a výstupných updates

- Update pre vstupujúce cesty (prefixy) môže byť upravený pred vložením do BGP tabuľky
 - prechodom cez prefix lists, filter-lists, a route mapy v smere IN
- Podobne pre vystupujúce smery skôr ako je updates odoslaný
 - prechodom cez route-mapy v smere out (outgoing), filter listy, a prefix listy



Filtrovanie pri redistribúcii

- Aplikáciou prefix listov, route máp na proces redistribúcie



Aplikácia BGP filtra (filter-list)

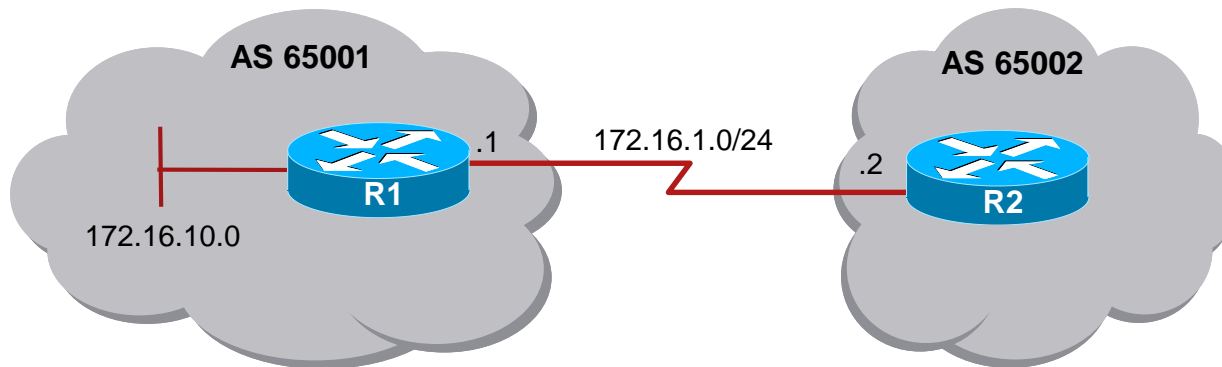
- Aplikuj filter list na cesty z alebo na suseda

Router (config-router) #

```
neighbor {ip-address | peer-group-name} filter-list  
access-list-number {in | out}
```

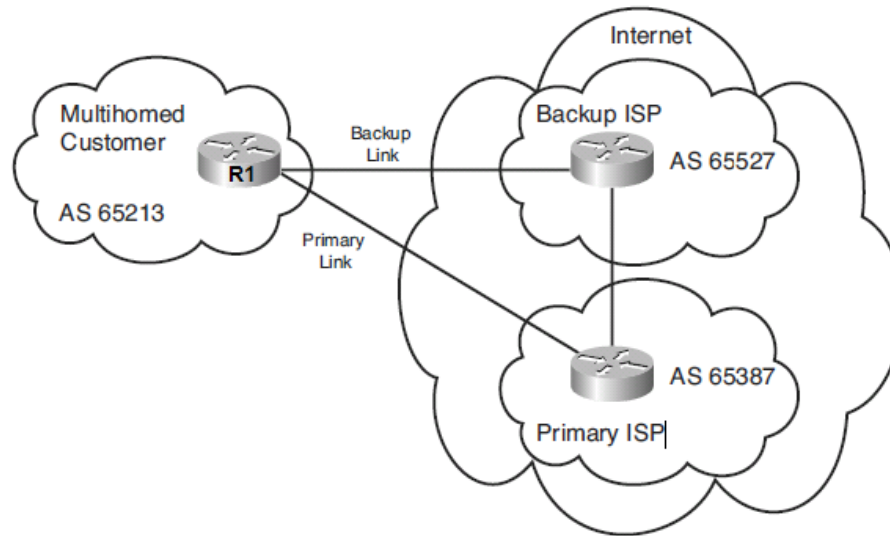
Parameter	Popis
<i>ip-address</i>	IP addressa BGP suseda
<i>peer-group-name</i>	Meno BGP peer groupy
<i>access-list-number</i>	Číslo AS-path access listu.
in	Aplikácia na incoming routes.
out	Aplikácia na outgoing routes.

Príklad BGP Filtra s využitím Prefix Lists



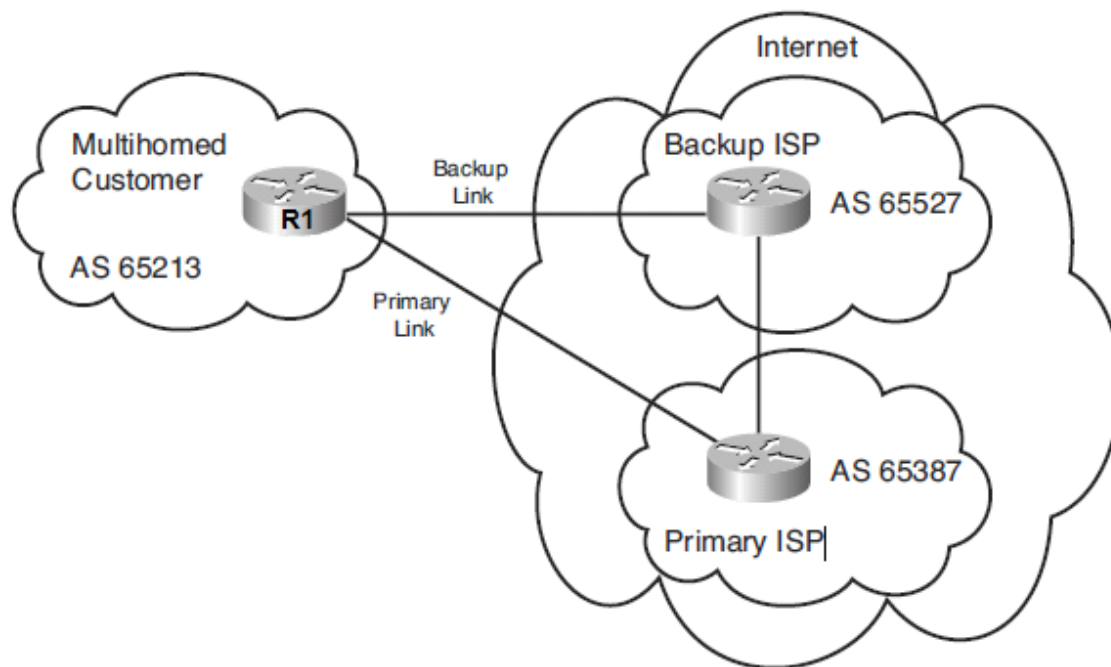
```
R1(config)# ip prefix-list ANY-8to24-NET permit 0.0.0.0/0 ge 8 le 24
R1(config)# router bgp 65001
R1(config-router)# neighbor 172.16.1.2 remote-as 65002
R1(config-router)# neighbor 172.16.1.2 prefix-list ANY-8to24-NET in
R1(config-router)# end
R1#
R1# show ip prefix-list detail ANY-8to24-NET
ip prefix-list ANY-8to24-NET:
Description: test-list
count: 1, range entries: 1, sequences: 10 - 10, refcount: 3
seq 10 permit 0.0.0.0/0 ge 8 le 24 (hit count: 0, refcount: 1)
```

Príklad BGP Filtra s využitím Route Maps



```
R1(config)# ip as-path access-list 10 permit _65387$
R1(config)# ip prefix-list DEF-ONLY seq 10 permit 0.0.0.0/0
R1(config)#
R1(config)# route-map FILTER permit 10
R1(config-route-map)# match ip address prefix-list DEF-ONLY
R1(config-route-map)# match as-path 10
R1(config-route-map)# set weight 150
R1(config-route-map)#
R1(config-route-map)# route-map FILTER permit 20
R1(config-route-map)# match ip address prefix-list DEF-ONLY
R1(config-route-map)# set weight 100
R1(config-route-map)# exit
```

Príklad BGP Filtra s využitím Route Maps



```
R1(config)# router bgp 65213
R1(config-router)# neighbor 10.2.3.4 remote-as 65527
R1(config-router)# neighbor 10.2.3.4 route-map FILTER in
R1(config-router)# neighbor 10.4.5.6 remote-as 65387
R1(config-router)# neighbor 10.4.5.6 route-map FILTER in
R1(config-router)#
```

