Dynamické směrování

Michal Minařík, Y36SPS

Směrování

Obecně můžeme říct, že pod pojmem směrování (anglicky routování) si můžeme představit hledání cesty v počítačových sítích. Úkolem je dopravit paket přes někdy i velmi komplikovanou infrastrukturu k místu jeho určení. O směrování se starají aktivní síťové prvky, které nazýváme směrovače neboli routery, které propojují jednotlivé sítě (subnety). Tyto síťové prvky pracují na třetí (síťové) vrstvi OSI modelu. Ke směrování se využívá IP protokol.

Směrování můžeme rozdělit na několik základních kategorií, rozdělení odpovídá tomu, jak vznikala routovací tabulka v routeru.

Statické směrování

Do routovací tabulky je přímo napsáno, které sítě jsou připojeny ke konkrétnímu interfacu. Tento typ směrování je bezpečný, ovšem není schopen reagovat na změny topologie sítě.

Defaultní směrování

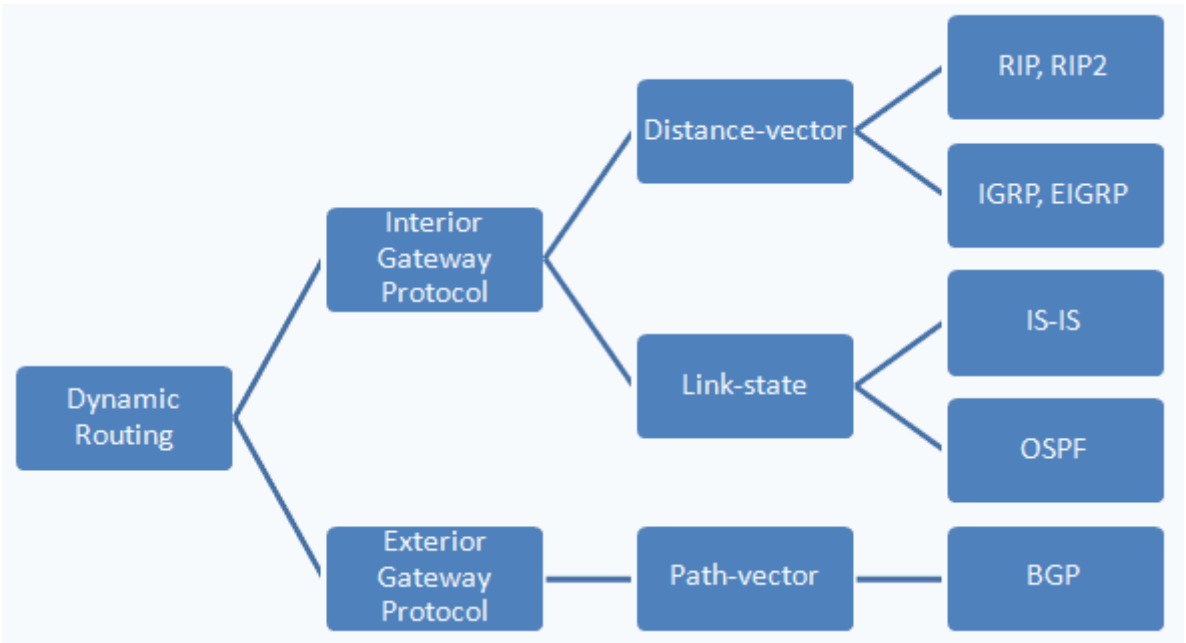
Defaultní směrování je taková cesta, kterou router zvolí, pokud neví kam daný packet zařadit.

Dynamické směrování

Reaguje na změny topologie v síti, routery si mezi sebou vyměnují informace a automaticky přizpůsobují svoje routovací tabulky aktuálnímu stavu sítě. A právě tímto směrováním bych se chtěl více zabývat.

Dělení dynamického směrování

Dynamické směrování je opět možno rozdělit do několika různých kategorií. Nejzákladnější dvě jsou, link-state routing protokol a distance-vector routing protokol. O nich si povíme něco více dále. Druhým dělením routovacích protokolů je, zda jsou určeny pro práci uvnitř autonomního systému (lokální sítě) nebo fungují napříč autonomními systémy. Takové dělení nám potom rozděluje protokoly na interior (interní) a exterior (externí) gateway protokoly. Celé dělení i příklady jednotlivých protokolů názorně ukazuje následující obrázek.

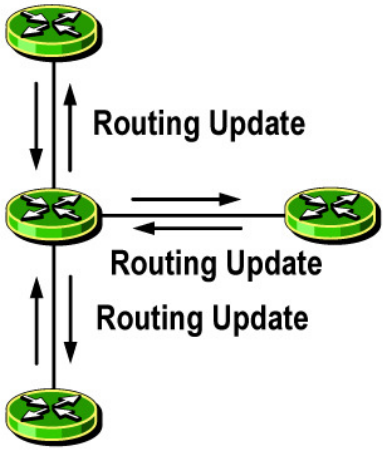


Distance-vector

Sousední zařízení si v těchto typech protokolu vyměňují navzájem routovací tabulky. Na základě takto získaných informací si zařízení vytváří nebo upravuje svoje záznamy v routovací tabulce. Úpravou je myšlena změna distance vector čísla, které např. (u RIP) udává počet hopů k dané adrese (subnetu). K výměně a úpravě těchto údajů dochází pravidelně např. u protokolu RIP je tomu tak jednou za 30 sekund. Nevýhodou tohoto typu směrování je, že zařízení znají topologii sítě jen na základě informací od svých bezprostředních sousedů. Mezi zástupce této rodiny protokolů můžeme zařadit např. protokoly RIP (routing information protokol), IGRP, EIGRP.

Routing information protokol (RIP)

Je směrovací protokol, který používá distance-vector, tedy dochází zde k výměně směrovacích tabulek. Poprvé byl tento protokol definován v roce 1988 (RFC 1058) a od té doby prošel vývojem, vznikla jeho novější verze RIP 2, byla také definována nová verze protokolu RIPng (RIP new generation), která zvládá i IPv6. Tento protokol se i v dnešní době stále používá, ale nutno říct, že je již překonaný např. protokolem OSPF z rodiny link-state protokolů.



RIP v2 podporuje na rozdíl od RIP v1 masky sítí, což umožňuje používat beztřídní směrování. Update směrovacích tabulek umožňuje autentizaci, je tedy bezpečnější. A v neposlední řadě také podporuje VLSM. Svoje updaty posílá RIP v2 v multicastu (224.0.0.9).

RIP funguje na principu počítání hopů na cestě k jednotlivým sítím. Jde vlastně o průchod grafem, při kterém je použit Bellman-Fordův algoritmus, což ovšem vede k tomu, že můžou vznikat nekonečné smyčky (stále by se přičítaly hopy). Proto je nezávisle na rozloze sítě stanoven maximální počet hopů u RIP protokolu na 15. Tedy číslo 16 reprezentuje nekonečno, tedy nedostupný prvek sítě. Routovací tabulky si routery vyměnují v 30 sekundových intervalech, a aby nedošlo k zahlcení sítě každou půl minutu, každé zařízení se na začátku inicializuje náhodně. Hned je zřejmá další nevýhoda tohoto protokolu a to je pomalá konvergence v případě změny topologie.

Jak již bylo řešeno problémem rodiny distance-vector jsou smyčky, se kterými je nutno se vypořádat. Jsou proto zavedeny některé techniky, které používá např. i RIP.

Split horizon

Split horizon znamená, že router neposílá update zpět na interface, ze kterého přišel.

Hold down timer

Je čas, po který router ignoruje další updaty. Tím se sice zpomalí konvergence sítě, ale zabrání se tak např. šíření špatného updatu mezi zařízeními.

Time to live (TTL)

Je číslo, které si sebou nese packet. Toto číslo se s každým hopem sníží o jedničku, pokud dosáhne nuly je paket prohlášen za mrtvý a je zahozen.

Route poisoning

Pokud je v síti zjištěna chybná cesta (výpadek/porucha), tak jsou všechny routery v síti na tuto skutečnost upozorněny tím, že počet hopů je roven 16, což je pro tento protokol nekonečno.

Základní příkazy pro konfiguraci na CISCO

Pro nakonfigurování protokolu RIP nám stačí naučit router přímo připojené sítě a další se již naučí z updatů, které mu pošlou ostatní routery.

```
ROUTER(config)#router rip
ROUTER(config-router)#version 2                  //pokud chceme RIP v2
ROUTER(config-router)#network 10.0.1.0
ROUTER(config-router)#network 10.0.2.0
ROUTER(config-router)#debug ip rip               //zapne debugování updatů
```

Interior Gateway Routing Protocol (IGRP)

Další z rodiny DV, tento protokol byl zaveden firmou CISCO a byl vytvořen, aby odstranil nedostatky protokolu RIP (15 hopů, maximální velikost). Maximální počet hopů u IGRP je stanoven na 255, již zde je vidět značné zlepšení oproti protokolu RIP. Dalším významným prvek protokolu IRGP je to, že podporuje více metrik (RIP pouze jedna metrika – počet hopů), IGRP má propustnost, zatížení, zpoždění, MTU a spolehlivost. Ovšem při porovnávání dvou cest se všechny tyto metriky přepočítají podle předem nastavených vzorců do jedné, která se je potom použita pro porovnání.

Protokol nepodporuje beztřídní směrování, které umí např. RIP v2. Nejenom díky tomuto nedostatku, byl tento protokol nahrazen svojí vylepšenou verzí EIGRP (enhanced IGRP), který již např. podporuje VLSM. Od Cisco IOS verze 12.3 není protokol IGRP vůbec podporován je nutné použít místo něj EIGRP.

Pro zvýšení jsou opět definovány některé metody již zmíněné u RIP. Jako je hold down timer, split horizon a poison reverse. V Cisco implementaci protokolu IGRP jsou posílány reverse-poison updaty pokud metrika stoupne o koeficient 1.1 a výš.

Základní příkazy pro konfiguraci

Při konfigurace je nutné uvést číslo autonomního systému, v našem případě jsme použili číslo 10.

```
ROUTER(config)#router igrp 10
ROUTER(config-router)#network 10.0.1.0
ROUTER(config-router)#network 10.0.2.0
ROUTER(config-router)#debug ip igrp              //zapne debugování updatů
```

Enhanced Interior Gateway Routing Protocol (EIGRP)

EIGRP využívá principy IGRP, které dále obohacuje a zlepšuje. EIGRP distance-vector protokol obohacený o prvky link-state routing protokolu. Vyznačuje se rychlejší konvergencí než IGRP, vetší rozšiřitelností a eliminuje smyčky. Pro eliminaci smyček používá EIGRP algoritmus DUAL (Diffusing Update ALgorithm). Opět podpora IPv6 a beztřídního (classless) směrování.

Router vytváří a udržuje vztahy se soudy pomocí hello packetů. Tyto packety posílá každých 5 sekund (na pomalých sítích 60 sekund). Aby mohly být routery sousedy, musí být ve stejném autonomním systému a musí mít stejně nastavené konstanty pro výpočet metriky. Vzorec pro výpočet metriky je následující:

$$\left[\left(K_1 \cdot \text{Bandwidth} + \frac{K_2 \cdot \text{Bandwidth}}{256 - \text{Load}} + K_3 \cdot \text{Delay}\right) \cdot \frac{K_5}{K_4 + \text{Reliability}}\right] \cdot 256$$

EIGRP používá k uchovávání informací o síti tři tabulky: tabulku sousedů, topologickou tabulku, směrovací tabulku.

Tabulka sousedů

Obsahuje seznam přilehlých routerů (je obdobou adjacency table u OSPF). Pokud se vyskytne nový soused, zaznamená se jeho adresa a rozhraní, ke kterému je připojen. Ve chvíli, kdy soused vyšle hello paket, pošle informaci o tzv. "hold time" - což je doba, po kterou se router považuje za dosažitelný a aktivní. Jestliže během hold time nepřijde hello paket, pak hold time vyprší (délka hold time je většinou trojnásobná než délka intervalu pro vysílání hello paketů). Po vypršení tohoto času se spuští DUAL - Diffusing Update Algorithm (což je EIGRP distance vector algoritmus), který přepočítá novou topologii.

Topologická tabulka

Je vytvořena ze všech směrovacích tabulek v autonomním systému. DUAL použije informace z tabulky sousedů a topologické tabulky a spočítá nejvýhodnější cesty do všech sítí (s nejnižší cenou, bez smyček). Nejlepší cesta se označuje za successor route (následující cesta). Tato informace se zaznamená také do topologické tabulky.

Směrovací tabulka

Obsahuje nejlepší cesty k cílovým sítím.

EIGRP posílá částečné, omezené updaty, čímž šetří propustnost linky. EIGRP router neposílá celé tabulky, ale jen rozdílové updaty. Narozdíl od OSPF routerů neposílá updaty všem routerům v oblasti, ale jen těm, které informace potřebují. EIGRP router posílá pravidelně malé hello pakety (multicast adresa 224.0.0.10), které nijak nezatěžují linku a umožňují routerům vzájemnou minimální komunikaci.

EIGRP používá vlastní transportní protokol, tzn. není zavislý na TCP/IP (jako RIP, IGRP a OSPF). Tento protokol se nazývá Reliable Transport Protocol (RTP).

Základní příkazy pro konfiguraci

```
ROUTER(config)#router eigrp 100                    // zapnutí EIGRP, 100 = číslo AS
ROUTER(config-router)#network 10.1.10.0 0.0.0.255 // které interfacy se zapojí do EIGRP (podle
čísla sítě), u masky se používá wildcard mask
ROUTER(config-router)#no auto-summary              // vypnutí autosumarizace
ROUTER(config-router)#variance number              // nestejnoměrné rozvažování (unequal cost
path load balancing),  akceptuje vše < min_metric*variance
ROUTER(config-router)#distance eigrp 80 130        // změna AD, slouží k zabránění smyčkám mezi
vice AS při vícecestné redistribuci, 80 = internal-distance, 130 = external-distance
ROUTER(config-router)#passive-interface serial0/0 // neoznamuje routovací informace na daném
interface - přestane vysílat i přijímat hello pakety (tedy routovací updaty)

ROUTER(config-if)#ip summary-address eigrp 100 192.1.0.0 255.255.0.0  // manuální definice
sumarizace, 100 = číslo AS
ROUTER(config-if)#bandwidth 56                       // max. kbits datový tok interfacem, pro
určování metriky
ROUTER(config-if)#ip bandwidth-percent eigrp 1 200   // kolik procent může využít EIGRP, AS =
1, 200% (počítá se z bandwidth)

ROUTER#show ip eigrp traffic    // statistiky EIGRP paketů
ROUTER#show ip eigrp topology   // zobrazí záznamy z tabulky topologie
```

Link-state routing protokol

Druhou významnou skupinou protokolů je rodina link-state. Tyto protokoly si udržují povědomí o celé topologii sítě, jsou si vědomi jednotlivých zařízení a jsou schopny rychle reagovat na změny

topologie. Nevýhodou je vyšší náročnost na hardware (více paměti, rychlejší procesory) a dále také zahlcení sítě na začátku, kdy všechny routery potřebují vytvořit směrovací a topologické tabulky.

Při změně topologie sítě dojde k odeslání LSA (Link State Advertisements) paketu zařízením, které tuto změnu detekovalo všem ostatním zařízením na síti, tím dojde k přepočítání směrovacích tabulek. Konvergence sítě při změně topologie, je tedy u těchto protokolů velmi rychlá.

Pokud nenastává žádná změna topologie, komunikují spolu routery přes hello pakety, ve kterých posílají informace o sobě. Je používáná komplexní metrika a také Dijkstrův algoritmus, pro nalezení nejkratší cesty. Pro ještě zefektivnění výkonu je celá síť rozdělena na menší oblasti.

Do této rodiny patří např. OSPF, IS-IS.

Open shortest path first (OSPF)

Je protokol z rodiny link-state. Celý systém funguje na principu hierarchických oblastí. Každá oblast (area) je připojena k páteřní oblasti (backbone). Pokud komunikujeme v rámci oblasti, probíhá vše naprosto normálně, pokud ovšem potřebujeme komunikovat s jinou oblastí, musí být tato komunikace směrována přes páteřní oblast (area 0).

Routery si v rámci této oblasti posílají LSP (link-state packety), takže si každý router udržuje tabulku topologie pro danou oblast. Jak je vidět, není zde tedy omezení na počet hopů (jako tomu je u prokotolu RIP), výsledou cestu si spočítá každý router pomocí Dijkstretova algoritmu, kterou následně zapíše do svojí routovací tabulky. Tento protokol je určen pro rozsáhlé heterogenní sítě a podporuje VLSM.

Pro komunikaci se používá protokol IP 89, hello pakety jsou posílány jednou za 10 sekund (využívá se multicast 224.0.0.6, 224.0.0.5, 224.0.0.2). Aby se mohly stát routery sousedy je třeba, aby se schodovaly v čísle oblasti, typu oblasti, subnetu (a masce), hello a dead timeru a autentizacích údajích. LSA se posílá jednou za 30 minut a to na všechny strany (šíří se záplavově).

Typy oblastí v OSPF

- **Standardní oblast** – přijímá updaty a sumární routy i externí
- **Backbone** – páteř sítě, připojení ke všem oblastním (area 0), jinak stejné vlastnosti jako standardní
- **Stub** – nepřijímá routy z ostatních AS, pro směrování mimo AS se použije defaultní routa

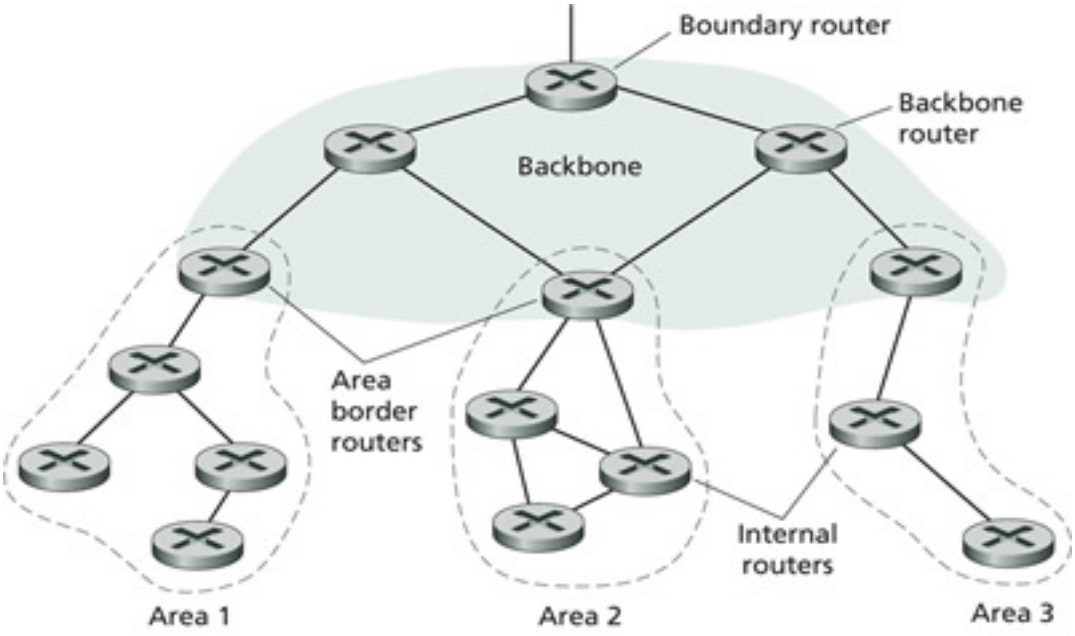
Typy routerů

Pokud jsme si takto zavedli typy oblastí, můžeme rozlišovat i různé typy routerů.

- **Area Border Router - ABR** - má interfacy ve více oblastech, pro každou oblast má separátní tabulku link-state, připojuje oblasti do backbone
- **Autonomous System Border Router - ASBR** - má interfacy ve více AS, slouží k distribuci route z jiné AS, většinou zde běží i BGP
- **Internal Router** - běžný, pouze v jedné oblasti
- **Backbone Router** - alespoň jeden interface v Area 0

DR/BDR routery

- **Designated Router - DR** - posílá LSA 2 všem sousedům v multi-access, je to router (přesněji interface routeru), který se volí v rámci segmentu u multiaccess (speciálními technikami i u NBMA), slouží k redukci síťového provozu, DR je zdrojem routovacích updatu, udržuje si kompletní tabulku topologie, všechny ostatní routery s ním navazují spojení
- **Backup Designated Router - BDR** - stává se DR, pokud selže původní DR, má druhou nejvyšší prioritu v době volby



Základní konfigurace

```
ROUTER(config)#router ospf 1     // 1 je process-id, můžeme provozovat více procesů na routeru
ROUTER(config-router)#network 192.168.5.4 0.0.0.3 area 1 // používá wildcard masku, 1 je číslo
oblasti
ROUTER(config-router)#neighbor 192.168.5.4      // určení souseda, může obsahovat prioritu/cenu
ROUTER(config-router)#summary-address 10.1.0.0 255.255.0.0 // sumarizace sítí na ASBR
ROUTER(config-router)#redistribute connected [subnets]  // posílá všechny lokální interfacy,
subnets - pošle i subneted routes
ROUTER(config-router)#passive-interface Serial0/0       // daný interface neposílá a nepřijímá
updaty
ROUTER(config-router)#area 1 stub               // určí oblast 1 jako stub
ROUTER(config-router)#area 1 stub no-summary    // neposílá LSA 3 a 4 = totally stuby
ROUTER(config-router)#area 1 nssa no-summary    // neposílá LSA 3 a 4 = NSSA totally stuby
ROUTER(config-router)#area 1 range 10.1.0.0 255.255.0.0 // sumarizace sítí na ABR, můžeme
určit, zda se má zveřejňovat - klíčové slovo advertise
ROUTER(config-router)#area 16 virtual-link 8.187.175.82 // virtuální link na IP (třeba
loopback) do backbone

ROUTER(config-if)#ip ospf network point-to-multipoint    // nastaví mod na interface, další
možnosti broadcast, non-broadcast, point-to-point
ROUTER(config-if)#encapsulation frame-relay              // nastaví encapsulaci interfacu
ROUTER(config-if)#ip ospf priority 10                    // nastaví prioritu pro volbu DR
ROUTER(config-if)#ip ospf cost 10        // nastavení cenu odchozím paketům na interfacu
ROUTER(config-if)#frame-relay map ip 10.1.1.1 200 broadcast // 200 = DLCI (Data-link
connection identifier), broadcast potřeba pro Frame Relay

ROUTER#show ip ospf                 // hlavní informace o OSPF procesu, oblasti, apod.
ROUTER#show ip ospf border-routers  // interní routovací záznamy do ABR a ASBR
ROUTER#show ip ospf neighbor        // informace o sousedech per interface včetně link state,
bez ABR, ASBR, SPF
ROUTER#show ip ospf interface       // informace z daného interfacu související s OSPF (link
state .) - router ID, vztah se sousedy
ROUTER#show ip ospf virtual-link    // info o virtul link do backbone
ROUTER#show ip ospf database        // info o topologii, link state, LSA
```

Intermediate System to Intermediate System (IS-IS)

Tento protokol opět síť rozděluje na hierarchické oblasti. Topologie je počítána pomocí Dijkstrova algoritmu a údaje jsou následně zaznamenávány do směrovací tabulky. Umožňuje autentizaci, podporuje beztřídní směrování. Pro posílání hello packetů nebo update packetů se stejně jako u OSPF využívá multicast.

Tento protokol byl definován jako mezinárodně uznávaný standard ISO/IEC 10589:2002. Protokol IS-IS pracuje stejně jako OSPF na 3 síťové vrstvě ISO OSI modelu, avšak pro přenos packetů nepoužívá IP protokol. Každý router, jak vyplývá z podstaty link-state algoritmů, si je vědom všech dosažitelných zařízení, pro které má vypočítánu cestu, podle které packety forwarduje.

Protokol IS-IS méně zatěžuje síť něž OSPF, proto může zvládnout obsloužit více zařízení, tedy může obsloužit obsáhlejší sítě něž protokol OSPF. Další hlavní výhodou je tohoto protokolu, je fakt, že je mu jedno, jaký ty adres routuje. Díky tomu je např. snadný přechod mezi IPv4 a IPv6. Pokud opět chceme srovnat tento protokol s protokolem OSPF, IPv6 je v OSPF podporována až od verze 3.

Typy routerů

U IS-IS máme pouze jeden typ routerů. Designated Router, volí se podle priority (0-127), potom vyšší SNPA (Subnetwork Point of Attachment - MAC v LAN nebo DLCI ve Frame Relay), pak system id v NSAP, po změně dochází hned k přepočítání (nové volbě)

Základní konfigurace

```
ROUTER(config)#router isis 0   // pokud máme více procesů specifikujeme je pomocí tagu, default 0

ROUTER(config-router)#net 49.1234.1111.1111.1111.00 // nastavení adresy routeru
ROUTER(config-router)#is-type level-1 // nastavení instanci na router L1 (level-1-2, level-2-only)
ROUTER(config-router)#redistribute eigrp 100 level-1 metric 50  // propojení z ISIS do EIGRP
ROUTER(config-router)#redistribute isis level-1 metric 512 10 255 1 1500 // propojení z EIGRP do ISIS

ROUTER(config-if)#ip router isis            // zapnutí pro interface, defaultně zapne Level 1-2
ROUTER(config-if)#isis circuit-type level-1 // pouze L1 nebo level-2-only pouze L2, nebo level-1-2
ROUTER(config-if)#isis metric 35 level-2    // nastavení metriky pro L2

ROUTER#show clns neighbors     // zobrazí System ID všech známých (sousedních) ISIS routerů (IS/ES)
ROUTER#show clns interface e0  // podrobnosti o routování na Intel., pokud je nenulové Circuit ID, tak je DIS
ROUTER#show isis database      // zobrazí routovací info o L1/L2 LSP
ROUTER#show isis route         // zobrazí info o L1 routách
ROUTER#show ip route isis      // ISIS routy L1/L2, su - summary route (když vytvoříme summary route, tak cisco automaticky vytvoří Null 0 interface a napojí na něj summary
```

Border gateway protokol (BGP)

Tento protokol se řadí jako jediný z popisovaných do kategorie externích routovací protokolů. Slouží tedy k routování mezi systémy (sítěmi), které nespravují stejné organizace. Byl vytvořen, aby nahradil starší protokol EGP Je postaven na základě Path Vector Protokolu, který nesmí obsahovat smyčky a udává posloupnost AS k cílové síti. Pro komunikaci se používá protokol TCP na portu 179

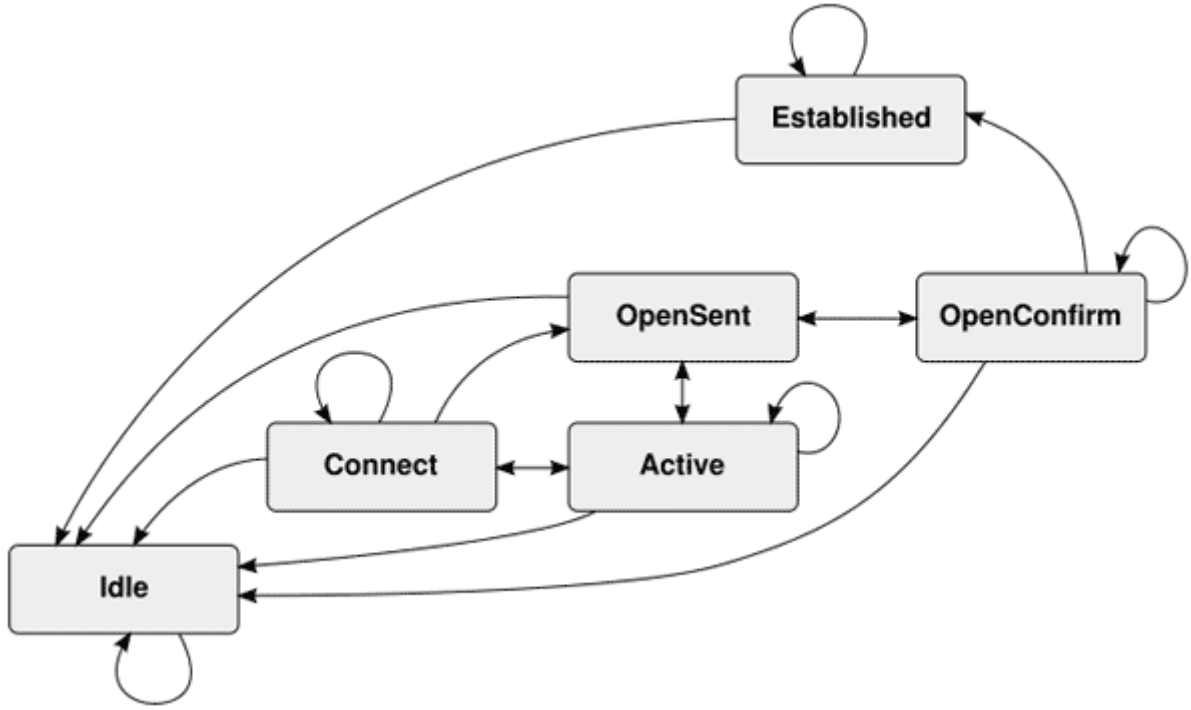
Směrování mezi autonomními systémy má charakteristické požadavky, které se nevyskytují v interním směrování. Směrovací tabulky obsahují stovky tisíc záznamů, nejdůležitějším kritériem nebývá vzdálenost, ale posuzují se nastavitelné parametry zohledňující například cenu a dodatečná

pravidla aplikovaná v závislosti na zdroji, cíli, seznamu tranzitních autonomních systémů a dalších atributech.

Vzhledem k velkému počtu záznamu se v případě změn v topologii vyměňují pouze informace o změnách, nikoliv celé směrovací tabulky jako je tomu v případě protokolu.

Stavy komunikace

- **Idle** - odmítá spojení, připravuje se na vysílání, přejde do Connect, nastavuje se ručně nebo sem padá po chybě
- **Connect** - vytváří spojení se sousedem, odešle BGP OPEN, přejde do OpenSent
- **Active** - přišlo spojení od souseda (BGP OPEN) , přejde do OpenSent
- **OpenSent** - čeká na OPEN od souseda, tu analyzuje určuje jestli patří do stejné AS a jestli je validní, po přijetí odešle KEEPALIVE
- **OpenConfirm** - čeká na KEEPALIVE od souseda
- **Established** - bylo vytvořeno obousměrné spojení, začne posílat UPDATE a KEEPALIVE



Základní konfigurace

```
ROUTER(config)#router bgp 300              // aktivuje BGP, 300 je číslo AS

ROUTER(config-router)#neighbor 170.10.20.1 remote-as 1005 //určí sousedy, s kterými se vytvoří
spojení, v daném AS
ROUTER(config-router)#neighbor 170.10.20.1 next-hop-self  // nastaví danou adresu jako next
hop
ROUTER(config-router)#neighbor 170.10.20.1 send-community // odešle community atributy
sousedovi
ROUTER(config-router)#neighbor 170.10.20.1 update-source loopback 1 // jako zdrojový interface
nastavíme loopback, pro IBGP můžeme chtít, aby spojení stále běželo a nezáleží přes který
interface, tak použijeme adresu loopbacku (ten nikdy nejde down)
ROUTER(config-router)#neighbor 170.10.20.1 route-reflector-client   // nastaví, že tento
router je reflector, a určí jeho klienta
ROUTER(config-router)#no synchronization     // vypne synchronizaci
ROUTER(config-router)#bgp always-compare-med // donutí router porovnávat metriky cest z jiných
AS

ROUTER#clear ip bgp *    // vymaže BGP tabulky a session, * znamená všechny, jinak se zadá IP
```

Dělení protokolů interní/externí

Protokoly můžeme rozdělit, jak již bylo řečeno úvodem podle pole jejich působnosti na dvě základní kategorie.

Interní - Interior gateway protokoly

Tyto jsou určeny pro sítě, které se nacházejí pod administrativní správou jedné organizace. Jejich účelem je hledání nejoptimálnější trasy v rámci dané sítě a jejich metrik.

Do této kategorie patří: RIP, RIP2, IGRP, EIGRP, OSPF, IS-IS

Externí – Exterior gateway protokoly

Jsou designovány k dynamickému propojování sítí, které se nacházejí pod administrativní správou různých organizací. Typicky jsou to sítě různých poskytovatelů internetových připojení.

Sem patří protokoly: BGP, EGP3

Srovnání interních směrovacích protokolů

Na závěr interních směrovacích protokolů uvádím stručný přehled jejich základní charakteristiky.

Protokol	RIP	RIP v2	OSPF	IGRP	EIGRP
typ	distance-vector	distance-vector	link-state	distance-vector	hybrid
konvergence	pomalá	pomalá	rychlá	pomalá	rychlá
zatížení směrovače	malé	malé	velké	malé	malé
zatížení sítě	velké	velké	malé	velké	malé
VLSM	ne	ano	ano	ne	ano
metrika	počet hopů	počet hopů	cena	složená	složená
sumarizace	automatická, podle třídy IP adresy	automatická, je možné vypnout	ruční	automatická, podle třídy IP adresy	automatická, je možné vypnout
autentikace	ne	ano	ano	ne	ano
proprietární	ne	ne	ne	ano	ano

Zdroje

http://netacad.fit.vutbr.cz/texty/ccna-moduly/ccna2-6.pdf

http://www.ivasp.info/pages/pocitacove-site-4/pocitacove-site-4-obsah.php

http://en.wikipedia.org/wiki/Enhanced_Interior_Gateway_Routing_Protocol

http://www.samuraj-cz.com/clanky-kategorie/site/

http://www.mathiaz.com/index.php?n=Routage.Routage