1.3 Směrování v počítačových sítích a v Internetu.

Abychom mohli paketovou sítí směrovat pakety od zdroje k cíli, potřebujeme správným způsobem naplnit směrovací tabulky všech směrovačů na trase. V malých sítích nebo v sítích, z nichž veškerý provoz ven odchází po jediné implicitní (default) cestě, lze toto vyřešit manuálním vložením potřebných informací, tj. statickým směrováním. V rozlehlejších sítích s měnící se topologií (z nichž největší je bezesporu Internet) jsou však nutné dynamické směrovací protokoly, které zajistí správné naplnění směrovacích tabulek automaticky na základě výměny informací mezi směrovači.

1.3.1 Hierarchické směrování, autonomní systémy

Současný Internet je natolik rozsáhlý a proměnlivý, že není reálné udržovat ve směrovačích úplnou informaci o jeho topologii. Tato informace by navíc byla velmi nestabilní, protože by se měnila s výpadkem nebo zapojením linky kdekoli na světě. Proto bylo rozhodnuto směrování v Internetu řešit hierarchickým způsobem. Předpokladem jeho použití je rozdělení Internetu do tzv. autonomních systémů (AS). Autonomním systémem rozumíme souvislou skupinu sítí a směrovačů, které jsou pod společnou správou a řídí se společnou směrovací politikou. Pod společnou směrovací politikou si představme zejména dohodnutý vnitřní směrovací protokol (např. OSPF nebo RIP), ale také speciální požadavky administrátorů na směrování některých druhů provozu (traffic engineering, load balancing). Příkladem autonomního systému tak může být autonomní systém jednoho konkrétního poskytovatele Internetu (ISP) nebo velké firmy.

Princip hierarchického směrování spočívá v tom, že z pohledu směrování mezi AS jsou autonomní systémy chápány jako základní jednotky, jejichž struktura již není mimo hranice autonomního systému známa. Z každého autonomního systému se pouze do okolí sděluje, které adresy sítí autonomní systém obsahuje. Autonomní systémy jsou číslovány celosvětově jednoznačnými šestnáctibitovými čísly.

Cílem hierarchického směrování je vždy nejprve doručit paket určený pro některou ze sítí autonomního systému na hranice tohoto autonomního systému. O další směrování ke konkrétní síti uvnitř AS se již postará vnitřní směrovací protokol, který topologii (nebo alepoň cesty ke všem sítím) svého vlastního AS zná. Směrovač, který je na hranici autonomního systému a účastní se jak na směrování mezi AS tak ve směrovacím protokolu svého AS, se nazývá hraniční směrovač (angl. border gateway).

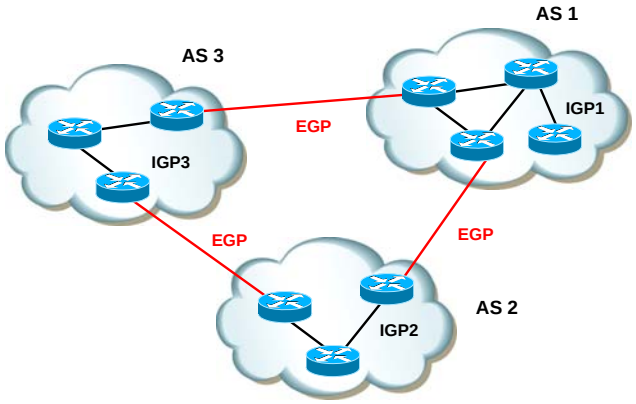
Podle počtu linek, kterými je autonomní systém připojen k okolnímu světu, můžeme autonomní systémy rozdělit na tzv. single-homed a multi-homed . Single-homed autonomní systém je připojen jedinou linkou k jinému AS (typicky poskytovateli Internetu), zatímco multi-homed systém je připojen více linkami. Linky multi-homed autonomního systému mohou vést k tomutéž ISP nebo (častěji) k více různým ISP. Autonomní systém, který dovoluje průchod provozu, který v něm nezačíná ani nekonční, se nazývá tranzitní autonomní systém. Tranzitní samozřejmě mohou být pouze multi-homed AS. Ne každý multi-homed AS je však používán jako tranzitní - multi-homed AS je např. každý AS firmy, která chce mít záložní spojení k více než jednomu ISP. Firma však nemusí mít zájem na tranzitu cizího provozu přes svou síť. Naopak tranzitní systém samozřejmě musí být multi-homed (single-homed AS je připojen pouze jedinou linkou, takže přes něj nemůže žádný provoz procházet).

Pokud k tomu není speciální důvod, nepřiděluje se síti každého zákazníka zvláštní AS. Naopak sítě zákazníků bývají často součástí AS poskytovatele. K žádosti o vlastní jednoznačné číslo AS zákazník typicky přistupuje pouze v případě, že se v budoucnu hodlá připojit k více různým poskytovatelům.

Všimněme si, že z pohledu počtu přeskoků, resp. cen spojů či jiné obvyklé metriky není směrování v Internetu optimální. Optimalita směrování však v praxi není z mnoha důvodů dosažitelná a kvůli potřebě aplikace různých směrovacích politik ani žádaná. Základní technickou komplikací pro dosažení optimálního směrování je neexistence společně interpretované metriky - každý vnitřní směrovací protokol používá svou, s ostatními nesrovnatelnou, metriku (např. počet přeskoků u RIP, cena cesty u OSPF, kompozitní metrika u Cisco IGRP). Suboptimální hierarchické směrování kompenzuje tuto nevýhodu tím, že omezuje počet záznamů ve směrovací tabulce s využitím agregace a položky default pro všechny sítě mimo autonomní systém.

1.3.1.1 Vnitřní a vnější směrovací protokoly

Při směrování v rámci jednotlivých autonomních systémů se používají tzv. vnitřní směrovací protokoly - Interior Gateway Protocols, IGP. Naopak pro směrování mezi autonomními systémy se používají vnější směrovací protokoly - Exterior Gateway Protocols, EGP. Situace je vyobrazena na obr. 1.17. Typickými vnitřními směrovacími protokoly jsou dnes např. OSPF nebo starší RIP, jako vnější směrovací protokol se používá téměř výhradně protokol BGP.



Obr. 1.17 – Interní a externí směrovací protokoly

1.3.2 Vnitřní směrovací protokoly

V dnešní době se používá celá řada vnitřních směrovacích protokolů. V následující kapitole si popíšeme, jak se tyto protokoly rozdělují, jaké jsou jejich základní principy, jak lze optimalizovat cesty nalezené těmito protokoly a na závěr si představíme typické reprezentanty jednotlivých tříd vnitřních směrovacích protokolů.

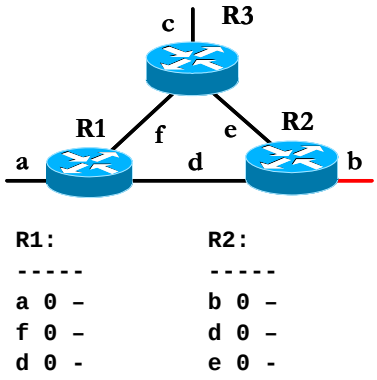
1.3.2.1 Klasifikace vnitřních směrovacích protokolů

Během vývoje byla postupně navrženo a aplikováno mnoho vnitřních směrovacích protokolů. Funkcionalita novějších protokolů zpravidla překonává protokoly starší, avšak z důvodu historických nebo pro jednoduchost implementace či konfigurace nebo výhodnosti pro konkrétní použití jsou stále prakticky využívány i protokoly starší. Při volbě vhodného směrovacího protokolu musíme zvážit zejména tato kritéria:

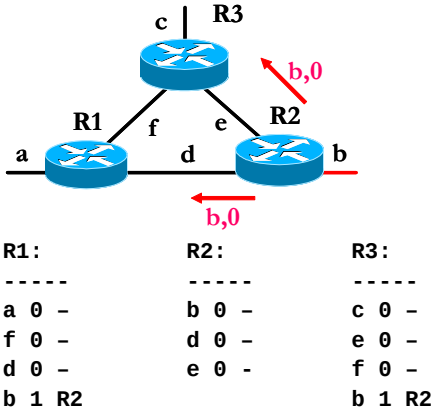
- Základní princip funkce a míra informovanosti o topologii sítě. Rozlišujeme dvě základní třídy směrovacích protokolů: protokoly založené na vektoru vzdálenosti (distance-vector) a na stavech linek (link-state)
- Doba konvergence, tedy po jak dlouhé době od změny topologie se upraví a ve všech směrovačích ustálí nové směrovací tabulky
- Použitá metrika, tedy kritérium, podle něhož směrovací protokol vybírá nejlepší z alternativních cest. Může jít např. o počet přeskoků (tj. směrovačů) na cestě, součet manuálně nakonfigurovaných cen linek tvořících cestu nebo třeba i kombinaci šířky pásma, zatížení, zpoždění a spolehlivosti linek
- Podpora pro vyvažování zátěže přes alternativní cesty a to případně i přes alternativní cesty s různou cenou
- Zda směrovací protokol šíří s adresami sítí i jejich masku podsítě nebo spoléhá na to, že maska podsítě bude odvozena podle dnes již téměř nepoužívané třídy IP adresy

Algoritmy třídy distance vector jsou historicky starší a jejich implementace je jednodušší. Pracují na principu Bellman-Fordova algoritmu, kdy si sousední směrovače navzájem vyměňují své směrovací tabulky a doplňují si informace, které se naučí od sousedů. Topologii celé sítě však neznají, musí se spokojit s adresami sousedů, přes která mají posílat pakety do jednotlivých cílových sítí a vzdálenostmi k těmto sítím, které společně tvoří tzv. distanční vektory.

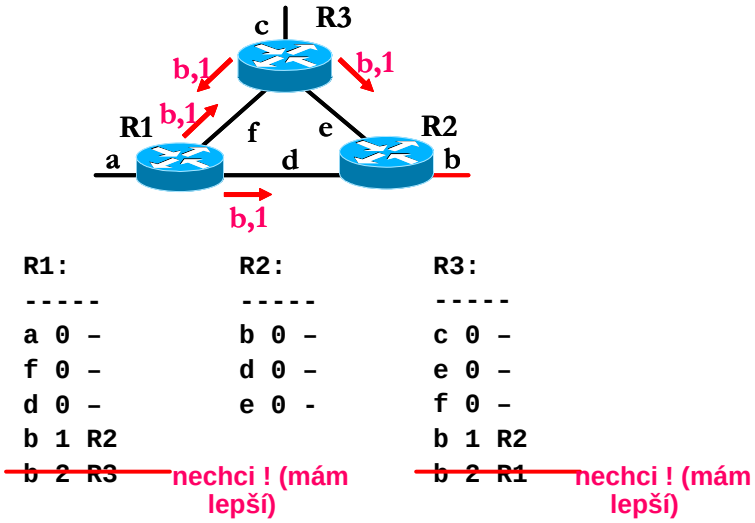
Na začátku směrovací tabulka každého směrovače obsahuje pouze adresy přímo připojených sítí, které jsou staticky nakonfigurovány administrátorem. Směrovací tabulka je periodicky zasílána všem sousedů. Každý směrovač si z došlých směrovacích tabulek sousedů (obsahujících vzdáleností sousedů od jednotlivých cílových sítí) výběrem nejlepší cesty do každé sítě postupně doplňuje a upravuje svou směrovací tabulku. Jestliže je mu sousedem nabízena cesta do sítě, kterou dosud nemá, do směrovací tabulky si ji přidá. Pokud soused nabízí cestu, kterou směrovač již má, ale má ji s horší metrikou, do směrovací tabulky se místo ní zaznamená lepší cesta od souseda. Ostatní nabízené cesty jsou ignorovány.
Odstraňování již neaktuálních cest se děje tak, že informace o každé cestě musí být sousedem pravidelně občerstvována - pokud cesta nebyla delší dobu sousedem inzerována, ze směrovací tabulky se odstraní.
Příklad šíření cesty do sítě symbolicky označené jako **b** a postupné doplňování směrovacích tabulek směrovačů R1, R2 a R3 o cestu do sítě **b** je vidět na obr. 1.18a až 1.18c. Paralelně s tím se samozřejmě šíří informace o ostatních sítích, ta však není z důvodu přehlednosti do příkladu zahrnuta.



Obr. 1.18 a – Informace předkonfigurované ve směrovacích tabulkách směrovačů



Obr. 1.18 b – R2 poslal směrovací tabulku sousedům, ti ji zkombinovali se svými



Obr. 1.18 c – R3 a R2 poslali své směrovací tabulky sousedům, ti je zkombinovali se svými

Metrikou protokolů třídy distance vector je typicky počet přeskoků na cestě do cílové sítě. Ten však nezohledňuje skutečné parametry linek (zejména přenosovou rychlost), které se mohou i velmi zásadně lišit. Konvergence při změnách topologie je velmi pomalá, o změně v topologii směrovač informuje až v příští periodě pro vysílání směrovací tabulky sousedům. Všechny uzly sítě jsou navíc zahlcovány rozesíláním kompletních směrovacích tabulek formou broadcastu. Směrovací protokoly třídy distance vector jsou také až příliš "optimistické" - směrovač se rychle učí dobré cesty, ale pomalu „zapomíná“ cesty nefunkční, protože musí čekat na vypršení časového limitu, po který cesta není sousedem inzerována.
Protokoly třídy distance vector trpí také mnohými riziky, při nichž může dojít k zasmyčkování nebo šíření nesprávné informace. Přestože dnes existují osvědčené mechanismy, jak tato rizika výrazně omezit a také zrychlit šíření informace o změně topologie, je z důvodu aplikace různých pomocných časovačů pro zabránění tvorby smyček v mnohých případech doba konvegence i u malých sítí v řádu mnoha minut.

Novější protokoly třídy link state mají sice složitější implementaci, ale výrazně rychlejší konvergenci v řádu desítek sekund a netrpí tolik problémy chování ve „speciálních situacích“. Směrování zde probíhá na základě znalosti "stavu" jednotlivých linek sítě (funkčnost, cena) s tím, že směrovače znají topologii celé sítě, kterou si udržují v topologické databázi. Topologická databáze je tvořena záznamy o linkách vedoucích k sousedům ode všech směrovačů a udržuje se tak, že každý směrovač neustále sleduje stav a funkčnost k němu připojených linek a při změně okamžitě a spolehlivým mechanismem šíří informaci o aktuálním stavu svého okolí všem ostatním směrovačům. Všechny směrovače tak stále udržují identickou topologickou databázi. Na základě topologické databáze si každý směrovač počítá strom nejkratších cest ke všem ostatním směrovačům a k nim připojeným sítím pomocí Dijkstrova algoritmu. Z vypočteného stromu pak snadno zkonstruuje správnou směrovací tabulku, kterou na rozdíl od protokolů třídy distance vector všechny směrovače počítají na základě stejných a úplných dat

Výhodou algoritmů třídy link state je, že v síti se šíří pouze informace o změnách a není zapotřebí žádné periodické rozesílání směrovacích tabulek jako u protokolů třídy distance vector. Informace o změně se rozšíří prakticky okamžitě do celé sitě, takže konvergence je velmi rychlá.

1.3.2.2 Standardizované vnitřní směrovací protokoly používané v Internetu

Z vnitřních směrovacích protokolů, které jsou definovány otevřenými standardy a poskytují interoperabilitu mezi nejrůznějšími platformami směrovačů je dnes asi nejčastěji používán směrovací protokol OSPF, u menších sítí také již velmi starý protokol RIP. Každý z nich je typickým reprezentantem jedné ze tříd směrovacích protokolů, proto se nyní podíváme podrobněji na jejich vlastnosti.

1.3.2.2,1 RIP

Protokol RIP (Routing information protocol), definovaný v RFC1058 je velmi starý, ale pro jednoduchost implementace a prakticky nulové nároky na znalosti správce stále často používaný v malých sítích. Jeho metrikou je počet směrovačů na cestě k cílové síti. Jelikož eliminuje problém vzniku smyček mechanismem tzv. „počítání do nekonečna“ [4] omezením metriky na hodnotu 15,

nemůže být použit v rozsáhlejších sítích, kde by počet směrovačů na kterékoli cestě mohl být větší než 15. V současné době jsou do implementací RIP běžně zahrnovány různé pomocné mechanismy jako Triggered Updates, Split Horizon, Holddown nebo Route Poissoning [4], které výrazně zrychlují konvergenci okamžitým šířením změn bez čekání na čas periody rozesílání směrovací tabulky, omezují riziko vzniku smyček nebo okamžitě informují o cestách, které se staly nedostupnými. V některých případech zajišťují okamžité odstranění neaktuálních cest bez čekání na vypršení časového limitu, po který nebyla cesta sousedem inzerována.
Směrovací tabulka se v RIP rozesílá každých 30 sekund. Pokud směrovač neslyšel o cestě po dobu 180 sekund, zahájí proces jejího odstranění.
Některé implementace RIP podporují rozkládání zátěže mezi cestami se stejnou metrikou, což je dosaženo jednoduše tím, že si směrovač ukládá do směrovací tabulky ne pouze jednu, ale všechny cesty do cílové sítě s nejnižší hodnotou metriky, pokud je mu takovýchto cest inzerováno více.

Původní protokol RIP nešířil masky podsítě, takže jej nebylo možné použít v sítích s adresací s maskou podsítě proměnné délky (VLSM). Proto byla v RFC2453 později definována verze 2 tohoto protokolu, která mimo šíření masek podsítě propagovaných sítí dále podporuje autentizaci sousedů a umožňuje distribuovat směrovací tabulky multicastem místo broadcastu, čímž se omezí rušení činnosti stanic provozem směrovacího protokolu RIPv2 mezi směrovači.

1.3.2.2.2 OSPF

Protokol OSPF (Open Shortest Path First) byl vytvořen organizací IETF přibližně v letech 1988 až 1991 a je dnes jedním z nejpoužívanějších směrovacích protokolů. Jeho nejnovější verze je definována v RFC2328. Jméno protokolu je odvozeno jednak z toho, že jde o otevřený standard a jednak z faktu, že směrovač nejprve vypočte strom nejkratších cest a až pak z něj vytvoří směrovací tabulku.
OSPF je typickým představitelem směrovacího protokolu typu link state. Vytváří tedy v paměti směrovače kompletní mapu celé sítě, označovanou jako topologická databáze. Nad touto databází potom pomocí Dijkstrova algoritmu provádí výpočty potřebné k nalezení nejvýhodnější cesty do jednotlivých sítí. Protokol OSPF používá metriku označovanou jako cena (angl. cost). To je číslo v rozsahu 1 až 65535, přiřazené ke každému rozhraní směrovače. Čím menší číslo, tím má linka lepší metriku a bude tedy více preferována. Standardně je ke každému rozhraní přiřazena cena automaticky a je nepřímo úměrná šířce pásma linky na daném rozhraní.

OSPF šíří informace o sítích včetně masek podsítí, takže může být použit i v sítích používajících VLSM adresování. Pro provoz používá multicastových adres. Tím je zajištěno, že rámce nesoucí OSPF pakety budou přijímat pouze směrovače podporující tento protokol a nebudou zbytečně přijímány pracovními stanicemi. OSPF může zvládat vyvažování zátěže mezi cestami se stejnou cenou a autentizaci výměny směrovací informace mezi sousedními směrovači.

Ve velmi zjednodušené podobě můžeme funkci protokolu OSPF popsat následovně :

1. Směrovač vysílá přes svá rozhraní tzv. Hello pakety. Pokud se dva navzájem propojené routery provozující protokol OSPF pomocí těchto paketů dohodnou na určitých společných

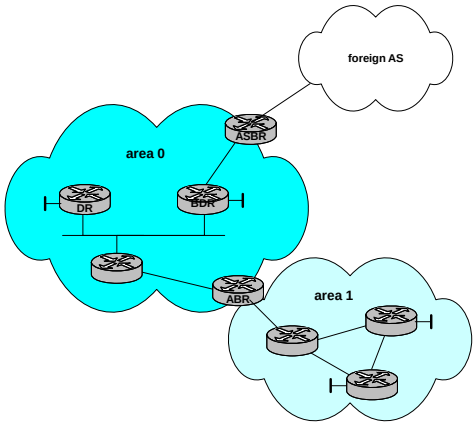
parametrech, stávají se sousedy. Na segmentech, ke kterým je připojeno současně více směrovačů, je zvolen tzv. pověřený směrovač (designated router), který bude koncentrovat výměnu informacé mezi směrovači na segmentu, aby si je nemusely vyměňovat každý s každým a tím zbytečně zatěžovat síť.

2. Sousední směrovače si vzájemně vyměňují pakety označované jako LSA (Link State Advertisement) informace. Ty popisují stav dvoubodových rozhraní jednotlivých směrovačů včetně identifikace k nim připojených sousedů nebo obsahují seznam směrovačů připojených k síti, na kterou může být připojeno současně více směrovačů (typicky jde o segment Ethernetu)
3. Všechny směrovače si ukládají přijaté LSA do své lokální topologické databáze a zároveň je přeposílají na ostatní přilehlé směrovače. Tím se informace postupně rozšíří mezi všechny směrovače v síti. Výsledkem bude shodná topologická databáze na všech směrovačích.
4. Po naplnění databáze každý směrovač provede výpočet pomocí SPF (Dijkstrova) algoritmu. Jeho výsledkem bude nalezení nejkratší cesty ze směrovače do každé známé sítě.
5. Na základě vypočtených dat je možné naplnit směrovací tabulku směrovače.
6. Pokud dojde ke změně topologie sítě, směrovač na kterém ke změně došlo odešle přilehlým směrovačům informaci v podobě LSA paketu. Ta se postupně rozšíří po celé síti a každý směrovač upraví svou topologickou databázi a provede nový výpočet SPF algoritmu.

Výpočet SPF algoritmu představuje pro směrovač poměrně velkou zátěž a je žádoucí, aby neprobíhal příliš často. K tomu by mohlo dojít například v případě, že některá z linek je nestabilní a opakovaně nabíhá a vypadává. Proto bývá definován minimální časový interval mezi dvěmi výpočty.

Velkou výhodou protokolu OSPF proti starším směrovacím protokolům je jeho schopnost pracovat v relativně velkých sítích. Toho se dosáhlo zavedením dvou úrovní hierarchie. Síť je rozdělené na takzvané oblasti (area). Oblast je logická skupina směrovačů a linek mezi nimi. LSA se šíří pouze uvnitř dané oblasti a také výpočet SPF algoritmu se spouští pro každou oblast samostatně. Směrovače v oblasti znají detailně pouze topologii sítě ve své oblasti a z ostatních oblastí dostávají jen souhrnné informace. Změna topologie sítě v jedné oblasti tedy nevyvolá přepočet SPF algoritmu v ostatních oblastech.

Oblasti jsou navzájem propojeny pomocí tzv. hraničních směrovačů (Area Border Router, ABR). K jiným autonomním systémům pak může být autonomní systém s protokolem OSPF připojen pomocí hraničního směrovače autonomního systému (AS Border Router, ASBR), který může do OSPF redistribuovat externí cesty. Příklad sítě rozdělené na více oblastí, která je navíc přes ASBR připojena k dalšímu AS, je vidět na obr. 1.19.



Obr. 1.19: Síť složená z více oblastí

Každá oblast je označena 32 bitovým číslem, které může být uvedeno ve dvou formátech. Buď jako běžné číslo (např. area 10) nebo ve formátu IP adresy (area 0.0.0.10). Zvláštní úlohu mezi oblastmi hraje area 0, někdy označována jako páteřní oblast (backbone). Ta navzájem propojuje všechny ostatní oblasti. Veškerý provoz, který teče z jedné oblasti do druhé, musí procházet přes oblast 0 a každá oblast musí být přes ABR napojená na oblast 0.

Adresování v síti s oblastmi je vhodné navrhnout tak, aby adresy jednotlivých sítí propagovaných sumárně ven z oblasti bylo možné sumarizovat a propagovat jako jedinou sumární cestu (tzv. supernet). Například čtyři za sebou následující sítě 192.168.4.0/24, 192.168.5.0/24, 192.168.6.0/24 a 192.168.7.0/24 můžeme propagovat jako cestu do supernetu 192.168.4.0/22.

Oblasti v OSPF mohou být konfigurovány jako oblasti různých typů podle toho, jaké informace chceme mít ve směrovacích tabulkách směrovačů v oblasti. Můžeme tak snížit požadavky na paměť i procesor směrovačů v dané oblasti. Nejčastěji se setkáme s oblastmi typu „stub“ nebo s jejch modifikací, oblastmi typu „totally stubby“.

Do oblasti typu „stub“ nebudou ABR směrovačem propagovány externí cesty, ale jen cesty nalezené OSPF protokolem v daném autonomním systému. Směrování do externích sítí bude řešeno pomocí implicitní (default cesty), která je do oblasti automaticky propagována z ABR. Jako stub je nejvýhodnější konfigurovat takovou oblast, ze které vede jen jediná cesta ven. Je to však možné i v případě, že cest z oblasti je více, pak ale nemusí pakety směrované do externích sítí procházet nejkratší možnou cestou.
Koncepce oblasti typu „totally stubby“ rozvíjí myšlenku stub oblasti ještě dále. Pokud z oblasti existuje jen jediná cesta ven, proč do ní propagovat cesty z ostatních oblastí stejného AS ? Do oblasti typu „totally stuby“ tedy bude z ABR propagována pouze a jedině default cesta. Ve směrovací tabulce směrovačů uvnitř oblasti typu totally stubby tak nalezneme jen cesty uvnitř oblasti a default cestu, která bude použitá pro dosahování cílů v ostatních oblastech autonomních systémů i sítí mimo autonomní systém.

Posledním typem oblasti je oblast typu NSSA (Not-So-Stubby Area), která má všechny vlastnosti oblasti typu stub, avšak připouští, aby v ní ležel hraniční směrovač autonomního systému (ASBR) a redistribuoval prostřednictvímm této stub oblasti do páteřní oblasti externí cesty.

1.3.2.2.2.1 Optimalizace směrování u vnitřních směrovacích protokolů

Vnitřní směrovací protokoly volí používané cesty do cílových sítí automaticky jako nejkratší cesty vypočtené na základě metriky daného směrovacího protokolu. V některých případech je však výhodné výpočet nejkratších cest směrovacího protokolu vhodným způsobem ovlivnit. Podle konkrétního použitého směrovacího protokolu lze toto realizovat nejrůznějšími způsoby. U směrovacích protokolů třídy link state jde zpravidla o nastavením cen linek na rozhraních směrovačů a vhodné definice oblastí a sumarizace informace propagované mezi oblastmi. Naopak u směrovacích protokolů třídy distance vector lze snadno vyfiltrovat informaci o některých cestách propagovaných do určitých částí sítě nebo definovat, že pro určité propagované nebo přijímané cesty bude uměle zvýšen počet přeskoků, čímž bude cestě dána nižší preference.

Pro konkrétní směrovač můžeme také definovat lokální pravidla pro směrování paketů, která budou mít přednost před dynamickými směrovacími protokoly. Ty mohou mimo adresy cílové sítě zahrnovat i jiná kritéria, jako zdrojovou adresu nebo třídu služby přicházejících paketů.

V sítích, které z různých důvodů používají současně více směrovacích protokolů, můžeme volit vzájemnou prioritu protokolů pro případ, že by směrovač získal různé cesty do téže sítě od různých směrovacích protokolů. Prioritu můžeme dát i "záložní" staticky nakonfigurované cestě, která bude normálně neaktivní a použije se pouze v případě, že se ztratí primární cesta propagovaná dynamickým směrovacím protokolem.

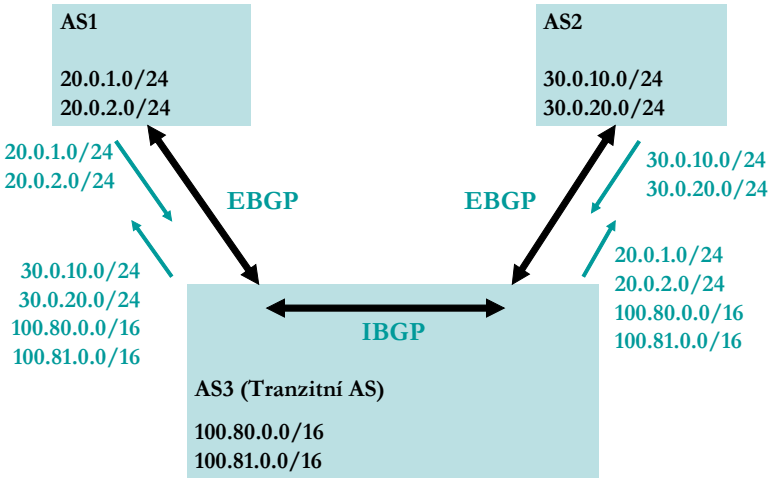
Směrovací informaci mezi částmi sítě používajícími různé směrovací protokoly také můžeme mezi protokoly redistribuovat, přičemž musíme dát pozor na možnost zacyklení a řešit problém zpravidla nekompatibilních metrik jednotlivých protokolů.

1.3.3 Vnější směrovací protokoly

V následující kapitole se podíváme na protokoly pro směrování mezi autonomními systémy. Po krátkém shrnutí vlastností vnějších směrovacích protokolů se zaměříme na protokol BGP, který se v současném Internetu pro směrování mezi AS používá výhradně.

1.3.3.1 Charakteristika vnějších směrovacích protokolů

Vnější směrovací protokoly propagují z každého autonomního systému všechny sítě, které mají být z vnějšího světa dostupné nebo do kterých nechává autonomní systém přes sebe procházet tranzitní provoz (obr. 1.20). Protože sítí propagovaných z autonomního systému může být velké množství, je výhodné, když sítě v rámci autonomního systému mají společný prefix adresy a mohou být propagovány společně jako jediná supersíť.



Obr. 1.20 – Propagování vlastních nebo dosažitelných prefixů sítí z AS

Směrovací informace se mezi autonomními systémy vyměňuje prostřednictvím hraničních routerů, angl. border gateway. Z tohoto názvu je odvozeno i jméno v současnosti prakticky jediného používaného vnějšího směrovacího protokolu: Border Gateway Protocol - BGP. Pomocí BGP si hraniční směrovače vyměňují informace o sítích v jednotlivým autonomních systémech a o tom, přes které autonomní systémy se lze k jednotlivým sítím dostat. V dnešní době se používá téměř výhradně protokol BGP ve verzi 4 RFC2283 nebo 4+, nazývaný někdy Multiprotocol BGP.

1.3.3.2 Směrovací protokol BGP

Protokol BGP nepracuje s grafem propojení jednotlivých směrovačů a sítí (jako to dělá např. OSPF), ale s grafem propojení autonomních systémů. V tomto grafu jsou pak vyhledávány cesty mezi sítěmi v různých autonomních systémech. Cestou (AS PATH) k nějaké síti se v terminologii BGP rozumí posloupnost čísel autonomních systémů, přes které se lze k cílové síti dostat.

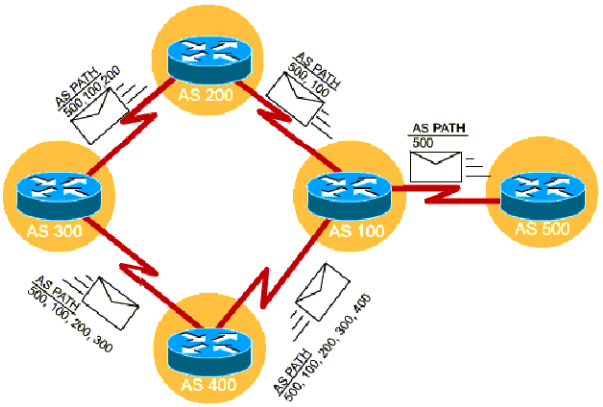
Na rozdíl od vnitřních směrovacích protokolů nemá BGP jednoznačnou metriku, podle níž by za všech okolností automaticky volil nejkratší cesty do jednotlivých cílových sítí. Při směrování mezi AS totiž směrujeme provoz přes cizí AS, jejichž provozovatelé mají nejrůznější zájmy a provozní i obchodní podmínky. Respektováním všech těchto faktorů pak určíme tzv. směrovací politiku (angl. routing policy). Směrovací politika například určuje

- do kterých AS necháme tranzitovat provoz přes náš AS
- ze kterých zdrojových AS necháme tranzitovat provoz přes náš AS
- kterou výstupní linkou z našeho AS necháme odcházet provoz k daným sítím
- kterou vstupní linkou do našeho AS necháme vstupovat provoz ke kterým sítím

Protože je třeba do konfigurace protokolu BGP všechny faktory směrovací politiky zahrnout, je jeho konfigurace mnohem více manuální, než jsme zvyklí z protokolů třídy IGP. U protokolů IGP jsou obvykle sousední směrovače vyhledávány automaticky a předpokládá se, že komunikovat spolu mohou všechny nalezené směrovače a že cesty do jednotlivých cílových sítí nejsou omezeny

žádnými dodatečnými podmínkami a hledá se vždy cesta s minimální hodnotou metriky. Naopak u BGP jsou sousední směrovače konfigurovány manuálně, stejně jako transformace prováděné nad cestami předávanými mezi jednotlivými sousedy.

Vnitřní směrovací protokoly se tradičně rozdělují na distance-vector a link-state. Z pohledu úrovně znalosti topologie sítě, způsobu předávání i obsahu směrovací informace se protokol BGP řadí na jejich pomezí. Někdy bývá označován jako protokol speciální třídy, nazývané path-vector. Termínem path vector rozumíme posloupnost čísel autonomních systémů, přes které vede cesta k nějaké síti. Spolu s každou cestou je šířen i její path vector, který se postupně prodlužuje tím, jak každý AS, přes který cesta projde, na začátek path vectoru připojí své číslo. Protože cesta nesmí obsahovat smyčku, může se číslo autonomního systému v path vector objevit nejvýše jednou. Smyčky při předávání směrovací informace se automaticky eliminují tak, že AS zahazuje nabízené cesty, které již v path vectoru obsahují jeho vlastní číslo autonomního systému (obr. 1.21).



Obr. 1.21 – zamezení smyčkám při předávání směrovací informace pomocí Path vector

Path vector také slouží k výběru nejkratší cesty do jednotlivých sítí. Za nejkratší bude považována ta cesta, která prochází nejmenším počtem autonomních systémů. Při výběru z alternativních cest do sítí tedy budou preferovány ty cesty, jejichž path vector je kratší.

Směrovací informace (routing updates) se v BGP vyměňuje vždy mezi sousedními směrovači. BGP směrovače jsou vždy na hranicích autonomního systému[1]. Každému BGP směrovači jsou při konfiguraci manuálně přiřazeni sousedé, se kterými si bude směrovací informaci vyměňovat. Aby výměna směrovací informace byla spolehlivá, probíhá s použitím protokolu TCP (port 179). Po navázání spojení mezi sousedy se mezi těmito sousedy vymění kompletní směrovací informace, která je oběma známa. Dále se pak již předávají pouze změny (směrovač propaguje novou přes něj dostupnou síť nebo odvolává dostupnost dříve inzerované sítě).

Předávané směrovací informace mají tvar dvojic <*prefix_adresy_sítě, délka_prefixu*>. Sítě s IP adresou začínající inzerovaným prefixem jsou dostupné přes AS, jehož hraniční směrovač prefix inzeruje. Ke každé cestě může být dále přiřazeno libovolné množství tzv. atributů, pomocí nichž se realizují směrovací politiky, jak bude vysvětleno dále

[1] výjimku tvoří tzv. interní BGP v tranzitních AS

BGP směrovače si periodicky (obvykle 1x za minutu) testují dostupnost každého svého nakonfigurovaného souseda pomocí tzv. keepalive zpráv. Pokud soused přestane být dostupný, musí směrovač odstranit všechny cesty vedoucí přes tohoto souseda a informovat ostatní sousedy o nedostupnosti cest

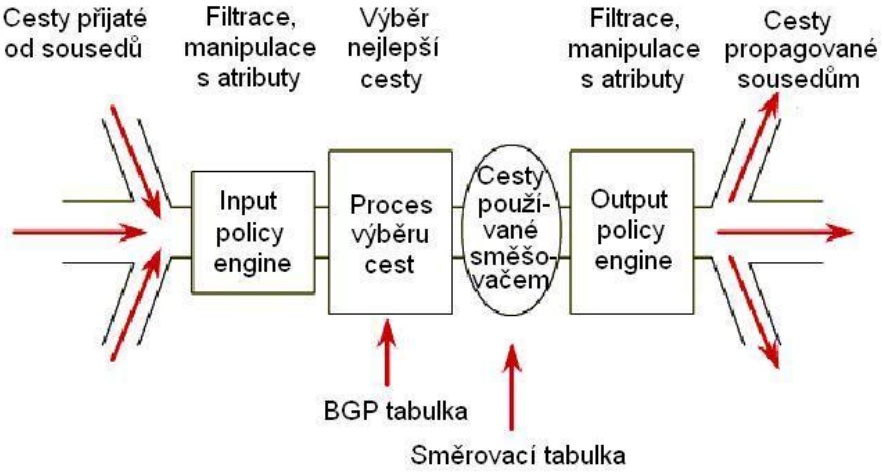
Informace o cestách získaných od sousedů se směrovač ukládá do databáze, nazývané někdy BGP tabulkou. Do každé sítě směrovač zvolí některou z cest uložených v databázi a vloží si tuto cestu jako záznam směrovací tabulky. Cesty, které směrovač sám používá (tj. vybral si je do směrovací tabulky) pak propaguje pomocí BGP svým sousedům. Výběr cesty z BGP tabulky do směrovací tabulky se děje jednak na základě délky path vector a jednak na základě hodnot atributů, jak si vysvětlíme dále.

1.3.3.2.1 Optimalizace směrování v protokolu BGP

Vyhledávání cest na základě algoritmu path-vector umožní nalézt nejkratší cesty do všech autonomních systémů. Abychom však byli schopni explicitně ovlivňovat směrovací politiky, je potřebný mechanismus, kterým bychom vyjádřili preferenci, resp. zákaz některých cest podle nejrůznějších kritérií. K tomuto účelu v protokolu BGP slouží atributy, které můžeme každé propagované cestě k cílové síti přiřadit. Je definováno několik atributů, z nichž některé jsou povinné a některé nepovinné. Dále je definován způsob zacházení s případnými neznámými atributy tak, aby postupně mohly být dodefinovávány další.

Ke každé cestě propagované v BGP musí být přiřazeny jisté povinné atributy (přinejmenším atribut AS_PATH nesoucí path vector příslušné cesty) a dále volitelně i další atributy, k jejimž hodnotám může být při propagování cesty mezi AS a při výběru nejlepší cesty do směrovací tabulky přihlíženo.

Realizace směrovací politiky mezi AS se děje pomocí manipulací s atributy cest přijímaných od jednotlivých sousedů, resp. propagovaných k jednotlivým sousedů a také filtraci těchto cest. Atributy jsou nastavovány a zpracovávány při přijetí směrovací informace od souseda v tzv. Input Processing Engine a také před odesláním informace sousedovi v tzv. Output Processing Engine BGP směrovače (obr. 1.22). Mezi tím jsou uchovávány v tzv. BGP tabulce, odkud jsou na základě pevně stanoveného algoritmu zohledňujícího hodnoty atributů vybírány do směrovací tabulky.
U každého souseda můžeme samostatně stanovit, jaké transformace atributů se mají dít při příchodu informace od tohoto souseda a také při formulování informace pro tohoto souseda. Hodnoty atributů a prefixy cest mohou být testovány na různé podmínky a při shodě vhodným způsobem měněny hodnoty jiných atributů. Na základě hodnot atributů nebo prefixů cest mohou být také některé cesty filtrovány. Toto testování a nastavování se děje nezávisle při přijetí cesty a před její propagací.



Obr. 1.22 – Zpracování atributů příchozích a odchozích cest v BGP

Mezi nejpoužívanější atributy patří atribut AS_PATH, NEXT_HOP, LOCAL_PREFERENCE, MED a COMMUNITY. Atribut AS_PATH obsahuje řetězec čísel autonomních systémů, přes které vede cesta k cílové síti. Pokud se vybírá z cest lišících se pouze v AS_PATH, volí se vždy cesta s "kratším" AS_PATH, tedy s hodnotou AS_PATH obsahující menší počet čísel AS. Textový řetězec v AS_PATH lze testovat na výskyt definovaného podřetězce pomocí regulárních výrazů. Takto lze např. vyloučit nebo naopak preferovat cesty procházející přes některé konkrétní AS. Také lze řetězec AS_PATH modifikovat, např. uměle jej prodlužovat vícenásobným vkládáním čísla AS, kterým propagovaná cesta prochází.

Druhým z atributů, který musí být povinně uveden u každé cesty, je atribut NEXT_HOP. Je v něm obsažena adresa rozhraní hraničního směrovače sousedního AS, který informaci o cestě do AS zaslal (tedy adresa směrovače z cizího AS). Tím se BGP značně odlišuje od protokolů třídy IGP, kde se předpokládá, že adresa nejbližšího souseda na cestě k cílové síti (next hop) je vždy adresa některého bezprostředně sousedícího směrovače. Cestu k rozhraní hraničního směrovače propagovaného v atributu NEXT_HOP musí směrovač zjistit z vnitřního směrovacího protokolu svého autonomního systému, který tak musí mít informaci i o adrese sítě spojovací linky mezi AS.

S použitím nepovinného atributu LOCAL_PREFERENCE se mohou směrovače (multihomed) autonomního systému dohodnout na společné volbě cesty k nějaké cizí síti, která je dostupná přes více alternativních linek. Naopak atributem MED (Multi-Exit Discriminator) lze ovlivnit volbu cesty používanou sousedním AS pro dosažení jednotlivých sítí uvnitř našeho AS, resp. za naším AS pokud dovolíme přes náš AS tranzitovat provoz.
Atributem COMMUNITY můžeme cestě přiřadit „značku“ a dále v síti pak cesty označené jistou značkou zpracovávat speciálním způsobem, například je filtrovat nebo naopak preferovat.

Pokud má směrovač v BGP tabulce k dispozici více alternativních cest k nějaké síti, musí vybrat jednu z nich do směrovací tabulky. Při jistém zjednodušení vypadá pořadí kritérií ovlivňujících výběr nejlepší cesty následovně:

1. Hodnota atributu LOCAL_PREFERENCE
2. Preference cesty generované směrovačem samotným (a pocházející z jeho AS) – např. cesty získané redistribucí z IGP
3. Kratší AS_PATH (menší počet čísel AS v hodnotě AS_PATH)
4. Hodnota atributu MED
5. Propagovaný next_hop dostupný přes kratší cestu vnitřkem AS
6. Nižší hodnota identifikátoru směrovače, od kterého byla cesta získána, tzv. Router ID. Router ID se používá pro definitivní rozhodnutí v případě, že podle žádného "rozumnějšího" kritéria nebylo možné rozhodnout.

Je užitečné si uvědomit, že počet AS v atributu AS_PATH je až čtvrtým kritériem v pořadí.

Na závěr naší diskuse o BGP si všimněte, že manipulací s atributy můžeme nezávisle ovlivňovat, kterými linkami bude procházet provoz ven z AS a provoz dovnitř do AS. Můžeme také dosáhnout jistého vyvažování zátěže, např. použitím jedné linky pro provoz k některým sítím a druhé linky k sítím ostatním. Při tom však musíme mít na zřeteli, že z mnoha důvodů je velmi dobré udržet symetrii směrování, tj. provoz do sítí odcházející určitou linkou by se měl touto linkou také vracet. Požadavky na vyvažování zátěže a symetrii směrování bývají často protichůdné, proto je obvykle třeba hledat vhodný kompromis. Na rozdíl od vnitřních směrovacích protokolů se tak v BGP vyvažování zátěže do jednoho cíle více linkami příliš často nepoužívá.