Směrování, směrovací algoritmy a protokoly

Petr Grygárek

Sítě s přepínáním okruhů a s přepínáním paketů (WAN)

Sítě s přepínáním okruhů

- historicky starší (vyvinuly se z telefonních sítí)
- explicitní žádost síti o vytvoření/zrušení okruhu koncovým zařízením
  - síť může vracet identifikátor okruhu (pokud může koncové zařízení lze vytvořit více okruhů)
- přenosová kapacita rezervována po celou dobu existence okruhu
  - jediná cesta – odpadá nebezpečí přeházení pořadí
- výhodné pro uživatele - garance kvality služby
- neekonomické pro síť (hlavně u přenosu dat)
  - shlukový charakter datových přenosů, nevyužité časové úseky
- při výpadku a rozpadu okruhu nutno žádat síť o nové vytvoření okruhu
  - síť musí najít cestu, která obejde vypadlé prvky
  - procedura nového navázání spojení mezi koncovými zařízeními po zřízení okruhu může být časově náročná
    - (handshake analogových modemů)

Sítě s přepínáním paketů

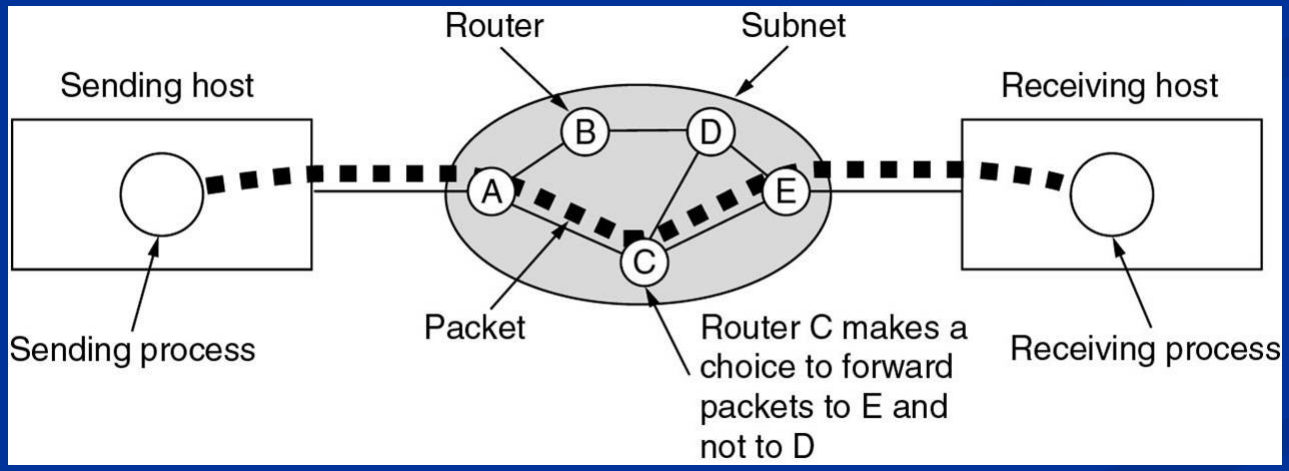
- Vyvinuto v rámci vojenského projektu ARPA
  - cílem odolnost proti výpadkům, rychlé zotavení
  - vyvinulo se do dnešního Internetu
- Polygonální struktura s redundandními spoji založená na směrovačích
- Datová jednotka – paket – se předává mezi směrovači nezávisle na ostatních
  - obsahuje identifikaci příjemce
  - každý paket může jít jinou cestou
    - bezpečné proti výpadkům spojů a směrovačů
    - nebezpečí změny pořadí přenášených paketů (příp. duplikace)
  - paket může být pozdržen v paměti směrovače na uživatelem neovlivnitelnou dobu
- Předávání paketu skok po skoku („hop by hop“)

Virtuální kanál

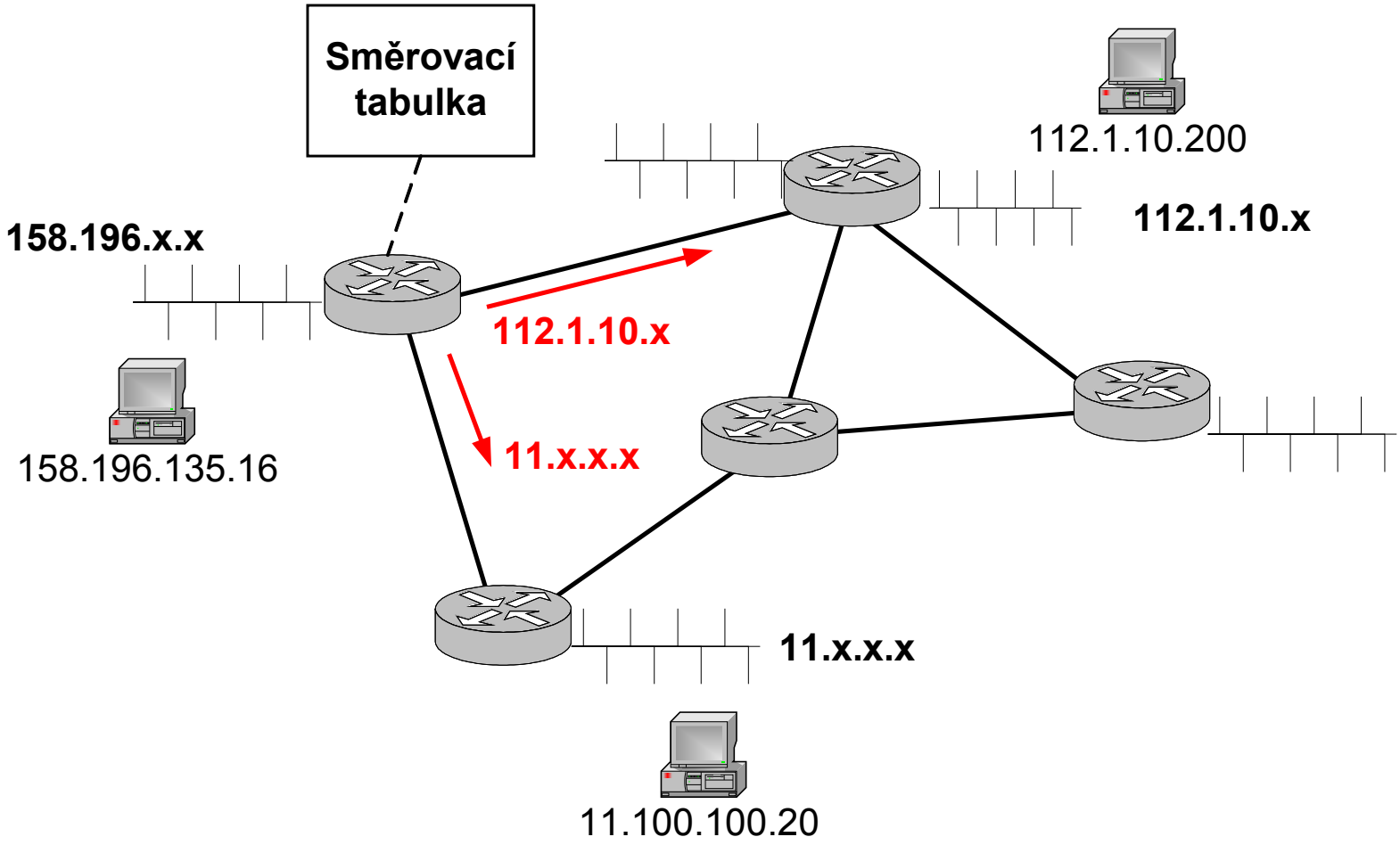
- Vytvoření logického okruhu nad sítí s přepínáním paketů
  - (kompromis)
- Virtuální okruh zřízen (a zrušen) na základě žádosti koncového zařízení
- Hledání cesty probíhá jen při vytváření okruhu, směrovače si výsledek poznačí – urychlení
  - Při zpracování dalších paketů se pracuje s identifikátorem virtuálního okruhu
    - Nesen v každém paketu
    - Obdrží koncové zařízení od sítě po vytvoření okruhu
    - Je klíčem do přepínací tabulky směrovacích prvků
- Všechny pakety jdou stejnou cestou – nemohou se přeházet do nesprávného pořadí

Směrování

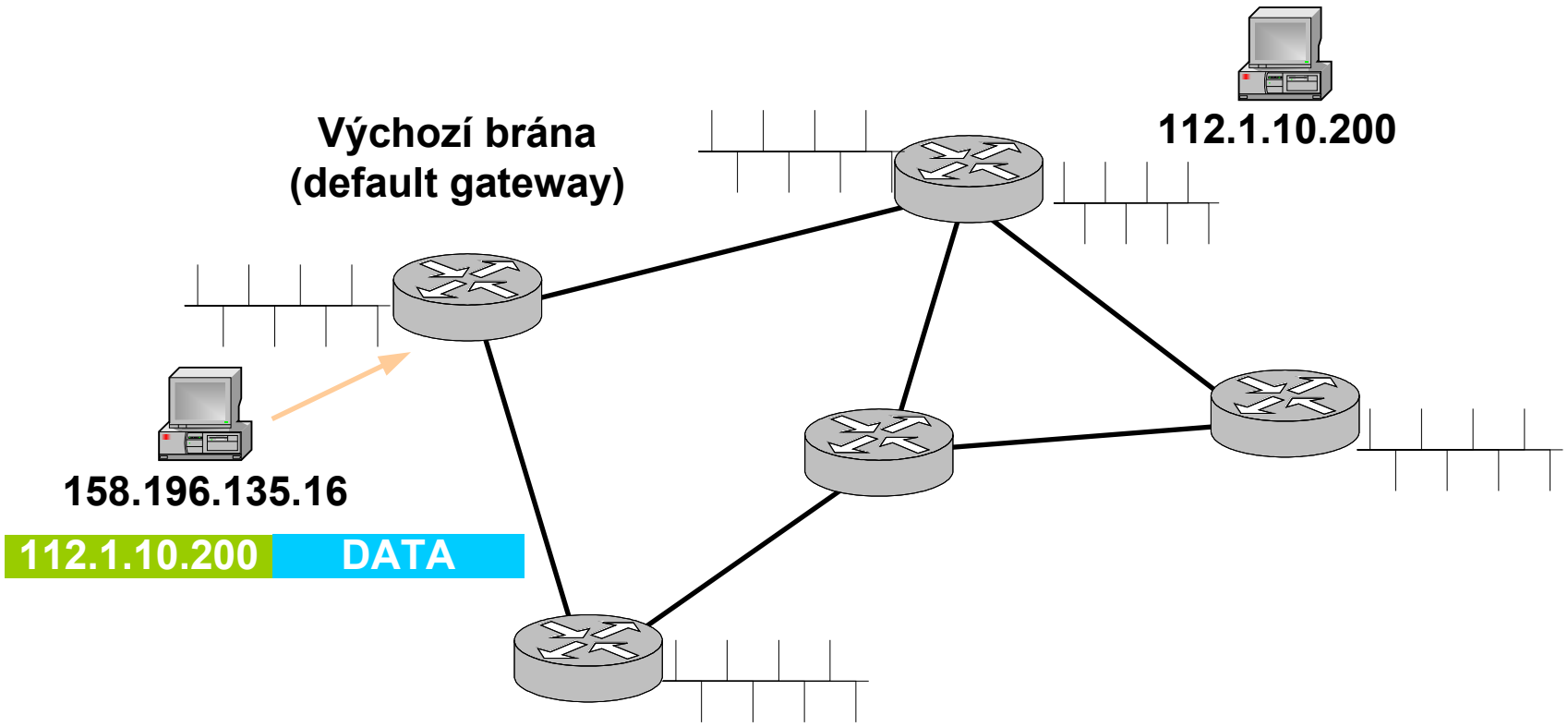
- Hledání cesty sítí
  - ve spojově orientovaných sítích při vytváření spojení
    - nastavování spojovacích polí přepínacích prvků na cestě
    - budování přepínacích tabulek virtuálního kanálu
  - v sítích s přepínáním paketů (obvykle obecně polygonálních a s alternativními cestami) při přenosu každého jednotlivého paketu = > **každý paket může jít jinou cestou**



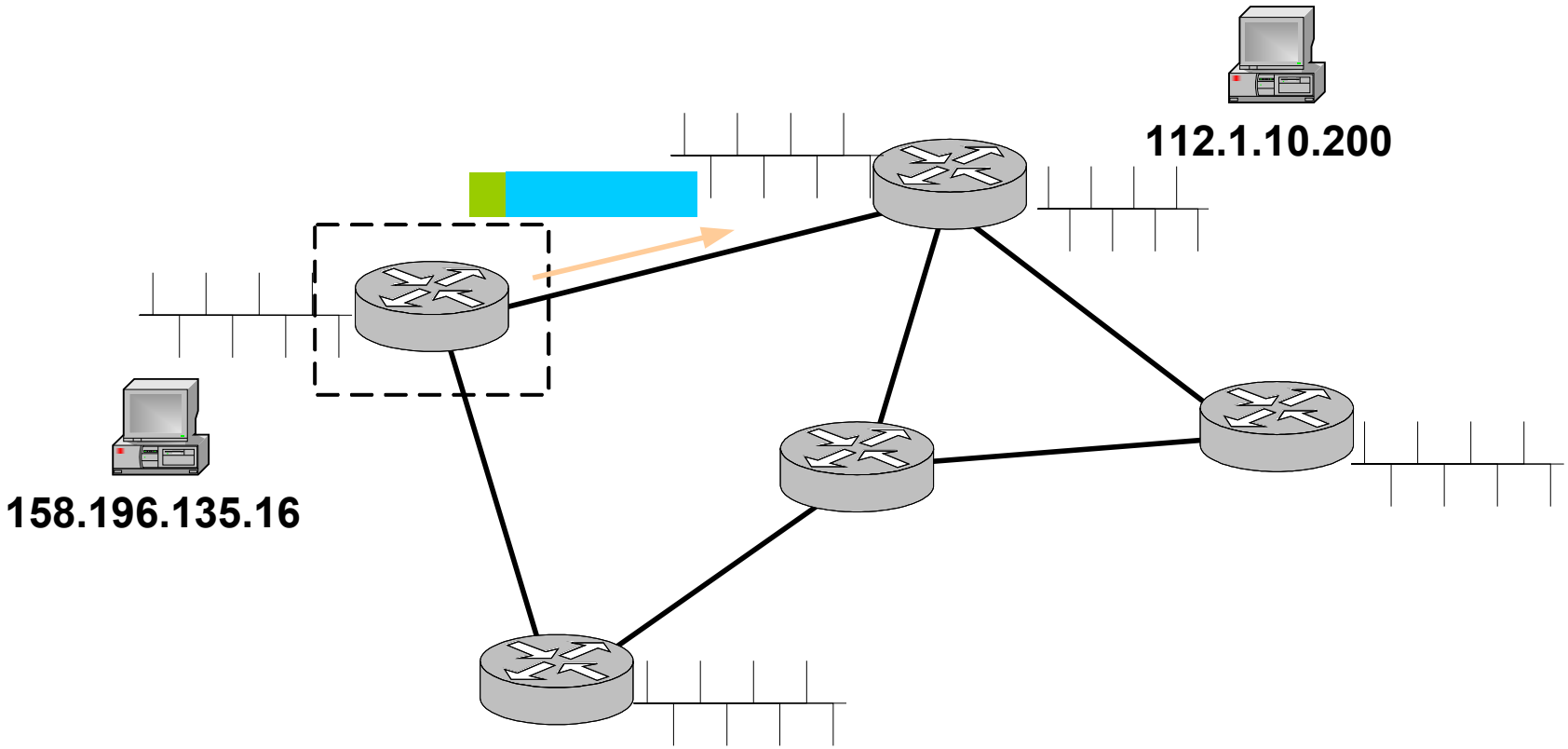
Směrování v síti s přepínáním paketů (1)



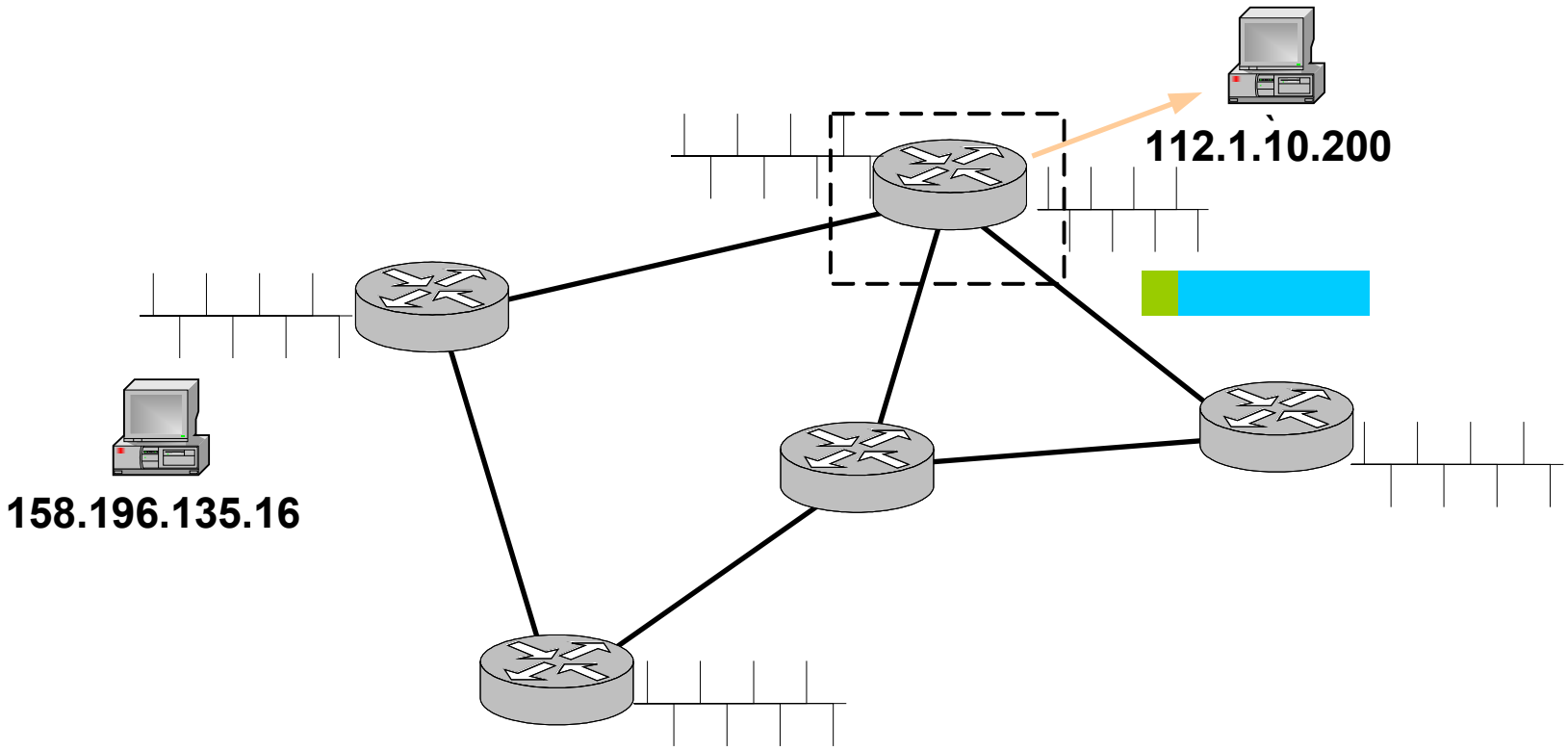
Směrování v síti s přepínáním paketů (2)



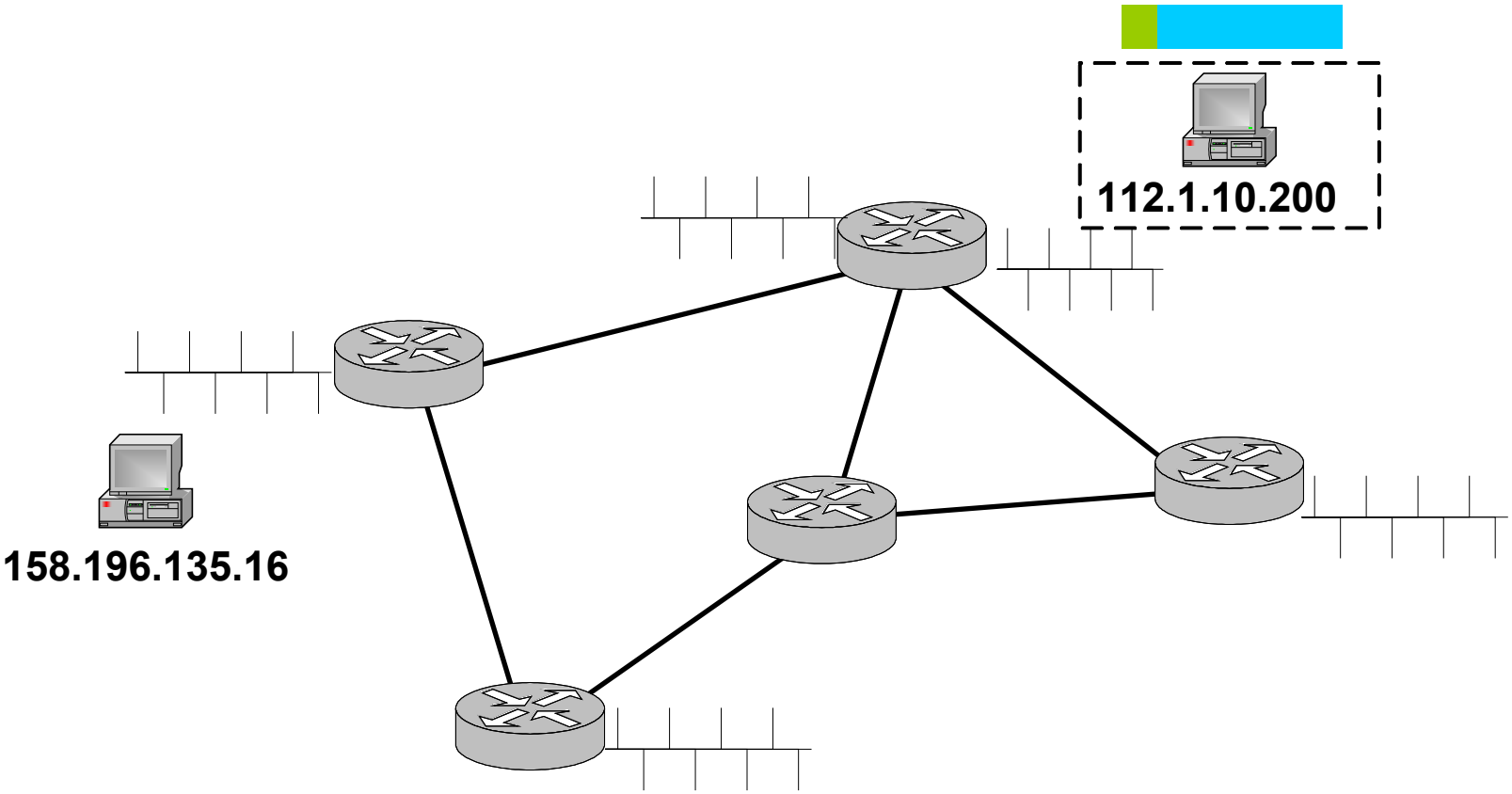
Směrování v síti s přepínáním paketů (3)



Směrování v síti s přepínáním paketů (4)



Směrování v síti s přepínáním paketů (5)



Směrovací algoritmy

Směrovací algoritmus

- část software 3. vrstvy OSI RM rozhodující, kterým rozhraním se má odeslat příchozí paket nebo kudy zřídit požadovaný (virtuální) okruh
- rozhoduje na základě staticky konfigurovaných informací nebo informací od **směrovacího protokolu**
- implementován obvykle s použitím **směrovací tabulky**

Poznámka:

- Tvorba (úprava) směrovacích tabulek pomocí směrovacího protokolu a směrování podle těchto tabulek může probíhat současně (a obvykle tomu tak je).

Požadované vlastnosti směrovacího algoritmu (směrovacího protokolu)

- jednoduchost
- robustnost
- stabilita
- férovost
- optimalita

Některé v protikladu, rozhoduje se, které upřednostnit

Směrovací tabulka

- Záznamy ve tvaru

  <cílová adresa(+maska), výstupní rozhraní/ next_hop, metrika>

- jako cílová adresa může být uvedena síť, podsíť nebo uzel
  - v prostředí IP jsou adresy sítí různé délky
- v prostředí IP se použije cesta, která se shoduje s adresou cíle v paketu na co největší počet míst

  => **nutnost projít vždy celou tabulku** => časově náročný proces

Implicitní cesta (default route)

- typicky pro sítě připojené jediným rozhraním k hierarchicky vyšší síti
- veškeré pakety, pro jejichž cílovou adresu neexistuje položka ve směrovací tabulce, se posílají na default route
- smyslem snížení počtu záznamů ve směrovací tabulce
- next-hop se konfiguruje na přilehlé rozhraní sousedního směrovače do vyšší hierarchické úrovně (typicky směrovač ISP)

Typy a přístupy ke směrování

Přístupy ke směrování

- centralizované
- distribuované
- (izolované)

Typy směrování

- *Neadaptivní směrování* – statické
- *Adaptivní (dynamické) směrování* - mění se podle:
  - okamžité topologie sítě
  - okamžitého zatížení jednotlivých částí sítě
    - (zde ale nebezpečí rozkmitání směrování, prakticky používáno zřídka)

Centralizované a distribuované směrování

Centralizované směrování

- v síti existuje centrální prvek RCC (Routing Control Center), který shromažďuje informace o okolí od všech směrovačů, kombinuje z nich topologii sítě, počítá směrovací tabulky pro všechny směrovače a předává jim je
- jednotlivé směrovače posílají periodicky do RCC stavové informace (seznam "živých" sousedů, délky front na jednotlivých rozhraních, celková zpracovávaná zátěž, ...)
- problém s distribucí tabulek od RCC do jednotlivých směrovačů - distribuce postupná, směrování v době distribuce nekonzistentní (staré a nové verze tabulek současně)

Dnes není prakticky používáno.

Distribuované směrování

- každý směrovač zná "vzdálenost" (ceny linek) ke všem svým sousedům a stav těchto linek (funkčnost, přenosovou rychlost, okamžitou zátěž, ...)
- každý směrovač si vyměňuje své informace o směrování s jinými směrovači (prostřednictvím sousedů)
- ze získaných informací si směrovač vytvoří směrovací tabulku

Používá se v mnoha variantách v Internetu i v intranetech

Izolované směrování

Založeno pouze na lokálně dostupné informaci

- **Horký brambor** - paket se vkládá do nejkratší výstupní fronty (cílem je se jej co nejrychleji zbavit)
- **Backward learning** - do paketů se vkládá identifikace zdrojové sítě a počítadlo, při průchodu každým směrovačem se počítadlo zvýší - směrovač z příchozího paketu zjistí, jak daleko je přes dané rozhraní ke zdrojové síti
- **Záplavové směrování (flooding)** - paket se vyšle na všechny linky mimo té, ze které přišel. V paketu je počítadlo přeskoků, likvidace paketu při dosažení limitu jako opatření proti zahlcení.
  - vždy vybírá nejkratší cestu (resp. všechny nejkratší cesty paralelně)
  - použití pro spolehlivé a rychlé rozšíření informace v síti s neznámou a příp. proměnnou topologií - vojenské aplikace
  - používá se jako "normál" pro srovnání ostatních algoritmů

Pouze pro speciální účely.

Statické a dynamické směrování

Statické směrování

- směrovací tabulky v jednotlivých směrovačích konfigurovány "ručně" - pracnější
- odpadá režie směrovacích protokolů
  - zabraná šířka pásma, čas na zpracování
- bezpečnější (omezení možnosti generování falešných směrovacích informací, odposlouchání topologie sítě)
- při výpadku linky nutný ruční zásah
- použitelné, pokud se topologie příliš často nemění (vlivem výpadků a modifikací sítě)

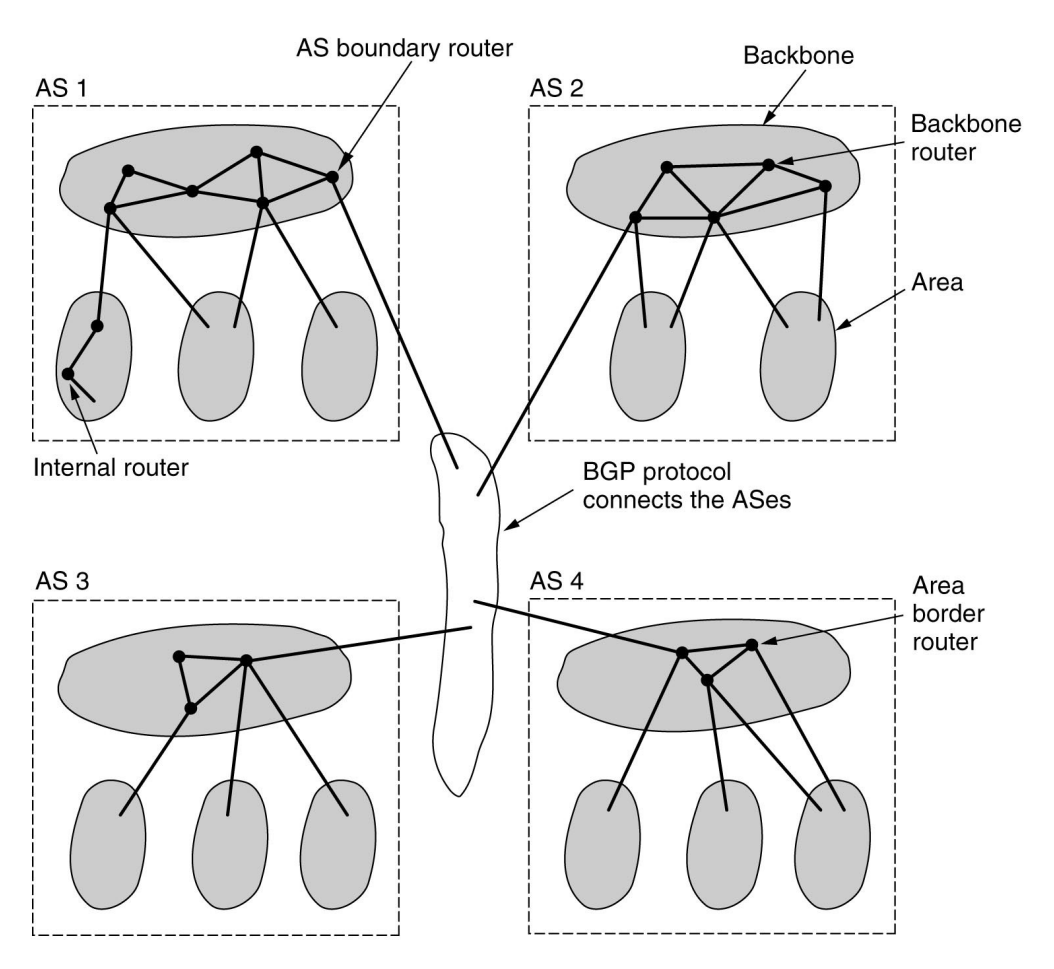
V intranetech používáno až překvapivě často

Dynamické směrování

- automaticky reaguje na poměry v síti (topologie, zátěž, ...)
- nutnost provozu směrovacích protokolů
- užitečné při častých změnách (příp. i obecně neznámé) topologie sítě (typicky v Internetu)

V praxi často používána kombinace statického a dynamického směrování, staticky nakonfigurované cesty mají obvykle přednost.

Hierarchické směrování



Hierarchické směrování - princip

- rozdělení sítě do hierarchicky organizovaných celků
  - výhodné i pro administrativní rozdělení kompetencí při správě sítě
- směrovače v jednotlivých celcích znají jen topologii svého celku, cestu do vyššího celku a seznamy sítí v hierarchicky nižších celcích
  - (nikoli jejich vnitřní strukturu)

Smyslem omezení rozsahu směrovacích tabulek (agregace a implicitní cesta)

Směrovací algoritmy

Čím se směrovací algoritmy liší ?

- použitá metrika
- úroveň informovanosti směrovačů o topologii sitě
- mechanismus šíření směrovací informace
  - (výměna mezi sousedy, flooding všem, ...)
- technická realizace zasílání směrovacích informací
  - (broadcast/multicast, perioda)

Cílem krátká doba konvergence

= doba do nalezení alternativních cest a stabilizace směrovacích tabulek při náběhu/výpadku linky nebo směrovače

Základní dělení směrovacích algoritmů

- Třída Distance Vector (vektory vzdáleností)
  - jednoduchá implementace, historicky starší
- Třída Link State (stavy spojů)
  - složitější implementace, ale rychlejší konvergence a chování ve „speciálních situacích“

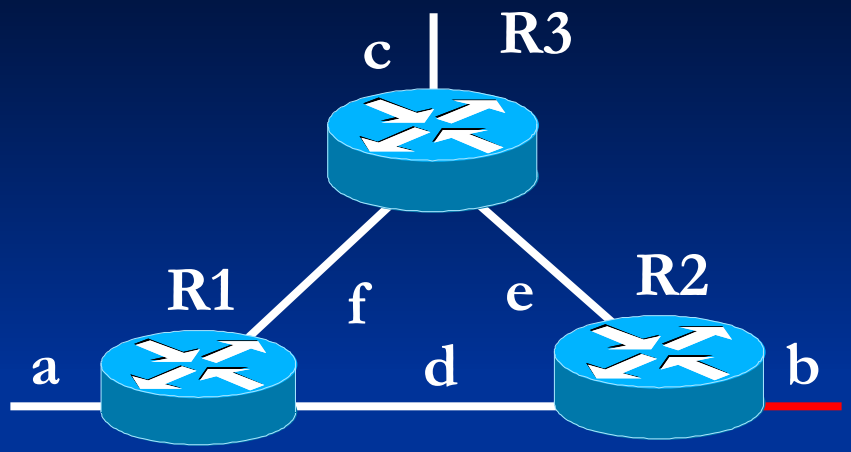
Algoritmy vektorů vzdáleností (distance vector algorithms - DVA)

Základní princip DVA

Směrovače neznají topologii sítě, pouze rozhraní (adresy sousedů), přes která mají posílat pakety do jednotlivých sítí a vzdálenosti k těmto sítím (tzv. **distanční vektory**)

- na začátku směrovací tabulka obsahuje pouze přímo připojené sítě
  - staticky nakonfigurováno administrátorem
- periodické zasílání směrovací tabulky sousedům
- z došlých směrovacích tabulek sousedů (vzdáleností sousedů od jednotlivých sítí) a výběrem nejlepší cesty si směrovač postupně upravuje svou směrovací tabulku
  - pokud cesta nebyla delší dobu sousedem inzerována, ze směrovací tabulky se odstraní

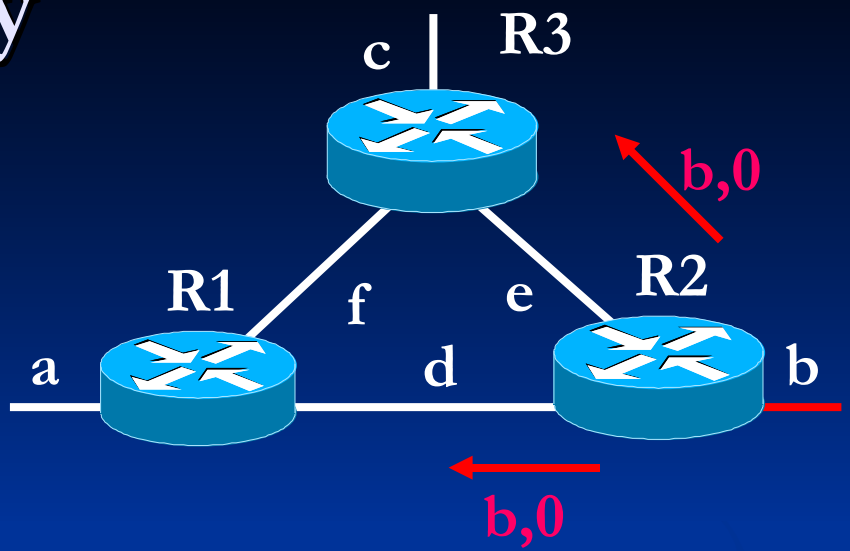
Příklad: šíření cesty do sítě b (1)



Předkonfigurované informace:

```
R1:          R2:          R3:
-----        -----        -----
a 0 -        b 0 -        c 0 -
f 0 -        d 0 -        e 0 -
d 0 -        e 0 -        f 0 -
```

Příklad: šíření cesty do sítě b (2)

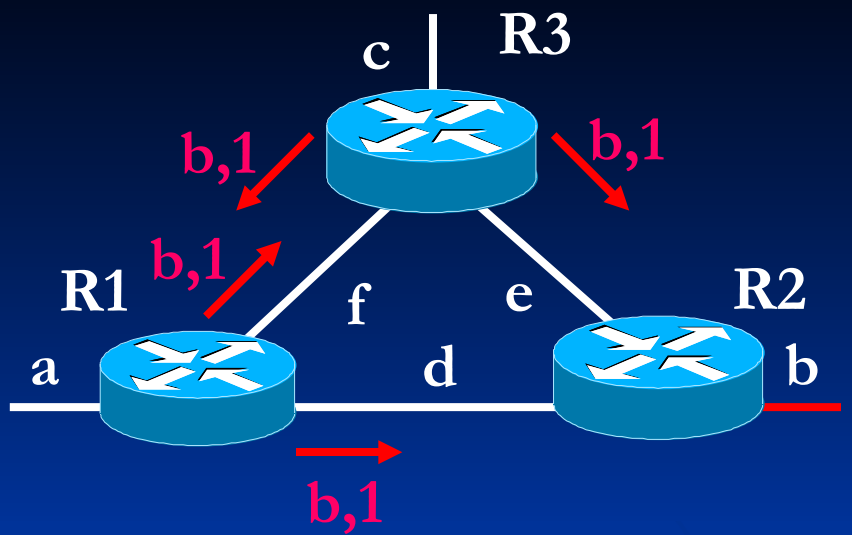


R2 pošle směrovací tabulku sousedům, ti zkombinují se svými:

```
R1:          R2:          R3:
-----        -----        -----
a 0 -        b 0 -        c 0 -
f 0 -        d 0 -        e 0 -
d 0 -        e 0 -        f 0 -
b 1 R2                    b 1 R2
```

Příklad: šíření cesty do sítě b (3)

R3 a R2 pošlou své směrovací tabulky sousedům, ti zkombinují se svými:




```
R1:           R2:           R3:
-----         -----         -----
a 0 -         b 0 -         c 0 -
f 0 -         d 0 -         e 0 -
d 0 -         e 0 -         f 0 -
b 1 R2                      b 1 R2
b 2 R3        nechci ! (mám lepší)   b 2 R1   nechci ! (mám lepší)
```

Kombinování směrovacích tabulek z přijatých informací

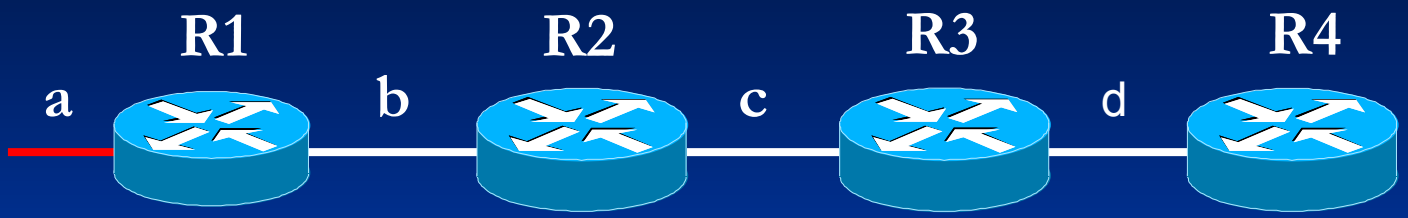
- Nabízena cesta do sítě, kterou směrovač nemá:
  => přidat do tabulky
- Nabízena cesta do sítě, kterou směrovač má, ale má ji s horší metrikou
  => ve směrovací tabulce změnit na nabízenou cestu
  - Pokud je nabízena cesta se stejnou metrikou, jako směrovač už má, ignoruje se

Výjimka:
Pokud směrovač, který je dalším přeskokem některé cesty, začne cestu inzerovat s horší metrikou, sousední směrovač si metriku u této cestu musí zhoršit

Vlastnosti metod DVA

- metrikou je počet "přeskoků" (hop count) na cestě mezi zdrojem a cílem
  - nezohledňuje parametry jednotlivých linek
    - (přenosovou rychlost, zpoždění, okamžitou zátěž, ...)
- pomalá konvergence při změnách topologie
  - o změně se informuje až při příštím periodickém broadcastu směrovací tabulky
- zátěž od broadcastu celých směrovacích tabulek
- příliš "optimistické" - směrovač se rychle učí dobré cesty, ale špatně zapomíná při výpadcích
  - čekání na timeout cesty, která přestala být inzerována
  - žádný směrovač nikdy nemá metriku horší, než minimum z metrik sousedů + 1 => pomalé šíření špatných zpráv

Rychlost konvergence

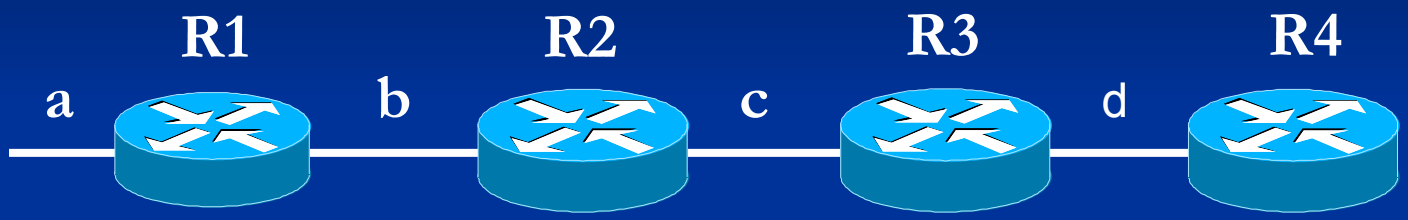


Náběh sítě **a** při periodě posílání směrovací tabulky 30s

- R2 se o síti **a** dozví po (max) 30s
- R3 se o síti **a** dozví po (max) 60s
- R4 se o síti **a** dozví po (max) 90s = 1.5 min

Problém počítání do nekonečna (1)

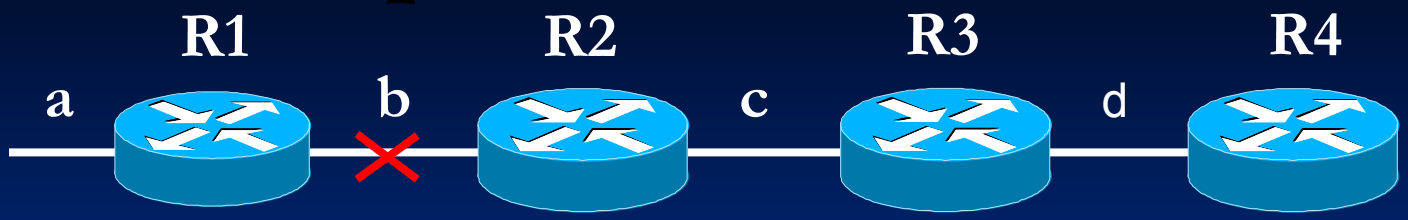
Směrovač, který se dozví od souseda o cestě k nějakému cíli, neví, že po výpadku původní cesty nově nabízená cesta vede přes něj samotného:



Počty přeskoků a sousedi z jednotlivých směrovačů do sítě a:

R2: 1,R1 R3: 2,R2 R4: 3, R3

Problém počítání do nekonečna (2)



Vypadne linka R1-R2

R2: ∞,- R3: 2,R2

R3 oznámí R2, že přes něj je síť **a** za 2 přeskoky

(ale nesdělí, že přes R2 ;-))

R2: 3,R3 R3: 2,R2

R2 oznámí R3, že přes něj je síť **a** za 3 přeskoky

(R3 měl v tabulce cestu za 2 přes R2, ale R2 teď inzeruje jinak, čili si přepíše)

R2: 3,R3 R3: 3, R2

R3 oznámí R2, že přes něj je síť **a** za 3 přeskoky

R2 měl v tabulce cestu za 3 přes R3, ale R3 teď inzeruje jinak, čili si přepíše

R2: 4,R3 R3: 3, R2

=> Postupné zvyšování metriky R2 a R3 do sítě a do nekonečna.

Řešení problému počítání do nekonečna

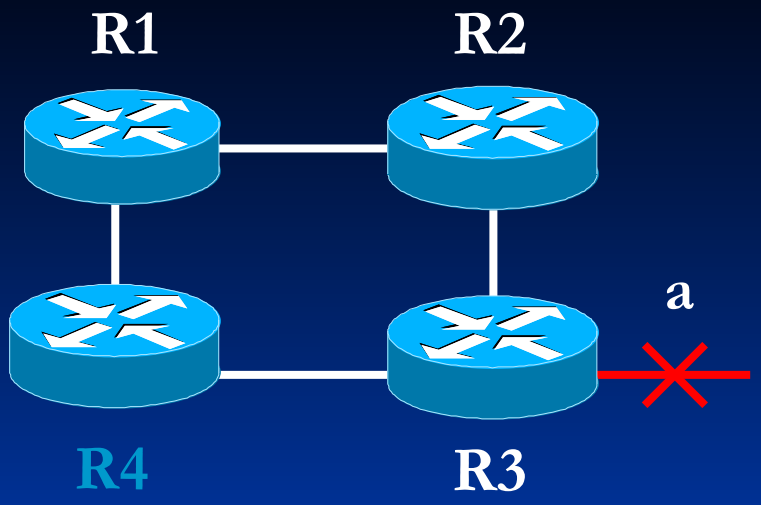
- Nekonečno číselně nahradíme průměrem sítě + 1, položky směrovací tabulky s metrikou nekonečno se nepoužijí
- **Split horizon** - směrovač neposílá informace o síti do toho rozhraní, které sám používá pro dosažení této sítě.

První způsob řeší i cykly přes více směrovačů, které Split horizon nedetekuje, proto se zpravidla kombinuje obojí

Možná vylepšení metod DVA

- **Triggered update** - po výpadku/náběhu přilehlé linky směrovače nebo změně směrovací tabulky po příchodu update od souseda se nečeká na periodu časovače, ale nová směrovací tabulka se sousedům zašle ihned.
- **Poisson reverse** - jako split horizon, ale cesta do sítě se na rozhraní používaném pro dosažení této sítě nejen neposílá, ale posílá s metrikou nekonečno.
- **Hold down** - po ztrátě nejlepší cesty do sítě po dobu časového intervalu (holddown) nepřijímáme cesty do téže sítě od jiných směrovačů. Mohlo by totiž jít o falešné cesty využívající linek, jejichž výpadku (o němž my už víme) si nabízející směrovače dosud nepovšimly.

Hold down - příklad



Aplikace hold down na R4:

- R3 informuje o nedostupnosti sítě **a**
  - (triggered update)
- R1 nabízí cestu do sítě **a** za 3 přeskoky
  - nevěříme mu po dobu holddown timer !
  (možná k němu jen informace o nedostupnosti sítě **a** ještě nedorazila)

Reprezentanti metod DVA

- Routing information protocol (RIP)
  - Velmi starý, stále však často implementovaný v malých sítích
    - hodnota "nekonečna" řešící problém počítaní do nekonečna a v technice Poisson reverse reprezentována číslem 16
  - Jednoduchý na implementaci
  - Prakticky nulové nároky na znalosti správce
- Interior Gateway Routing Protocol (IGRP)
  - Cisco proprietary
  - Lepší metrika, než pouhý počet přeskoků
    - bandwidth, delay, (reliability, load)
  - Není omezení na 16 přeskoků (zvýšeno na 255)
- …

Algoritmy stavů spojů (link state)

Charakteristika algoritmů LSA

- směrování na základě znalosti "stavu" jednotlivých linek sítě (funkčnost, cena)
- směrovače (uzly grafu) znají topologii celé sítě (graf) a ceny jednotlivých linek (ohodnocení hran). Tyto informace udržují v topologické databázi.
  - všechny směrovače mají stejnou topologickou databázi
- každý směrovač počítá strom nejkratších cest ke všem ostatním směrovačům (a k nim připojeným sítím) pomocí Dijkstrova algoritmu
  - na rozdíl od DVA všechny směrovače počítají na základě stejných a úplných dat

Funkce algoritmu LSA

- každý směrovač neustále sleduje stav a funkčnost k němu připojených linek
  - testováním linek k sousedním směrovačům – Hello protokol
- při změně okamžitě šíří informaci o aktuálním stavu svého okolí všem ostatním směrovačům. Ty si ji vloží do topologické databáze
  - šíří se pouze změny (ale do celé sítě) - žádné periodické rozesílání směrovacích tabulek

Okamžitá reakce na změnu stavu linek (výpadek, náběh) => rychlá konvergence

Topologická databáze

- Složena ze záznamů ve tvaru
  - ID směrovače
  - seznam přilehlých linek k sousedním směrovačům, u každé ID souseda
  - Seznam koncových sítí připojených ke směrovači

  K lince vždy připojena i její cena a adresa sítě
- Záznamy generovány jednotlivými směrovači při změnách stavu linek a šířeny do celé sítě
- Ze záznamů topologické databáze lze zkonstruovat graf topologie sítě

Reprezentanti metod LSA

- Open Shortest Path First (OSPF)
  - otevřený (open) standard
  - směrovač nejprve vypočte strom nejkratších cest (shortest path first), pak z něj vytvoří směrovací tabulku
  - podporuje hierarchické směrování (oblasti)
  - dnes jeden z nejpoužívanějších směrovacích protokolů
- IS-IS
  - ISO standard, koncepčně obdobný OSPF

Algoritmy teorie grafů používané při směrování

Směrování: základní algoritmy

- **Dijkstra** - pro zvolený uzel hranově ohodnoceného grafu nalezne strom nejkratších cest do ostatních uzlů. V modifikované variantě základ protokolu OSPF.

- **Floyd** - vychází z matice cen grafu, počítá matici vzdáleností uzlů grafu a úplné směrovací tabulky pro jednotlivé uzly

- **Ford-Fulkerson** - určení nejkratších vzdáleností ze všech uzlů do jednoho společného uzlu. V distribuované variantě základ protokolu RIP.

Směrování v prostředí TCP/IP

Směrovací protokoly v prostředí TCP/IP

Vnitřní (uvnitř autonomního systému):

- Otevřené standardy:
  - RIP (Routing Information Protocol) - distance vector
  - OSPF (Open Shortest Path First) - link state
- Firemní (proprietární)
  - IGRP, EIGRP - Cisco
  - NLSP – Novell

Vnější (mezi autonomními systémy):

- BGP (Border Gateway Protocol) - tzv. path vector

Beztřídní IP adresy a směrovací protokoly

Směrovací protokoly, které nepropagují s adresou sítě i masku (ale spoléhají na třídy) vyžadují

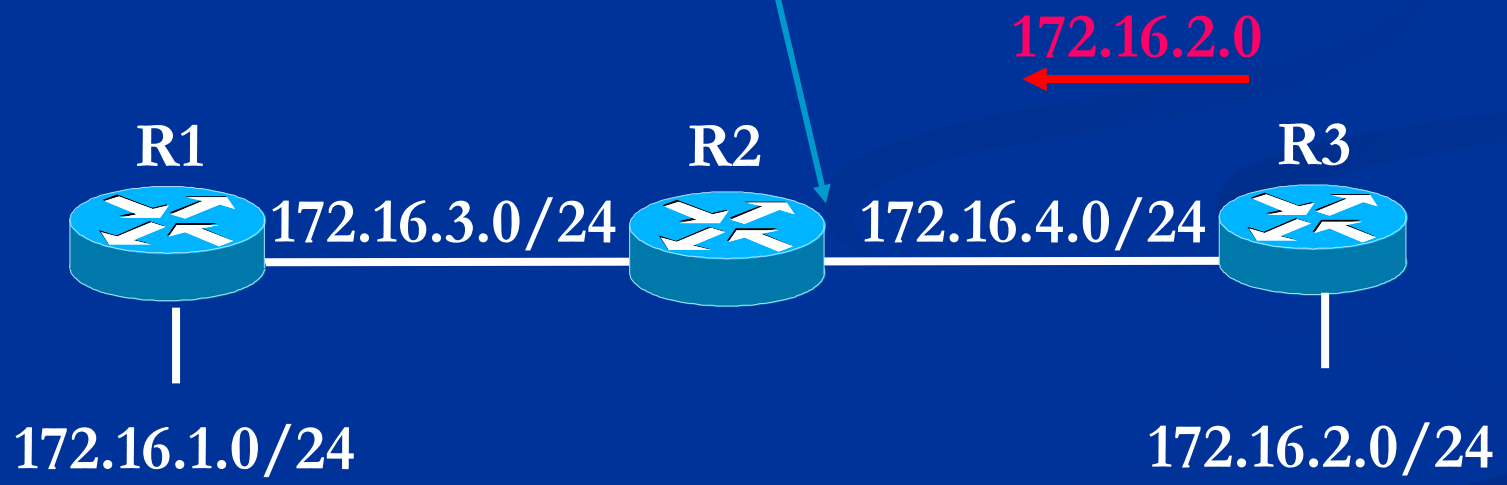
- společnou (konstantní) masku podsítě v rámci jedné (třídní) sítě
- spojitost podsítí (=podsítě téže adresy sítě nesmí být odděleny segmentem jiné sítě).
  - protože na hranici třídní sítě se podsítě agregují na adresu třídní sítě

Problém se starými DV protokoly (RIP, IGRP)

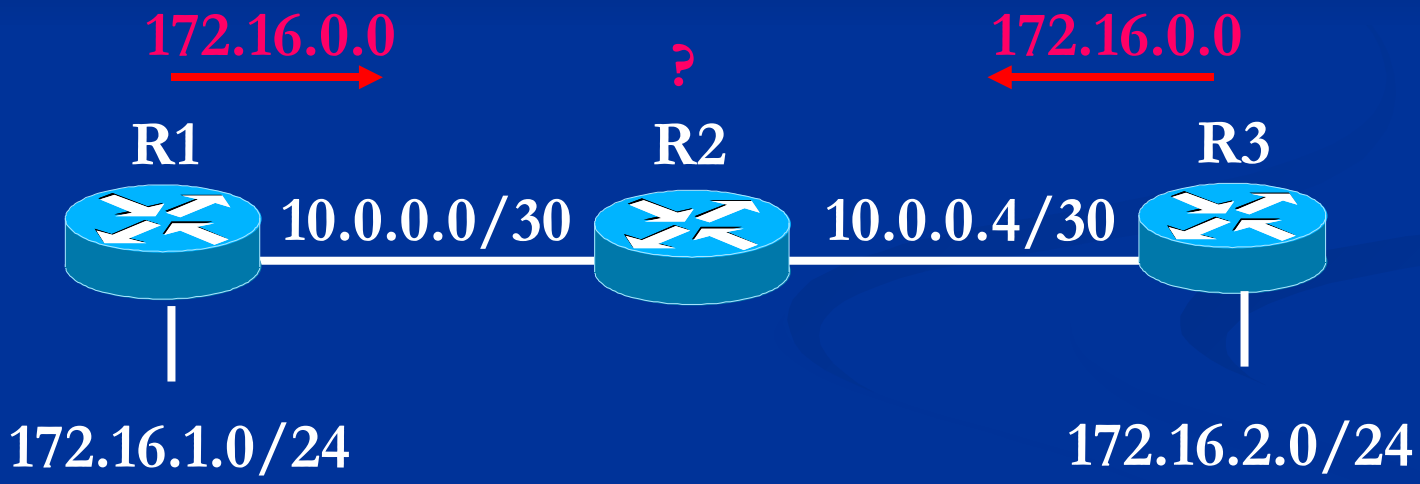
Třídní adresování a maska podsítě konstantní délky

172.16.2.0, ale s jakou maskou ?

Převezmu masku z rozhraní, kudy informace přišla !



Třídní adresování a nespojité podsítě



VLSM: Variable-Length Subnet Mask

- Původně se v rámci jedné podsíťované IP sítě používala stejná maska podsítě.
  - Neefektivní v případě sítě se segmenty o velmi různém počtu stanic
    - (např. Ethernet LAN segment / dvoubodový spoj)
- VLSM (RFC 1009) dovoluje v podsítích jedné sítě používat více rozdílných masek podsítí
  - výsledné adresy však samozřejmě nadále musí zůstat jednoznačné
- Použitelné jen se směrovacími protokoly, které spolu s adresou sítě propagují i její masku podsítě (OSPF, ISIS, RIP v.2).
  - ve směrovacích tabulkách uloženy adresy cílových sítí vždy s příslušnými maskami podsítí
  - použije se položka, která se shoduje s cílovou adresou ze směrovaného paketu na největší počet bitů