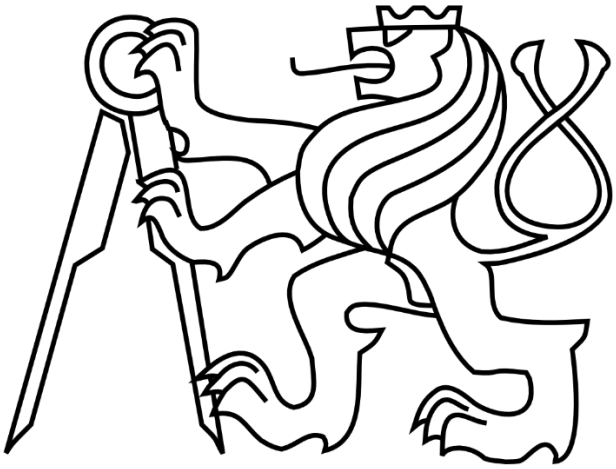
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická

Katedra telekomunikační techniky



Poskytování pokročilých telekomunikačních služeb v rozsáhlých podnikových sítích

Vedoucí práce: Ing. Pavel Troller, CSc.

Student: Bc. Lukáš Knížek

Leden 2016

Poděkování

Na tomto místě bych rád poděkoval Ing. Pavlu Trollerovi za vedení této práce, čas, který mi věnoval, i ochotu a rady při řešení dílčích problémů. Dále pak Stanislavu Petrovi za všechny odborné konzultace.

Čestné prohlášení

Prohlašuji, že jsem zadanou diplomovou práci zpracoval sám s přispěním vedoucího práce a konzultanta a používal jsem pouze literaturu v práci uvedenou. Dále prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé diplomové práce nebo její části se souhlasem katedry.

Datum: 11. 1. 2016

.....

České vysoké učení technické v Praze
Fakulta elektrotechnická

katedra telekomunikační techniky

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: **Bc. Lukáš Knížek**

Studijní program: Komunikace, multimédia a elektronika
Obor: Komunikační systémy

Název tématu: **Poskytování pokročilých telekomunikačních služeb v rozsáhlých podnikových sítích**

Pokyny pro vypracování:

Vypracujte projekt implementace vnitřní datové sítě rozsáhlého podniku (Enterprise Solution) tak, aby byly splněny následující požadavky:

- Heterogenní síťové prostředí s protokoly IPv4 i IPv6.
- Zajištění QoS pro oba protokoly pro současný provoz hlasových a datových služeb.
- Zajištění bezpečnosti dle požadavků zákazníka s možností separace jednotlivých druhů provozu (např. prostřednictvím VLAN či MPLS).
- Návrh topologie sítě odolný proti výpadku části přenosové technologie s využitím vhodných směrovacích protokolů.
- Návrh vhodných síťových prvků a laboratorní ověření jejich použitelnosti v reálném nasazení.
- Simulace navržené sítě ve vhodném prostředí dle vlastního výběru.

Seznam odborné literatury:

[1] THOMAS, R. M.: Lokální počítačové sítě (překlad Libor Spěvák). 1.vydání, Praha : Computer Press, 1996. 277 s. ISBN 80-85896-45-1.
[2] PENIAK, P.; KÁLLY, F.: Počítačové sítě LAN/MAN/WAN. 1.vydání, Praha : Grada, 199. 311 s. ISBN 80-7169-407-X.
[3] TRULLOVE, J.: Sítě LAN - hardware, instalace a zapojení (překlad Tomáš Znamenáček). 1.vydání, Praha : Grada, 2009. 384 s. ISBN 978-80-247-2098-2.

Vedoucí: Ing. Pavel Troller, CSc.

Platnost zadání: do konce letního semestru 2015/2016

prof. Ing. Boris Šimák CSc.
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 3. 12. 2015

Anotace:

Tato diplomová práce se zabývá návrhem a konfigurací rozsáhlé podnikové sítě určené primárně pro poskytování VoIP služeb. Jejím cílem je implementace sítě, která bude použita v komerčním prostředí. Práce danou problematiku zvažuje z technického i ekonomického pohledu, a díky své komplexnosti tak může být užitečnou pomocí při řešení stejného nebo obdobného problému.

Klíčová slova: VoIP, IP, IPv6, IPsec, GRE, BGP, MikroTik

Summary:

This diploma thesis deals with design and configuration of a large scale enterprise network supposed to be used primarily for providing VoIP services. Its aim is to implement a network that will be used in commercial sphere. This thesis considers the given subject technically as well as economically and thanks to its complexity serve as a great help for solving the same or a similar problem in the future.

Index terms: VoIP, IP, IPv6, IPsec, GRE, BGP, MikroTik

Obsah

Seznam zkratek a odborných termínů

Obecné zkratky nejsou rozváděny. Ty, které nejsou dále v textu zmiňovány, jsou rozvedené.

ATM – Asynchronous Transfer Mode

BFD – Bidirectional Forwarding Detection

BGP – Border Gateway Protocol

BGP Session – spojení protokolu BGP

Default Route – defaultní směr pro cíle bez konkrétnějšího záznamu

DSCP – Differentiated services code point

EIGRP – Enhanced Interior Gateway Routing Protocol

Firewall – zabezpečovací systém pro filtrování síťového provozu na základě předdefinovaných pravidel

Frame Realy – protokol 2. vrstvy specifikovaný v RFC 1490

GRE – Generic Routing Encapsulation

HTTP – Hypertext Transfer Protocol

IANA – Internet Assigned Numbers Authority

IETF – Internet Engineering Task Force

IGP – Interior Gateway Protocol

IPsec – Internet Protocol Security

IPv4 – Internet Protocol version 4

IPv6 – Internet Protocol version 6

IS-IS – Intermediate System to Intermediate System

L2TP – Layer 2 Tunneling Protocol

LAN – Local area network

Link-state Routing Protocol – typ routovacích protokolů, každý za zapojených routerů zná kompletní topologii sítě a počítá nejvýhodnější cesty

MTU – Maximum transmission unit

NAT – Network address translation

Next Hop – sousední router použitý při směrování paketu do dané destinace

OpenVPN – VPN protokol

OSPF – Open Shortest Path First

Overhead – nadbytečná informace, kterou je nutné s užitečnými daty přenášet, například záhlaví IP protokolu

PPTP – Point-to-Point Tunneling Protocol

RFC – Request for Comments

RIB – Routing Information Base

RIP – Routing Information Protocol

Router – často používaný český ekvivalent je směrovač, síťové zařízení pro routování paketů, pracuje na 3. vrstvě

Routování – často používaný český ekvivalent je směrování

SFP – Small Form-factor Pluggable

SSTP – Secure Socket Tunneling Protocol

Stub network – síť s pouze jedním odchozím a příchozím směrem

Switch – často používaný český ekvivalent je přepínač

TCP – Transmission Control Protocol

TCP/IP model – model síťového provozu

UDP – User Datagram Protocol

VLAN – Virtual LAN

VoIP – Voice over IP

VPLS – Virtual Private LAN Service

VPN – Virtual Private Network

Úvod

Cílem této diplomové práce je navrhnout síť vhodnou pro poskytování primárně VoIP služeb zákazníkům s více geograficky oddělenými pobočkami, správnost návrhu otestovat a následně celý návrh implementovat v produkčním prostředí. Požadované řešení zahrnuje páteřní technologie a jednotlivé vzdálené lokality připojené k Internetu pomocí zálohované konektivity. K propojení jednotlivých lokalit a páteřní technologie nebudou sloužit dedikované spoje, ale virtuální privátní sítě využívající šifrování provozu pro jeho zabezpečení, navázané přes standardní připojení k Internetu. Pro plně funkční řešení bude nutné vybrat vhodné protokoly pro vytvoření virtuálních sítí, zajistit správné routování všeho provozu a v neposlední řadě také řešit implementaci prioritizace určitého typu provozu, v tomto konkrétním případě VoIP. Součástí zadání je kromě realizace v síti IPv4 také souběžný provoz IPv6.

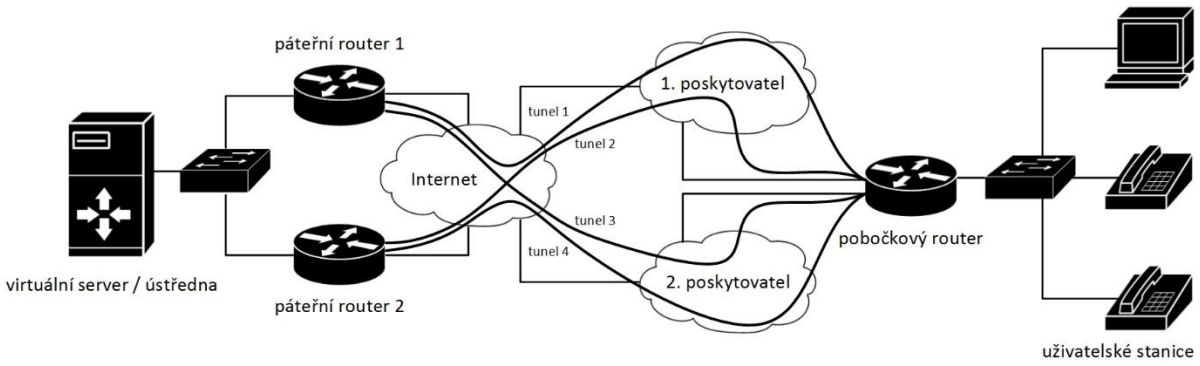
Při popisu dané problematiky se setkáme s velkým množstvím anglických pojmů, které nemají běžně používaný český ekvivalent. Rozhodl jsem se proto nepoužívat jejich doslovný překlad, ale uvádět je raději v jejich původní formě, anglicky. Prosím proto váženého čtenáře, aby v případě potřeby nahlédl do seznamu zkratek a odborných termínů. Pro jednotu projevu budu anglicky uvádět i termíny, jejichž český ekvivalent, v případě že existuje, není v praxi běžně používán.

1 Zapojení sítě

Jak již bylo uvedeno v úvodu práce, bude navržená síť sloužit primárně pro poskytování VoIP služeb zákazníkům s jednou či více pobočkami. Zadání vychází z projektu jedné pražské firmy. Práce tak bude mít své reálné využití a funkčnost celého návrhu bude dobře prověřena.

Funkci ústředny bude zajišťovat software Asterisk pracující na virtuálním serveru. Část serveru je mimo rozsah této práce, a proto se budu dále zabývat pouze infrastrukturou od páteřních routerů směrem k pobočkám zákazníka. Pro zjednodušení můžeme uvažovat, že je virtuální server (ústředna) připojen do jedné LAN sítě s páteřními routery.

Na následujícím obrázku (obr. 1) je zobrazeno výše popsané zapojení sítě pro jednu pobočku. V případě více poboček bude použito více pobočkových routerů a navázáno více tunelů na páteřní routery. Na obrázku je výše zmíněný virtuální server v jedné LAN síti s oběma páteřními routery, které jsou přes veřejný Internet a sítě obou poskytovatelů připojení k Internetu dostupné z pobočkového routeru. Tunel 1 je skrze primární připojení k Internetu navázán k prvnímu páteřnímu routeru a tunel 2 ke druhému. Analogicky jsou tunel 3 a tunel 4 navázány k oběma páteřním routerům skrze sekundární připojení k Internetu.



obr. 1: Zapojení sítě

V pravé části obrázku je naznačena LAN síť v pobočce zákazníka s uživatelskými stanicemi, které se standardně skládají z počítače a VoIP telefonu. Infrastruktura na této straně pobočkového routeru nebude v této práci dále popisována.

2 Výběr výrobce zařízení

Při výběru výrobce použitých zařízení je důležité zohlednit cenu, která výrazně ovlivní možnosti využití daného řešení, stejně jako postavení výrobce na trhu a jeho přístup k zákazníkům. Mezi nejvýznamnější výrobce síťových prvků zahrnutých do výběru patří Cisco, Juniper a MikroTik.

2.1 Cisco

Firma Cisco je jedním z největších a nejznámějších výrobců síťových zařízení. Byla založena v roce 1984 v San Franciscu, v Kalifornii, ve Spojených státech amerických, odtud také její název. V historii došlo k několika významným akvizicím, a tak je záběr vyráběných zařízení opravdu široký, od routerů a switchů, přes VoIP telefony až po set top boxy.

Jako páteřní routery od firmy Cisco je možné vybrat například router z řady 29XX a jako pobočkový například router z řady 8XX. Router Cisco řady 29XX je v České republice možné zakoupit za cenu okolo 80 000 Kč a router z řady 8XX za cenu okolo 20 000 Kč. Předpokladem je, že při odběru většího množství kusů je možné dosáhnout výrazné slevy ve výši až 50 %.

2.2 Juniper

Firma Juniper byla založena v roce 1996 v Kalifornii, ve Spojených státech amerických. Zabývá se výhradně výrobou síťových zařízení a její produkty je možné srovnávat s produkty firmy Cisco.

Jako páteřní routery od firmy Juniper je možné vybrat například router SRX1400 a jako pobočkový router Juniper SRX210. Router Juniper SRX1400 je v České republice možné zakoupit za cenu okolo 120 000 Kč a router SRX210 za cenu okolo 20 000 Kč. Předpokladem je, že při odběru většího množství kusů je možné dosáhnout výrazné slevy ve výši až 50 %.

2.3 MikroTik

Firma MikroTik byla založena v Lotyšsku v roce 1995 a v roce 2002 začala s výrobou svých vlastních síťových zřízení RouterBOARD se systémem RouterOS postaveným na

Linuxovém jádře. Jde o poměrně dynamického výrobce, který je schopný svižně reagovat na oznámené chyby v systému, vydávat k nim opravy a podporovat komunitu zákazníků.

Jako páteřní routery byly vybrány routery MikroTik Cloud Core Router CCR1016-12G. Jde o výkonné routery se šestnácti-jádrovým procesorem taktovaným na 1,2 GHz, 2 GB RAM paměti a dvanácti porty o rychlosti 1 Gbit/s. Teoretická propustnost takového zařízení dosahuje až ke 12 Gbit/s. Poskytují tak dostatečný výkon k zadanému řešení a dokonce jej převyšují pro pozdější rozšíření celé sítě.

Jako routery pro jednotlivé pobočky byly vybrány routery MikroTik RouterBOARD 750GL a 2011UiAS. Routery RB750 postačují se svým nižším výkonem pro použití v menších pobočkách a pro použití ve větších pobočkách generujících větší provoz je možné použít router RB2011UiAS. Ten disponuje i SFP portem pro případné připojení do optické sítě.

Router CCR1016 je v České republice možné zakoupit za cenu do 15 000 Kč a router RB750 za cenu do 1 500 Kč. Cena produktů výrobce MikroTik je tak výrazně výhodnější a pro projekt v začátcích jako jediná ze zde popsaných variant i ekonomicky možná. Pro vypracování této práce budou tedy použita zařízení výrobce MikroTik.

Výběr výkonnějšího routeru byl po spuštění reálného provozu upraven na MikroTik Cloud Core Router CCR1009. Konkrétní důvody jsou popsány v kapitole zabývající se reálným provozem.

3 Tunely

Pro zabezpečení přenášených dat a zvýšení kvality poskytované služby je nutné použít VPN (Virtual Private Network). Z protokolů poskytovaných routery MikroTik je možné využít protokoly PPTP (Point-to-Point Tunneling Protocol), L2TP (Layer 2 Tunneling Protocol), SSTP (Secure Socket Tunneling Protocol), OpenVPN, GRE (Generic Routing Encapsulation), VPLS (Virtual Private LAN Service) a IPsec (Internet Protocol Security).

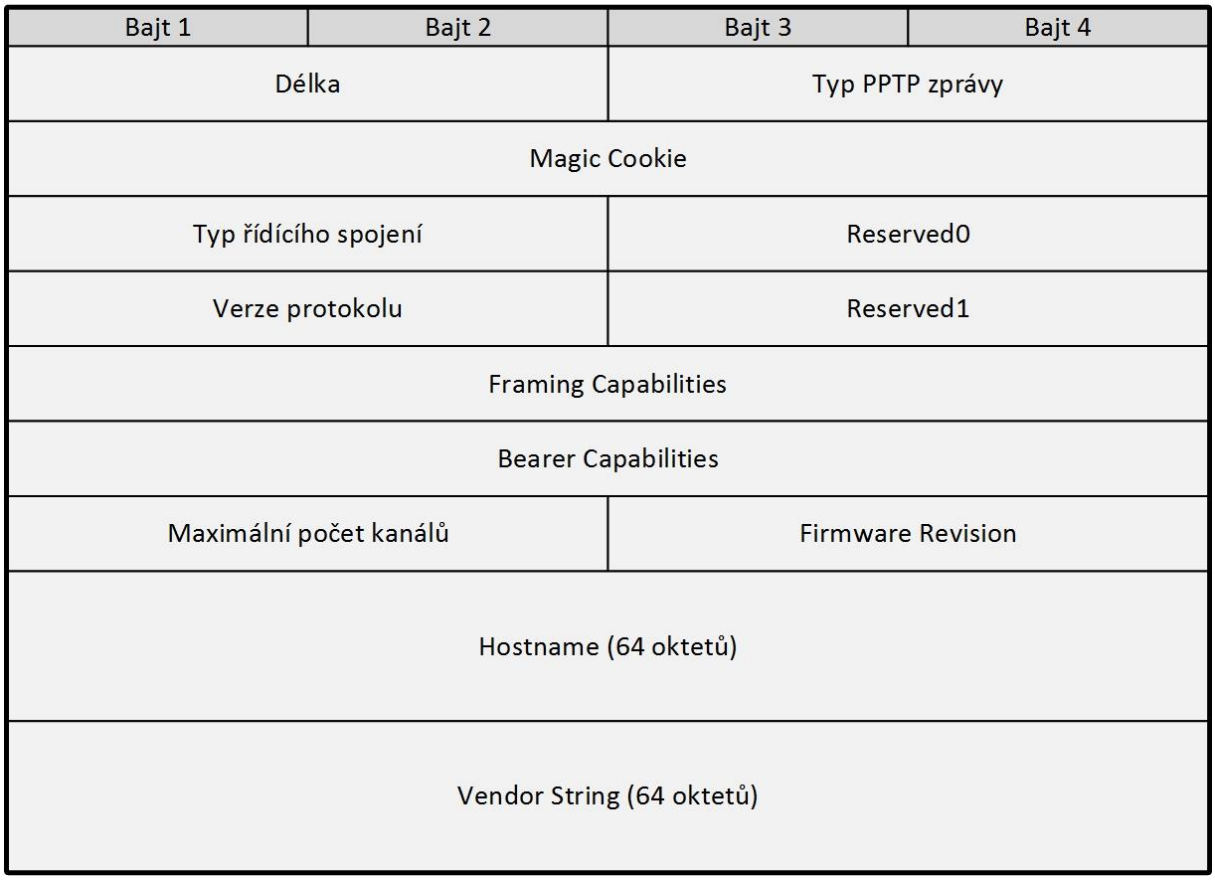
3.1 Protokol PPTP

Protokol PPTP (Point to Point Tunneling Protocol)je jedním z nejpoužívanějších tunelovacích protokolů. Jeho spojení funguje na modelu server klient. Hlavními dvěma součástmi protokolu jsou mechanizmus pro kontrolu spojení a vlastní IP tunel. Jeho specifikaci můžeme najít v RFC 2637.

3.1.1 Řídící spojení

Řídící spojení musí být vždy navázáno dříve, než je možné spojit vlastní tunel. Jde o standardní TCP spojení, přes které jsou přenášeny informace správy a řízení protokolu PPTP. Řídící spojení existuje vedle každého vytvořeného tunelu. Je zodpovědné za navázání, řízení a ukončení spojení tunelu.

Na začátku každé PPTP komunikace musí proběhnout sestavení takového spojeni. K tomu slouží požadavek o start řídícího spojení.



obr. 2: Záhlaví PPTP řídící zprávy

Délka – délka PPTP zprávy v bajtech.

Typ PPTP zprávy – 1 pro zprávu řídící spojení.

Magic Cookie – vždy nabývá hodnoty 0x1A2B3C4D. Slouží pro kontrolu „příčetnosti“ výrobce. Pokud je tato hodnota nastavena správně, je reálné, že jsou i ostatní pole implementována dle specifikace.

Typ zprávy řídícího spojení – 1 pro start řídícího spojení.

Reserved0 – pole rezervované pro pozdější použití, musí být vždy 0.

Verze protokolu – Verze PPTP protokolu, kterou chce odesílatel používat.

Reserved1 – pole rezervované pro pozdější použití, musí být vždy 0.

Odpověď na žádost o start řídícího spojení obsahuje kód popisující výsledek pokusu o sestavení spojení. Jeho hodnota je 1, pokud bylo sestavení úspěšné, 2 v případě chyby,

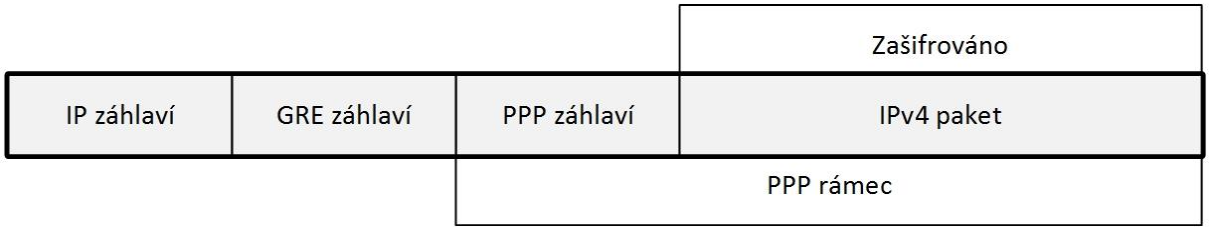
3 pokud byl kanál sestaven již dříve, 4 v případě špatné autorizace nebo 5 pokud není podporována verze protokolu, kterou používá žadatel.

Žádost o ukončení spojení obsahuje důvod ukončení. Ten může být 1 pro obecný důvod, 2 pokud nemůže být protokol druhé strany podporován nebo 3 pokud se odesílající zařízení vypíná.

Paket odpověď na žádost o ukončení spojení obsahuje kód výsledku. 1 pokud vše proběhlo standardně, 2 pokud došlo k chybě. Chyba ukončení je blíže specifikována v poli Error Code.

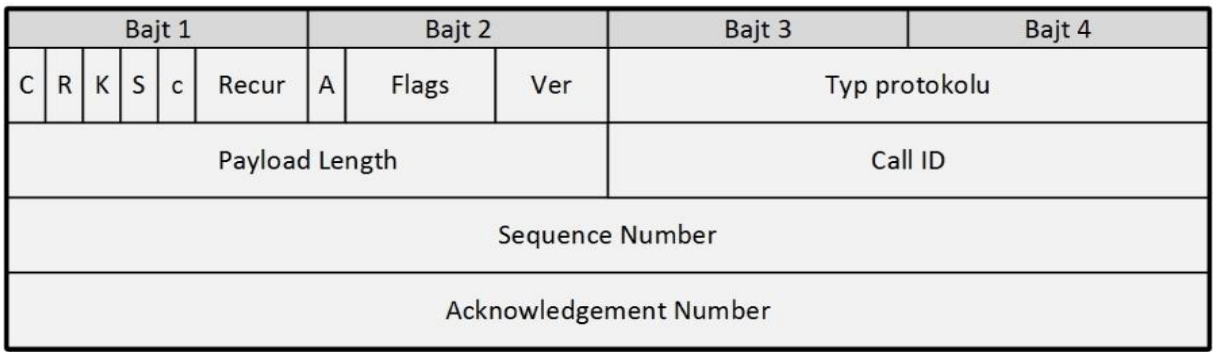
3.1.2 Protokol tunelu

Vlastní tunel využívá rozšířený protokol GRE (Generic Routing Encapsulation) a protokol PPP (Point to Point Protocol). K paketu určenému k zapouzdření a přenosu tunelem je připojeno záhlaví protokolu PPP, čímž vznikne PPP rámec, a k PPP rámci je připojeno záhlaví protokolu GRE. Celý paket je pak opatřen IP hlavičkou a odeslán. Struktura takového paketu je na obrázku (orb. 3).



obr. 3: Zapouzdření PPTP rámce

Protokol GRE je pro použití v PPTP rozšířen zejména o pole Acknowledgement Number sloužící pro potvrzení přijetí paketu. Potvrzení není použito pro opakování odeslání paketu, ale o řízení rychlosti, jakou jsou pakety tunelem odesílány.



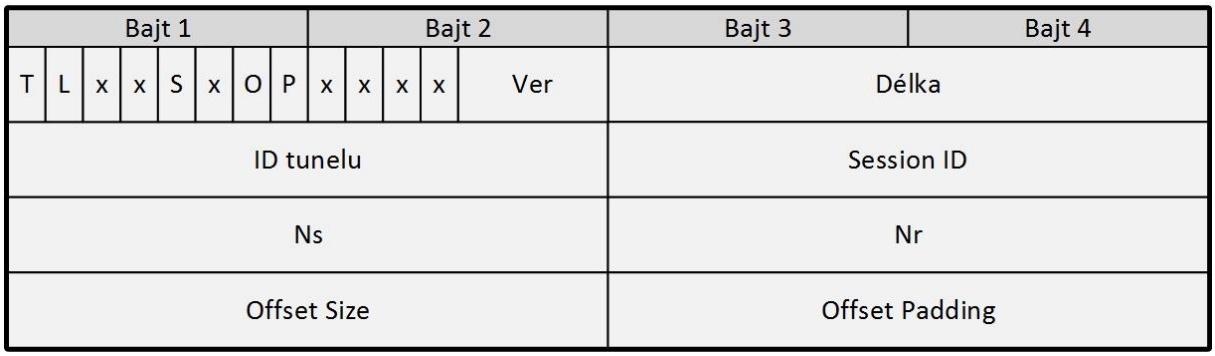
obr. 4: Záhlaví PPTP datové zprávy

3.2 Protokol L2TP

Protokol L2TP (Layer Two Tunneling Protocol) sám o sobě neposkytuje žádné šifrování provozu. K tomu je možné využít jej v kombinaci s protokolem IPsec. Na transportní vrstvě využívá protokol UDP na portu 1701 a je definován v RFC 2661. Protokol je možné provozovat v jiných než IP sítích (například Frame Relay, ATM a další).

3.2.1 Záhlaví L2TP

Protokol L2TP využívá dva druhy zpráv. Řídící zprávy jsou využity k sestavení, udržení a rušení spojení, zatímco datové zprávy zapouzdřují přenášená data. Řídící zprávy využívají kontrolního spojení pro kontrolu doručení. Při nedoručení však nejsou přeposílány. Záhlaví obou druhů zpráv jsou obdobná, liší se pouze v nepovinných polích. Záhlaví protokolu L2TP je zobrazeno na následujícím obrázku (obr. 5).



obr. 5: Záhlaví L2TP

Pole Verze musí být vždy 2.

Pole Délka určuje délku celé zprávy v bajtech.

Pole ID tunelu označuje tunel příjemce zprávy. Odesílatel i příjemce mohou mít odlišná ID jednoho tunelu.

Pole Session ID označuje jeden konkrétní přenos (session) v rámci tunelu z pohledu příjemce zprávy. Stejně jako Tunnel ID se i Session ID jednoho přenosu mohou na straně odesílatele a příjemce zprávy lišit.

Ns je sekvenční číslo zprávy. Při odeslání každé další zprávy je v tomto poli hodnota o 1 vyšší.

Nr je použito pouze u řídících zpráv. V datových zprávách musí být nastaveno na 0. Jde o předpokládanou hodnotu Ns v příští přijaté řídící zprávě. Je to tedy Ns poslední přijaté řídící zprávy zvýšené o 1.

Pole Offset Size určuje, v kolikátém bajtu od začátku začínají přenášená data (a zároveň končí L2TP záhlaví). Pokud je pole Offset Size přítomno, začínají přenášená data za poslední bajtem pole Offset Padding.

3.2.2 Řídící zprávy

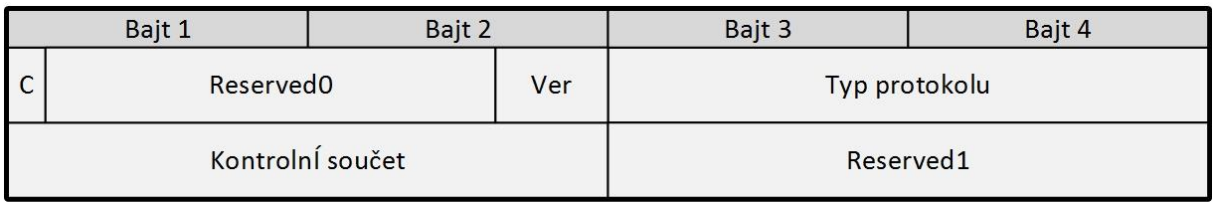
Řídící zprávy zařizují sestavení, udržení a ukončení spojení. Při sestavování spojení (tunelu) pošle jedno ze zařízení požadavek na vytvoření spojení. Protistrana mu odpoví zprávou odpovědi a proces sestavování je ukončen potvrzovací zprávou, kterou odesílá zařízení, které o spojení žádalo. Při ukončení spojení pošle jedna ze stran oznámení o ukončení spojení, na kterou není definována žádná odpověď a spojení je okamžitě ukončeno i se všemi aktivními přenosy.

Při aktivním spojení odesílá jedna ze stran Hello zprávu. Doporučená doba pro odesílání takových zpráv je 60 sekund.

Sestavení i ukončení přenosu uvnitř spojení probíhá podobně, jako u spojení. Jedna ze stran odešle požadavek na přenos, druhá strana na něj odpoví a ve třetím kroku odešle žadatel potvrzovací zprávu. Tím je přenos navázán. Pro ukončení přenosu a uvolnění prostředků je použita řídící zpráva oznámení o ukončení přenosu. Po přijetí takové zprávy musí být přenos okamžitě ukončen a není odesláno žádné potvrzení, stejně jako při ukončení spojení.

3.3 Protokol GRE

Protokol GRE je vytvořen pro jednoduché zapouzdření provozu a poslání přes IP síť. Je definován v RFC 2784. Neposkytuje žádnou formu šifrování. Je protokolem transportní vrstvy, a nevyužívá tak pro přenos protokol TCP ani UDP. Při přenosu IP sítí je v IP hlavičce označen číslem protokolu 47. Při použití protokolu GRE je před zapouzdřený paket vloženo záhlaví GRE a paket je dále zpracován, je k němu přidáno záhlaví protokolu IP a je odeslán. Hlavička protokolu GRE je na obrázku (obr. 6).



obr. 6: Záhlaví GRE

Pokud je bit C (Checksum Present) nastaven na 1, musí být v záhlaví přítomna pole Checksum a Reserved1

Pole Reserved0 Je rezervováno pro použití v budoucnu.

Pole Verze musí být nastaveno na 0.

Pole Typ protokolu určuje typ protokolu, který je v daném paketu zapouzdřen. Hodnoty pro různé protokoly jsou definované v RFC 1700. Pokud je přenášeným protokolem IP, je hodnota tohoto pole 47.

Pole Kontrolní součet obsahuje kontrolní součet prvních 16 bitů záhlaví GRE. Pro výpočet kontrolního součtu je toho pole počítáno s hodnotou 0.

Pole Reserved1 je vyhrazeno pro použití v budoucnu. Pokud je přítomné, musí být všechny bity nastaveny na 0.

3.4 Protokol IPsec

Protokol IPsec slouží k zajištění zabezpečení provozu pouze na síťové vrstvě v protokolu IP. Síťová zařízení využívající protokol IPsec jsou si rovna, ne jako server klient. Při využití protokolu IPsec je vytvořena hranice, na které se uplatňují pravidla, která jsou definovaná v databázi bezpečnostních pravidel pro práci s jednotlivými pakety. Ty mohou projít beze změny, být zabezpečeny nebo zahozeny. Chování záleží na konkrétní konfiguraci. Zabezpečený spoj může být mezi dvěma hosty, mezi hostem a zabezpečovací branou nebo mezi dvěma zabezpečovacími branami.

K zabezpečení provozu využívá protokol IPsec protokoly ESP (Encapsulating Security Payload) a AH (Authentication Header). Zatímco protokol AH poskytuje autentizaci dat, ale data nešifruje, protokol ESP je vhodný pro zajištění obou funkcí. Další fundamentální součástí protokolu IPsec jsou logické kanály (SA – Security Associations).

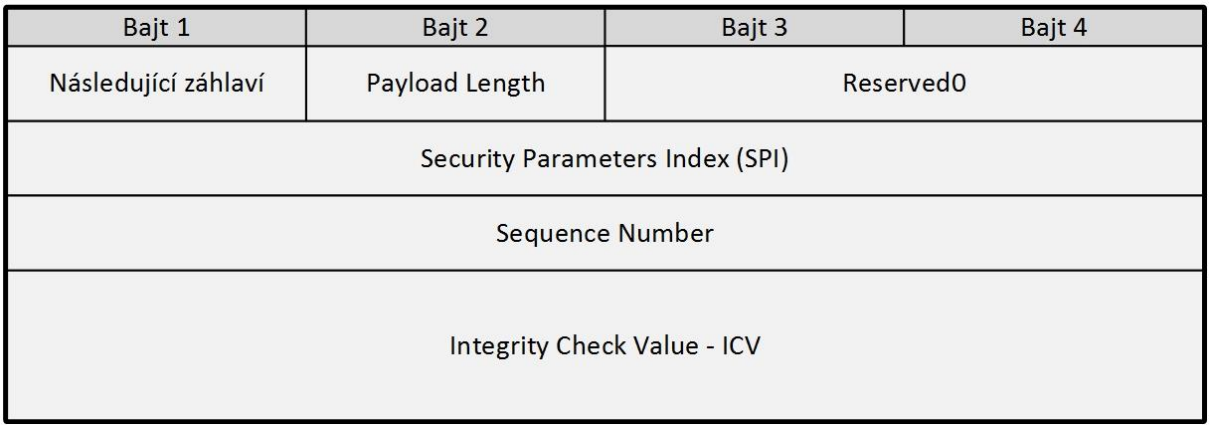
3.4.1 SA

Logické kanály (SA) v protokolu IPsec tvoří soubor algoritmů a parametrů použitých pro autentizaci a šifrování dat. SA jsou vždy simplexní, pro přenos dat v obou směrech mezi dvěma zařízeními je tak nutné vytvořit dva SA. Jeden SA může pracovat pouze s protokolem AH nebo ESP, pokud je provoz zabezpečen oběma protokoly, je potřeba jeden SA pro každý z nich.

3.4.2 Authentication Header

Authentication Header (AH) je definováno v RFC 4302. Jeho cílem je zajistit integritu a autentizaci dat. AH poskytuje autentizaci záhlaví IP paketu, ovšem pouze těch polí, která se při přenosu sítí nemohou měnit.

AH je možné využít v transportním módu nebo v módu tunelu. Při použití transportního módu je AH vloženo mezi záhlaví IP a záhlaví transportní vrstvy. Při použití módu tunelu je AH vloženo před zpracovávaný IP paket a je připojeno nové IP záhlaví, ve kterém je adresa IPsec protějšku použita jako cílová adresa. Záhlaví IP protokolu předcházející AH bude mít v poli protokol hodnotu 51 označující AH. Formát tohoto záhlaví je na obrázku (obr. 7).



obr. 7: Záhlaví IPsec Authentication Header

Pole Následující záhlaví určuje záhlaví, které následuje po AH, například 4 pro IPv4 a 41 pro IPv6.

Payload Length je počet 32 bitových slov v AH mínus 2.

Pole Reserved je rezervováno pro použití v budoucnu a musí být nastaveno na 0.

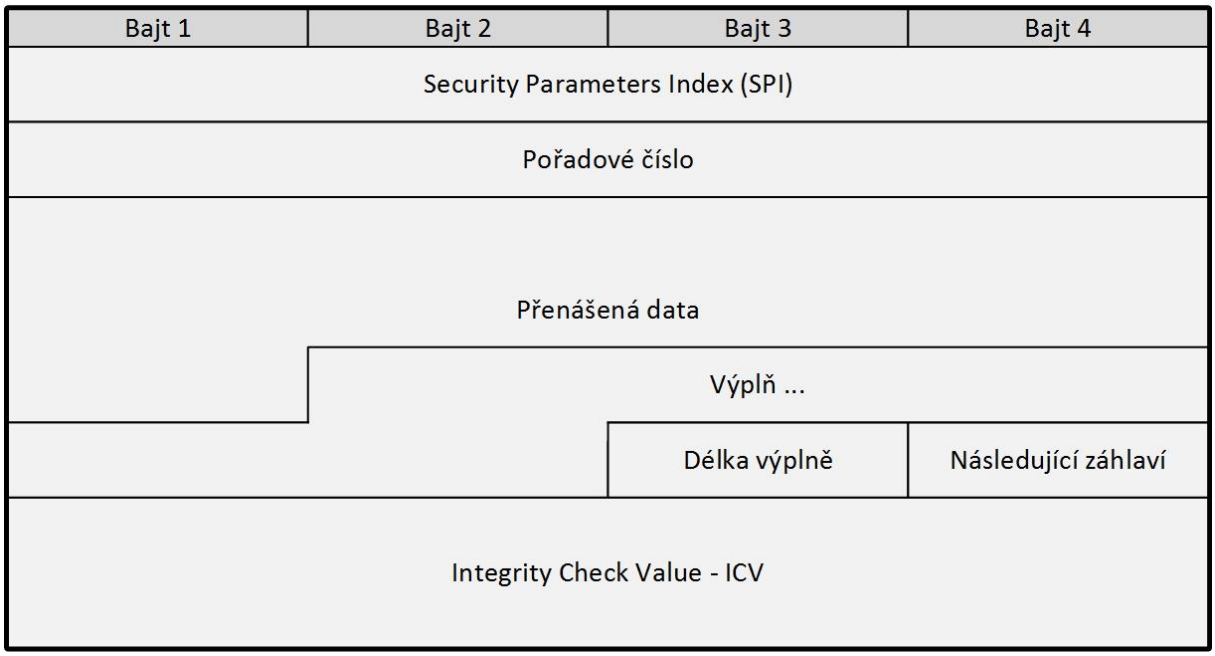
Pole SPI využije příjemce pro identifikaci logického kanálu, ke kterému paket patří.

Sequence Number pole je využito pro počítání pořadí odeslaného paketu. S každým odeslaným paketem se hodnota zvýší o 1.

Pole Integrity Check Value slouží ke kontrole integrity celého paketu. Jeho délka musí být násobkem 32 bitů.

3.4.3 Encapsulating Security Payload

Encapsulating Security Payload (ESP) je definováno v RFC 4303. Je navrženo pro poskytnutí bezpečnostních služeb v sítích IPv4 a IPv6. ESP může být, stejně jako AH, použito v transportním módu i v módu tunelu. Vložení záhlaví ESP do paketu v obou módech se řídí stejnými pravidly jako u AH. V hlavičce IP protokolu v poli protokolu identifikující ESP je vložena hodnota 50. Výsledný paket při použití ESP je zobrazen na obrázku (obr. 8).



obr. 8: Záhlaví IPsec Encapsulating Security Payload

Pole SPI je použito přijímacím zařízením pro identifikaci logického kanálu, do kterého daný paket patří.

Pole Pořadové číslo označuje pořadové číslo odeslaného paketu. Pořadí je vynulováno při navázání logického kanálu.

Pole Payload Data obsahuje vlastní přenášený paket.

Pole výplně je použito, pokud je pro zašifrování nutný určitý násobek počtu bajtů nebo pokud je žádoucí zajistit, aby pole obsahující přenášený paket končilo na hranici 4 bajtů.

Pole Délka výplně určuje počet bajtů výplně.

Pole Následující záhlaví určuje protokol přenášený v ESP. Hodnota je stejná, jako hodnota v poli Protocol v hlavičce protokolu IPv4.

Integrity Check je kontrolní součet celého paketu.

4 Výběr routovacího protokolu

Mezi nejvíce používané routovací protokoly, nad kterými je potřeba při výběru použitého přemýšlet, patří bezesporu RIP (Routing Information Protocol), EIGRP (Enhanced Interior Gateway Routing Protocol), OSPF (Open Shortest Path First), IS-IS (Intermediate System to Intermediate System) a BGP (Border Gateway Protocol). Protokol RIP není pro řešení zadání vhodný kvůli své jednoduchosti a špatné škálovatelnosti ve větších sítích. Protokol EIGRP (Enhanced Interior Gateway Routing Protocol) je CISCO proprietární a není ho tedy možné na zařízeních MikroTik použít. Protokol IS-IS se svými vlastnostmi velmi blíží protokolu OSPF, není routery MikroTik podporován a kvůli jeho nižší oblibě by bylo obtížnější hledat podporu při řešení nestandartních situací. Proto jeho použití nebudu zvažovat. Dále se tedy budu věnovat pouze možnostem, které nabízejí protokoly OSPF a BGP

4.1 OSPF

Protokol OSPF byl vyvinut skupinou pro vývoj OSPF v rámci komise pro technickou stránku Internetu (Internet Engineering Task Force - IETF). Byl navržen konkrétně pro prostředí TCP/IP s podporou CIDR (Classless Inter-Domain Routing – beztřídní routování, tzn. routování sítí s jinou maskou než maskou jejich třídy). Vývoj byl soustředěn na protokol, který rychle reaguje na změny v topologii a není náročný na přenosové pásmo. Mezi hlavní nevýhody protokolu OSPF patří vysoká výpočetní náročnost a vyšší využití paměti z důvodu ukládání duplikátních informací. Specifikaci OSPF můžeme najít v RFC 2328.

4.1.1 Link-State databáze

Link-State databáze je napočítanou reprezentací topologie oblasti, ve které daný router funguje. Skrze tuto databázi router „vidí“ stav spojení (anglicky „link“) v dané oblasti, odtud také název Link-State databáze.

Databáze je svým způsobem tabulka routerů a sítí se zaznamenanými spoji mezi nimi. Můžeme si ji představit jako obrázek znázorňující topologii sítě, ve kterém jsou routery a sítě zobrazeny jako body a spojení mezi nimi jako spojnice. V Link-State databázi je zaznamenáno nejen to, které routery jsou navzájem v přímém dosahu a které sítě mají

v dosahu, ale i cena (cost) jednotlivých spojů. Cena spoje může být svázána s jakýmkoli významným kritériem a je využita jako metrika pro hodnocení výhodnosti spoje.

4.1.2 OSPF Area

Protokol OSPF rozděluje autonomní systémy do jednotlivých oblastí (Area) pro jasnou správu celého systému. V každé z oblastí běží vlastní instance routovacího algoritmu s vlastními daty oddělenými od dat ostatních oblastí. Topologie každé oblasti není vidět zvenku, pouze zevnitř. Žádný z routerů, který není v dané oblasti, tak nezná její topologii, zná však kompletní topologii své vlastní oblasti. Tento princip umožňuje minimalizaci přenosu routovacích informací ve srovnání s přístupem, ve kterém bychom celý autonomní systém považovali za jednu oblast.

Všechny routery v jedné oblasti mají vždy identickou Link-State databázi. Pokud router patří do dvou a více oblastí, má samostatnou databázi pro každou oblast a nazýváme jej hraničním routerem.

4.1.3 Výpočet routovací tabulky

Pro výpočet nejlepších cest do různých destinací je sestaven strom všech cest v dané oblasti. Nejprve jsou uvažovány pouze spoje mezi routery a do tranzitních sítí. Takový strom je počítán pomocí Dijkstrova algoritmu z Link-State databáze. Později jsou do stromu zahrnuty i sítě, ve kterých provoz vždy končí (stub networks). Dále jsou vypočítány cesty do oblastí, kterých není daný router součástí.

4.2 BGP

Protokol BGP se řadí mezi exterior gateway protokoly a je určen především pro výměnu routovacích informací mezi většími celky (autonomními systémy), uvnitř kterých je standardně použit některý z interior gateway protokolů (jako třeba již zmíněný OSPF), určený pro routování uvnitř autonomního systému. Každý z routerů je zde označován jako peer a má také své router ID. Ve srovnání s protokolem OSPF se v BGP neposílá kompletní informace o topologii celé sítě, ale pouze informace o směrech do jednotlivých sítí. K těm jsou opět přidány různé priority a metriky, na základě kterých je možné vybrat „lepší“ ze dvou informací o dostupnosti jedné sítě.

Routovací informace v BGP podporují pouze routování na základě cílové adresy. Všechna pravidla použitá v BGP musí tomuto způsobu routování odpovídat.

Protokol BGP využívá k přenostu protokol TCP, a nemusí proto definovat mechanismy pro fragmentaci, opakování přenosu nebo zajištění doručení ve správném pořadí. BGP naslouchá na portu 179.

4.2.1 IBGP vs EBGP

Autonomní systém je skupina routerů pod správou jedné organizace, která má z pohledu ostatních autonomních systémů jednotný interní routovací plán a prezentuje sítě, které jsou skrze něj dostupné. Pokud mluvíme o IBGP (internal BGP), mluvíme o instanci BGP mezi routery v jednom autonomním systému. Oproti tomu EBGP (external BGP) je instance BGP mezi routery různých autonomních systémů.

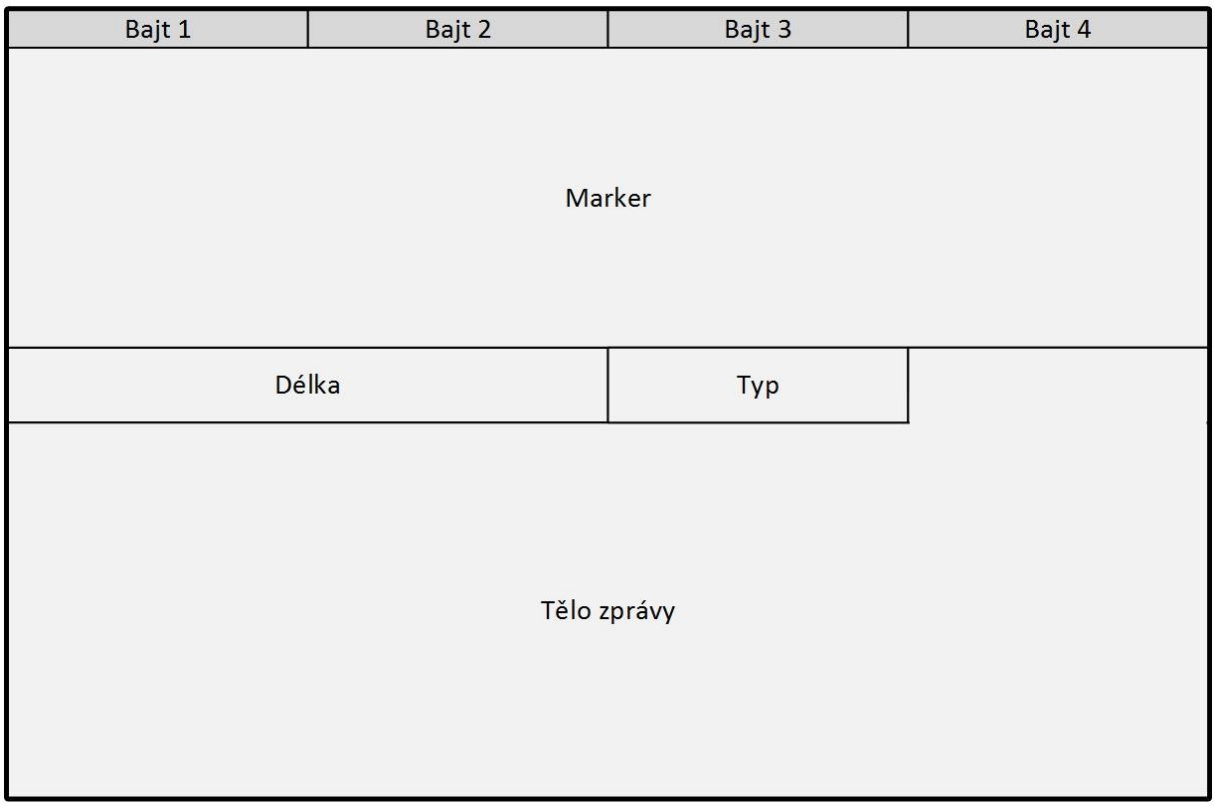
Číslo autonomního systému bylo až do roku 2007 definováno jako 16 bitové a poté byla jeho délka zvýšena na 32 bitů. Čísla autonomních systémů jsou přidělována organizací IANA. Privátní rozsahy čísel autonomních systémů jsou 64512 – 65534 a 4200000000 – 4294967.

4.2.2 RIB

V RIB (Routing Information Base) jsou uloženy všechny routovací informace. RIB je rozdělena do tří částí. Adj-RIBs-In obsahuje routovací informace přijaté v příchozích UPDATE zprávách. Tyto informace budou použity v rozhodovacím procesu. Loc-RIB obsahuje lokální routovací informace po aplikování lokálních pravidel na informace z Adj-RIBs-In. Adj-RIBs-Out obsahuje routovací informace, které byly lokálně vybrány pro propagaci a které budou propagovány odchozími UPDATE zprávami.

4.2.3 Formát BGP zpráv

Všechny BGP zprávy jsou přenášeny pomocí TCP protokolu. Nejdelší zpráva může mít 4096 bajtů, zatímco nejkratší 19 bajtů.



obr. 9: Formát BGP zprávy

Záhlaví BGP zprávy je pevně definovaná a pro všechny zprávy má stejný formát a velikost (19 bajtů). Zpráva může, ale také nemusí, obsahovat tělo zprávy v závislosti na svém typu. Zpráva bez těla bude mít velikost dříve zmíněných 19 bajtů.

Pole Marker v záhlaví BGP zprávy je vytvořené pro kompatibilitu, má velikost 16 bajtů a musí být vždy nastaveno na samé jedničky.

Délka indikuje délku celé zprávy včetně záhlaví.

Typ je jednobajtové pole, které určuje typ zprávy. Obsahuje jednu z hodnot: 1 – OPEN, 2 – UPDATE, 3 – NOTIFICATION nebo 4 – KEEPALIVE.

4.2.3.1 Zpráva OPEN

Poté, co je navázáno TCP spojení, je odeslána zpráva OPEN. Ta je potvrzena zprávou KEEPALIVE. Kromě záhlaví obsahuje i další pole a její minimální velikost je 29 bajtů.

Verze označuje použitou verzi protokolu BGP. Aktuální verze je 4. Pole můj autonomní systém indikuje autonomní systém odesílatele. Hold Time je navrhovaná maximální doba mezi přijetím jednotlivých KEEPALIVE a UPDATE zpráv. Po přijetí zprávy OPEN je

výsledná hodnota počítána z konfigurované a přijaté hodnoty hold time. BGP identifikátor je identifikátor odesílatele. Je nastaven na jeho IP adresu. Délka volitelných parametrů určuje celkovou délku polí volitelných parametrů. Musí být nastavena na 0, pokud nejsou použity žádné volitelné parametry. V poli volitelných parametrů jsou všechny použité volitelné parametry zakódovány do trojic (ty parametru, délka parametru, hodnota parametru).

4.2.3.2 Zpráva UPDATE

Pro přenos routovacích informací se používají zprávy typu UPDATE. Tato zpráva je využívána k oznamování dostupnosti a nedostupnosti jednotlivých směrů. Kromě povinného záhlaví BGP zprávy obsahuje informaci o nedostupných odebraných směrech (Withdrawn Routes), atributy cesty (Path Attributes) a jednotlivé dostupné směry (NLRI – Network Layer Reachability Information). V jedné zprávě UPDATE mohou být dohromady oznamovány pouze směry se stejnými atributy cesty.

4.2.3.3 Zpráva NOTIFICATION

V případě chyby je odeslána zpráva NOTIFICATION a BGP spojení je okamžitě ukončeno. Obsahuje pole kódu chyby (Error Code), upřesňující kód chyby (Error Subcode) a data pro hlubší diagnostiku problému.

4.2.3.4 Zpráva KEEPALIVE

Pro udržení jednotlivých spojení si mezi sebou jednotlivé routery posílají KEEPALIVE zprávy. Posílají je periodicky a interval odesílání je vypočten z dříve dohodnuté hodnoty hold time. Rozumný interval je třetina hodnoty hold time, ale konkrétní implementace může být odlišná.

4.2.4 BGP Path Attributes

BGP Path Attributes jsou atributy jednotlivých směrů oznamované v BGP zprávách. Jsou rozděleny na všem známé povinné, všem známé volitelné, doplňkové přenositelné a doplňkové nepřenositelné. Všem známé atributy musí být rozpoznávány všemi implementacemi BGP nezávisle na výrobci. Všem známé povinné atributy musí být obsaženy v každé zprávě obsahující informaci o dostupnosti jednotlivých směrů (NLRI), zatímco volitelné atributy v takových zprávách mohou, ale také nemusí, být obsaženy. Na rozdíl od všem známých atributů jsou doplňkové atributy plně závislé na konkrétní

implementaci protokolu BGP. Takový atribut může být přenositelný nebo nepřenositelný, platný pouze lokálně.

4.3 BFD

BFD (Bidirectional Forwarding Detection) není sám o sobě routovacím protokolem. Je však možné jej využít v úzké souvislosti s jedním z nich. Zajišťuje detekci chyby komunikace. BFD naslouchá na portu 3784 a jeho provoz je přenášen pomocí protokolu UDP.

V pravidelném, předem dohodnutém intervalu, odesílá řídící zprávy, na které protistrana odpovídá pakety odpovědi.

5 Prioritizace provozu

Pokud bychom se vším provozem v síti zacházeli podle stejných pravidel, mohlo by dojít k vyčerpání síťových prostředků méně důležitým provozem, a zajištění důležitých služeb by tak mohlo být ohroženo. Pro zajištění kvality služeb je tedy nutné přistupovat různě k různým typům provozu a vytvořit několik stupňů priorit provozu. Důležitější služby musí sítí procházet s vyšší prioritou, než služby méně důležité, a může jim být vyhrazena část rychlosti konektivity. Při takovém řešení každá z využívaných služeb nebo jejich skupin využívá vlastní část rychlosti a o případně nerozdělenou část se v případě potřeby rozdělí podle přiřazené priority.

5.1 Využití IP adres

Základní možností pro identifikaci různých typů provozu a jejich skupin je rozdělení podle zdrojové a cílové IP adresy v záhlaví použitého IP protokolu. Zásadní nevýhodou takového řešení je nemožnost identifikace typu provozu při využití různých služeb na jednom serveru jedním klientem. Pokud například sídlí VoIP i HTTP server na jednom fyzickém serveru pod jednou IP adresou, není podle analýzy IP adres možné určit, ke které ze služeb daný provoz patří.

5.2 Využití DSCP

Další možností identifikace typu provozu je využití hodnoty DSCP (Differentiated Services Code Point). Jedná se o hodnotu přenášenou v poli Type of Service v záhlaví protokolu IPv4 a v poli Traffic Class v hlavičce protokolu IPv6. Význam DSCP je právě v určení priority provozu. Jednotlivá koncová zařízení mohou při odeslání paketu určit, s jakou prioritou má sítí procházet nastavením, právě nastavením hodnoty DSCP na určitou hodnotu. Výhodou použití hodnoty DSCP je, že není kromé IP potřeba analyzovat další záhlaví v paketu. Nevýhodou je nestejná implementace DSCP různými výrobci. Hodnota DSCP také jednoznačně neurčuje přenášenou službu, ale pouze její prioritu.

5.3 Využití čísla protokolu

Při zjišťování přenášeného protokolu aplikační vrstvy je nutné analyzovat záhlaví transportní vrstvy. To přináší zvýšenou výpočetní náročnost a tím i zpomalení. Při takové analýze je zjištěno číslo protokolu, který je transportní vrstvou přenášen, a je tak možné

jednoznačně určit, o jakou službu se jedná. Je možné určit protokoly, kterým bude přiřazena vyšší priorita, a zároveň nastavit hodnotu DSCP, aby byla analýza provozu při průchodu dalším routerem jednodušší a rychlejší.

6 IPv6

IPv6 protokol byl vyvinut jako nová generace IP protokolu. Původní verzí je dnes běžně používaný a v Internetu nejrozšířenější IPv4 protokol. Jedním z hlavních důvodů pro změnu protokolu je nedostatečná kapacita adres IPv4, která je již prakticky vyčerpaná. Tento problém je v IPv6 vyřešen zvětšením velikosti adresy z 32 bitů na 128 bitů. Protokol IPv6 je definován v RFC 2460.

6.1 Adresace

Jak je uvedeno výše, IPv6 adresa má velikost 128 bitů. Její zápis je v pro lidi čitelné formě osm hexadecimálních čtveřic oddělených dvojtečkami. Zápis podléhá pravidlům pro zjednodušení a jednoznačnost. Nula nebo více nul na začátku každé čtveřice mohou být vynechány a jedna nebo více za sebou jdoucích čtveřic nul mohou být nahrazeny dvojtečkou. Adresa 2001:0002ac:fe01:0000:0010:0000:0000 tak může být zkrácena na 2001:2:ac10:fe01:0:10::.

6.1.1 Adresní rozsahy

Stejně jako v IPv4 definuje i v IPv6 konkrétní adresní rozsah maska sítě. Maska sítě říká, kolik prvních bitů adresy určuje síť a kolik bitů je možné v rámci sítě měnit. Doporučení je, aby byly masky násobky čtyř. Jednotlivé menší rozsahy je pak možné jednodušeji spočítat a lidská práce se tak zjednoduší. Dalším doporučením je nepřidělovat jednotlivým sítím rozsahy menší než s maskou /64. Takových sítí je v celém adresním rozsahu IPv6 možné vytvořit okolo 2^{64}.

6.1.2 Typy adres IPv6

V IPv6 jsou definovány i adresní rozsahy určené pro specifické použití. Těmi jsou:

::/128 – tato adresa může být použita pouze jako zdrojová při inicializaci, pokud zařízení nemá žádnou adresu přidělenou. Ekvivalent v IPv4 je adresa 0.0.0.0/32.

::1/128 – adresa loopbacku. Ekvivalent v IPv4 je rozsah 127.0.0.1/32.

::ffff:0:0/96 – speciální rozsah pro fungování IPv4 a IPv6 zároveň. IPv4 adresa je uložena v posledních 32 bitech, například ::ffff:8.8.8.8. Nemá ekvivalent v IPv4

fc00::/7 – lokální rozsah. Adresy z tohoto rozsahu nemusí být globálně unikátní a jsou použity pouze v lokálních sítích domácností nebo podniků. Tyto adresy nejsou routovány ve veřejném Internetu. Ekvivalent v IPv4 jsou rozsahy 10.0.0.0/8, 172.16.0.0/12 a 192.168.0.0/16.

fe80::/10 – adresy unikátní v jedné fyzické síti. Pakety se zdrojovou nebo cílovou adresou z tohoto rozsahu nesmí být přeposílány. Ekvivalent v IPv4 je rozsah 169.254.0.0/16.

2001::/32 – rozsah vyhrazený pro protokol Teredo pro přechod z IPv4 na IPv6. Nemá ekvivalent v IPv4.

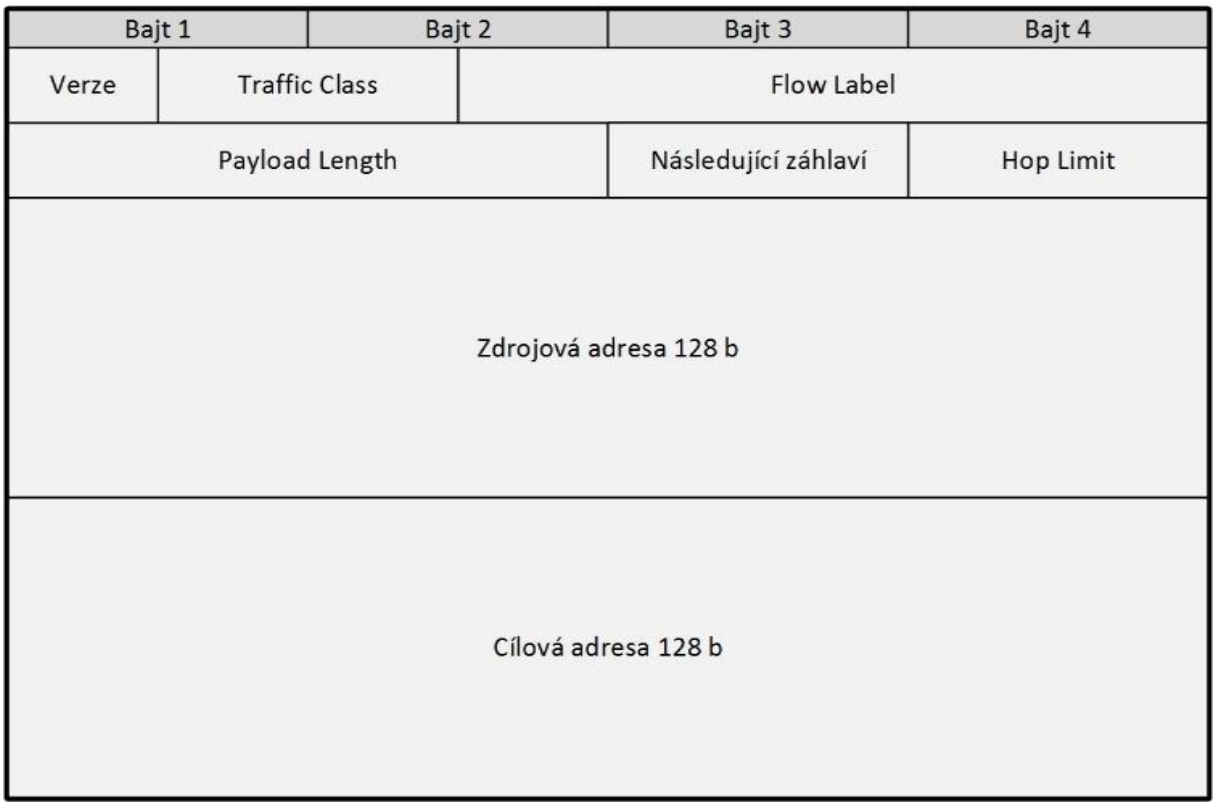
2002::/16 - rozsah vyhrazený pro protokol 6to4 pro přechod z IPv4 na IPv6. Nemá ekvivalent v IPv4.

2001:db8::/32 – Adresy vyhrazené pro dokumentaci a příklady. Nikdy nesmí být použity jako zdrojová nebo cílová adresa. Ekvivalent v IPv4 jsou rozsahy 192.0.2.0/24, 198.51.100.0/24 a 203.0.113.0/24.

ff00::/8 – Rozsah adres vyhrazený pro multicast. Ekvivalent v IPv4 je rozsah 224.0.0.0/4.

6.2 Záhlaví IPv6 paketu

Záhlaví definované v protokolu IPv6 je proti původní verzi protokolu IP výrazně větší, ale zároveň značně zjednodušené pro rychlejší a efektivnější zpracování při směrování. Větší velikost záhlaví je dána výrazným zvětšením adresního pole z 32 bitů na 128 bitů. Celková velikost záhlaví se tak zvýšila z 24 bajtů (v IPv4) na 40 bajtů. Změna však není pouze ve velikosti dvou adresních polí (zvětšení o 24 bajtů), ale také v odebrání některých polí záhlaví IPv4 (zmenšení o 8 bajtů). Záhlaví IPv6 je na na obrázku (obr. 10).



obr. 10: Záhlaví IPv6 Paketu

Pole Verze označuje verzi protokolu. Pro IPv6 je jeho hodnota vždy 6. Velikost pole version je 4 bity.

Pole Traffic Class je použito pro hodnoty DSCP (Differentiated Services Code Point), známou z IPv4, a ECN (Explicit Congestion Notification), která umožňuje notifikaci zahlcení sítě. Velikost pole Traffice Class je 8 bitů.

Hodnota Flow Label je určena pro řízení směru toku paketů. Pokud je tato hodnota nastavena na nenulovou hodnotu, pakety do jednoho cíle mají odcházet vždy stejným směrem, aby nedošlo k jejich přeházení. Velikost pole Flow Label je 20 bitů.

Payload Lenght je velikost IPv6 paketu za záhlavím v bajtech. Velikost pole Payload Length je 16 bitů.

Pole Následující záhlaví určuje typ záhlaví, které následuje IPv6 záhlaví. Hodnoty jsou stejné, jako v poli Protocol v záhlaví protokolu IPv4 (například 6 pro TCP nebo 17 pro UDP). Velikost pole Next Header je 8 bitů.

Hodnota Hop Limit je snížena o 1 při každém přeposlání paketu. Pokud je hodnota po snížení 0, je paket zahozen. Hodnota při odeslání paketu určuje maximální počet routerů, přes které může paket projít, než dorazí do cíle. Jedná se o ekvivalent pole Time to Live v záhlaví protokolu IPv4. Velikost pole Hop Limit je 8 bitů.

Zdrojová a cílová adresa jsou dříve diskutované IPv6 adresy zdrojového a cílového zařízení.

6.3 Neighbour Discovery Protocol

Neighbor Discovery Protocol (NDP) je protokol pro sestavení a udržení spojení mezi IPv6 zařízeními v jedné síti. NDP má 5 typů zpráv: Router Solicitation, Router Advertisement, Neigbor Solicitation, Neighbor Advertisement (ty jsou rozesílány pomocí protokolu ICMPv6) a zpráva Redirect.

Když se síťové zařízení připojí k sítí, pokusí se nejdříve najít router. Pošle tedy Router Solicitation zprávu, která je odeslána na multicast IPv6 adresu všech routerů ff02::2. Zdrojová adresa takového paketu je buď IPv6 nespecifikovaná (::), nebo adresa síťového rozhraní unikátní pro jednu fyzickou síť (například fe80::aa:bb:cc:dd).

Router Advertisement zprávu posílá okamžitě po přijetí Router Solicitation zprávy, ale také v pravidelných intervalech na multicast adresu všech síťových zařízení (ff02::1). Pokud je zpráva odeslána v odpovědi na Router Solicitation zprávu, je cílová adresa adresa síťového prvku, který Router Solicitation zprávu poslal. Obsahuje informace o IPv6 adresním rozsahu použitém v síti, který může zařízení využít, MTU (maximální velikost paketu pro přenos), specifické routy a informaci, je-li možné využít protokol SLAAC pro automatickou konfiguraci IPv6 adresy. Zdrojová adresa takového paketu je adresa síťového rozhraní routeru unikátní pro jednu fyzickou síť (například fe80::aa:bb:cc:ee).

Zpráva Neighbor Solicitation nahrazuje funkci protokolu ARP používaného v IPv4 sítích. Taková zpráva může být poslána pro zjištění fyzické adresy rozhraní jiného síťového zařízení nebo k ověření, zda je adresa uložená v cache paměti stále platná. Odpověď na Neighbor Solicitation zprávu je zpráva Neighbor Advertisement.

Zpráva Redirect říká síťovému zařízení, že určitá síť je dosažitelná přes jiný router nebo v místní fyzické síti.

6.4 DHCPv6

Dynamic Host Configuration Protocol version 6 (DHCPv6) je protokol pro dynamické přidělování IPv6 adres nahrazující protokol DHCP, který je použit pro přidělování adres IPv4. RA zpráva NDP protokolu obsahuje různé příznaky. Příznak M indikuje, zda je k dispozici DHCPv6 server přidělující IPv6 adresy. Pokud server přiděluje pouze DNS další konfiguraci bez adres, je nastaven příznak O. Příznak A indikuje, zda je možné použít pro konfiguraci IPv6 adresy protokol SLAAC. Aby síťové zařízení nepoužilo pro konfiguraci defaultně využívaný protokol SLAAC ale DHCPv6 server, musí být nastaveny příznaky S a A.

6.5 SLAAC

Stateless Address Autoconfiguraction (SLAAC) je protokol pro automatickou konfiguraci IPv6 adres. Při použití SLAAC si síťové zařízení vygeneruje EUI-64 identifikátor síťového rozhraní. Ten je definován jako MAC adresa, do které je přesně doprostřed vloženo ff:fe. Dále se vygeneruje modifikovaný EUI-64 identifikátor invertováním sedmého bitu EUI-64 identifikátoru. Výsledných 64 se připojí k adrese sítě. Pokud je třeba vytvořit adresu z lokálního rozsahu pouze pro komunikaci na místní síti, připojí se adresa již dříve zmíněného lokálního rozsahu (fe80::). Pokud se vytváří adresa pro použití v Internetu, použije se adresa rozsahu získaná při komunikaci pomocí protokolu NDP (pokud je možnost použití SLAAC povolena).

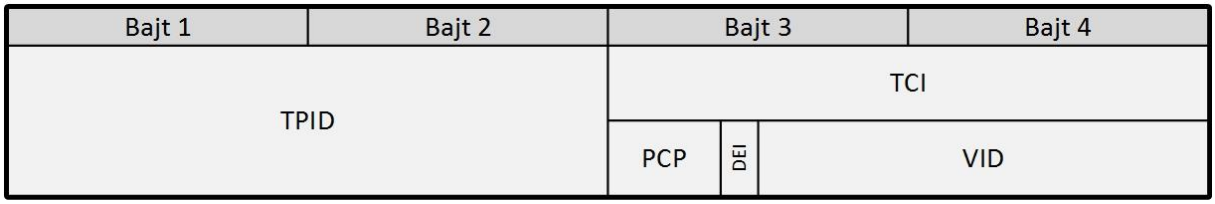
Pokud je MAC adresa zařízení například 00:53:42:7e:bf:27 bude EUI-64 identifikátor 0053:42ff:fe7e:bf27. Po modifikaci pak 0253:42ff:fe7e:bf27. Adresa z lokálního rozsahu pak bude fe80::253:42ff:fe7e:bf27.

7 Oddělení provozu

Pro zvýšení zabezpečení v síti LAN je vhodné separovat jednotlivé druhy provozu. Nejčastějším způsobem pro oddělení provozu v jedné fyzické síti je použití VLAN. Použití VLAN rozdělí provoz na druhé vrstvě. Odstraní tak potřebu budování druhé fyzické sítě a pořizování dalších zařízení pracujících na druhé vrstvě.

Funkce VLAN je definovaná v IEEE 802.1Q. Při použití VLAN je Ethernet rámec označen číslem VLAN. Tato hodnota je uložena v hlavičce rámce. Proto musela být specifikací IEEE 802.3ac zvýšena maximální velikost Ethernet rámce z původních 1518 bajtů na 1522 bajtů.

Číslo VLAN je v poli protokolu 802.1Q v záhlaví Ethernet rámce mezi polem zdrojové adresy a polem označujícím velikost rámce. Jeho velikost je 4 bajty a skládá se z polí TPID a TCI. Pole TCI je dále rozděleno na hodnoty PCP, DEI a VID. Celé pole protokolu 802.1Q je vidět na obrázku (obr. 11):



obr. 11: Struktura pole při použití VLAN

Pole TPID je vždy nastaveno na hodnotu 0x8100.

PCP (Priority Code Point) je tříbitové pole označující třídu služby podle její priority.

DEI (Drop Eligible Indicator) označuje rámce, které mohou být při zahlcení sítě zahozeny a jeho velikost je 1 bit.

Pole VID (VLAN Identifier) obsahuje přímo číslo VLAN. Má velikost 12 bitů a čísel VLAN je tak 4094. Hodnota 0x000 indikuje, že je rámec přenášen bez čísla VLAN a hodnota 0xFFF je rezervována pro speciální použití.

8 Konfigurace sítě

Jak již bylo zmíněno v úvodu práce, cílem projektu je vytvořit síť vhodnou pro poskytování pokročilých telekomunikačních služeb s vysokými nároky na stabilitu, zabezpečení a dostupnost.

Pro realizaci sítě se jako nejvýhodnější výrobce zařízení jeví, zejména kvůli ceně zařízení, MikroTik. Konkrétní kalkulace a porovnání s ostatními navrhovanými výrobci jsou popsány výše. Mezi vybranými routery budou nejprve nakonfigurovány tunely pro zabezpečení provozu. Vytvořenou VLAN bude možné využít pro spuštění routovacího protokolu, který routerům umožní routovat všechna data požadovanými směry mezi koncovými stanicemi. Pro zajištění plynulého provozu v síti bude následně nutné vytvořit pravidla pro jeho prioritizaci. V síti bude také umožněn provoz protokolu IPv6, který se pro relevantní použití prakticky všech sítových aplikací v budoucnu jeví jako nutný.

Pro konfiguraci předpokládám, že ve vzdálené lokalitě zákazníka je nainstalován router s primárním a záložním připojením k Internetu. Dále pak dva páteřní routery pro routování provozu připojené k Internetu konektivitou s vysokou propustností, aby nedošlo k jejímu přetížení.

8.1 Konfigurace tunelů

Prvním krokem implementace navržené sítě je konfigurace tunelů. Jde o vytvoření virtuálních rozhraní a spojů mezi routery a dá se přirovnat k vytvoření fyzických spojů. Virtuální rozhraní budou fungovat prakticky stejně, jako fyzická, bude možné přiřadit jim IP adresu a budou v aktivním, případně neaktivním stavu.

Vybranou technologií je GRE tunel přenášený protokolem IPsec. Protokol GRE byl vybrán pro svou jednoduchost a málo neužitečných informací přidaných k provozu. Svým způsobem jej využívá i protokol PPTP, jehož sestavení je ovšem složitější, má větší overhead a stejně jako GRE nepodporuje šifrování provozu. Pokud by byl k realizaci vybrán protokol L2TP, bylo by i tak nutné pro šifrování využít protokol IPsec.

Při konfiguraci rozhraní protokolu GRE se konfiguruje pouze IP adresa zdroje a cíle provozu, MTU, DSCP vytvořeného provozu a povolení fragmentace. Protokol GRE je bezstavový, a proto je jeho rozhraní aktivní vždy, když zná cestu k cílové adrese, a to

i v případě, že je cílové zařízení nedostupné. Právě proto výrobce MikroTik přidal možnost keepalive, která umožní ověření dostupnosti cíle. Tuto možnost však nevyužiji a ověření správné funkčnosti tunelu bude realizováno pomocí funkce routovacího protokolu (bude popsáno dále).

Protokol GRE sám o sobě umí pracovat, pouze pokud jsou IP adresy obou zařízení přímo dostupné, ani jedna z nich tedy nesmí být privátní v síti využívající NAT. Faktem je, že hodně připojení lokálních poskytovatelů služeb využívají NAT a koncovému uživateli přidělují pouze privátní IP adresu. Toto omezení však bude odstraněno použitím protokolu IPsec, který takovou konfiguraci umožňuje.

Právě kvůli častějšímu využití připojení s NAT bude nutné tento fakt při konfiguraci IPsec zvážit.

8.2 Konfigurace routovacího protokolu

V tomto bodě, kdy je již k dispozici virtuální propojení (tunely) mezi routery, je možné začít s konfigurací vybraného routovacího protokolu. Tím vybraným je protokol BGP, především protože poskytuje nejširší možnosti, co se týče řízení směru toku provozu, z uvažovaných protokolů. Konkrétně bude využit pro interní routování, tedy jako iBGP.

Pro základní konfiguraci protokolu BGP je nutné nejprve nastavit jeho instanci a poté jednotlivé protějšky (peer). Při nastavování instance protokolu BGP je nutné zadat číslo autonomního systému, do kterého daný router patří. Při nastavování protějšku je nutné zadat číslo autonomního systému, které je nakonfigurované v jeho instanci. V našem případě jsou obě čísla stejná, jedná se o verzi IBGP.

Toho nastavení bylo později změněno. Konkrétní důvody a nové nastavení jsou uvedeny v kapitole o reálném provozu.

Ve chvíli, kdy jsou oba routery nakonfigurované správně, je BGP Session navázána. Pro předávání routovacích informací je však nutné nastavit, které mají být použity. To je v případě routerů MikroTik možné nastavit přímo v nastavení instance, kde nastavujeme distribuci informací o připojených sítích, statických směrech, případně distribuci informací z jiných routovacích protokolů, i v samostatném nastavení distribuovaných informací.

Tam je možné přidávat jednotlivé sítě, které budou propagovány jako dostupné bez ohledu na to, zda opravdu jsou.

8.2.1 Filtry

V tuto chvíli si routery vyměňují routovací informace. Dalším krokem je nastavení filtrů. Ty přiřazujeme každému peer samostatně pro směr dovnitř a ven, tedy pro odesílání a přijímání informací. Filtry budou důležité při použití stejné konfigurace pro více zákazníků, kteří nemají mít propojené sítě. V takovém případě musí být informace o dostupnosti sítí jednotlivých zákazníků odděleny a známé pouze routerům v síti příslušných zákazníků.

8.2.2 Řízení směru provozu

Protokol BGP byl vybrán především kvůli pokročilým možnostem řízení směru toku provozu. Tyto možnosti je nutné využít kvůli řešení s primární a záložní konektivitou, kdy v ideálním stavu obě konektivity fungují, a každá je využita pro jiný typ provozu. Rozdělení provozu do správných směrů bude zajištěno pomocí filtrů, pomocí kterých jsou jednotlivým routovacím informacím nastaveny různé BGP atributy (například Local Preference) pro určení priority. Když tedy páteřní router přijme informaci o směru do sítě LAN pobočky od routeru v lokalitě dvakrát (přes primární i sekundární konektivitu), vybere tu, která vede přes primární konektivitu (má nastavenou vyšší hodnotu Local Preference).

Provoz do sítě LAN pobočky zároveň musí být rozdělen podle typu provozu primární a záložní konektivitou. To je poměrně nestandardní úkol, který může být vyřešen dvojí propagací adresního rozsahu sítě LAN. Jednou bude použit standardně s prioritou nastavenou tak, aby provoz procházel záložní konektivitou. Podruhé bude nastavena priorita opačně, pro provoz primární konektivitou, a zároveň bude takové routovací informaci přidělena routovací značka (bude přidělena do zvláštní routovací tabulky). Všechen prioritní provoz do LAN sítě pobočky bude routován podle takto vytvořené routovací tabulky a půjde tak primární konektivitou.

8.2.3 BFD

Další funkcí, která bude v konfiguraci využita, je BFD. Ta umožní detekci neprůchozího tunelu GRE, jehož rozhraní jsou vždy v aktivním stavu. Při konfiguraci je nutné zadat

interval, v jakém budou kontrolní pakety odesílány, v jakém mají být přijímány a konstantu, kterou bude dohodnutý interval vynásoben. V případě že není přijat kontrolní paket v intervalu konstanta x dohodnutý interval, je spojení považováno za nefunkční.

8.2.4 Provoz do Internetu

Kromě provozu ve vytvořené privátní síti, kterou tvoří pobočkové sítě LAN a jednotlivé tunely, je potřeba zabývat se i provozem, který směřuje z pobočky přímo do veřejného Internetu, případně zpět. Protože je žádoucí, aby byl Internet dostupný i v případě výpadku jedné z konektivit, není možné takový provoz routovat staticky například záložní konektivitou. V případě jejího výpadku by do ní dále odesílal data, i když by byla nefunkční. Pokud bychom brali v potaz pouze výpadek fyzické sítě na úrovni routeru (například odpojení kabelu), bylo by řešení poměrně jednoduché a stačilo by do routovací tabulky přidat druhou statickou routu s vyšší administrativní vzdáleností. Při výpadku v síti poskytovatele je však fyzická síť pro router stále dostupná a je nutné zapojit složitější mechanismus.

Default Route (směr pro provoz do Internetu) tak bude propagována z páteřního routeru oběma konektivitami. Pomocí filtru bude poté změněn Next Hop daného záznamu, aby provoz do Internetu neprocházel tunely, ale přímo danou konektivitou. Dále bude pomocí filtru nastavena příslušná priorita záznamu podobně, jako je popsáno výše. V případě že jsou obě konektivity funkční, bude provoz do Internetu směrován záložní konektivitou, a v případě výpadku jedné z konektivit, bude směrován tou funkční. Toto řešení však neřeší situaci, ke které by došlo v případě výpadku obou páteřních routerů najednou. Proto bude nastavena statická route skrze primární konektivitu s vyšší administrativní vzdáleností. Ta bude v takovém případě použita, a provoz do Internetu tak zůstane zachován.

8.3 Konfigurace prioritizace provozu

Jak je zmíněno výše, základní rozdělení provozu probíhá na základě rozdělení toku mezi primární a záložní konektivitu. Tím je prakticky výhradně rezervována přenosová rychlost primární konektivity pro prioritní provoz. To ovšem neznamená, že prioritizaci není potřeba dále řešit. Při návrhu řešení musí být vzata v potaz i situace, kdy bude na jedné z konektivit výpadek a všechen provoz, prioritní i méně prioritní, bude muset procházet

tou druhou. Je tedy nutné v každé konektivitě jednoznačně rozlišit typ provozu a určit jeho prioritu, s jakou může být přenášen.

8.3.1 Logické rozdělení provozu

Nejprioritnějším v každém tunelu bude vždy provoz protokolu BGP, bez kterého by router neměl prakticky žádné routovací informace a nemohl by routovat provoz (kromě provozu do Internetu, popsáno výše jako výpadek obou páteřních routerů). Stejně prioritní musí být i provoz BFD, bez kterého by v navržené konfiguraci nebyl funkční protokol BGP. Tento provoz můžeme lehce rozpoznat jako provoz se zdrojovým nebo cílovým portem 179 pro BGP a 3784 pro BFD, případně podle zdrojové a cílové adresy routerů, které jsou dopředu známé a mezi nimiž jiný provoz není.

Na další úrovni priorit je prioritní provoz zákazníka, v našem případě VoIP. Ten je primárně poskytovanou službou a pro zachování vysoké kvality hovoru je nutné dodržet nulovou ztrátovost, stejně jako nízké a konstantní zpoždění. Provoz VoIP je možné rozpoznat podle nastavené hodnoty DSCP (26 pro SIP a 46 pro RTP) a podle IP adresy ústředny.

Poslední kategorií provozu v tunelu je ostatní provoz. Do této kategorie může patřit například komunikace s FTP serverem klienta, který je zabezpečený a tunely prochází. Takový provoz je jednoduše provoz, který je odesílán rozhraním tunelu a nespadá do žádné z výše uvedených kategorií.

Provoz na fyzickém rozhraní je rozdělen pouze na provoz tunelu a ostatní provoz do veřejného Internetu. Rozdělení může probíhat na základě zdrojové a cílové IP adresy provozu.

Navržené rozdělení provozu bylo později mírně upraveno. Konkrétní úprava a její důvody jsou vysvětleny v kapitole o reálném provozu.

8.3.2 Prioritizace provozu

Nyní je již provoz logicky rozdělen, jak je popsáno výše. Rozdělení je nyní nutné aplikovat přímo v nastavení routeru. V případě systému výrobce MikroTik je možné rozdělení provést označením jednotlivých paketů na úrovni firewallu s využitím funkce mangle, která umožňuje označovat pakety značkou paketu pro následné zpracování. Takové značky

nejsou dále přenášeny a jsou platné pouze lokálně v daném routeru. Kromě takového označování je možné měnit i hodnoty v hlavičce IP paketu, které jsou přenášeny dále, jako například DSCP.

Právě pomocí Firewall Mangle je prioritnímu provozu ve směru do pobočky přidělená routovací značka, jak je zmíněno výše. Dále zde bude změněna hodnota DSCP proritního provozu, respektive provozu tunelu, aby byla prioritizována v síti poskytovatelů připojení k Internetu, které takovou prioritizaci podporují.

Provoz, který byl v předchozím kroku označen značkou paketu, je možné přiřadit do jedné z front. Konfigurace front se skládá z nastavení priority fronty, rychlosti a typu fronty. Typů front je více a nové mohou být konfigurovány. V zásadě jde o určení pořadí, v jakém budou pakety odesílány, a způsob jejich zahazování pro předejití zahlcení sítě. Typ pořadí pro odesílání může být například FIFO (First In First Out – provoz je odesílán v pořadí, v jakém byl přijat), SFQ (Stochastic Fairness Queuing – provoz je rozdělen na spojení podle IP adresy a portu zdroje a cíle a každému je odesílán se stejnou prioritou) nebo jejich obměny.

8.4 Konfigurace oddělení provozu

Oddělení provozu bude realizováno použitím VLAN. Pro lokální LAN síť bude použito číslo VLAN 2000 pro VoIP služby. Pro uživatelské stanice nebude nastavena žádná VLAN, respektive číslo VLAN 1. Jejich provoz neponese informaci o číslě VLAN.

Tímto je provoz oddělen na fyzické vrstvě. Pro oddělení provozu i na síťové vrstvě je nutné nastavit pravidlo pro filtraci provozu. Pravidlo bude zakazovat provoz mezi IP adresami přidělovanými ve VLAN 2002 a IP adresami přidělovanými bez VLAN. Před konfigurací VLAN byly všem zařízením v pobořce přiděleny adresy z jednoho bloku /24. Nyní bude tento blok rozdělen na dva bloky /25 pro filtraci provozu. Bylo by možné přidělovat ve VLAN 2002 adresy z dalšího bloku /24, ten by však bylo nutné přidat ve všech pravidlech pro routing, jak bylo zmíněno dříve.

8.5 IPv6

Jedním z požadavků na vytvořenou síť je i podpora IPv6. Vzhledem k tomu, že velká část poskytovatelů připojení k Internetu v současnosti nepodporuje IPv6, musí být takový

provoz veden jinudy, než přímo jedním z takových připojení. Všechen provoz protokolu IPv6 bude směrován GRE tunely přes páteřní routery ven do Internetu.

Každé z lokalit bude pro využití v síti LAN přidělen rozsah adres s maskou 64, tedy více než 10^{20} adres. Tyto adresy budou routovány stejně jako privátní rozsahy IPv4 pomocí protokolu BGP. Protokol BGP umožňuje propagaci IPv6 adres pomocí stávajících IPv4 peerů.

I pro IPv6 bude nutné rozdělit směry provozu pro prioritní provoz, pro který je vyhrazena primární konektivita, a ostatní provoz, který je směrován záložní konektivitou. Stejně jako v případě IPv4 bude vyřešena také prioritizace provozu.

9 Testování konfigurace

Stejně jako jakýkoli jiný návrh řešení je i řešení podle zadání této práce nutné před implementací v reálném provozu patřičně otestovat. Testováním ověříme, že všechny systémy fungují podle dokumentace výrobce, a zároveň že nedošlo ke špatnému návrhu a do uvažování byly zahrnuty všechny eventuality. Ze zkušenosti s podobnými implementacemi vyplývá, že i když jsou návrh i testování provedeny opravdu důkladně, při nasazení řešení v reálném provozu vyplynou situace, které nebyly v původním návrhu řešeny.

Otázkou, kterou je potřeba před započetím testování vyřešit, je samotný způsob jakým bude konfigurace testována. První zvažovanou variantou bylo použití virtualizačního nástroje, ve kterém by bylo možné vytvořit virtuální routery s virtuálními propoji mezi nimi. Takový způsob umožňuje věrnou simulaci přímo na software výrobce při velmi nízkých nákladech. Při dostatečném výkonu je možné virtualizaci provozovat na domácím PC nebo na virtuálním serveru spuštěném jen po dobu simulace. Hlavní nevýhodu použití takového způsobu je praktická nemožnost použití vytvořeného projektu jako přílohy této práce. Do přílohy by bylo možné dát pouze testovanou konfiguraci routerů a opětovné testování by bylo obtížné. Dalším problémem bylo licencování operačního systému routerů MikroTik, RouteOS. Systém je dostupný online, ale jeho použití bez licence je limitováno na 24 hodin a zakoupení licence je poměrně nákladné.

Z těchto důvodů jsem se rozhodl testovat navržené řešení na reálných zařízeních. Tím sice nebude vyřešen problém přiložení projektu k této práci, ale bude možné dosáhnout přesnějších výsledků díky fyzickému uspořádání, které bude prakticky totožné z reálným provozem. Takové řešení nebude ani levnější než použití virtualizace a nakoupení licencí RouterOS, ale investice do použitých zařízení bude zhodnocena při jejich použití u zákazníka. Právě proto jsem se rozhodl využít pro testování právě tuto možnost.

Další možností, jejíž použití jsem v jednu chvíli zvažoval, bylo využití nějakého simulačního programu, například Packet Tracer. Právě Packet Tracer by byl vhodný pro svou jednoduchost, podporuje však pouze router výrobce Cisco. Bylo by tak možné otestovat základní využité principy, ale ne velké množství detailů, které jsou pro řešení důležité a jsou často výrazně závislé na výrobci.

9.1 Zapojení pro testování

Po rozhodnutí, že testování proběhne na fyzických zařízeních, bylo zjevně nejjednodušší a ekonomicky nejvýhodnější použít pro testování dostatečné výkonné routery, které mohou být později použité v reálném provozu.

Jak již bylo zmíněno dříve, jako páteřní routery a koncentrátory provozu byly vybrány routery výrobce MikroTik z řady Cloud Core Router, které poskytují dostatečný výkon na procesoru a zároveň podporují hardwarové šifrování provozu. Ani zvýšený provoz z většího množství zákaznických poboček tunely IPsec tak pro ně nebude výkonově problém. Dva routery MikroTik Cloud Core Router CCR1016-12G byly tedy zakoupeny a umístěny v datovém centru. Každému z nich byly přiděleny dvě veřejné IP adresy potřebné k navržené konfiguraci.

Jako pobočkový router byl pro testování využit výše zmíněný router MikroTik RB750 zapojený v kanceláři. Jako primární konektivita byla použita konektivita běžně používaná v kanceláři, která umožňuje přidělit testovacímu routeru nevyužitou veřejnou IP adresu. Sekundární konektivitou se stala firemní sít LAN s privátní IP adresou. Takové zapojení je prakticky totožné s předpokládaným zapojením v reálném provozu. Jediným nedostatkem je sdílená rychlost obou konektivit. Hlavním předmětem testování však není zahlcení obou připojení, a proto takové omezení není překážkou.

9.2 Průběh a výsledky testování

Hlavním předmětem testování bylo ověření funkčnosti navržené konfigurace. Nejprve byly nakonfigurovány IP adresy pro základní fungování použitých routerů v síti. Poté tunely a po ověření jejich funkčnosti také routovací protokoly, které pro svůj provoz tunely potřebují. Jsou jimi posílány všechny informace potřebné pro správnou funkci routování.

Již brzy po konfiguraci routovacích protokolů se v síti projevil první problém. Testovací pobočkový router v nepravidelných intervalech, vždy přibližně do jedné hodiny, přestával reagovat. Všechna jeho spojení se rozpadla a neodpovídal na dotazy ping ze vzdálené ani z lokální sítě. Po delším času se stejný problém projevil i na páteřních routerech CCR. Po delším času testování a pokusů o detekci problému se mi podařilo připojit se do grafického konfiguračního rozhraní routeru za pomoci nástroje Winbox pro správu routerů MikroTik, který umožňuje připojení k routeru bez využití protokolu IP.

Po připojení do rozhraní routeru se vše zdálo být v naprostém pořádku. Nefunkční byl jen všechen provoz protokolu IP. Po komunikaci s výrobcem MikroTik jsem zjistil, že je chyba způsobena chybou systému jako takového. Chyba se týkala detekce smyček při použití dvou a více tunelů a routovacího protokolu BGP zároveň a způsobovala zaplnění routing cache.

Routing cache je prostor v operační paměti vyhrazený pro informace o routování. Nejedná se o routovací tabulku. Obsahuje spíše informace o způsobu zpracování paketů, které vychází právě z routovací tabulky a již zpracovaných paketů. Routing cache využívá linuxové jádro pro zrychlení routování. Standardně k zaplnění routing cache nemá nikdy dojít. O to se stará garbage collector, nástroj pro správu routing cache, který ji podle předem definovaných pravidel promazává. Z důvodu další chyby však tento mechanismus nepracoval zcela správně a k zaplnění došlo. V takové chvíli nebyl postižený router schopen jakéhokoli provozu na síti IP.

Tento problém byl odstraněn v nové verzi operačního systému RotuerOS. Po jeho vydání a nasazení se tato chyba dále neprojevovala, a byla tak odstraněna jediná chyba odhalená při testování.

10 Reálný provoz

Při testování bylo vytvořeno připojení testovací pobočky a páteřní routery byly nakonfigurovány pro produkční použití. Při konfiguraci routeru pro pobočku zákazníka je tedy možné použít stávající konfiguraci testovacího routeru a upravit použité IP adresy.

Předpokladem pro implementaci řešení v reálném provozu je, že je navržená konfigurace plně funkční, což bylo potvrzeno v průběhu testování. Je však možné, že bude v průběhu implementace nutné upravit některé parametry, na které nebyl v průběhu testování brán zřetel. Reálný provoz je často velice specifický, zvláště při vyšším počtu uživatelů, a zahrnout do testování všechny možné situace je často velice náročné, vzhledem k omezenému času testování i nereálné.

10.1 Chyby odhalené v produkčním provozu

I přes důkladné testování celé konfigurace v prostředí hodně podobném produkčnímu prostředí se mi nepodařilo vyhnout se všem problémům. Zde popíši všechny změny, které bylo nutné po započetí produkčního provozu změnit.

10.1.1 Routování

Při spuštění provozu na více pobočkách byla zjištěna nesprávná propagace adres protokolem BGP. Páteřní routery sice měly informace o všech adresách, ale jednotlivé pobočkové routery neměly informace o sobě navzájem. Tato chyba nebyla odhalena v rámci testování, protože probíhalo pouze s jedním pobočkovým routerem bez zapojení dalších poboček. Chyba byla způsobena použitím varianty BGP protokolu iBGP, kdy mají oba spolu komunikující routery stejné číslo autonomního systému. V takovém případě páteřní router nepřeposílá informace z jedné pobočky na druhou.

Tento problém byl způsoben chybou v návrhu a jeho odstranění je poměrně snadné, změna konfigurace z iBGP na eBGP. Pobočkovým routerům byla přidělena jiná čísla AS než páteřním routerům.

10.1.2 Tunely

Další úpravou, která byla na původně navržené konfiguraci provedena, byla změna definice IPsec pravidel a peerů na páteřních routerech. S přibývajícím množstvím poboček přibývalo i připojení, která nedovolovala použití pevné veřejné IP adresy. Byl proto

změněn způsob přidávání IPsec peerů ze statického na dynamický. Po změně konfigurace je každý IPsec peer vždy pevně definován na straně pobočkového routeru a na straně páteřního routeru je IPsec peer včetně pravidla pro tunelování a šifrování přidán automaticky. Taková konfigurace umožňuje výrazně snazší konfiguraci nových poboček zjednodušením konfigurace páteřních routerů.

10.1.3 Prioritizace provozu

V produkčním provozu také v jednu chvíli došlo k výraznému snížení rychlosti služby web. Po kontrole aktuálního provozu na routeru jsem zjistil, že fronta pro neprioritní provoz v tunelu je přeplněná a nezanedbatelné množství paketů musí být zahozeno. Provoz, který tunelem procházel, byl provoz na port 80 (HTTP) ve směru na virtuální server, který neplní pouze roli VoIP, ale i funkci webového a databázového serveru.

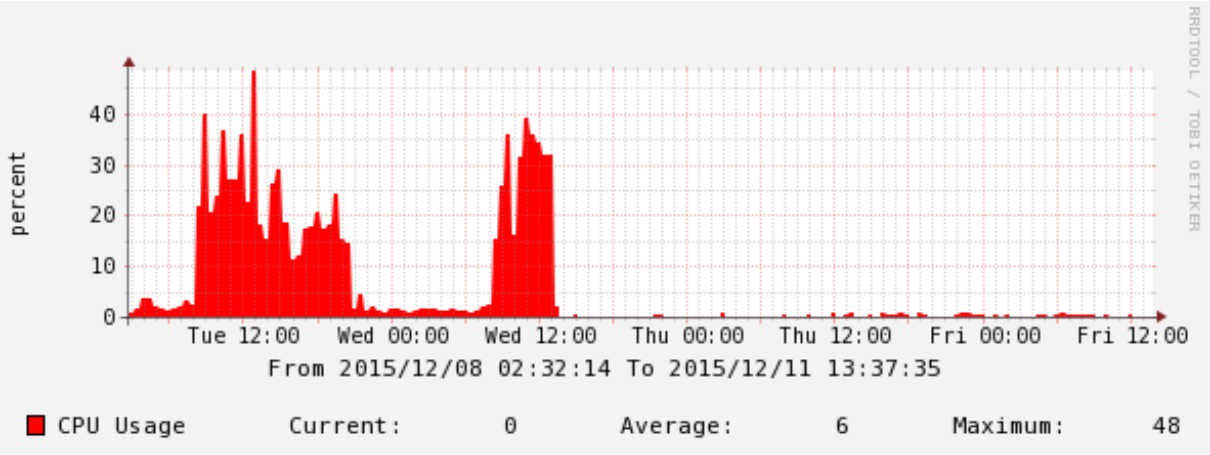
Řešení této situace spočívalo opět ve změně konfigurace. Tentokrát byla změněna pravidla pro prioritizaci provozu. Webovému provozu byla přiřazena vlastní fronta, a byl tak vyřazen z fronty pro všechen ostatní provoz. Tak mu mohla být přiřazena větší část přenosové rychlosti. Stejná úprava musela být samozřejmě provedena i v druhém směru, z ústředny na pobočku, v konfiguraci páteřního routeru.

10.1.4 Výběr zařízení

Až po delší době od spuštění provozu na jedné z poboček začala pobočka růst a monitorovací systém zaznamenal zvýšenou latenci některých ping dotazů. Kvalita hovorů však zůstala zachována. Problém se projevoval v nepravidelných intervalech a zcela náhodně. Analyzoval jsem konektivitu a nezjistil jsem žádný zvýšený provoz nebo sníženou celkovou kvalitu připojení k Internetu.

Problém jsem po dalším zkoumání odhalil ve výkonu routeru MikroTik RB750. Veškeré šifrování na něm z důvodu absence dedikovaného čipu obstarává procesor. Při vyšších rychlostech provozu v tunelech IPsec se výrazně zvyšuje využití procesoru. V pětiminutových průměrech tak hodnota využití procesoru dosahovala 50 %. Maximální rychlost v tunelu dosahovala 1 Mbit/s, což není extrémně vysoká hodnota, ale byla tvořena relativně malými pakety přenášejícími hlas. Zpracování velkého množství malých paketů vytížilo procesor routeru natolik, že v některých situacích nestihl odbavovat všechen provoz standardní rychlostí.

Z tohoto důvodu byl router MikroTik RB750 nahrazen routerem MikroTik Cloud Core Router RRC1009, který podporuje hardwarové šifrování provozu, a provoz IPsec tunelů tak nezatěžuje žádné z devíti jader jeho procesoru. Po výměně dále k žádným problémům nedocházelo. Změna vytížení procesoru je zřejmá z následujícího obrázku (obr. 12). K výměně došlo přibližně ve 13:00.



obr. 12: Změna využití CPU po výměně routeru za výkonnější model

Závěr

V průběhu zpracovávání této diplomové práce jsem se podrobně seznámil s problematikou protokolů a řešení vhodných pro návrh sítě, síť navrhl a řešení implementoval. Úspěšně jsem tak dokončil zadaný projekt a zajistil poskytování pokročilých telekomunikačních služeb několika významným zákazníkům. Po několikaletých zkušenostech v oboru sítí jsem měl příležitost pracovat samostatně na takto rozsáhlém projektu a vidět jeho konkrétní výsledky v produkčním prostředí.

Dostal jsem šanci podílet se na výběru zařízení, zamyslet se nad celým problémem z ekonomického pohledu, studovat do hloubky použité protokoly a konzultovat problematiku se seniornějšími kolegy. Největším zadostiučiněním je pro mne vidět, že výsledek mé práce je možné reálně použít a mohu s ním dále dennodenně pracovat.

Seznam použité literatury

[1] http://wiki.mikrotik.com/wiki/Manual:Interface/Gre

[2] http://wiki.mikrotik.com/wiki/Manual:Interface/L2TP

[3] http://wiki.mikrotik.com/wiki/Manual:Interface/PPTP

[4] http://wiki.mikrotik.com/wiki/Manual:IP/IPsec

[5] http://wiki.mikrotik.com/wiki/Manual:IPv6_Overview

[6] http://wiki.mikrotik.com/wiki/Manual:Queue

[7] http://wiki.mikrotik.com/wiki/Manual:Routing/BGP

[8] http://wiki.mikrotik.com/wiki/Manual:Routing/OSPF

[9] http://www.cisco.com/web/partners/downloads/765/tools/quickreference/routerperformance.pdf

[10] http://www.tcpipguide.com/free/t_OSPFBasicTopologyandtheLinkStateDatabase-3.htm

[11] https://academy.ripe.net/

[12] https://tools.ietf.org/html/rfc2637

[13] https://tools.ietf.org/html/rfc2661

[14] https://tools.ietf.org/html/rfc2784

[15] https://tools.ietf.org/html/rfc4301

[16] https://www.ietf.org/rfc/rfc2328.txt

[17] https://www.ietf.org/rfc/rfc2460.txt

[18] https://www.ietf.org/rfc/rfc4271.txt

[19] https://www.ietf.org/rfc/rfc4302.txt

[20] https://www.ietf.org/rfc/rfc4303.txt

[21] KÁLLAY, Fedor a Peter PENIAK. Počítačové sítě a jejich aplikace. Vyd. 1. Praha: Grada, 1999, 311 s. ISBN 80-7169-407-x.

[22] THOMAS, Robert M. Lokální počítačové sítě. Vyd. 1. Praha: Computer Press, 1996, 277 s. ISBN 80-85896-45-1.

[23] TRULOVE, James. Sítě LAN: hardware, instalace a zapojení. 1. vyd. Praha: Grada, 2009, 384 s. Profesionál. ISBN 978-80-247-2098-2.

Seznam příloh

I. Konfigurace pobočkového routeru

V konfiguraci jsou všechny veřejné IP adresy nahrazeny privátními, IPv6 adresy nahrazeny IPv6 adresami určenými pro dokumentaci a příklady a hesla změněna.

```
/interface bridge
add mtu=1500 name=lan
/interface ethernet
set [ find default-name=ether1 ] comment=uplink_pri
set [ find default-name=ether2 ] comment=uplink_sec
/interface gre
add clamp-tcp-mss=no !keepalive local-address=192.168.10.1 mtu=1476 name=gre-1ccr1 remote-
address=192.168.0.1
add clamp-tcp-mss=no !keepalive local-address=192.168.10.1 mtu=1476 name=gre-1ccr2 remote-
address=192.168.0.3
add clamp-tcp-mss=no !keepalive local-address=192.168.20.1 mtu=1476 name=gre-2ccr1 remote-
address=192.168.0.2
add clamp-tcp-mss=no !keepalive local-address=192.168.20.1 mtu=1476 name=gre-2ccr2 remote-
address=192.168.0.4
/interface vlan
add interface=lan l2mtu=1594 name=voip-vlan vlan-id=2000
/ip ipsec proposal
add enc-algorithms=des name=proposal1
/ip pool
add name=pc ranges=10.135.0.1-10.135.0.125
add name=voip-vlan ranges=10.135.0.129-10.135.0.253
/ip dhcp-server
add address-pool=pc disabled=no interface=lan lease-time=1d name=pc
add address-pool=voip-vlan disabled=no interface=voip-vlan lease-time=1d name=voip-vlan
/queue type
set 0 pfifo-limit=500
add kind=sfq name=sfq-pptp sfq-allot=1936 sfq-perturb=0
/queue tree
add limit-at=100M max-limit=100M name=uplink1-down parent=global queue=default
add limit-at=18M max-limit=18M name=uplink1-up parent=global queue=default
add limit-at=27M max-limit=27M name=uplink2-down parent=global queue=default
add limit-at=27M max-limit=27M name=uplink2-up parent=global queue=default
add name=uplink1-tunnel-down packet-mark=uplink1-tunnel-down parent=uplink1-down priority=1
queue=default
add name=uplink2-tunnel-down packet-mark=uplink2-tunnel-down parent=uplink2-down priority=1
queue=default
add limit-at=15M max-limit=15M name=uplink1-tunnel-up parent=uplink1-up priority=1
queue=default
add limit-at=25M max-limit=25M name=uplink2-tunnel-up parent=uplink2-up priority=1
queue=default
add name=uplink1-other-down packet-mark=uplink1-down parent=uplink1-down priority=4
queue=default
add name=uplink2-other-down packet-mark=uplink2-down parent=uplink2-down priority=4
queue=default
add name=uplink2-other-up packet-mark=uplink2-up parent=uplink2-up queue=default
add name=uplink1-other-up packet-mark=uplink1-up parent=uplink1-up queue=default
```

```
add name=uplink2-bgp_traffic packet-mark=tun-uplink2-bgp_traffic parent=uplink2-tunnel-up
priority=1 queue=default
add name=uplink2-other-in-tunnel packet-mark=tun-uplink2-other-in-tunnel-up parent=uplink2-
tunnel-up priority=7 queue=default
add name=uplink2-voice packet-mark=tun-uplink2-voice-in-tunnel-up parent=uplink2-tunnel-up
priority=2 queue=sfq-pptp
add name=uplink1-voice packet-mark=tun-uplink1-voice-in-tunnel-up parent=uplink1-tunnel-up
priority=2 queue=sfq-pptp
add name=uplink1-other-in-tunnel packet-mark=tun-uplink1-other-in-tunnel-up parent=uplink1-
tunnel-up priority=7 queue=default
add name=uplink1-bgp_traffic packet-mark=tun-uplink1-bgp_traffic parent=uplink1-tunnel-up
priority=1 queue=default
/routing bgp instance
add as=64601 client-to-client-reflection=no name=cust1 out-filter=bgp_out router-
id=10.135.0.254
/interface bridge port
add bridge=lan interface=ether3
add bridge=lan interface=ether4
add bridge=lan interface=ether5
/ip address
add address=10.135.0.126/25 interface=lan network=10.135.0.0
add address=10.135.0.254/25 interface=voip-vlan network=10.135.0.128
add address=192.168.10.1/30 interface=ether1 network=192.168.10.0
add address=192.168.20.1/30 interface=ether2 network=192.168.20.0
add address=10.134.6.1/30 interface=gre-1ccr1 network=10.134.6.0
add address=10.134.6.5/30 interface=gre-2ccr1 network=10.134.6.4
add address=10.134.7.1/30 interface=gre-1ccr2 network=10.134.7.0
add address=10.134.7.5/30 interface=gre-2ccr2 network=10.134.7.4
/ip dhcp-server network
add address=10.135.0.0/24 dhcp-option=provisioning dns-server=10.134.2.1
gateway=10.135.0.254 wins-server=10.134.2.1
add address=10.135.250.0/24 dns-server=10.135.250.254 gateway=10.135.250.254 netmask=24
/ip firewall filter
add action=reject chain=forward dst-address=10.135.0.0/25 src-address=10.135.0.128/25
add action=reject chain=forward dst-address=10.135.0.125/25 src-address=10.135.0.0/25
/ip firewall mangle
add action=change-dscp chain=postrouting dst-address=192.168.0.0/29 new-dscp=46
add action=mark-packet chain=prerouting in-interface=ether1 new-packet-mark=uplink1-tunnel-
down passthrough=no src-address=192.168.0.0/29
add action=mark-packet chain=forward in-interface=ether1 new-packet-mark=uplink1-wifi-guest-
down out-interface=wifi-guest passthrough=no
add action=mark-packet chain=prerouting in-interface=ether1 new-packet-mark=uplink1-down
passthrough=no
add action=mark-packet chain=prerouting in-interface=ether2 new-packet-mark=uplink2-tunnel-
down passthrough=no src-address=192.168.0.0/29
add action=mark-packet chain=forward in-interface=ether2 new-packet-mark=uplink2-wifi-guest-
down out-interface=wifi-guest passthrough=no
add action=mark-packet chain=prerouting in-interface=ether2 new-packet-mark=uplink2-down
passthrough=no
add action=mark-packet chain=postrouting dst-address=192.168.0.0/29 new-packet-mark=uplink1-
tunnel-up out-interface=ether1 passthrough=no
```

```
add action=mark-packet chain=forward in-interface=wifi-guest new-packet-mark=uplink1-wifi-guest-up out-interface=ether1 passthrough=no
add action=mark-packet chain=postrouting new-packet-mark=uplink1-up out-interface=ether1 packet-mark=no-mark passthrough=no
add action=mark-packet chain=postrouting dst-address=192.168.0.0/29 new-packet-mark=uplink2-tunnel-up out-interface=ether2 passthrough=no
add action=mark-packet chain=forward in-interface=wifi-guest new-packet-mark=uplink2-wifi-guest-up out-interface=ether2 passthrough=no
add action=mark-packet chain=postrouting new-packet-mark=uplink2-up out-interface=ether2 passthrough=no
add action=mark-packet chain=postrouting new-packet-mark=tun-uplink1-up out-interface=gre-1ccr1
add action=mark-packet chain=postrouting new-packet-mark=tun-uplink1-up out-interface=gre-1ccr2
add action=mark-packet chain=postrouting new-packet-mark=tun-uplink2-up out-interface=gre-2ccr1
add action=mark-packet chain=postrouting new-packet-mark=tun-uplink2-up out-interface=gre-2ccr2
add action=mark-packet chain=postrouting dscp=46 dst-address=10.134.8.6 new-packet-mark=tun-uplink1-voice-in-tunnel-up packet-mark=tun-uplink1-up passthrough=no
add action=mark-packet chain=postrouting dscp=26 dst-address=10.134.8.6 new-packet-mark=tun-uplink1-voice-in-tunnel-up packet-mark=tun-uplink1-up passthrough=no
add action=mark-packet chain=postrouting dscp=46 dst-address=10.134.8.6 new-packet-mark=tun-uplink2-voice-in-tunnel-up packet-mark=tun-uplink2-up passthrough=no
add action=mark-packet chain=postrouting dscp=26 dst-address=10.134.8.6 new-packet-mark=tun-uplink2-voice-in-tunnel-up packet-mark=tun-uplink2-up passthrough=no
add action=mark-packet chain=postrouting dst-address=10.134.6.0/24 new-packet-mark=tun-uplink1-bgp_traffic packet-mark=tun-uplink1-up passthrough=no
add action=mark-packet chain=postrouting dst-address=10.134.6.0/24 new-packet-mark=tun-uplink2-bgp_traffic packet-mark=tun-uplink2-up passthrough=no
add action=mark-packet chain=postrouting dst-address=10.134.7.0/24 new-packet-mark=tun-uplink1-bgp_traffic packet-mark=tun-uplink1-up passthrough=no
add action=mark-packet chain=postrouting dst-address=10.134.7.0/24 new-packet-mark=tun-uplink2-bgp_traffic packet-mark=tun-uplink2-up passthrough=no
add action=mark-packet chain=postrouting new-packet-mark=tun-uplink1-other-in-tunnel-up packet-mark=tun-uplink1-up passthrough=no
add action=mark-packet chain=postrouting new-packet-mark=tun-uplink2-other-in-tunnel-up packet-mark=tun-uplink2-up passthrough=no
/ip firewall nat
add action=masquerade chain=srcnat out-interface=ether1
add action=masquerade chain=srcnat out-interface=ether2
/ip ipsec peer
add address=192.168.0.1/32 secret=heslo
add address=192.168.0.2/32 secret=heslo
add address=192.168.0.3/32 secret=heslo
add address=192.168.0.4/32 secret=heslo
/ip ipsec policy
add dst-address=192.168.0.1/32 sa-dst-address=192.168.0.1 sa-src-address=192.168.10.1 src-address=192.168.10.1/32 tunnel=yes
add dst-address=192.168.0.2/32 sa-dst-address=192.168.0.2 sa-src-address=192.168.20.1 src-address=192.168.20.1/32 tunnel=yes
```

```
add dst-address=192.168.0.3/32 sa-dst-address=192.168.0.3 sa-src-address=192.168.10.1 src-address=192.168.10.1/32 tunnel=yes
add dst-address=192.168.0.4/32 sa-dst-address=192.168.0.4 sa-src-address=192.168.20.1 src-address=192.168.20.1/32 tunnel=yes
/ip route
add distance=1 gateway=10.134.4.1 routing-mark=global-avail
add distance=1 dst-address=192.168.0.1/32 gateway=192.168.10.2 routing-mark=pri
add distance=1 dst-address=192.168.0.2/32 gateway=192.168.10.2 routing-mark=pri
add distance=1 dst-address=192.168.0.3/32 gateway=192.168.20.2 routing-mark=sec
add distance=1 dst-address=192.168.0.4/32 gateway=192.168.20.2 routing-mark=sec
add distance=1 gateway=10.134.6.2 routing-mark=1ccr1
add distance=1 gateway=10.134.6.6 routing-mark=2ccr1
add distance=1 gateway=10.134.7.2 routing-mark=1ccr2
add distance=1 gateway=10.134.7.6 routing-mark=2ccr2
add comment=sec distance=15 gateway=192.168.20.2
add comment=pri distance=10 gateway=192.168.10.2
/ip route rule
add action=lookup-only-in-table dst-address=192.168.0.1/32 src-address=192.168.10.2/32 table=pri
add action=lookup-only-in-table dst-address=192.168.0.2/32 src-address=192.168.20.2/32 table=sec
add action=lookup-only-in-table dst-address=192.168.0.3/32 src-address=192.168.10.2/32 table=pri
add action=lookup-only-in-table dst-address=192.168.0.4/32 src-address=192.168.20.2/32 table=sec
add action=lookup-only-in-table src-address=10.134.6.1/32 table=1ccr1
add action=lookup-only-in-table src-address=10.134.6.5/32 table=2ccr1
add action=lookup-only-in-table src-address=10.134.7.1/32 table=1ccr2
add action=lookup-only-in-table src-address=10.134.7.5/32 table=2ccr2
/ipv6 dhcp-server
add address-pool=pc authoritative=yes disabled=no interface=lan name=pc
add address-pool=voip authoritative=yes disabled=no interface=voip-vlan name=voip
/ipv6 pool
add name=pc prefix=2001:db8::/64 prefix-length=128
add name=voip prefix=2001:db8:0:1::/64 prefix-length=128
/ipv6 address
add address=2001:db8:1:1::1 interface=gre-1ccr1
add address=2001:db8:1:2::1 interface=gre-2ccr1
add address=2001:db8:1:3::1 interface=gre-1ccr2
add address=2001:db8:1:4::1 interface=gre-2ccr2
add address=2001:db8::ffff:ffff:ffff:ffff interface=lan
add address=2001:db8:0:1:ffff:ffff:ffff:ffff interface=voip-vlan
/routing bgp network
add network=10.135.0.0/25 synchronize=no
add network=10.135.0.128/25 synchronize=no
add network=10.135.0.0/24 synchronize=no
add network=2001:db8::/64  synchronize=no
add network=2001:db8:0:1::/64  synchronize=no
/routing bgp peer
add address-families=ip,ipv6 in-filter=bgp_in_1ccr1 instance=cust1 name=1ccr1 remote-address=10.134.6.2 remote-as=64600 ttl=default update-source=gre-1ccr1 use-bfd=yes
```

```
add address-families=ip,ipv6 in-filter=bgp_in_2ccr1 instance=cust1 name=2ccr1 remote-
address=10.134.6.6 remote-as=64600 ttl=default update-source=gre-2ccr1 use-bfd=yes
add address-families=ip,ipv6 in-filter=bgp_in_1ccr2 instance=cust1 name=1ccr2 remote-
address=10.134.7.2 remote-as=64600 ttl=default update-source=gre-1ccr2 use-bfd=yes
add address-families=ip,ipv6 in-filter=bgp_in_2ccr2 instance=cust1 name=2ccr2 remote-
address=10.134.7.6 remote-as=64600 ttl=default update-source=gre-2ccr2 use-bfd=yes
/routing filter
add action=accept chain=bgp_in_1ccr1 prefix=0.0.0.0/0 set-distance=1 set-in-
nexthop=192.168.10.2
add action=accept chain=bgp_in_2ccr1 prefix=0.0.0.0/0 set-distance=1 set-in-
nexthop=192.168.20.2
add action=accept chain=bgp_in_1ccr2 prefix=0.0.0.0/0 set-distance=1 set-in-
nexthop=192.168.10.2
add action=accept chain=bgp_in_2ccr2 prefix=0.0.0.0/0 set-distance=1 set-in-
nexthop=192.168.20.2
add action=accept chain=bgp_in_1ccr1 set-in-nexthop=10.134.6.2 set-scope=20
add action=accept chain=bgp_in_2ccr1 set-in-nexthop=10.134.6.6 set-scope=20
add action=accept chain=bgp_in_1ccr2 set-in-nexthop=10.134.7.2 set-scope=20
add action=accept chain=bgp_in_2ccr2 set-in-nexthop=10.134.7.6 set-scope=20
add action=accept chain=bgp_out prefix=10.135.0.0/24 prefix-length=24-25
add action=discard chain=bgp_out
/system clock
set time-zone-autodetect=no time-zone-name=Europe/Prague
/system identity
set name=branch
```

II. Konfigurace prvního páteřního routeru

V konfiguraci jsou všechny veřejné IP adresy nahrazeny privátními, IPv6 adresy nahrazeny IPv6 adresami určenými pro dokumentaci a příklady a hesla změněna.

```
/interface gre
add !keepalive local-address=192.168.0.1 name=gre-cust1-site1 remote-address=192.168.10.1
add !keepalive local-address=192.168.0.2 name=gre-cust1-site2 remote-address=192.168.20.1
/interface vlan
add interface=h90 name=vlan11 vlan-id=11
/interface bonding
add mode=802.3ad name=h90 slaves=ether1,ether2 transmit-hash-policy=layer-2-and-3
/ip ipsec proposal
set [ find default=yes ] enc-algorithms=3des,aes-128-cbc
/queue type
set 0 pfifo-limit=500
add kind=sfq name=sfq-pptp sfq-allot=1936 sfq-perturb=0
/queue tree
add limit-at=15M max-limit=15M name=gre-cust1-site1-tun parent=global queue=default
add limit-at=25M max-limit=25M name=gre-cust1-site2-tun parent=global queue=default
add name=gre-lux-petriny1-voice packet-mark=gre-cust1-site1-voice-in-tunnel-up parent=gre-
cust1-site1-tun priority=2 queue=default
add name=gre-lux-petriny1-bgp packet-mark=gre-cust1-site1-bgp-in-tunnel-up parent=gre-
cust1-site1-tun priority=1 queue=default
add name=gre-lux-petriny1-other packet-mark=gre-cust1-site1-other-in-tunnel-up parent=gre-
cust1-site1-tun queue=default
add name=gre-lux-petriny2-voice packet-mark=gre-cust1-site2-voice-in-tunnel-up parent=gre-
cust1-site2-tun priority=2 queue=default
add name=gre-lux-petriny2-bgp packet-mark=gre-cust1-site2-bgp-in-tunnel-up parent=gre-
cust1-site2-tun priority=1 queue=default
add name=gre-lux-petriny2-other packet-mark=gre-cust1-site2-other-in-tunnel-up parent=gre-
cust1-site2-tun queue=default
/routing bgp instance
add as=64600 client-to-client-reflection=no name=cust1 redistribute-other-bgp=yes router-
id=10.134.8.1
/ip address
add address=192.168.0.1/29 interface=vlan11 network=192.168.0.0
add address=192.168.0.2/29 interface=vlan11 network=192.168.0.0
add address=10.134.6.2/30 interface=gre-cust1-site1 network=10.134.6.0
add address=10.134.6.6/30 interface=gre-cust1-site2 network=10.134.6.4
/ip dns
set servers=8.8.8.8,8.8.2.2
/ip firewall mangle
add action=change-dscp chain=postrouting new-dscp=46 src-address=192.168.0.0/29
add action=mark-packet chain=postrouting dscp=46 new-packet-mark=gre-cust1-site1-voice-in-
tunnel-up out-interface=gre-cust1-site1 passthrough=no
add action=mark-packet chain=postrouting dscp=24 new-packet-mark=gre-cust1-site1-voice-in-
tunnel-up out-interface=gre-cust1-site1 passthrough=no
add action=mark-packet chain=postrouting dscp=46 new-packet-mark=gre-cust1-site2-voice-in-
tunnel-up out-interface=gre-cust1-site2 passthrough=no
```

```
add action=mark-packet chain=postrouting dscp=24 new-packet-mark=gre-cust1-site2-voice-in-tunnel-up out-interface=gre-cust1-site2 passthrough=no
add action=mark-packet chain=postrouting dst-address=10.134.6.0/24 new-packet-mark=gre-cust1-site1-bgp-in-tunnel-up out-interface=gre-cust1-site1 passthrough=no
add action=mark-packet chain=postrouting dst-address=10.134.6.0/24 new-packet-mark=gre-cust1-site2-bgp-in-tunnel-up out-interface=gre-cust1-site2 passthrough=no
add action=mark-packet chain=postrouting new-packet-mark=gre-cust1-site1-other-in-tunnel-up out-interface=gre-cust1-site1
add action=mark-packet chain=postrouting new-packet-mark=gre-cust1-site2-other-in-tunnel-up out-interface=gre-cust1-site2
/ip ipsec peer
add address=0.0.0.0/0 generate-policy=port-strict local-address=192.168.0.1 secret=heslo send-initial-contact=no
add address=0.0.0.0/0 generate-policy=port-strict local-address=192.168.0.2 secret=heslo send-initial-contact=no
/ip route
add distance=1 gateway=192.168.0.6
/ip route rule
add action=lookup-only-in-table dst-address=10.135.0.0/24 src-address=10.134.8.6/32 table=for_cust1
/ipv6 address
add address=2001:db8:10::ffff:ffff:fffe:fffe advertise=no interface=vlan11
add address=2001:db8:1:1:ffff:ffff:ffff:ffff interface=gre-cust1-site1
add address=2001:db8:1:2:ffff:ffff:ffff:ffff interface=gre-cust1-site2
/ipv6 route
add distance=254 gateway=2001:db8:10::ffff:ffff:ffff:ffff
/queue interface
set ether1 queue=ethernet-default
set ether2 queue=ethernet-default
/routing bgp instance
add as=64600 client-to-client-reflection=no name=cust1 redistribute-other-bgp=yes router-id=10.134.8.1
/routing bgp network
add network=10.134.8.0/29 synchronize=no
add network=10.134.6.0/30 synchronize=no
add network=10.134.6.4/30 synchronize=no
add network=2001:db8:1:1:: synchronize=no
add network=2001:db8:1:2:: synchronize=no
/routing bgp peer
add default-originate=always in-filter=bgp_in_site1 instance=cust1 name=petriny1 out-filter=bgp_cust1_out_site1 remote-address=10.134.6.1 remote-as=64601 route-reflect=yes ttl=default update-source=gre-cust1-site1 use-bfd=yes
add default-originate=always in-filter=bgp_in_site2 instance=cust1 name=petriny2 out-filter=bgp_cust1_out_site2 remote-address=10.134.6.5 remote-as=64601 route-reflect=yes ttl=default update-source=gre-cust1-site2 use-bfd=yes
/routing filter
add action=accept chain=bgp_cust1_in_site1 prefix=10.135.0.0/21 prefix-length=25 set-bgp-local-pref=120 set-routing-mark=for_cust1
add action=accept chain=bgp_cust1_in_site1 set-bgp-local-pref=100
add action=accept chain=bgp_cust1_in_site2 prefix=10.135.0.0/21 prefix-length=25 set-bgp-local-pref=100 set-routing-mark=for_cust1
add action=accept chain=bgp_cust1_in_site2 set-bgp-local-pref=120
```

```
add action=accept chain=bgp_cust1_out_site1 prefix=10.134.8.0/29 set-bgp-local-pref=120
add action=jump chain=bgp_cust1_out_site1 jump-target=cust1-passthrough set-bgp-local-
pref=100
add action=accept chain=bgp_cust1_out_site2 prefix=10.134.8.0/29 set-bgp-local-pref=100
add action=jump chain=bgp_cust1_out_site2 jump-target=cust1-passthrough set-bgp-local-
pref=120
add action=accept chain=cust1-passthrough prefix=10.134.0.0/21 prefix-length=21-30
add action=accept chain=cust1-passthrough prefix=10.134.8.0/21 prefix-length=21-30
add action=accept chain=cust1-passthrough prefix=10.135.0.0/21 prefix-length=21-30
add action=jump chain=cust1-passthrough jump-target=everyone-passthrough
add action=accept chain=everyone-passthrough prefix=0.0.0.0/0
add action=accept chain=everyone-passthrough prefix=10.134.8.8/29
add action=discard chain=everyone-passthrough
/system clock
set time-zone-autodetect=no time-zone-name=Europe/Prague
/system identity
set name=ccr1
```

III. Konfigurace druhého páteřního routeru

V konfiguraci jsou všechny veřejné IP adresy nahrazeny privátními, IPv6 adresy nahrazeny IPv6 adresami určenými pro dokumentaci a příklady a hesla změněna.

```
/interface gre
add !keepalive local-address=192.168.0.3 name=gre-cust1-site1 remote-address=192.168.10.1
add !keepalive local-address=192.168.0.4 name=gre-cust1-site2 remote-address=192.168.20.1
/interface vlan
add interface=h90 name=vlan11 vlan-id=11
/interface bonding
add mode=802.3ad name=h90 slaves=ether1,ether2 transmit-hash-policy=layer-2-and-3
/ip ipsec proposal
set [ find default=yes ] enc-algorithms=3des,aes-128-cbc
/queue type
set 0 pfifo-limit=500
add kind=sfq name=sfq-pptp sfq-allot=1936 sfq-perturb=0
/queue tree
add limit-at=15M max-limit=15M name=gre-cust1-site1-tun parent=global queue=default
add limit-at=25M max-limit=25M name=gre-cust1-site2-tun parent=global queue=default
add name=gre-lux-petriny1-voice packet-mark=gre-cust1-site1-voice-in-tunnel-up parent=gre-
cust1-site1-tun priority=2 queue=default
add name=gre-lux-petriny1-bgp packet-mark=gre-cust1-site1-bgp-in-tunnel-up parent=gre-
cust1-site1-tun priority=1 queue=default
add name=gre-lux-petriny1-other packet-mark=gre-cust1-site1-other-in-tunnel-up parent=gre-
cust1-site1-tun queue=default
add name=gre-lux-petriny2-voice packet-mark=gre-cust1-site2-voice-in-tunnel-up parent=gre-
cust1-site2-tun priority=2 queue=default
add name=gre-lux-petriny2-bgp packet-mark=gre-cust1-site2-bgp-in-tunnel-up parent=gre-
cust1-site2-tun priority=1 queue=default
add name=gre-lux-petriny2-other packet-mark=gre-cust1-site2-other-in-tunnel-up parent=gre-
cust1-site2-tun queue=default
/routing bgp instance
add as=64600 client-to-client-reflection=no name=cust1 redistribute-other-bgp=yes router-
id=10.134.8.2
/ip address
add address=192.168.0.3/29 interface=vlan11 network=192.168.0.0
add address=192.168.0.4/29 interface=vlan11 network=192.168.0.0
add address=10.134.6.2/30 interface=gre-cust1-site1 network=10.134.6.0
add address=10.134.6.6/30 interface=gre-cust1-site2 network=10.134.6.4
/ip dns
set servers=8.8.8.8,8.8.2.2
/ip firewall mangle
add action=change-dscp chain=postrouting new-dscp=46 src-address=192.168.0.0/29
add action=mark-packet chain=postrouting dscp=46 new-packet-mark=gre-cust1-site1-voice-in-
tunnel-up out-interface=gre-cust1-site1 passthrough=no
add action=mark-packet chain=postrouting dscp=24 new-packet-mark=gre-cust1-site1-voice-in-
tunnel-up out-interface=gre-cust1-site1 passthrough=no
add action=mark-packet chain=postrouting dscp=46 new-packet-mark=gre-cust1-site2-voice-in-
tunnel-up out-interface=gre-cust1-site2 passthrough=no
```

```
add action=mark-packet chain=postrouting dscp=24 new-packet-mark=gre-cust1-site2-voice-in-tunnel-up out-interface=gre-cust1-site2 passthrough=no
add action=mark-packet chain=postrouting dst-address=10.134.7.0/24 new-packet-mark=gre-cust1-site1-bgp-in-tunnel-up out-interface=gre-cust1-site1 passthrough=no
add action=mark-packet chain=postrouting dst-address=10.134.7.0/24 new-packet-mark=gre-cust1-site2-bgp-in-tunnel-up out-interface=gre-cust1-site2 passthrough=no
add action=mark-packet chain=postrouting new-packet-mark=gre-cust1-site1-other-in-tunnel-up out-interface=gre-cust1-site1
add action=mark-packet chain=postrouting new-packet-mark=gre-cust1-site2-other-in-tunnel-up out-interface=gre-cust1-site2
/ip ipsec peer
add address=0.0.0.0/0 generate-policy=port-strict local-address=192.168.0.3 secret=heslo send-initial-contact=no
add address=0.0.0.0/0 generate-policy=port-strict local-address=192.168.0.4 secret=heslo send-initial-contact=no
/ip route
add distance=1 gateway=192.168.0.6
/ip route rule
add action=lookup-only-in-table dst-address=10.135.0.0/24 src-address=10.134.8.6/32 table=for_cust1
/ipv6 address
add address=2001:db8:10::ffff:ffff:ffff:fffe advertise=no interface=vlan11
add address=2001:db8:1:3:ffff:ffff:ffff:ffff interface=gre-cust1-site1
add address=2001:db8:1:4:ffff:ffff:ffff:ffff interface=gre-cust1-site2
/ipv6 route
add distance=254 gateway=2001:db8:10::ffff:ffff:ffff:ffff
/queue interface
set ether1 queue=ethernet-default
set ether2 queue=ethernet-default
/routing bgp instance
add as=64600 client-to-client-reflection=no name=cust1 redistribute-other-bgp=yes router-id=10.134.8.2
/routing bgp network
add network=10.134.8.0/29 synchronize=no
add network=10.134.7.0/30 synchronize=no
add network=10.134.7.4/30 synchronize=no
add network=2001:db8:1:3:: synchronize=no
add network=2001:db8:1:4:: synchronize=no
/routing bgp peer
add default-originate=always in-filter=bgp_in_site1 instance=cust1 name=petriny1 out-filter=bgp_cust1_out_site1 remote-address=10.134.7.1 remote-as=64601 route-reflect=yes ttl=default update-source=gre-cust1-site1 use-bfd=yes
add default-originate=always in-filter=bgp_in_site2 instance=cust1 name=petriny2 out-filter=bgp_cust1_out_site2 remote-address=10.134.7.5 remote-as=64601 route-reflect=yes ttl=default update-source=gre-cust1-site2 use-bfd=yes
/routing filter
add action=accept chain=bgp_cust1_in_site1 prefix=10.135.0.0/21 prefix-length=25 set-bgp-local-pref=130 set-routing-mark=for_cust1
add action=accept chain=bgp_cust1_in_site1 set-bgp-local-pref=110
add action=accept chain=bgp_cust1_in_site2 prefix=10.135.0.0/21 prefix-length=25 set-bgp-local-pref=110 set-routing-mark=for_cust1
add action=accept chain=bgp_cust1_in_site2 set-bgp-local-pref=130
```

```
add action=accept chain=bgp_cust1_out_site1 prefix=10.134.8.0/29 set-bgp-local-pref=130
add action=jump chain=bgp_cust1_out_site1 jump-target=cust1-passthrough set-bgp-local-
pref=110
add action=accept chain=bgp_cust1_out_site2 prefix=10.134.8.0/29 set-bgp-local-pref=110
add action=jump chain=bgp_cust1_out_site2 jump-target=cust1-passthrough set-bgp-local-
pref=130
add action=accept chain=cust1-passthrough prefix=10.134.0.0/21 prefix-length=21-30
add action=accept chain=cust1-passthrough prefix=10.134.8.0/21 prefix-length=21-30
add action=accept chain=cust1-passthrough prefix=10.135.0.0/21 prefix-length=21-30
add action=jump chain=cust1-passthrough jump-target=everyone-passthrough
add action=accept chain=everyone-passthrough prefix=0.0.0.0/0
add action=accept chain=everyone-passthrough prefix=10.134.8.8/29
add action=discard chain=everyone-passthrough
/system clock
set time-zone-autodetect=no time-zone-name=Europe/Prague
/system identity
set name=ccr2
```