

NSWI184 – Řízení počítačových sítí

Přednáška čtvrtá: Babel

Ondřej Zajíček, Kateřina Kubecová

zaskakuje Maria

22. října 2025

Jak RIP funguje

Distance Vector algoritmus

- ▶ Každý router udržuje routovací tabulku
- ▶ Inicializuje tabulku lokálními routami
- ▶ Pravidelně ohlašuje celou tabulku sousedům
- ▶ Když obdrží update od souseda, připočte k metrikám cenu rozhraní
- ▶ Pokud je nová routa lepší než stávající (má nižší metriku), nahradí ji
- ▶ Distribuovaný Bellman-Ford algoritmus

Jak Babel funguje

Distance Vector algoritmus

- ▶ Každý router udržuje routovací tabulku
- ▶ Inicializuje tabulku lokálními routami
- ▶ Pravidelně ohlašuje celou tabulku sousedům
- ▶ Když obdrží update od souseda, připočte k metrikám cenu rozhraní
- ▶ Pokud je nová routa lepší než stávající (má nižší metriku), nahradí ji
- ▶ Distribuovaný Bellman-Ford algoritmus

Babel Hello

- ▶ Ohlašování sousedům: Hello pakety
 - ▶ mohou být multicast i unicast
 - ▶ obsahují sekvenční číslo (rostoucí)
 - ▶ obsahují slib, kdy přijde další
- ▶ Nazpátek se posílá IHU (I Heard You)
 - ▶ může být unicast i multicast
 - ▶ aktuální rxcost spočítaná podle toho, kolik hellos skutečně přišlo
 - ▶ kdy dorazí další

Cena linky

- ▶ Různé druhy metrik – wired, wireless, rtt
- ▶ Vstup pro dynamickou metriku dodává druhá strana v IHU jako $rxcost$
- ▶ Babel nspecifikuje konkrétní metriku, pouze doporučuje, a má požadavky:
 - ▶ 16-bit unsigned integer.
 - ▶ Pokud nám nechodí hellos, $rxcost = 65535 = \infty$.
- ▶ Z dodané hodnoty $rxcost$ a lokálně nastavené $txcost$ se počítá výsledná cena.
- ▶ Doporučená $rxcost$: $\lfloor \frac{256N}{K} \rfloor$, kdy z posledních N hellos dorazilo K .

Doporučená Babel metrika – wired

Fixní cena linky

- ▶ Maximální metrika: $2^{16} - 2 = 65534$
- ▶ Metrika 65535 = nedosažitelné
- ▶ Sčítání stejně jako v RIPv
- ▶ Linka je up = přišlo K z posledních N hellos

Doporučená Babel metrika – wireless

Čím větší ztrátovost, tím větší cena
... cože, wifi je ztrátová?

Jak funguje wifi

Nutná fakta o wifi IEEE 802.11 pro pochopení protokolu Babel

- ▶ Každá linka je ztrátová
- ▶ Každý* wifi rámeček se musí ACKnout
- ▶ Když není ACK, zkouší se to znova ... až 8×

Jak funguje wifi

Nutná fakta o wifi IEEE 802.11 pro pochopení protokolu Babel

- ▶ Každá linka je ztrátová
- ▶ Každý unicast wifi rámec se musí ACKnout
- ▶ Když není ACK, zkouší se to znova ... až 8×
- ▶ Ztracený multicast nikoho netrápí

Ztrátovost na multicastu

- ▶ výrazně vyšší než na unicastu
- ▶ vždy nenulová (na unicastu lze dosáhnout prakticky nuly)

Jak si zobrazit wifi rámce

```
# iw dev wlp0s20f3 interface add mon0 type monitor
# ip link set mon0 up
# tcpdump -i mon0 -e -vv
```

```
11:16:16.992414 1605101682458us tsft 5500 MHz 11a -54dBm signal [bit 22] Protected
    36us CF +QoS DA:<censored> BSSID:<censored> SA:<censored> Data IV:823d Pad 20 KeyID 0
11:16:17.037917 1605101727911us tsft 5500 MHz 11a -53dBm signal [bit 22] Retry Protected
    36us CF +QoS DA:<censored> BSSID:<censored> SA:<censored> Data IV:823e Pad 20 KeyID 0
11:16:17.037940 1605101727911us tsft 5500 MHz 11a -53dBm signal [bit 22] Retry Protected
    36us CF +QoS DA:<censored> BSSID:<censored> SA:<censored> Data IV:823f Pad 20 KeyID 0
11:16:17.037945 1605101727911us tsft 5500 MHz 11a -53dBm signal [bit 22] Retry Protected
    36us CF +QoS DA:<censored> BSSID:<censored> SA:<censored> Data IV:823f Pad 20 KeyID 0
11:16:17.049295 1605101739257us tsft 5500 MHz 11a -53dBm signal [bit 22] Protected
    36us CF +QoS DA:<censored> BSSID:<censored> SA:<censored> Data IV:8240 Pad 20 KeyID 0
```

Doporučená Babel metrika – wireless

Čím větší ztrátovost, tím větší cena

- ▶ Zajímá nás úspěšnost doručování **od nás ven**.
- ▶ Linka má základní metriku = cenu v ideálním případě (txcost)
- ▶ Kolik přišlo multicast hellos z posledních N (obvykle $N = 16$)?

$$\text{rxcost} = \left\lfloor \frac{256N}{K} \right\rfloor ; \quad \text{cost} = \frac{\text{rxcost} \star \text{txcost}}{256} \approx \text{txcost} \star \frac{N}{K}$$

Doporučená Babel metrika – wireless

Čím větší ztrátovost, tím větší cena

- ▶ Zajímá nás úspěšnost doručování **od nás ven**.
- ▶ Linka má základní metriku = cenu v ideálním případě (txcost)
- ▶ Kolik přišlo multicast hellos z posledních N (obvykle $N = 16$)?

$$\text{rxcost} = \left\lfloor \frac{256N}{K} \right\rfloor ; \quad \text{cost} = \frac{\text{rxcost} \star \text{txcost}}{256} \approx \text{txcost} \star \frac{N}{K}$$

- ▶ To by hezky fungovalo, kdyby neexistoval **AARF**

AARF

- ▶ Animal Allies Rescue Foundation
- ▶ Alaskan Animal Rescue Friends
- ▶ AARF Animal Rescue Foundation
- ▶ All About Rescue and Fixin', Inc.
- ▶ ...

AARF

- ▶ Animal Allies Rescue Foundation
- ▶ Alaskan Animal Rescue Friends
- ▶ AARF Animal Rescue Foundation
- ▶ All About Rescue and Fixin', Inc.
- ▶ ...
- ▶ Adaptive Auto Rate Fallback

Adaptive Auto Rate Fallback

- ▶ Stanice se *dohadují* na rychlosti přenosu
- ▶ Implementováno v hardwaru a firmwaru
- ▶ Ztrácí se pakety moc často? Snížíme rychlost, zlepšíme spolehlivost!
- ▶ RFC 8966, appendix A.2.2: *By default, the IEEE 802.11 MAC performs Automatic Repeat Query (ARQ) and rate adaptation on unicast frames, but not on multicast frames, which are sent at a fixed rate with no ARQ; therefore, measuring the loss rate of multicast frames yields a useful estimate of a link's quality.*
- ▶ IEEE 802.11, sekce 9.7.5 (ohledně rate adaptation) je velmi komplikovaná
- ▶ Moje AP adaptuje rychlost pro multicast stejně jako pro unicast
- ▶ AARF měří ztrátovost unicastu (selhání ARQ).
- ▶ Obvyklý threshold pro AARF: kolem 15% selhání unicastu.

Babel metrika – wireless

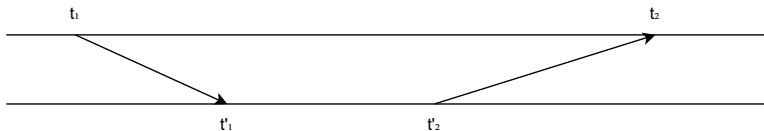
Čím větší ztrátovost, tím větší cena

- ▶ Běžný multicast packet loss na linkách: mezi 1% a 15%
- ▶ Doporučená metrika je příliš lineární.
- ▶ Téma na bakalářku / diplomku? Naměřit skutečné chování současných wifin a navrhnout lepší metriku pro Babel!

Babel metrika – RTT

Čím delší odezva, tím větší cena

- ▶ Hodí se na tunelech a VPN
- ▶ Hello obsahuje navíc lokální timestamp odesílatele t_1 .
- ▶ Při přijetí Hello se uloží lokální timestamp příjemce t'_1 .
- ▶ IHU obsahuje navíc t_1 a t'_1 a v *tomtéž* *paketu* se pošle taky Hello z druhé strany, timestampované t'_2 . Při jeho přijetí se uloží timestamp t_2
- ▶ $rtt = (t_2 - t_1) - (t'_2 - t'_1)$



Babel metrika – RTT

Jak převést RTT na cenu linky

- ▶ RTT je kolísavé, metrika potřebuje vyhlazovat.
- ▶ Doporučení:
 - ▶ $\text{rtt}_{n+1} = \alpha \text{rtt} + (1 - \alpha) \text{rtt}_n$
 - ▶ pro dostatečně malé rtt je cena konstantní (dost dobrá linka)
 - ▶ pro dostatečně velké rtt taktéž (dost mizerná linka)
 - ▶ v meziprostoru lineární závislost

Ochrana proti smyčkám

Horší routy nepřijímáme, sorry jako!

- ▶ Horší routa může být naše vlastní, která přišla nazpátek
- ▶ *Feasibility*: metrika menší než současná

Ochrana proti smyčkám

Horší routy nepřijímáme, sorry jako!

- ▶ Horší routa může být naše vlastní, která přišla nazpátek
- ▶ *Feasibility*: metrika menší než současná
- ▶ *Feasibility*: vyšší seqno než současná

Sekvenční číslo: nová vlna updatů

- ▶ Došly feasible routy? Pošlu požadavek na reflood (seqno request)!
- ▶ Posílá se **proti routám** ... i když nejsou feasible.
- ▶ Nutná kontrola na duplicitní requesty (aby se nezacyklily).
- ▶ Resend unless satisfied.

Zdroje

- ▶ RFC 8966 – Babel
- ▶ RFC 9616 – Round Trip Time metric
- ▶ IEEE 802.11