Vysoké učení technické v Brně
Fakulta informačních technologií

Síťové aplikace a správa sítí - projekt

Nástroje monitorující a generující zprávy jednoduchých distance-vector protokolů

19. listopadu 2018

Vojtěch Čurda (xcurda02)

Obsah

1 Jednoduché distance-vector směrovací protokoly

1.1 Obecně

RIP (Routing Information Protocol) je směrovací protokol, pomocí kterého komunikují směrovače mezi sebou a tím dokáží reagovat na změny topologie sítě.

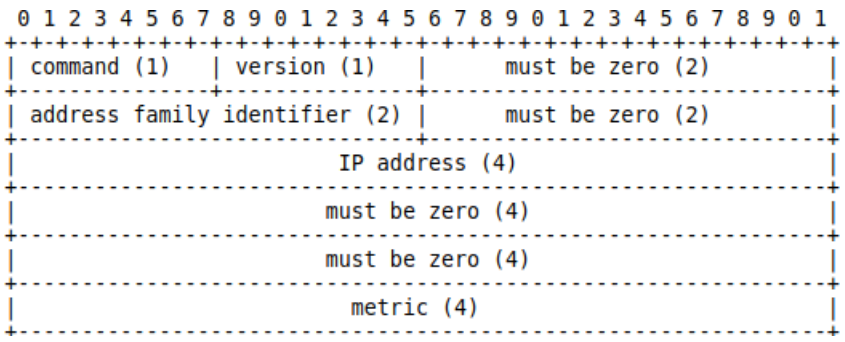
Je to protokol typu distance-vector využívající Bellmanův-Fordův algoritmus pro určení nejkratší cesty v síti. Metrikou směrování je počet skoků k cílové síti. Maximální počet skoků je omezen na 15, přičemž 16 znamená nekonečno (používá se k označení nedostupných sítí).

Rozlišují se dva typy zpráv: požadavky (Request) a odpovědi (Response). Směrovač, který byl např. nově přiřazen do sítě pošle požadavek okolním směrovačům. Ty mu obratem posílají odpověd v podobě jejich směrovací tabulky, nebo pouze částí tabulky, podle toho, jak byl formulován požadavek. Dále každý směrovač posílá požadavky na okolní sítě periodicky.

Jako transportní protokol je používáno UDP. Hlavička paketu všech RIP protokolů je jednotná. Obsahuje informaci o typu zprávy (command) a verzi (version).

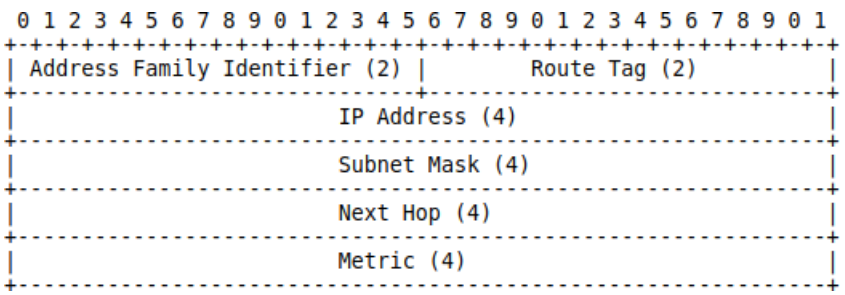
1.2 RIPv1

RIP protokol verze 1 pracuje s IPv4 adresami podle původních tříd A,B a C. Každých 30 vteřin posílají směrovače na broadcastovou adresu `255.255.255.255` požadavek pro všechna okolní zařízení. RIP entry nese informace jako je IP adresa a metrika. Struktura RIP paketu včetně hlavičky je následující:

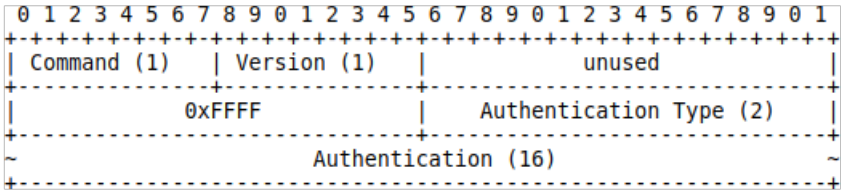


1.3 RIPv2

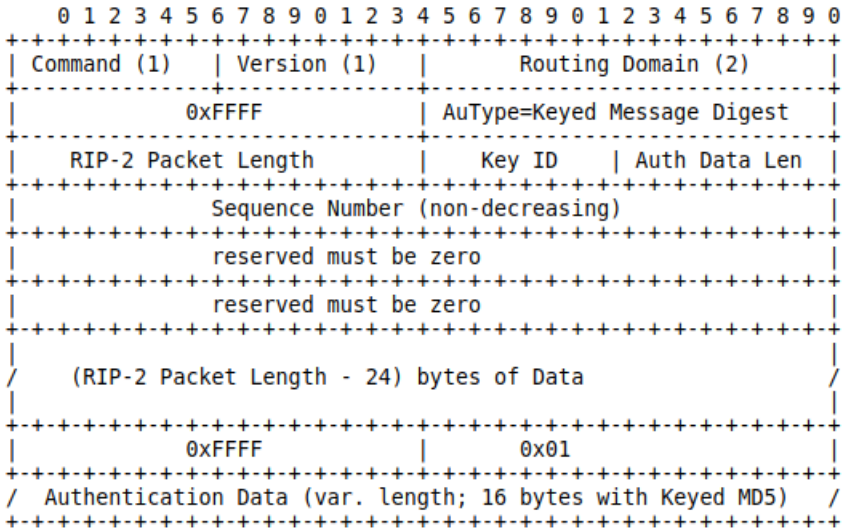
RIP protokol verze 2 přidává a rozšiřuje funkcionalitu předchozí verze. Uvnitř RIP entry se nově dá přenášet i maska sítě a Next-Hop adresa. Přidán je také nový typ entry, který se stará o autentizaci zprávy. Struktura RIP entry nesoucí adresu:



Struktura autentizační entry včetně RIP hlavičky:

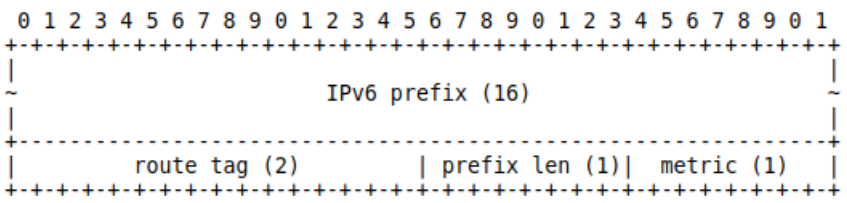


RIPv2 umožňuje také autentizaci pomocí MD5, kdy pro autentizaci je vymezena první a poslední entry paketu:

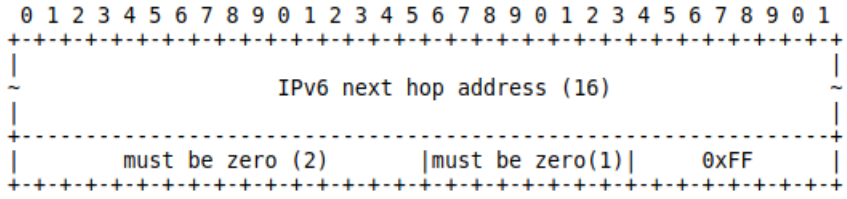


1.4 RIPng

RIPng je oproti RIPv2 vylepšeno o práci s IPv6 adresami. Narozdíl od předchozích dvou verzí už nepoužívá UDP port 520 ale UDP port 521.
RIPng zpráva má může obsahovat dva typy entry:
Route Table Entry:



Next-hop Route Table Entry:



2 Implementace

2.1 Úvod

Jako implementační jazyk bylo vybráno C++ pro jednodušší práci s řetězci a také proto, že jazyk C nepřináší pro naše účely výrazné výhody, spíše naopak. Implementace obsahuje soubory `myripsniffer.cpp`, `myripresponse.cpp` a `myriprequest.cpp`, které obsahují hlavní logiku aplikací a hlavičkový soubor `rip_defs.h` obsahující definice konstant a potřebných struktur, který využívají všechny tři aplikace.

2.2 Sniffer RIP zpráv

Sniffer (zachytávač) RIP zpráv využívá funkcí knihovny PCAP. Je možné zachytávat RIP pakety jak online, tak offine (z .pcap souboru). Mód zachytávání se určuje pomocí spouštěcího parametru `-i` a hodnotou která reprezentuje buď název rozhraní nebo již zmíněný soubor.

V případě že na vstupním parametru bylo zadáno rozhraní, je nejprve otevřeno funkcí `pcap_open_live()`. Pokud bylo rozhraní úspěšně otevřeno, je na ně nastaven filtr `portrange 520-521 and udp` funkcí `pcap_set_filter()`. Tento filtr propouští pouze pakety transportního protokolu UDP na portu 520 a 521.

Funkcí `pcap_loop()` je spuštěno zpracovávání příchozích RIP paketů. Tato funkce posílá příchozí pakety na zpracování dokud nenarazí na konec souboru (v případě čtení paketů ze souboru) nebo je aplikace násilně ukončena ukončovacím signálem. Případné ukončovací signály jsou odchyceny a je spuštěna funkce `terminate()` starající se o úklid prostředků a ukončení aplikace.

2.3 Podvrhávač RIPng respose zpráv

Vstupní parametry:

`-i <interface>`	povinný - rozhraní, ze kterého bude vyslána zpráva
`-r <IPv6>/[16-128]`	povinný - IP adresa podvrhávané sítě, za lomítkem maska
`-n <IPv6>`	IP adresa Next-hopu
`-m <metric>`	metrika (počet hopů), implicitně 1
`-t <route-tag>`	hodnota Route Tagu, implicitně 0
`-h`	Nápověda

Podvrhávač RIPng response zpráv má za úkol vyslat útočný paket protokolu RIPng na sousední směrovač a přidat tím IP adresu zadanou ve vstupních parametrech do jeho směrovací tabulky.

Aplikace nejprve naplní entry údajemi zadanými vstupními parametry. Pokud byla zadána Next-hop adresa, paket bude obsahovat dvě entry. Poté se vyplní hlavička paketu a připojí se k ní již vytvořené entry. Dále je vytvořen socket, který se nabinduje k zadanému rozhraní. Funkcí `sendto()` je poté packet poslán na multicastovou adresu `FF02::9`

2.4 Podvrhávač RIPng request zpráv

Vstupní parametry:

`-i <interface>`	povinný - rozhraní, ze kterého bude vyslána zpráva
`-r <IPv6>/[0-128]`	povinný - IP adresa na dotazovanou síť, za lomítkem maska
`-m <metric>`	metrika (počet hopů), implicitně 1
`-t <route-tag>`	hodnota Route Tagu, implicitně 0
`-h`	Nápověda

Podvrhávač RIPng request zpráv simuluje router posílající dotaz na zjištění topologie sítě od ostatních směrovačů. Podle vstupních parametrů je naplněn paket a odeslán jako požadavek na multicastovou adresu `FF02::9`.

Implementačně je v něm plno kódu převzato z podvrhávače response zpráv, jelikož jediné věci, ve kterém se liší jsou: absence next-hop adresy u podvrhávače request zpráv a jiná hodnota atributu `command` v RIP hlavičce.

3 Ukázka činnosti programu

Testování bylo prováděno mezi freeBSD virtuálním strojem s nakonfigurovaným SW směrovačem Quagga a virtuálním strojem Ubuntu, přičemž jsou oba propojeny virtuální sítí.
Všechny tři aplikace je nutno spouštět jako administrátor.

3.1 Sniffer RIP zpráv

Sniffer při spuštění příkazem `sudo ./myripsniffer -i eth0` začíná naslouchat na rozhraní `eth0` a vypisuje případné RIP pakety. Na obrázku lze vidět zachycení periodické aktualizace od virtuálního směrovače. Nejprve se vypíší informace o paketu jako takovém včetně verze RIP protokolu, času přijetí paketu, zdrojové a cílové adresy, zdrojového a cílového portu, typu RIP zprávy a délky celého paketu. Poté se vypisují jednotlivé entry.

```
[RIPng]     Src IP: fe80::a00:27ff:fef6:6a86   Src Port: 521     Command: Response
(13:02:16)  Dst IP: ff02::9                    Dst Port: 521     Length:  166
──────────────────────────────────────────────────────────────────────────────
┌──────────────── Route Table Entry ─────────────────┐
 IPv6 Prefix:       fd00::
 Route tag:         0
 Prefix Length:     64
 Metric:            1
└────────────────────────────────────────────────────┘
┌──────────────── Route Table Entry ─────────────────┐
 IPv6 Prefix:       fd00:d3:2d78::
 Route tag:         0
 Prefix Length:     64
 Metric:            1
└────────────────────────────────────────────────────┘
┌──────────────── Route Table Entry ─────────────────┐
 IPv6 Prefix:       fd00:109:26ac::
 Route tag:         0
 Prefix Length:     64
 Metric:            1
└────────────────────────────────────────────────────┘
┌──────────────── Route Table Entry ─────────────────┐
 IPv6 Prefix:       fd00:95a:63::
 Route tag:         0
 Prefix Length:     64
 Metric:            1
└────────────────────────────────────────────────────┘
┌──────────────── Route Table Entry ─────────────────┐
 IPv6 Prefix:       fd00:960:16da::
 Route tag:         0
 Prefix Length:     64
 Metric:            1
```

3.2 Podvrhávač RIPng response zpráv

Pokus o podvržení adresy `2001:db8:0:abcd::/64` byl spuštěn příkazem

```
sudo ./myripresponse -i eth0 -r 2001:db8:0:abcd::/64
```

Podle obsahu směrovací tabulky routeru lze uznat pokus za úspěšný, jelikož podvrhovaná adresa ve směrovací tabulce směrovače přibyla.
Odchozí response zpráva zachycená odchytávačem:

```
[RIPng]     Src IP: fe80::a00:27ff:fe94:b41e   Src Port: 521     Command: Response
(13:05:36)  Dst IP: ff02::9                    Dst Port: 521     Length:  86
```

```
Route Table Entry
IPv6 Prefix:       2001:db8:0:abcd::
Route tag:         0
Prefix Length:     64
Metric:            1
```

Směrovací tabulka před zasláním response zprávy:

```
Routing> show ipv6 route
Codes: K - kernel route, C - connected, S - static, R - RIPng, O - OSPFv3,
       I - ISIS, B - BGP, * - FIB route.

K>* ::/96 via ::1, lo0, rej
C>* ::1/128 is directly connected, lo0
K>* ::ffff:0.0.0.0/96 via ::1, lo0, rej
C>* fd00::/64 is directly connected, em0
C>* fd00:d3:2d78::/64 is directly connected, lo0
C>* fd00:109:26ac::/64 is directly connected, lo0
C>* fd00:95a:63::/64 is directly connected, lo0
C>* fd00:960:16da::/64 is directly connected, lo0
K>* fe80::/10 via ::1, lo0, rej
C * fe80::/64 is directly connected, lo0
C>* fe80::/64 is directly connected, em0
K>* ff02::/16 via ::1, lo0, rej
```

Směrovací tabulka po zaslání response zprávy:

```
Routing> show ipv6 route
Codes: K - kernel route, C - connected, S - static, R - RIPng, O - OSPFv3,
       I - ISIS, B - BGP, * - FIB route.

K>* ::/96 via ::1, lo0, rej
C>* ::1/128 is directly connected, lo0
K>* ::ffff:0.0.0.0/96 via ::1, lo0, rej
R>* 2001:db8:0:abcd::/64 [120/2] via fe80::a00:27ff:fe94:b41e, em0, 00:00:51
C>* fd00::/64 is directly connected, em0
C>* fd00:d3:2d78::/64 is directly connected, lo0
C>* fd00:109:26ac::/64 is directly connected, lo0
C>* fd00:95a:63::/64 is directly connected, lo0
C>* fd00:960:16da::/64 is directly connected, lo0
K>* fe80::/10 via ::1, lo0, rej
C * fe80::/64 is directly connected, lo0
C>* fe80::/64 is directly connected, em0
K>* ff02::/16 via ::1, lo0, rej
```

3.3 Podvrhávač RIPng request zpráv

Na zaslání request zprávy s adresou `::/0` a metrikou 16 by měl směrovač regovat posláním jeho směrovací tabulky dotazovanému směrovači. Příkaz

```
sudo ./myriprequest -i eth0 -r ::/0 -m 16
```

spustí podvrhávač RIPng request zpráv, který pošle request zprávu na sousední směrovač, který okamžitě odpovídá a sdílí směrovací tabulku, jak je vidno z obrázku níže.

```
[RIPng]     Src IP: fe80::a00:27ff:fe94:b41e   Src Port: 521     Command: Request
(13:07:48)  Dst IP: ff02::9                    Dst Port: 521     Length:  86
──────────────────────────────────────────────────────────────────────────────
┌──────────────── Route Table Entry ─────────────────┐
 IPv6 Prefix:       ::
 Route tag:         0
 Prefix Length:     0
 Metric:            16

[RIPng]     Src IP: fe80::a00:27ff:fef6:6a86   Src Port: 521     Command: Response
(13:07:48)  Dst IP: fe80::a00:27ff:fe94:b41e   Dst Port: 521     Length:  166
──────────────────────────────────────────────────────────────────────────────
┌──────────────── Route Table Entry ─────────────────┐
 IPv6 Prefix:       fd00::
 Route tag:         0
 Prefix Length:     64
 Metric:            1
┌──────────────── Route Table Entry ─────────────────┐
 IPv6 Prefix:       fd00:d3:2d78::
 Route tag:         0
 Prefix Length:     64
 Metric:            1
┌──────────────── Route Table Entry ─────────────────┐
 IPv6 Prefix:       fd00:109:26ac::
 Route tag:         0
 Prefix Length:     64
 Metric:            1
┌──────────────── Route Table Entry ─────────────────┐
 IPv6 Prefix:       fd00:95a:63::
 Route tag:         0
 Prefix Length:     64
 Metric:            1
┌──────────────── Route Table Entry ─────────────────┐
 IPv6 Prefix:       fd00:960:16da::
 Route tag:         0
 Prefix Length:     64
 Metric:            1
```

Odkazy

[1] RFC 1058, *Routing Information Protocol*, C. Hendrik, The Internet Society (June 1988)

[2] RFC 2453, *RIP Version 2*, G. Malkin, The Internet Society (November 1998)

[3] RFC 2082, *RIP-2 MD5 Authentication*, F. Baker, R. Atkinson, The Internet Society (January 1997)

[4] RFC 2080, *RIPng for IPv6*, G. Malkin, R. Minnear, The Internet Society (January 1997)

[5] Wikipedia contributors. (2018, October 3). Routing Information Protocol. *In Wikipedia, The Free Encyclopedia*. Retrieved 13:25, November 14, 2018