

Směrování -RIP

Základy směrování

- ❑ Předpoklady:
 - Mějme směrovač X
 - Směrovač nemůže znát topologii celé sítě
 - X potřebuje určit směrovač pro přístup k ostatním subsítím v Internetu
 - Tato informace je uložena do směrovací tabulky směrovače
- ❑ Hlavní problémy směrování
 - Změny topologie ovlivňují rychlost konvergence a stabilitu
 - Rozšiřitelnost (škálovatelnost) velkého množství propojených sítí, směrovačů a linek
 - Která cesta je nejlepší?
 - Minimální počet mezilehlých uzlů
 - Minimální zpoždění
 - Maximální propustnost

2

Směrování kontra posílání

- ❑ Směrování(routing): proces vytváření směrovacích tabulek v každém směrovači
- ❑ Posílání (forwardování): zjištění cílové adresy paketu a poslání paketu na vybrané rozhraní směrovače
- ❑ Posílání vyžaduje přístup k lokální směrovací tabulce
- ❑ Někdy se vytváří tabulka pro forwardování, která se pak liší od směrovací tabulky
 - Forwardovací tabulka: optimalizovaná pro vyhledání cíle a posílání
 - Směrovací tabulka: optimalizovaná pro změny směrování, změny topologie

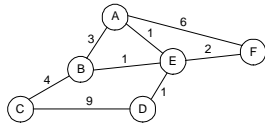
Net #	Next hop	Link Cost
10	171.69.245.10	2

Net #	Interface	MAC Address
10	if1	8:0:2b:e4:b:1:2

3

Směrování jako problém teorie grafů

- ❑ Uzly: směrovače jedné administrativní domény (vnitřní směrování), nebo různých sítí (vnější směrování)
- ❑ Hrany: vzájemné propojení směrovačů
- ❑ Ohodnocení hran: podle vzdálenosti, kapacity, zpoždění, ...
- ❑ Cíl: nalezení minimální cesty mezi libovolnými dvěma uzly
- ❑ Problém: nalezení minimální cesty decentralizovanou (nebo centralizovanou) metodou
- ❑ Rychlé a robustní reakce na změnu topologie



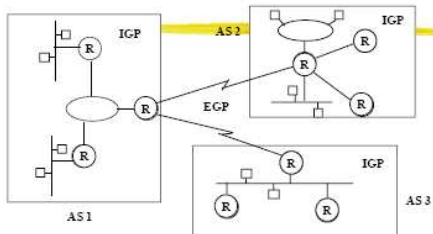
4

Typy algoritmů směrování

- ❑ „Statické“ směrování
 - Ruční nastavení směrovací tabulky
- ❑ „Dynamické“ směrování
 - Adaptivní algoritmy nastavení směrovací tabulky
 - Interní směrování (RIP, OSPF)
 - Externí směrování (BGP)
- ❑ Směrování podle vektoru vzdáleností (Distance Vector Algorithm)
 - Šíření obsahu směrovací tabulky sousedním směrovačům
- ❑ Směrování podle stavu linek (Link State Algorithm)
 - Šíření informace o stavu linek (hran grafu) sousedním směrovačům
- ❑ Hybridní směrování

5

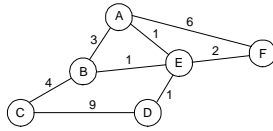
Propojení tří autonomních oblastí



6

Směrování podle vektoru vzdáleností

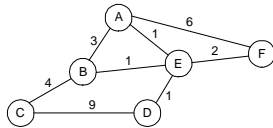
- ❑ Používá Bellman-Fordův algoritmus (dynamické programování)
- ❑ Vektor vzdáleností pro uzel X: minimální vzdálenost z uzlu X do všech ostatních uzlů
 - Např. pro uzel A je to {2,6,2,1,3}
- ❑ Každý uzel provádí následující 3 operace souběžně
 - Posílá vektor vzdáleností svým sousedům
 - Přijímá vektor vzdáleností od svých sousedů
 - Počítá nové vzdálenosti na základě přijatých vektorů
 - $distance(X,Z) = \min \{ distance(X,Y) + distance(Y,Z) \}$ pro všechny sousední uzly Y



7

Směrování podle vektoru vzdáleností

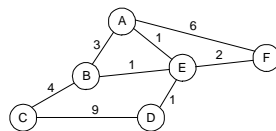
- ❑ Počáteční vektor vzdáleností vychází pouze ze znalosti vzdáleností k sousedním uzlům
 - Např. pro uzel A {3,∞,∞,1,6}
- ❑ Lokální výměna globální informace o dostupnosti
- ❑ Vektory vzdáleností jsou posílány
 - Periodicky (30s)
 - Při změně položky ve směrovací tabulce
- ❑ Uzel detekuje chyby uzlů a linek periodickou výměnou „Hello“ paketů nebo výměnou směrovací informace



8

Počáteční nastavení směrování

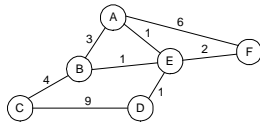
uzel	A	B	C	D	E	F
A	0	3	∞	∞	1	6
B	3	0	4	∞	1	∞
C	∞	4	0	9	∞	∞
D	∞	∞	9	0	1	∞
E	1	1	∞	1	0	2
F	6	∞	∞	∞	2	0



9

Počáteční finální směrovací tabulka uzlu A

Cíl (od A)	cena	Násl. uzel
B	3	B
C	∞	-
D	∞	-
E	1	E
F	6	F

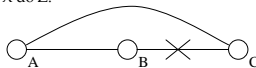


Cíl (od A)	cena	Násl. uzel
B	2	E
C	6	E
D	2	E
E	1	E
F	3	E

10

Změny topologie

- ❑ Problém „čítání do nekonečna“
- ❑ Možná řešení
 - Omezení horní meze pro čítání (maximální vzdálenost)
 - Split horizon (rozštěpený obzor)
 - X nesmí poslat do uzlu Y svou vzdálenost k uzlu Z, je-li uzel Y ve směru z X do Z.
 - Split horizon with poisoned reverse (rozštěpený obzor s otráveným zpětným směrem)
 - X posílá do uzlu Z jeho vzdálenost k uzlu Z je ∞ , je-li uzel Y ve směru z X do Z.



11

Změny topologie

- ❑ Bohužel, žádné z těchto řešení nezabrání cyklům
- ❑ Možné řešení: Před generováním a posíláním vektoru vzdáleností, který upravuje konektivitu k jinému uzlu, počkat nějakou dobu na informace o konektivité k tomuto uzlu od jiných uzlů
 - Může významně prodloužit dobu konvergence.
- ❑ Příčinou potíží je asynchronní výměna stavových informací
- ❑ Není zaručeno, že je ve všech uzlech konzistentní směrovací informace
- ❑ Urychlení konvergence: triggered update (okamžité spuštění opravy)

12

Routing Information Protocol (RIP)

- Implementace algoritmu „směrování podle vektoru vzdáleností“
- RFC 1058, UDP port 520
- Všechny ohodnocení linek jsou nastaveny na 1 (počet mezilehlých uzlů)
- Vektory vzdáleností vyměňovány každých 30 s
- Maximální možné ohodnocení je 15, 16 je nekonečno
- Omezení cyklů pomocí algoritmu „Split horizon with poisoned reverse“ (rozštěpený obzor s otráveným zpětným směrem)
- Urychlení konvergence pomocí „Triggered update“ (okamžitá oprava)
- Někdy se používá také „Hold down“ (pozdržení odeslání informace o výpadku uzlu nebo linky)
- Detekce výpadku uzlu nebo linky po 180 s
- Výmaz z nedostupnosti ze směrovací tabulky po 120 s
- Max. velikost datagramu 512 slabik – 25 cest

13

Záhlaví RIP

0	8	16	31
Command	Version	Must be zero	
Family of net 1		Address of net 1	
Address of net 1			
Distance to net 1			
Family of net 2		Address of net 2	
Address of net 2			
Distance to net 2			

14

Algoritmus opravy směrovací tabulky, RIP2

- Pokud je nově vypočtená vzdálenost
 - Menší – opravit
 - Stejná – nic neměnit
 - Horší
 - Na základě zprávy ze směrovače, který je sousední pro původní směrování – opravit (zhoršení ocenění)
 - Na základě zprávy z jiného směrovače – nic neměnit
- Aktivní režim (směrovač)
- Pasivní režim (hostitelský systém)
- RIP 2
 - Posílání subsíťové masky a adresy následujícího uzlu
 - Podpora skupinového doručování – snížení zátěže
 - Podpora ověřování pravosti - heslo

15
