FIT VUT v Brně

Síťové aplikace a správa sítí
2017/2018

Nástroje monitorující a generující zprávy jednoduchých distance-vector protokolů

David Endrych (xendry02)

Brno, 17. 11. 2018

Zadání

1) nastudovat si směrovací protokoly RIP a RIPng
2) naprogramovat sniffer RIPv1, RIPv2 a RIPng zpráv
3) naprogramovat podvrhávač falešných RIPng Response zpráv
4) za použití obou nástrojů, které jste si předpřipravili v předchozím bodě pak provést úspěšný útok

Obsah

Úvod do problematiky

Routing Information Protocol je směrovací protokol umožňující směrovačům komunikovat mezi sebou a reagovat na změny topologie. RIP protokol je směrovací protokol typu distance-vector (vektor vzdáleností). Pro určení nejkratší cesty v síti se využívá Bellmanův-Fordův algoritmus. Tenhle algoritmus počítá nejkratší cestu v ohodnoceném grafu, kde na rozdíl od Dijkstrova algoritmu můžou být hrany i záporné. Historicky první použití RIP protokolu bylo v roce 1969 v síti ARPANET.

Existují tři verze RIP protokolu RIP, RIPv2 a RIPng.

RIPv1

Používá směrování podle původních tříd IPv4 adres A, B nebo C. Periodické aktualizace neobsahují informace o masce sítě. To znemožňuje existenci různě velkých podsítí uvnitř jedné třídy IP adres. Všechny podsítě v téhle verzi musí mít stejnou masku sítě. Pro šíření směrovací tabulky sousedním směrovačům využívá RIP protokol broadcast.
V první verzi RIP protokol neobsahoval možnost autentizace, proto byl protokol velmi náchylný na různé útoky.

RIPv2

Jelikož první verze RIP protokolu měla nedostatky, tak vznikla druhá verze tohohle protokolu. Přibyla zde možnost přenášet informace o masce sítě. Kvůli zpětné kompatibilitě zde zůstal omezený hop count. RIPv2 vysílá směrovou tabulku sousedním směrovačům na adrese 224.0.0.9 (multicast). Přibyla zde autentizace routerů. V první verzi se hesla přenášela v nekódovaném textu, a proto později přibyla možnost autentizace pomocí MD5.

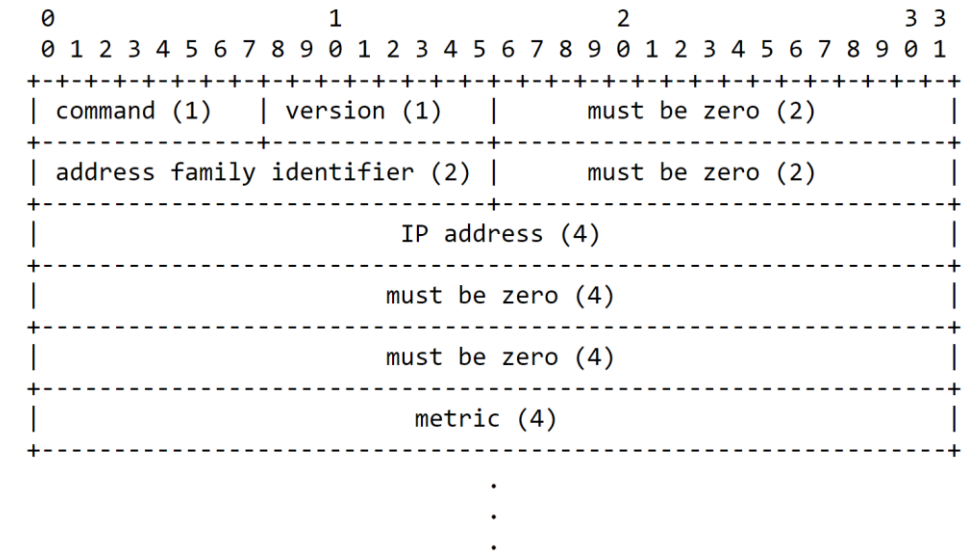
RIPng

Je rozšíření RIPv2 o podporu IPv6 síťování. Oproti předchozím verzím nepodporuje připojování libovolných tagů k směrovačům. Také zde chybí podpora aktualizovaných autentizací.

Struktura protokolů

RIPv1

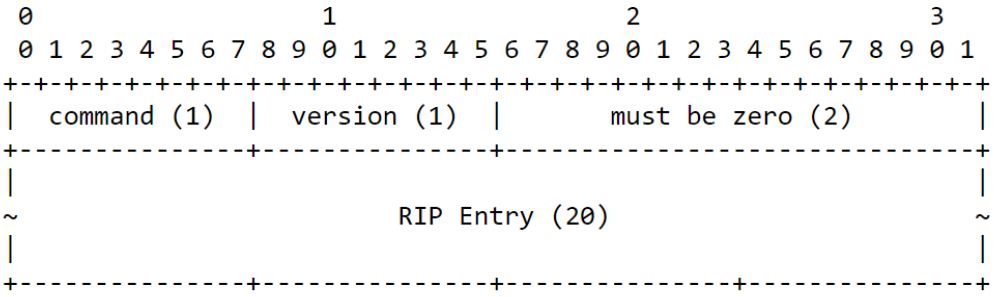
Packet RIP protokolu můžeme vidět na obrázku 1 níže. Uvedená čísla značí počet bajtů, ve kterých je informace obsažena. Jako první se v packetu nachází dva bajty, které udávají základní informace o protokolu. Command značí, jestli se jedná o požadavek nebo o odpověď. Zatímco verze nám udává, jestli se jedná o RIPv1 nebo RIPv2, zbytek prvního řádku musí povinně vyplňovat nuly. Dále se v packetu nachází informace o adresách a jejich metrice. Část od address family identifier po metric se může v packetu opakovat. Celkem tahle část může být v packetu až dvacet pětkrát.



Obrázek 1: Struktura RIPv1 protokolu

RIPv2

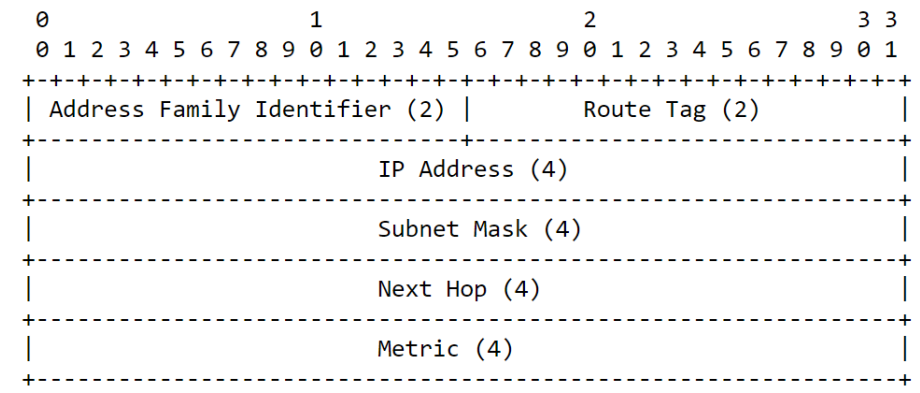
Na druhém obrázku můžeme najít, jak vypadá struktura RIPv2 protokolu. Stejně jako v prvním případě je na začátku command a version, jako v případě RIPv1 protokolu. Poté následuje RIP entry. Počet entry může být v rozsahu od jedné do dvaceti pěti.



Obrázek 2: Struktura RIPv2

Entry

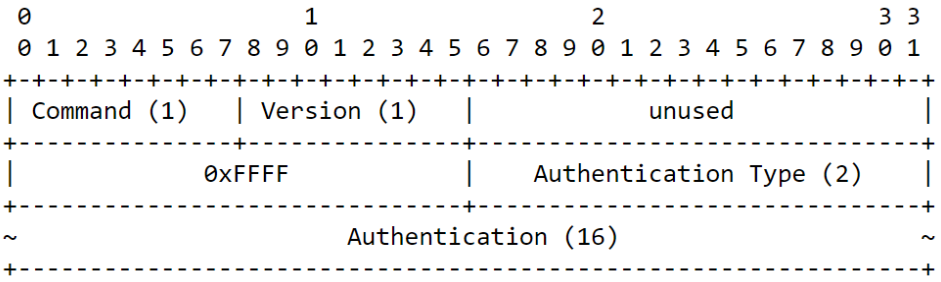
Entry může být v protokolu, jak už bylo dříve uvedeno až dvacet pětkrát za sebou. Minimálně však musí být uvedena jednou. Oproti RIPv1 zde byla přidání informace o router tagu a masky podsítě a adresy dalšího skoku.



Obrázek 3: Struktura RIPv2 Entry

Autentizace

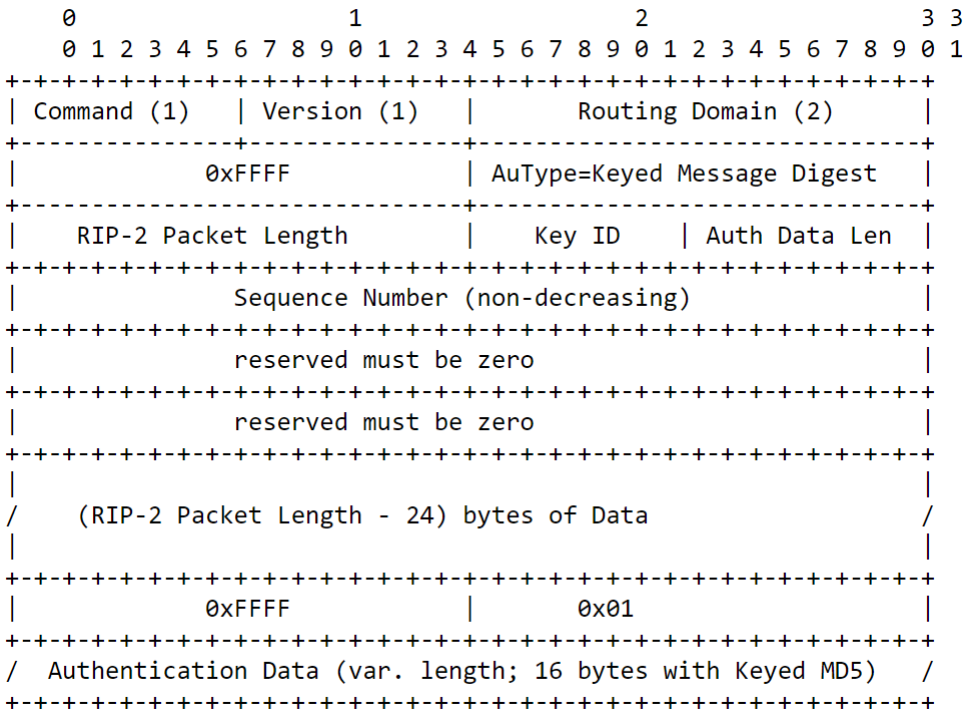
V RIPv2 můžeme najít i autentizaci. Základní autentizace se nachází na začátku packetu. Má stejnou velikost jako entry. Na místě address family identifier je hodnota 0xFFFF. Pozice router tagu má nyní význam typu autentizace. Následuje šestnáct bajtů, kde je obsaženo heslo v nekódovaném tvaru.



Obrázek 4: RIPv2 Autentizace

MD5 autentizace

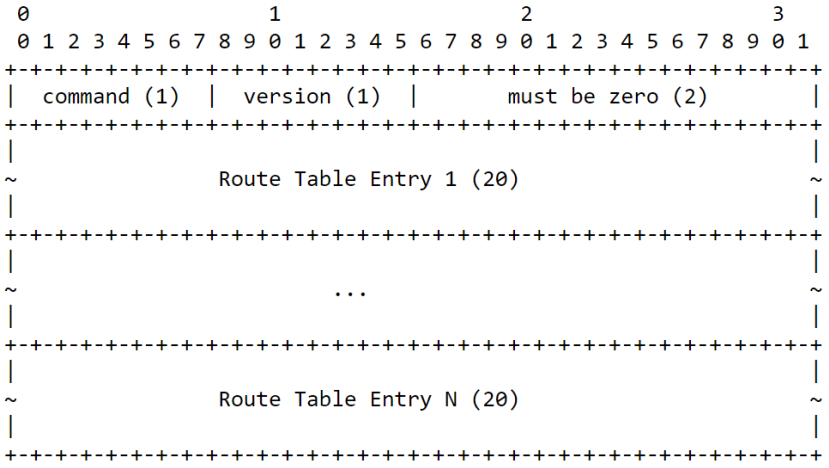
Další možností v RIPv2 protokolu je MD5 autentizace. Zde vidíme, že v prvním řádku nyní máme navíc položku routing domain, dále následuje první část autentizace pro MD5, kde můžeme vidět například id klíče, délku autorizačních dat nebo délku packetu. Po sekvenčním čísle a části, kde nalezneme pouze nuly můžeme najít jednotlivé entry a poté druhou část autentizace, kde nalezneme autentizační data.



Obrázek 5: Struktura RIPv2 packetu s MD5 autentizaci

RIPng

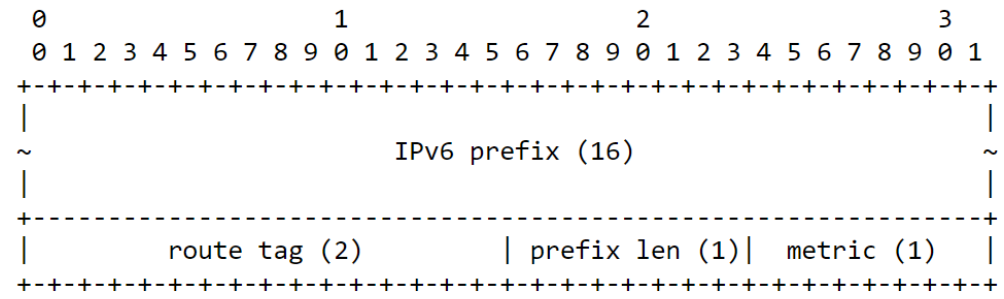
V případě RIPng protokolu můžeme vidět shodný první řádek s protokoly RIPv1 a RIPv2, poté následují jednotlivé entry.



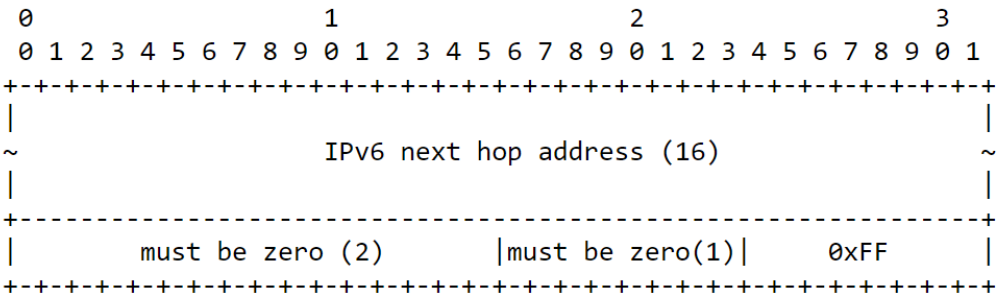
Obrázek 6: Struktura RIPng

Entry

V entry můžeme najít informace o prefixu IPv6 adresy, router tagu, délky prefixu a metrice. Další varianta je taková, že je zde adresa next hopu, poté na pozici router tagu a délky prefixu nalezneme nuly a hodnota metriky odpovídá hodnotě 0xFF.



Obrázek 7: Struktura RIPng entry s adresou



Obrázek 8: Struktura RIPng entry s adresou next hopu

Popis implementace

Sniffer

Na začátku pomocí funkce getopt načteme argumenty volaní programu. Pokud není zadán interface, tak dojde k výpisu nápovědy a ukončení programu. Dále jsou zde dvě varianty, buď se sniffer napojí na pcap soubor nebo na reálný interface. Dále následuje smyčka (pcap_loop), která zachycuje packety, které posílá do funkce parse_packet, kde se packety zpracovávají.

Ve funkci parse_packet se zpracovávají postupně jednotlivé packety, začínají zpracováním ethernetového protokolu, dále následuje zpracování IP protokolu, poté zde je UDP protokol, který obaluje RIP protokol. Podle toho, jestli se jedná o IPv6 komunikaci nebo IPv4 se RIP protokol zpracuje buď to ve funkci parse_rip nebo parse_ripng.

V parse_rip se jako první zpracuje začátek protokolu, jelikož ho verze protokolu mají stejný a poté se zpracovává zbytek packetu podle toho, jestli se jedná o RIPv1 nebo RIPv2. V RIPv2 poté musíme rozlišovat, jestli se jedná o klasické entry nebo autentizaci nebo MD5 autentizaci.

Response

Na začátku se inicializují proměnné, nastaví se jejich defaultní hodnoty, dále dojde k načtení argumentů volání programu pomocí funkce getopt. Pokud jsou všechny proměnné správně nastaveny může se přejít do funkce send_ripng_response.

Zde se postupně nastaví potřebné struktury protokolu RIPng. Dále se vytvoří packet, který se naplní daty struktur. Vytvoří se socket, který je potřeba správně nastavit. Poté může dojít k poslaní packetu na daném socketu. Socket se může nakonec uzavřít.

Popis zajímavých částí

Jedna ze zajímavějších částí je zpracování RIPv2 protokolu. První část protokolu se zpracovává ve společné funkci pro RIPv1 a RIPv2. Zbytek protokolu se vypisuje pomocí funkce print_ripv2_entry. Zde se jako první rozhoduje podle hodnoty address family identifier, to nám řekne, jestli se jedná o entry nebo o autentizaci. V případě autentizace se poté rozhoduje podle typu autentizace, jestli se jedná o klasickou autentizaci nebo md5 v případě md5 jsou zde dvě možnosti zpracování. Jedna zpracovává začátek autentizace a druhá konec autentizace.

Testování

Testování snifferu jsem jako první prováděl pomocí pcap souborů, pro každou verzi RIP protokolu jsem měl jeden pcap soubor. Poté co jsem odladil sniffer na těchto souborech, začal jsem testovat sniffer pomocí virtuálního počítače, který nám byl poskytnut k projektu. Na tomto virtuálním počítači běží SW směrovač Quagga. Na tomto stroji jsem otestoval sniffer na protokoly RIPv2 a RIPng.

Dále jsem také využil možnosti testování v labořatoři, kde se mi povedlo podvrhnout routu falešnou RIPng response zprávou. Dále jsem zde také otestoval sniffer, ten zde taky fungoval až na packety s MD5 autentizací, kterou jsem následně přidal do programu.

Při implementaci MD5 autentizace pro RIPv2 jsem znovu použil pro testování pcap soubor.

Základní informace o programu

Máme zde dva programy první z nich je sniffer RIP zpráv. Sniffer je program, který slouží pro sledování síťového provozu. Program umí zachycovat a zaznamenávat komunikaci v počítačové síti. V tomhle programu je schopný zachycovat a zpracovávat RIP protokoly.

Druhý program slouží pro vytváření RIPng zpráv typu response. Dokáže na dané rozhraní poslat RIPng response zprávu, která je naplněna hodnotami definovanými uživatelem v argumentech programu.

Návod použití

myripresponse

Spuštění

./myripsniffer -i <rozhraní>

Význam argumentů

-i <rozhraní> Rozhraní, na kterém má být prováděn odchyt packetů

myripsniffer

Spuštění

./myripresponse -i <rozhraní> -r <IPv6>/[16-128] {-n <IPv6>} {-m [0-16]} {-t [0-65535]}

Význam argumentů

-i: <rozhraní> Rozhraní, ze kterého má být útočný paket odeslán

-r: <IPv6> IP adresa podvrhované sítě a za lomítkem číselná délka masky sítě

-m: RIP Metrika, implicitně 1

-n: <IPv6> Adresa next-hopu pro podvrhovanou routu, implicitně ::

-t: Hodnota Router Tagu, implicitně 0

Bibliografie

HEDRICK, C. Routing Information Protocol. June 1988 [cit. 2018-11-17]. Dostupné z: https://tools.ietf.org/html/rfc1058

MALKIN, G. RIP Version 2 [online]. November 1988 [cit. 2018-11-17]. Dostupné z: https://tools.ietf.org/html/ rfc2453

MALKIN, G. RIPng for IPv6 [online]. November 1988 [cit. 2018-11-17]. Dostupné z: https://tools.ietf.org/html/rfc2080

MALKIN, G. RIP-2 MD5 Authentication [online]. January 1997 [cit. 2018-11-17]. Dostupné z: https://tools.ietf.org/html/rfc2082