Konfigurace a nastavení platformy RouterBoard

RouterBoard platform configuration

Bc. Libor Blaha

Diplomová práce
2011

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2010/2011

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Bc. Libor BLAHA**
Osobní číslo:	**A09500**
Studijní program:	**N 3902 Inženýrská informatika**
Studijní obor:	**Informační technologie**
Téma práce:	**Konfigurace a nastavení platformy RouterBoard**

Zásady pro vypracování:

1. **Vypracujte literární rešerši na dané téma.**
2. **Popište řešení VPN na bázi OpenVPN a IPsec.**
3. **Popište řešení směrování v síti IP protokolem BGP, OSPF, RIP.**
4. **Popište FUP, důvody použití a způsoby použití.**
5. **Zpracujte popis nastavení FUP na platformě RouterBoard.**
6. **Zpracujte popis nastavení platformy RouterBoard jako Hotspot.**
7. **Zpracujte popis nastavení VPN v součinnosti s platformou RouterBoard, Cisco, Linux.**
8. **Zpracujte popis nastavení směrování v IP sítích na platformě RouterBoard.**

Rozsah diplomové práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **MikroTik Routers and Wireless [online]. 2011 [cit. 2011-01-25]. Dostupný z WWW: [http://www.mikrotik.com/].**
2. **Routerboard.com [online]. 2011 [cit. 2011-01-25]. Dostupný z WWW: [http://www.routerboard.com/].**
3. **PUŽMANOVÁ, Rita. Bezpečnost bezdrátové komunikace : Jak zabezpečit Wi-Fi, bluetooth, GPRS či 3G. [s.l.] : [s.n.], 2005. 184 s. ISBN 80-251-0791-4.**
4. **PUŽMANOVÁ, Rita. TCP/IP v kostce. [s.l.] : [s.n.], 2004. 608 s. ISBN 80-7232-236-2.**
5. **SPORTACK, Mark. Směrování v sítích IP. [s.l.] : [s.n.], 2004. ISBN 80-251-0127-4. s. 351.**

Vedoucí diplomové práce: **doc. Ing. Martin Sysel, Ph.D.**
Ústav počítačových a komunikačních systémů

Datum zadání diplomové práce: **24. února 2011**

Termín odevzdání diplomové práce: **18. května 2011**

Ve Zlíně dne 24. února 2011

prof. Ing. Vladimír Vašek, CSc.
děkan

L.S.

UNIVERZITA TOMÁŠE BATI VE ZLÍNĚ
Fakulta aplikované informatiky

doc. Mgr. Roman Jašek, Ph.D.
ředitel ústavu

ABSTRAKT

Práce volně navazuje na bakalářskou práci autora z roku 2009 a představuje další možnosti využití platformy RouterBoard a Mikrotik RouterOS. Zaměřuje se zejména na popis a využití Mikrotik RouterOS jako bran do VPN, dále pak začlenění do systémů dynamického routování (BGP, OSPF) a v neposlední řadě jako systém zajišťující služby QoS. Případové studie v praktické části popisují využití RouterBoardu jako bránu pro veřejný přístupový bod (Hotspot), popisují dva způsoby realizace virtuální privátní sítě a konfiguraci RouterBoardu v sítích s využitím dynamických směrovacích protokolů. Poslední případová studie řeší využití RouterBoardu pro řízení sítě.

Klíčová slova: RouterBoard, BGP, OSPF, VPN, IPsec, OpenVPN, Hotspot, QoS.

ABSTRACT

The thesis is a free continuation of the bachelor thesis by the author of 2009 and introduces an additional possible usage of a platform RouterBoard and Mikrotik RouterOS. It focuses mainly on the description and use of Mikrotik RouterOS VPN gateway, as well as integration into the dynamic routing (BGP, OSPF) and finally as a system providing QoS. The case study also describes the use RouterBoard as a gateway for public access point (hotspot), describes two ways of implementing a virtual private network and configuration of RouterBoard in networks by using dynamic routing protocols. The last case study deals with the use RouterBoard network management.

Keywords: RouterBoard, BGP, OSPF, VPN, IPsec, OpenVPN, Hotspot, QoS.

Poděkování, motto

Rád bych na tomto místě poděkoval všem, bez kterých by tato práce nikdy nevznikla. Jsou to ti, kteří mi pomohli dobrou radou, ale také ti, kteří mě podporovali morálně a byli pro mě zázemím nezbytným k napsání této práce. Děkuji doc. Ing. Martinu Syslovi, Ph.D., za trpělivost, obětavost a cenné připomínky při vedení diplomové práce.

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl jsem seznámen s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování diplomové práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně 2. května 2011 …….

podpis diplomanta

OBSAH

ÚVOD

Cílem práce je představení platformy RouterBoard a operačního systému Mikrotik RouterOS jako technologie, která umožňuje realizovat poměrně složité síťové aplikace. Vzhledem k relativně nízké finanční náročnosti této technologie se může jevit využití platformy RouterBoard jako výhodné.

Práce je dělena na dvě části, kde první, teoretická část, popisuje obecně aplikace, ke kterým je vhodné a možné technologii RouterBoard použít a krátce popisuje práci s balíčky OS Mikrotik, jejich použití pro konkrétní danou aplikaci a seznamuje s hardwarovou konfigurací jednotlivých typů jednotek RouterBoard. Jedná se především o popis realizací virtuálních privátních sítí, kdy prostřednictvím veřejné telekomunikační sítě lze realizovat zabezpečené a šifrované propojení lokálních sítí. Práce se ve své teoretické části věnuje i směrování, a to jak statickému, tak i dynamickému a popisuje protokoly dynamického směrování, které se k tomuto využívají. Poslední dvě kapitoly teoretické části popisují parametry a metody aplikace QoS a využití a použití technických opatření řazených do skupiny Fair User Policy.

Druhá, praktická část práce, popisuje konkrétní případové studie, které vycházejí z teoretické části práce. První případová studie řeší využití RouterBoardu jako veřejného přístupového bodu do sítě Internet či jiné datové sítě. Druhá případová studie řeší konkrétní použití platformy RouterBoard pro realizaci VPN, a to jak VPN typu LAN-to-LAN, tak VPN typu remote access. Třetí případová studie popisuje konfiguraci dynamického směrování za použití protokolu OSPF a navázání OSPF komunikace mezi platformou RouterBoard a Cisco a komunikaci mezi platformou RouterBoard a softwarem quagga, který realizuje OSPF na Linuxu. Poslední případová studie řeší použití platformy RouterBoard jako realizátora služeb QoS.

I. TEORETICKÁ ČÁST

1 LITERÁRNÍ REŠERŠE

Virtuální neveřejné (privátní) sítě (Virtual Private Network – VPN) jsou novou kategorií sítí, které nejsou specifické svou technologií, ale způsobem efektivního využití veřejných sítí a komunikačních služeb. Virtuální privátní sítě jsou zásadní pro realizaci bezpečného vzdáleného přístupu, kdy uživatele, připojující se do LAN sítě prostřednictvím sítě Internet či obecné datové přenosové sítě, je nutno jednotně autentizovat a autorizovat jejich přístup k síti a jejím prostředkům. Virtuální privátní sítě je možné definovat jako neveřejné páteřní sítě, které využívají veřejnou (sdílenou) komunikační infrastrukturu, tedy např. síť Internet. Virtuální privátní síť je logická síť vytvořená v rámci veřejné infrastruktury, která si však zachovává charakter privátní sítě, kdy komunikace v rámci virtuální privátní sítě je zabezpečena (šifrovaná) a kvalita komunikace je zachována[1].

Směrování ve vzájemně propojených sítích je jedním ze základních principů propojování sítí. Je realizováno na třetí, síťové vrstvě architektury. Cílem směrování je určení cesty v síti a dopravení datového paketu k cílové stanici pokud možno co nejefektivnější cestou, přičemž nejefektivnější nemusí vždy znamenat nejkratší. Mezi odesílatelem a adresátem paketu je často velmi složitá síťová infrastruktura a směrování se většinou nezabývá celou cestou paketu v síti, ale řeší vždy jen jeden krok komu data předat jako dalšímu na cestě mezi odesílatelem a adresátem. Základním řešením směrování na úrovni hardware jsou směrovače a na úrovni software se jedná o datovou strukturu označovanou jako směrovací tabulka (routing table). Bez dynamického směrování není možné dnes provozovat žádnou rozlehlou datovou síť (např. Internet). Rozlišují se směrovací protokoly v rámci sítě provozované jedním provozovatelem (autonomní systém) a směrovací protokoly, které realizují směrování mezi sítěmi.

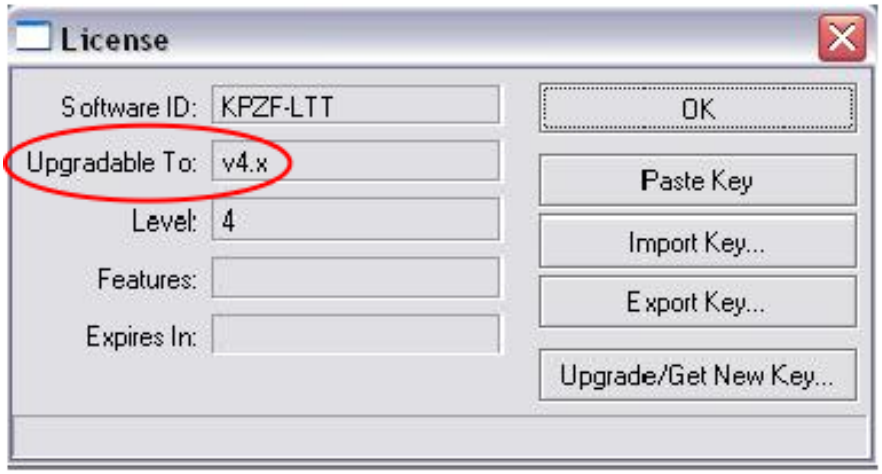
Quality of Service (QoS - kvalita služby) je soubor pravidel, která v kombinaci s Fair User Policy (FUP) představují technologii jak v počítačových sítích zajistit řízení datových toků a rezervaci přenosové kapacity spravedlivě pro všechny uživatele sítě tak, aby nedocházelo při zatížení sítě ke snížení kvality a dostupnosti síťových služeb. Tato pravidla jsou používána zpravidla v sítích, jejichž přenosová kapacita je omezena na poměrně nízké hodnoty a nejsou garantovány odezvy na požadavky v přenosové síti. Při přetěžování sítě konkrétním uživatelem dochází k aplikaci pravidel, která zajistí rozdělení technických prostředků sítě spravedlivě pro všechny uživatele stejně.

2 KONFIGURACE A NASTAVENÍ PLATFORMY ROUTERBOARD

Mikrotik RouterOS je distribuován a instalován ve formě jednotlivých balíčků (packages), kdy každý balíček představuje specifickou funkcionalitu systému Mikrotik RouterOS. Popis jednotlivých balíčků nutných k řešení zadaných úkolu je popsán dále [2] [3].

2.1 Softwarové balíčky

Softwarové balíčky je možné instalovat buď po jednotlivých balíčcích, nebo lze provést kompletní upgrade celého systému pomocí kombinovaného balíčku, kdy dojde k instalaci aktuální verze všech balíčků obsažených v instalaci. Poslední aktuálně dostupný firmware je vždy na webových stránkách společnosti Mikrotik na adrese *www.mikrotik.com* v sekci *download*. Pro každou skupinu RouterBoardů je nutné vybrat odpovídající typ kombinovaného balíčku. Upgrade je možné provádět dvěma způsoby. Pomocí aplikace WINBOX použitím funkce drag and drop v OS Windows a uložením požadovaného balíčku do složky *Files* a restartem zařízení. Po restartu dojde k automatické aktualizaci balíčků. Druhá možnost je použití příkazu FTP pro uložení požadovaného balíčku do Mikrotik RouterOS. Zde je nutné mít FTP povoleno v sekci IP/Services nebo povolit ftp příkazem CLI *ip services enable ftp* např. v terminálovém okně. Firmware není možné aktualizovat neomezeně. Maximální verzi, kterou je možné v daném routeru použít, specifikuje příkaz *system licence print*.



Obrázek 1 – Licence, maximální verze firmware

Po instalaci patřičného balíčku je nutné jej aktivovat v menu System/Packages nebo příkazem *system package hotspot enable* (pro balík hotspot). Nepotřebné balíčky je možné

pro přehlednost systémového menu deaktivovat. K aktivaci či deaktivaci definovaného balíčku dojde po následném startu systému. Přehled některých balíčků a popis jejich funkce uvádí následující tabulka [4].

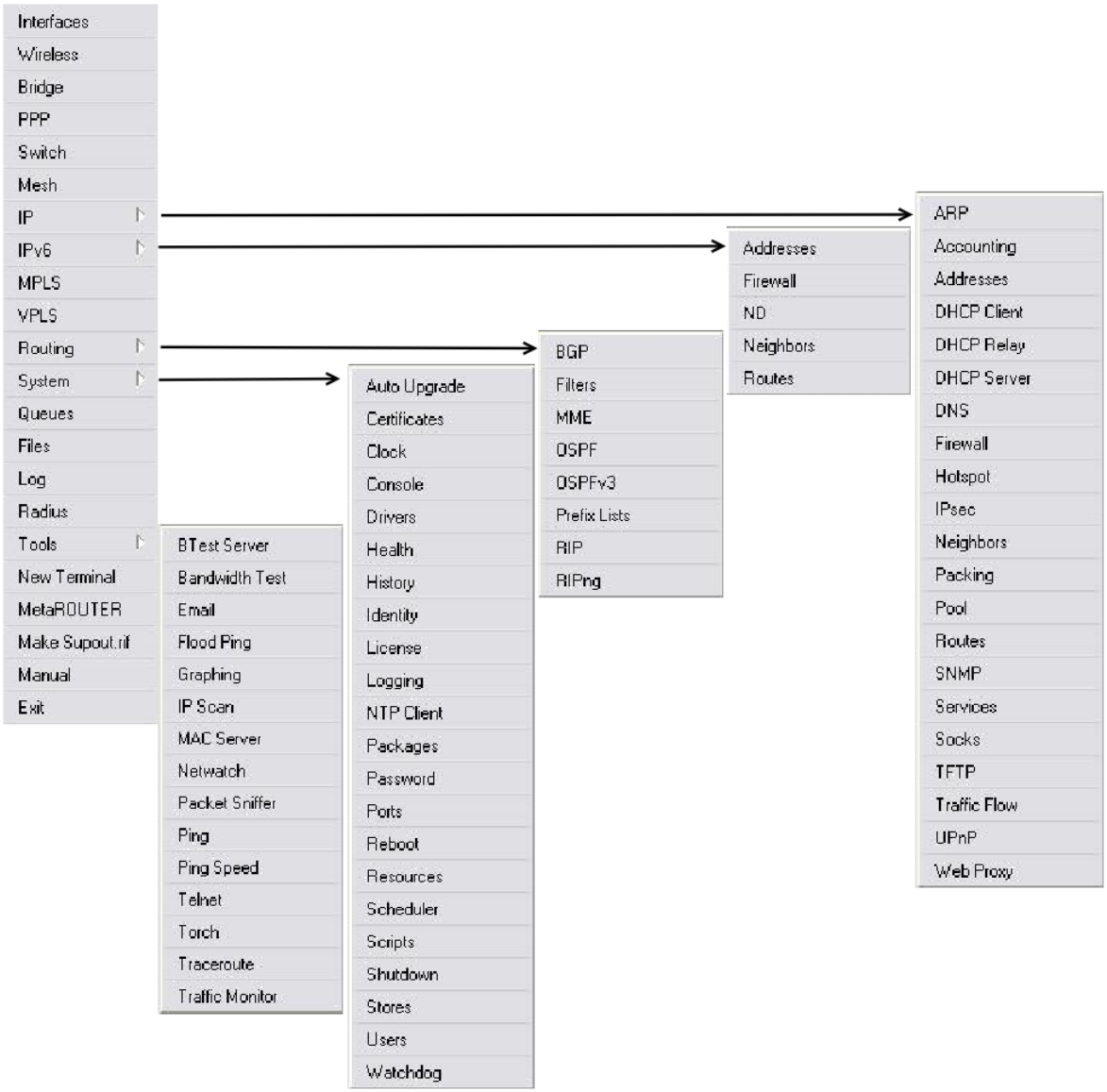
Tabulka 1 – Přehled balíčků

Název balíčku	Popis funkce
advanced-tools	Pokročilé nástroje. Ping, NetWatch, ip-scan, wake-on-lan
dhcp	Podpora protokolu DHCP klient a server
gps	Podpora zařízení GPS
hotspot	Správa uživatelů veřejného přístupového bodu
ipv6	Podpora adresování IPv6
mpls	Podpora Multi Protocol Labels Switching
ntp	Network Time Protocol klient i server
ppp	Podpora PPP, PPTP, L2TP, PPPoE klient i server
routerboard	Základní balík pro správu a přístup pro RouterBoard
routing	Balík pro dynamické směrovací protokoly
security	Podpora pro IPsec, SSH, Winbox
system	Balík základních funkcí. Statické routování, IP adresy, SNMP, Telnet, VLAN, Firewall, DNS, TFTP, SMTP
user-manager	Správa uživatelů Mikrotik RouterOS
wireless	Podpora bezdrátových rozhraní
kvm	Podpora vizualizace KVM (dostupné pouze v balíku x86)

2.2 Možnosti využití Mikrotik RouterOS

Aplikace, ke kterým je možné využít platformu RouterBoard s Mikrotik RouterOS, jsou patrné z výše uvedené tabulky funkcí. Konfigurace a nastavení provádíme pomocí aplikace WINBOX, což je grafické rozhraní pro nastavování systému Mikrotik RouterOS, nebo

pomocí Command Line Interface dostupný přes SSH nebo Telnet. Struktura příkazů odpovídá struktuře systémového menu WINBOXu a je patrná z následujícího obrázku.



Obrázek 2 – Struktura menu

Struktura menu je uvedena pro instalaci kombinovaného balíku verze 4.16 (mipsbe) a aktivaci všech instalovaných balíčků.

2.3 Hardwarová specifikace

Každá hardwarová specifikace (typ RouterBoardu) má definovaný typ softwarového balíčku, který se pro danou specifikaci používá. Existují čtyři různé verze (typy) softwarových balíčků. Následující tabulka uvádí, které specifikaci RouterBoardu odpovídá ten který typ balíčku [3].

Tabulka 2 – Specifikace hardware RouterBoard a software RouterOS

Typ HW specifikace	Typ balíčku
RB Crossroads	Mipsle
SXT	Mipsbe
RB 100 série	Mipsle
RB 200 série	x86
RB 300 série	Ppc
RB 400 série	Mipsbe
RB 500 série	Mipsle
RB 600 série	Ppc
RB 700 série	Mipsbe
RB 800 série	Ppc
RB 1000 série	Ppc
PC	x86

Z výše uvedené tabulky je patrné, že existuje několik hardwarových platforem, které se navzájem odlišují nejen technickými parametry, ale i úrovní svého vybavení. Jsou k dispozici základní jednotky, které jsou osazeny pouze jedním metalickým ethernetovým portem, jedním portem miniPCI a základní verzí procesoru s malou kapacitou paměti. Oproti tomu stojí jednotky s výkonným procesorem Power PC MPC 8533 1066 MHz, osazeny 13 porty 10/100/1000 Mbps, vybaveny slotem pro microSD a samozřejmě sériovým portem RS232. Ve střední řadě mohou být jednotky vybaveny rozhraním USB, slotem pro GSM kartu či slotem pro miniPCI pro 3G modemy. Základní parametry jednotlivých typů jsou popsány v následující tabulce. Kompletní hardwarová specifikace je pak uvedena v příloze této práce[3].

Tabulka 3 – Hardwarová specifikace I

	SXT	**250GS**	**750/G**	**1100**	**411**
Využití	P-t-P	Switch	Router	Router	Client
Procesor	400 MHz		AR7161	800MHz	300MHz
RAM	32MB		32MB	512MB	32MB
Architektura	MPIS-BE	RISC	MIPS-BE	PPC	MIPS-BE
LAN port	1x100Mbps	5x1Gbps	5x100/1000mbps	13x1Gbps	1x100Mbps
MiniPCI	Integrovaná				1
USB	Ano				Dle typu
Karta				microSD	
Licence	Level3		Level4	Level6	Level3
Napájení	PoE	9-28VDC	10-28VDC	12-24VDC	10-28VDC

Tabulka 4 – Hardwarová specifikace II

	433	**493**	**450/G**	**800**	**711**
Využití	AP/klient	AP/client	Router	AP/Router	Client
Procesor	300 MHz	300 MHz	300 MHz	800MHz	400MHz
RAM	64MB	64MB	32MB	256MB	32MB
Architektura	MPIS-BE	MIPS-BE	MIPS-BE	PPC	MIPS-BE
LAN port	3x100Mbps	9x100Mbps	5x100/1000mbps	3x1Gbps	1x100Mbps
MiniPCI	3	3		4	
USB	Dle typu	Dle typu			
Karta				1xCF	
Licence	Level4	Level4	Level4	Level6	Level3
Napájení	10-28VDC	10-28VDC	10-28VDC	10-56VDC	10-28VDC

3 VIRTUÁLNÍ PRIVÁTNÍ SÍŤ

Tato část práce popisuje možná řešení zabezpečeného propojení lokálních počítačových sítí ve vzdálených pobočkách společnosti. Popisuje možné způsoby realizace zabezpečeného propojení celých sítí prostřednictvím sítě Internet, ale i zabezpečeného propojení samostatného klienta, a jsou zde popsány i alternativní možnosti řešení nevyužívající síť Internet. Jsou zde popsány samotné způsoby realizace takového bezpečného propojení za použití technologie RouterBoard a MikrotikRouterOS, realizace pomocí technologie Cisco, pomocí opensource řešení i řešení pro OS Windows.

3.1 VPN – důvod použití

Je téměř nemožné si představit, že v dnešní době může fungovat výrobní podnik, školská zařízení, obchodní společnosti, ale i malá společnost bez informačního systému obsahujícího veškerá data o zákaznících, cenách výrobků, ekonomických a jiných informacích. Z výše uvedeného je zjevné, že se jedná o data, která jsou, když ne přímo tajná, tak určitě neveřejná. Problémem však je, jak zajistit přístup k datům uloženým na serveru v centrále firmy pracovníkům, kteří pracují v pobočce firmy v jiném městě či jiném státě, jak zajistit přístup k datům na serveru pro obchodní zástupce, kteří jsou každý den na jiném místě atd.

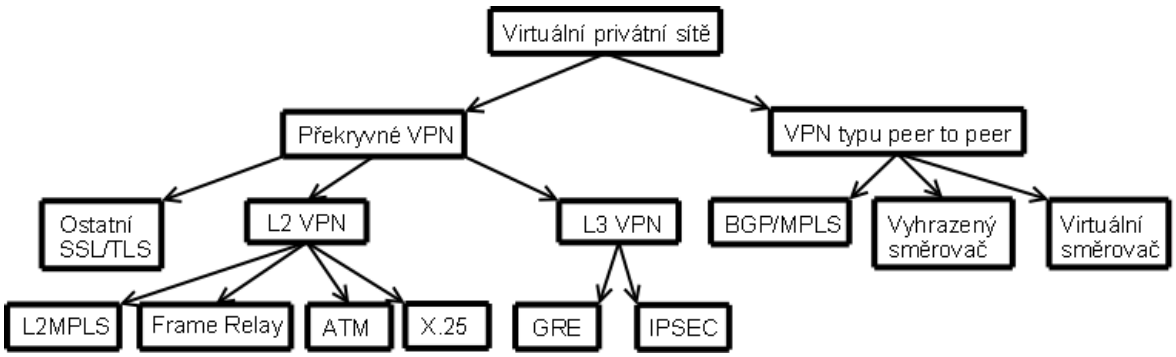
Pro propojení dvou poboček výrobního podniku je možné hledat řešení ve vybudování datového propojení za použití vlastní technologie, např. bezdrátový radioreléový spoj, či optická kabeláž, zde se ovšem potýkáme s ekonomikou celého řešení (realizace vlastní optické kabeláže mezi pobočkami v Praze a Zlíně se nemůže nikdy vyplatit) a také na technická omezení (překonat vzdálenost 10 km v členitém terénu může představovat tři i více kusů bezdrátových spojů). Budování vlastních datových vedení pro obchodního zástupce, který má v portfoliu zákazníky ve střední a východní Evropě, je nemožné.

V těchto případech je využívána nová kategorie podnikových sítí, a to virtuální neveřejné podnikové sítě, které jsou specifické efektivním využitím veřejných sítí. Virtuální privátní sítě (VPN, Virtual Private Network) využívají veřejnou či sdílenou komunikační infrastrukturu, např. Internetu, veřejné sítě na bázi protokolu IP, ale i na bázi veřejné sítě ATM. Jedná se tedy o zabezpečené propojení privátních (soukromých) sítí přes síť veřejnou, budovanou na úrovni druhé nebo třetí vrstvě síťové architektury. Veškerý

provoz, který je realizovaný v síti VPN, je šifrovaný a komunikace mezi jednotlivými stanicemi v rámci VPN je bezpečná. Taková síť pak vytváří dojem, že se jedná o komunikační infrastrukturu vyhrazenou pro potřebu konkrétní organizace, školy, či podniku, přestože je ve skutečnosti sdílená s dalšími uživateli, jelikož celé řešení je postaveno na veřejné infrastruktuře. Na takto realizovanou VPN je pak možné nahlížet jako na bezpečnou síť, kde jsou zdánlivě všichni uživatelé připojeni lokálně a všechny prostředky jsou dostupné.

3.2 Modely VPN

Modely virtuálních privátních sítí popisuje následující obrázek. Na sítích ATM nebo Frame Relay se virtuální okruhy budují mezi koncovými zařízeními (bránami VPN), podobně lze použít tunely u IP VPN. U VPN typu peer-to-peer se přesouvá poskytování služeb VPN z koncových zařízení na síť a o provoz VPN sítě se stará poskytovatel IP služby[1].



Obrázek 3 – Modely VPN

3.3 Typy VPN

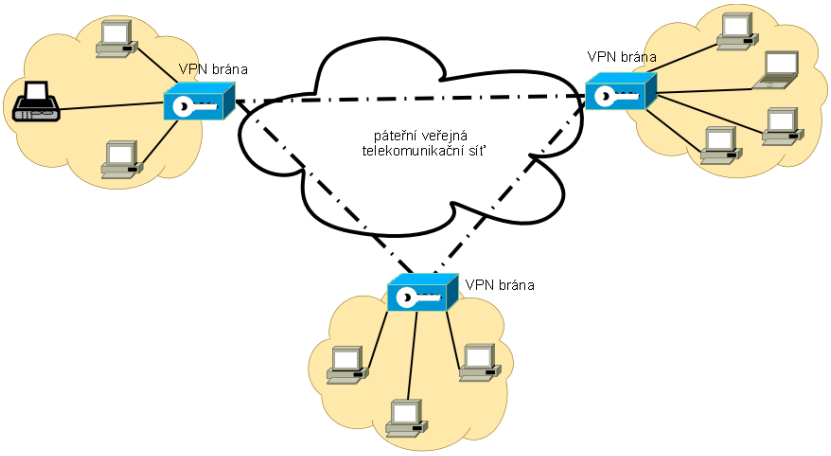
Z výše uvedeného je patrné, že se budou řešit dva různé typy požadavků na zabezpečené připojení do lokální sítě. V prvním případě bude nutné zajistit bezpečnou komunikaci přes veřejnou síť pro dva či více geograficky distribuovaných pobočkových intranetů do jednoho velkého firemního intranetu. V druhém případě se řeší připojení vzdáleného uživatele k podnikovému intranetu, využívající např. mobilní připojení či domácí připojení od nedůvěryhodného poskytovatele. V takovýchto případech pak VPN brána musí vykonávat ještě další funkce, jako např. DNS, DHCP atd. Bez ohledu na typ použité VPN je nutné si uvědomit, že VPN je tak kvalitní (z pohledu kvality služeb QoS), jak kvalitní

je připojení do veřejné komunikační sítě, přes kterou je pak VPN realizována. Klasická VPN je patrná z obrázku 4.

Obrázek 4 – Virtuální privátní síť

3.3.1 VPN typu LAN-to-LAN

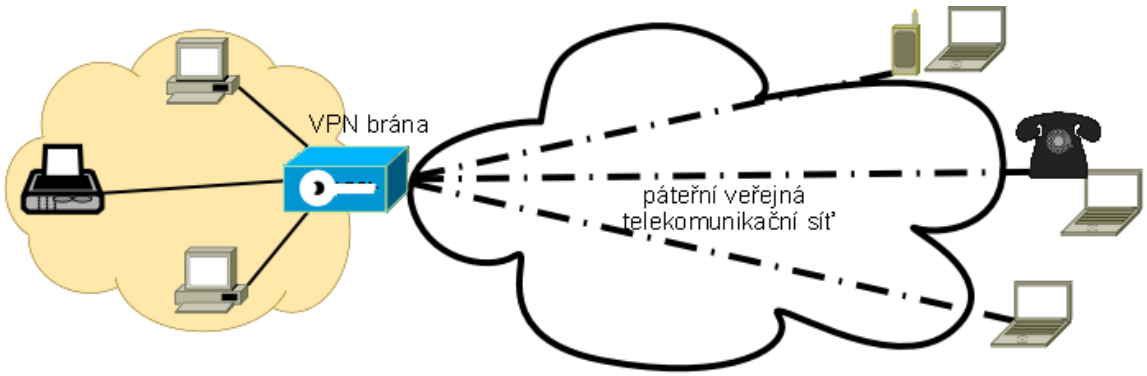
VPN typu LAN-to-LAN (site-to-site) představuje řešení, kdy jsou realizovány dvě samostatné sítě a následně vzniká potřeba propojení do jednoho velkého firemního intranetu. Z uživatelského pohledu se jedná o přívětivější variantu. Pro navázání a provoz VPN není nutný zásah uživatele, ani není nutné zajištění spouštění speciálního software na jednotlivých klientských stanicích. O chod VPN se starají VPN brány, které zajistí šifrované (bezpečné) spojení lokálních sítí připojených za těmito branami prostřednictvím veřejné telekomunikační sítě. Veškerá konfigurace a případná změna nastavení se provádí pouze na těchto branách a z pohledu uživatele v lokální síti nevyžaduje žádného zásahu v nastavení komunikace. Užívá se jak asymetrického, tak symetrického šifrování. Asymetrické šifrování je použito při navázání spojení a symetrické pak pro šifrování datového toku. Propojení sítí typu LAN-to-LAN je znázorněno na obrázku 5.



Obrázek 5 – VPN LAN-to-LAN

3.3.2 VPN typu vzdálený přístup

VPN typu vzdálený přístup (remote access) má za úkol zajistit zabezpečené připojení do firemního intranetu jednotlivým uživatelským stanicím, nikoli celé LAN. Z uživatelského pohledu se může jevit toto řešení jako složitější (pokud se za složitost považuje spuštění jednoho aplikačního souboru). U těchto typů VPN řešení je na rozhraní lokálního intranetu a veřejné telekomunikační sítě instalováno zařízení (VPN gateway) zajišťující navázání bezpečného spojení s klientem, který je většinou v podobě softwaru odpovídajícího konkrétnímu řešení. Jiného klienta používá řešení od společnosti Cisco, jiného samozřejmě řešení založené např. na opensource. Konfigurace a nastavení šifrovaného spojení je nutné provádět nejen v této bráně, ale i v každé konkrétní stanici, která má šifrovanou komunikaci používat. Konfigurace je poměrně jednoduchá a většinou ji provádí správce systému. U pracovní stanice se jedná o instalaci a konfiguraci softwarového klienta a instalaci vygenerovaného klíče. Obrázek 6 popisuje typ VPN vzdálený přístup.



Obrázek 6 – VPN typu remote access

3.4 Realizace VPN

3.4.1 Základní prvky IP VPN

Základními prvky IP VPN řešení jsou dle RFC 2764 následující komponenty:

- VPN brána (VPN gateway) – zařízení zajišťující propojení celé sítě k VPN. U VPN sítí typu LAN-to-LAN se jedná o zařízení, která jsou instalována na rozhraní lokální a veřejné sítě. U VPN typu remote access se jedná o zařízení, se kterým navazují šifrované spojení jednotliví klienti v podobě speciálního softwaru, příp.

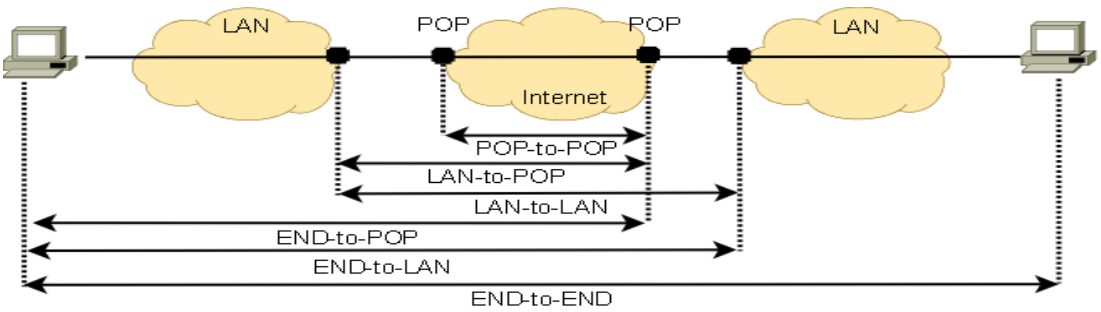
speciálního hardwarového zařízení, podobně jako u VPN typu LAN-to-LAN. Brána musí zajistit bezpečný přístup do lokální sítě pro oprávněné uživatele a udržet neoprávněný provoz vně sítě. Dále se musí postarat o šifrování komunikace mezi sítěmi a většinou i o překlad adres (NAT). Ve většině případů brány podporují různé autentizační mechanizmy. Mohou také zajišťovat služby QoS, kdy provádí klasifikaci a označování provozu na typu kvality služby. Brána je realizována prostřednictvím směrovače (routeru), firewallu či jiného zvláštního zařízení.

- VPN klient – většinou software, který se instaluje na koncová zařízení a který je protistranou pro VPN bránu.
- Autentizační server – jedná se o systémy, které plní funkci např. Radius serveru pro zajištění identity bran a VPN klientů.
- VPN server pro správu – zajišťuje monitoring a řízení VPN a taktéž se stará o zajištění informací o stavu VPN.
- Fyzický přenos – spojení po libovolné IP síti či Internetu.

Ne všechny prvky jsou pro správnou funkci nutné. Není možné ovšem realizovat VPN řešení bez VPN brány, VPN klienta a samozřejmě zajištěné přenosové trasy [5].

3.4.2 Tunely

VPN využívá tzv. tunelování mezi koncovými klientskými sítěmi, resp. mezi sítí a koncovým klientem, kdy tunel představuje logický dvoubodový spoj. Tunely mohou být realizovány mnoha způsoby s ohledem na typ VPN. Je možné realizovat spoj typu END-to-END, což je tzv. koncový tunel nebo např. tunel mezi přístupovými místy k Internetu (POP-to-POP). Následující obrázek 7 popisuje možnosti vytváření tunelů.



Obrázek 7 – Typy tunelů

Tunel je definován dvěma koncovými body (vstupem a výstupem z tunelu) a mechanizmem, který je používán při přenosu paketu tunelem. Koncový bod zajišťuje např. autentizaci, řízení přístupu a dojednávání dalších bezpečnostních služeb. Bezpečný tunel chráněný bránami VPN umožňuje klientům VPN přístup k privátním zdrojům. Tunelování lze použít jako transportní mechanizmus, ale lze jej použít i pro zajištění bezpečnosti, kdy se nezabezpečený paket vkládá do zabezpečeného (zašifrovaného) paketu.

3.4.2.1 Tunelování na druhé vrstvě

Tunelování na druhé (spojové) vrstvě představuje tunelování rámců, kdy spojení iniciuje buď klient sám, nebo jej může iniciovat přístupový server, aniž by došlo ze strany klienta k jakékoli iniciativě. U protokolů využívajících tunelování na druhé vrstvě síťové architektury je nutné, aby byl tunel vytvořen, spravován a ukončen protokolem druhé vrstvy. Typickými protokoly pro tunelování na druhé vrstvě je PPTP a L2TP.

3.4.2.2 Tunelování na třetí vrstvě

Tunelování na třetí (síťové) vrstvě vyžaduje zapouzdření IP datagramu do jiného datagramu. Tímto dojde k potlačení potencionálně problematické záležitosti adresace. Nejčastěji používaný bezpečnostní mechanizmus je IPSec (Internet Protokol Security). Konfiguraci tunelů je nutné provádět manuálně a předem.

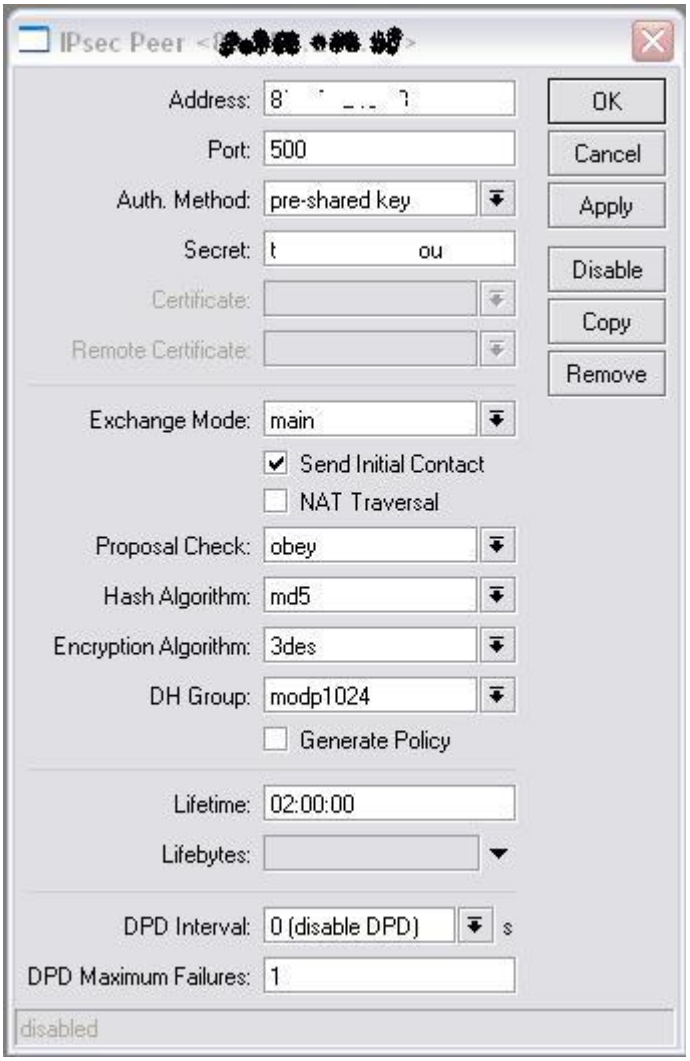
3.5 IPsec

Architekturu IPsec popisuje RFC2401. IPsec je v podstatě rozšíření protokolu IP, které zajišťuje bezpečnost IP protokolu a protokoly vyšších vrstev. IPsec pracuje ve dvou různých módech. Tunelový mód chrání celý datagram IP, naopak mód transportní pouze protokoly vyšších vrstev. Při použití tunelového módu je celý IP datagram zapouzdřený do nového IP datagramu, chráněného protokolem IPsec, kdežto při použití transportního módu je pouze část IP datagramu zpracovaná IPsec protokolem [1] [6]. V následujících odstavcích jsou pro základní orientaci uvedeny různé způsoby konfigurace.

3.5.1 IPsec na MikrotikRouterOS - Linux

Mikrotik RouterOS umožňuje využití architektury IPsec pro vytvoření VPN spojení mezi dvěma koncovými body přenosové soustavy. VPN řešení není závislé na použité platformě

a je možné navázat VPN spojení na bázi IPsec mezi různými platformami navzájem, Mikrotik RouterOS nevyjímaje. Konkrétní realizace je popsána v praktické části této práce. Konfiguraci VPN brány na MikrotikRouterOS popisuje následující obrázek.



Obrázek 8 – Ukázka konfigurace IPsec na platformě MikrotikRouterOS

Konfigurace OS Linux VPN brány je uložena v `/etc/ipsec.conf`, samotný klíč je pak uložen v `ipsec.secret`.

```
conn VPN-1
        authby=secret
        left=8x.x1.2xx.xx
        leftsubnet=192.168.1.0/24
        right=7y.y5.1yy.yy0
        rightsubnet=192.168.2.0/24
        pfs=no
        auto=start

/etc/ipsec.secret
8x.x1.2xx.xx 7y.y5.1yy.yy0: PSK "tesnepredzkouskou"
```

3.5.2 IPsec Linux – Cisco

Pro implementaci VPN brány na Linuxu je použit Openswan. Ve VPN bráně tvořené OS Linux je konfigurace uložena v `/etc/ipsec.conf`, samotný klíč je pak uložen v `ipsec.secret`.

```
config setup
conn alias
        authby=secret
        left=8x.xx1.2xx.x             #IP adresa levé strany
        leftsubnet=10.203.1.0/24      #IP rozsah na levé straně
        leftnexthop=8x.xx1.2xx.x2x    #IP adresa výchozí brány
        right=8y.yy1.2yy.y            #IP adresa pravé strany
        rightsubnet=10.203.2.0/24     #IP rozsah na pravé straně
        esp=3des                      #způsob šifrování uvnitř VPN
        ike=3des-md5-modp1536,3des-md5-modp1024      #autentizace
        auto=start

/etc/ipsec.secret
8x.xx1.2xx.x 8y.yy1.2yy.y: PSK "1234567890123456"
```

Konfigurace routeru Cisco bude vypadat takto:

```
sakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2

crypto isakmp key 1234567890123456 address 8x.xx1.2xx.x no-xauth

crypto ipsec transform-set FREESWAN esp-3des esp-md5-hmac

crypto map VPN 10 ipsec-isakmp
 set peer 8x.xx1.2xx.x
 set transform-set FREESWAN
 set pfs group2
 match address 100

access-list 100 permit ip 10.203.2.0 0.0.0.255 10.203.1.0 0.0.0.255
```

3.5.3 IPsec Linux - Linux

Konfigurace uložena v `/etc/ipsec.conf`, pro uložení klíče slouží `ipsec.secret`.

```
config setup
        interfaces="ipsec0=eth0"
        # Debug-logging controls:  "none" for (almost) none, "all" for
lots.
        klipsdebug=none
        plutodebug=none
        uniqueids=yes

conn %default
        keyingtries=0
        disablearrivalcheck=no
        authby=rsasig

conn uh-gub
        # Left security gateway, subnet behind it, next hop toward right.
```

```
left=xx.xx1.2xx.17
leftsubnet=192.168.1.0/24
leftnexthop=xx.xx1.2xx.xx4
leftid=@xxx.xxxx.uh.cz
leftrsasigkey=123456789
# Right security gateway, subnet behind it, next hop toward left.
right=yy.yy1.2yy.41
rightsubnet=192.168.2.0/24
rightnexthop=yy.yy1.2yy.yy6
rightid=@yyy.yyyy.uh.cz
rightrsasigkey=987654321
auto=start
```

Analogicky bude vytvořena konfigurace „protistrany“, kde se v konfiguračním souboru vymění levá strana za pravou a opačně.

3.5.4 IPsec MikrotikRouterOS – MS Windows

Realizace VPN mezi platformou RouterBoard a platformou Windows je popsána v případové studii v odstavci 8.1, která ovšem popisuje realizaci VPN za použití OpenVPN. Mikrotik RouterOS neumožňuje provedení konfigurace brány IPsec pro VPN typu remote access podobně jako např. Cisco a je nutné tento požadavek řešit realizací L2TP/IPsec.

3.6 OpenVPN

OpenVPN je řešení založené na SSL/TLS, které poskytuje stejnou funkci L3 VPN jako řešení založená na IPsec technologii. Toto řešení je dostupné na různých platformách (MS Windows, Linux, BSD) a není kompatibilní s žádnou další VPN technologií (IPsec, PPTP, L2TP). Podporuje komunikaci např. přes NAT a HTTP.

3.6.1 OpenVPN Linux - WIN

Ukázka konfigurace VPN brány pro jednoduchou konfiguraci.

```
local 192.168.1.10                          #vnější adresa VPN GW
ca ca.crt
cert server.crt                             #certifikát a klíče umístěny
key server.key                              #ve stejném adresáři jako conf
dh dh2048.pem
server 192.168.100.0 255.255.255.0          #adresace VPN klientů
push ”route 10.10.10.0 255.255.255.0”       #adresace vnitřní sítě
cipher AES-128-CBC                          #šifrovací algoritmus
comp-lzo
log /var/log/openvpn.log
```

Příklad konfigurace klienta pro takto definovanou bránu:

```
client
remote 192.168.1.10 1194                    #adresa VPN brány a port
```

```
ca ca.crt
cert klient.crt
key klient.key
cipher AES-128-CBC
comp-lzo
```

3.7 Další možnosti bezpečného propojení sítí

3.7.1 Vlastní telekomunikační infrastruktura

Pokud je realizováno propojení LAN-to-LAN prostřednictvím vlastní telekomunikační infrastruktury, lze využít různých technologií. Zpravidla je možné realizovat vlastní telekomunikační infrastrukturu pomocí následujících přenosových médií:

- Bezdrátové technologie
  - ve volném pásmu – pásmo 2,4 GHz, 5,4 GHz, 10 GHz, 70-80 GHz,
  - v licencovaném pásmu – pásmo 11 GHz, 18 GHz, 26 GHz, 34 GHz.
- Metalická kabeláž
  - technologie DSL,
  - ethernet.
- Optická vlákna
  - SM vlákna,
  - MM vlákna.

Každé z výše uvedených řešení má své výhody a nevýhody. Oproti realizaci propojení pomocí VPN není vytvořena závislost na dodavateli internetové konektivity a zabezpečení provozu je řešeno za použití vlastního technologického zařízení. U bezdrátových řešení je třeba se obeznámit se zabezpečením rádiového protokolu z důvodu možného odposlechu při nedostatečném stupni zabezpečení. Tímto způsobem je možné samozřejmě realizovat pouze propojení poboček, nikoli ovšem propojení pro uživatele typu remote access.

3.7.2 Vyhrazené datové kanály na infrastruktuře třetích stran

Zjednodušeně lze toto řešení popsat jako kombinaci již výše představených řešení. Nevyužívá se vlastní telekomunikační infrastruktura, ale ani veřejná telekomunikační síť či Internet. Využívá se zpravidla telekomunikační síť operátora, kde „oba konce“ a celá

přenosová trasa je pod kontrolou a dohledem jednoho subjektu. Tzn., že je přesně stanovená topologie sítě mezi dvěma připojovacími body, což představuje jistou výhodu oproti využití Internetu pro realizaci VPN.

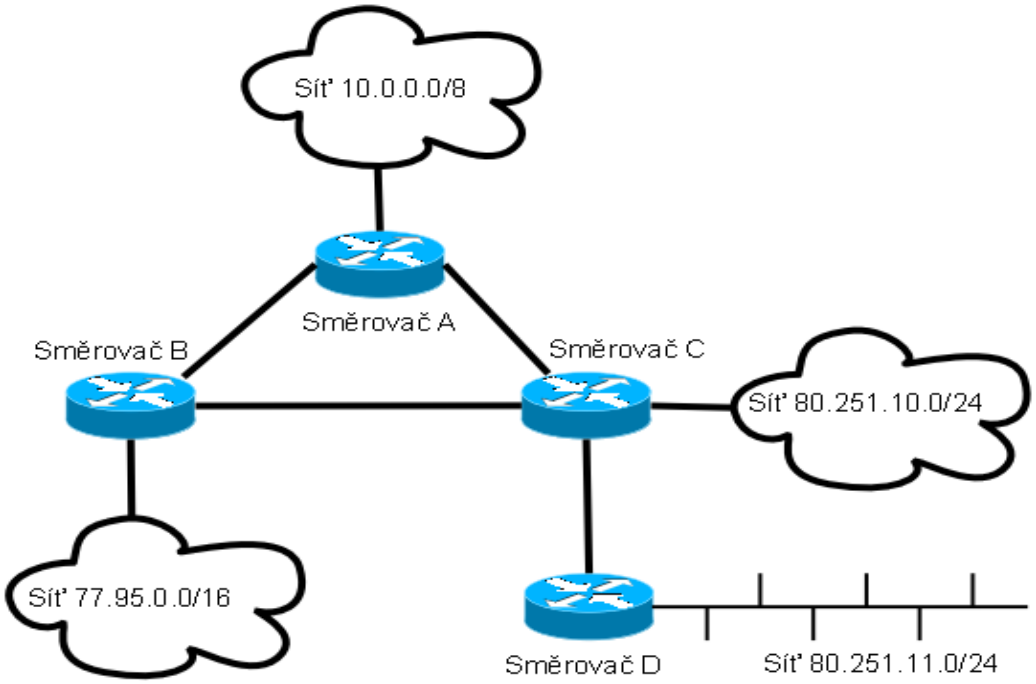
4 SMĚROVÁNÍ - ROUTING

Směrování je základním principem propojování sítí. Cílem směrování je nalezení cesty sítí, po níž se má datagram poslat k cílové stanici (nejčastěji se jedná o síť, ve které stanice sídlí), a to cesty takové, která je nejlepší podle stanovených kritérií. Tato kritéria jsou stanoveny na základě volby daného typu směrování či směrovacího protokolu. Směrování je proces zajišťující nalezení vhodné cesty a směrovač je zařízení, které provádí rozhodnutí o vhodnosti té které cesty. Při vzájemné komunikaci dvou počítačů v rámci jedné LAN (ve stejné IP síti) stačí pouze odvysílat patřičné datové rámce do přenosového média. Při navazování spojení mezi zdrojovým a cílovým počítačem ve WAN síti (např. Internetu) jsou od sebe oba počítače odděleny blíže neurčeným počtem síťových hardwarových zařízení. Je nemožné odeslat pakety na všechna tato zařízení a doufat, že paket nakonec do cíle nějak dorazí. Logickým řešením je provést identifikaci cesty v internetové síti od výchozího bodu do cíle. Nalezení cesty, resp. nalezení nejlepší cesty, se neobejde bez směrovače a vzájemné komunikaci mezi nimi. Není podmínkou, aby cesta mezi dvěma stanicemi ve WAN síti byla stejná pro oba směry komunikace, tzn., že pro komunikaci směrem tam je jiná cesta než pro komunikaci zpět [1] [7].

4.1 Směrovače

Směrovače jsou inteligentní zařízení v LAN síti, které na rozdíl od jiných aktivních prvků v LAN sítích pracují na třetí (síťové) vrstvě referenčního modelu OSI, nikoli pouze na prvních dvou (jako např. switche či huby). Díky této vlastnosti mohou směrovače vzájemně propojovat různé sítě LAN prostřednictvím adresování třetí vrstvy. Směrovač musí mít minimálně dvě rozhraní (většinou to bývá i více), která jsou připojena k různým LAN sítím. Směrovač zjišťuje adresy počítačů a sítí, které jsou připojeny k jednotlivým rozhraním, a jejich seznam ukládá do tabulky, ve které je definováno, v jakém vztahu jsou adresy třetí vrstvy a jednotlivé porty, k nimž jsou příslušné systémy přímo či nepřímo připojeny. Každý směrovač zajišťuje dvě základní funkce. První z nich je zjištění a výběr nejlepší cesty v síti a druhou je vnitřní přepínání paketů ze vstupního portu na výstupní. Taktéž protokoly, se kterými každý směrovač pracuje, jsou dva. Oba působí na třetí vrstvě a označují se jako směrované (směrovatelné) a směrovací protokoly. Směrované protokoly zapouzdřují uživatelské informace a data do podoby paketů. Asi nejznámější je protokol IP, jehož úkolem je zapouzdření aplikačních dat pro síťový přenos. Oproti tomu směrovací

protokoly běží jen mezi jednotlivými směrovači, které podle nich sestavují dostupnost a kvalitu cesty, vyměňují si o nich informace a následně po této cestě přeposílají pakety směrovaného protokolu. Směrovače tedy slouží k přeposílání datových paketů mezi jednotlivými zařízeními, která nemusí být připojena ke stejné lokální síti [7]. Topologie sítě se směrovači je patrná z následujícího obrázku.



Obrázek 9 – Síť se směrovači

4.2 Statické směrování

Jedná se o typ směrování, které se dnes ještě stále používá v menších lokálních sítích, kdy není třeba automaticky řešit výpadek jedné či více přenosových tras a kdy rozsah sítě nepředstavuje pro ruční konfiguraci směrování velkou zátěž. Statické směrování používá vždy aktuálně jen jednu cestu k cíli, která je předem manuálně zadána ve směrovači správcem systému a nepodporuje možnost alternativní cesty pro případy výpadku přenosové trasy či směrovače v dané cestě. Směrovač nemá možnost dynamického přesměrování, tzn., že žádný z vyslaných paketů nemá možnost dorazit do cíle, pokud nedojde k odstranění poruchy na přenosové trase nebo pokud správce systému manuálně neprovede změnu v konfiguraci směrování v jednotlivých směrovačích. U některých směrovačů je možné provést konfiguraci směrování více statických cest pro rozložení celkové zátěže mezi jednotlivými cestami. I když se může na první pohled zdát statické

směrování jako nevyhovující, existují případy, kdy je dobré či postačující je použít. Statické směrování je výhodné použít v případech, kdy je nutné především z důvodu bezpečnosti zajistit volbu konkrétní, předem známé a definované cesty. Taktéž se statické směrování volí v případech, kdy existuje pouze jedna jediná cesta do cílové sítě a tím pádem je dynamické směrování zbytečné (zbytečně by zatěžovalo směrovač i okolní směrovače při zjišťování možných alternativních cest). Statické směrování lze kombinovat v jedné síti se směrováním dynamickým. V takovémto případě má obecně statické směrování „přednost“ před směrováním dynamickým (toto je otázka nastavení a lze prioritu měnit) z toho důvodu, že cesta nastavená manuálně správcem systému je považována za lepší. V opačném případě, kdy dynamické směrování má přednost před statickým, využíváme statického směrování jako zálohu při výpadku směrovacího protokolu (nesmí ovšem zároveň při výpadku směrovacího protokolu dojít i k výpadku směrovače samotného). Konfigurace směrovače pro statické směrování může být v rozlehlých sítích poměrně komplikovaná a je nutné ji provádět s maximální přesností a obezřetností. Výhodou takto provedené konfigurace je např. nulová režie směrovacího protokolu (nepoužívá se) a směrovač nepotřebuje provádět aktualizace směrovací tabulky. Srovnání některých parametrů statického a dynamického směrování je uvedeno v následující tabulce [1].

Tabulka 5 – Vlastnosti statického a dynamického směrování

Vlastnost	Statické směrování	Dynamické směrování
Automatická reakce na změny v síti	nepodporuje	Ano
Možnost rozložení zátěže do více cest	lze (dle směrovače)	lze (dle protokolu)
Účast správce při rekonfiguraci	vysoká	Nízká
Dohled nad používanými cestami	vysoký	Malý
Výměna směrovacích informací	žádná	Vysoká
Zátěž směrovače	nízká	Vysoká
Zátěž paměti	nízká	Vysoká
Zátěž sítě	ne	Střední

4.2.1 Implicitní cesta

Implicitní cesta je speciálním případem statického směrování. Z praxe známe tento pojem jako „výchozí brána“ či „default gateway“. Implicitní cesta stanovuje cestu sítí pro všechny jinak nespecifikované sítě. Jedná se o nejznámější a nejrozšířenější funkci směrovače zapojeného v úrovni sítě LAN jako brány do sítě WAN či Internetu.

Výpis směrovací tabulky v systému MikrotikRouterOS vč. Implicitní cesty:

#	DST-ADDRESS	PREF-SRC	GATEWAY-STATE	GATEWAY	DISTANCE	INTERFACE
0 A S	0.0.0.0/0		reachable	80.251.244.126	1	ether1-wan
1 ADC	80.251.244.0/25	80.251.244.10			0	ether1-wan
2 ADC	192.168.1.0/24	192.168.1.100			0	bridge1
3 ADC	192.168.2.0/24	192.168.2.1			0	bridge2

Staticky definované cesty dle předchozího obrázku.

Tabulka 6 – Statická definice cest

Směrovač	Cíl	Další přeskok
A	77.95.0.0	B
A	80.251.0.0	C
B	10.0.0.0	A
B	80.251.0.0	C
C	10.0.0.0	A
C	77.95.0.0	B
C	80.251.11.0	D

4.3 Dynamické směrování

Dynamické směrování se automaticky stará o nalezení alternativní cesty v případě poruchy na původně vybrané cestě, proto někdy hovoříme o adaptivním směrování. Dynamické směrování používá k výběru nejlepší cesty do cílové sítě algoritmus, který je založený na aktuálních směrovacích informacích. Tyto informace dostává směrovač od sousedních

směrovačů v síti a zároveň předává své informace sousedním směrovačům v síti. Směrovací protokol řídí výměnu těchto informací, které se posílají podle typu protokolu buď v pravidelných intervalech, nebo v případě detekce změny síťové topologie (používá se i kombinace obou případů). Informace předávané mezi směrovači obsahují výčet dostupných sítí a hodnotu cesty, kterou se může datagram do cíle dostat. V případě dynamického směrování se používají dva základní způsoby (algoritmy) pro určení nejlepší cesty:

- Směrování s vektorem vzdáleností – směrovače předávají pravidelně kopie své směrovací tabulky bezprostředním sousedům v síti. Každý příjemce přičte k tabulce svůj vlastní vektor vzdáleností a předá je opět svým bezprostředním sousedům. Směrovače se postupnými kroky dozví o ostatních směrovačích a vytvoří si představu o vzdálenostech v síti. Podle výsledné tabulky se pak aktualizují směrovací tabulky jednotlivých směrovačů. O ostatních směrovačích, ani o skutečné topologii sítě, žádné informace směrovače nezískávají.
- Směrování se stavem linky – tento algoritmus směrování využívají protokoly obecně označované jako protokoly nejkratších cest (Shortest Path First - SPF). Zjišťují úplné informace o směrovačích v síti a způsobu jejich vzájemného propojení a dále si tyto informace udržují a udržují si i složitou databázi topologie sítě. Směrovače si s ostatními směrovači vyměňují oznámení o stavu linky (Link-State Advertisements – LSA). Směrovač si ze všech přijatých oznámení konstruuje databázi s topologií sítě (mapa je ve formě grafu, jehož uzly jsou směrovače sítě, hrany jsou vzájemné přímé propojení), pomocí algoritmu vypočte dosažitelnost jednotlivých cílů v síti a aktualizuje směrovací tabulku. Tento algoritmus dokáže rozpoznat změny v topologie sítě způsobené např. poruchou, či naopak rozšířením sítě.

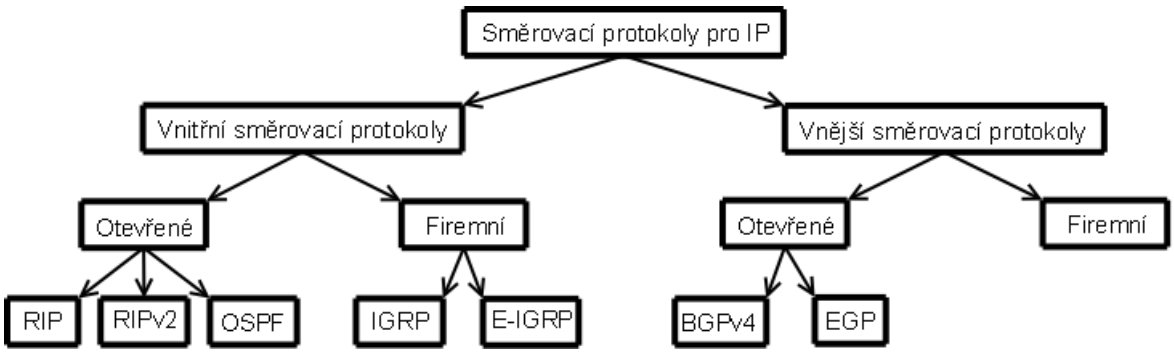
4.3.1 Autonomní systém

Autonomním systémem (Autonomous system - AS) označujeme skupinu sítí a směrovačů, které jsou řízeny z pohledu směrování paketů jednou autoritou. Směrovače uvnitř jednoho autonomního systému mohou využívat libovolné, ovšem předem definované vnitřní směrovací protokoly (Interior Gateway Protocol – IGP). Každý autonomní systému musí povinně oznamovat kořenovým směrovačům své vnitřní směrovací informace (seznam sítí,

které jsou prostřednictvím tohoto AS dostupné) prostřednictvím externího směrovače, který používá vnější směrovací protokol (Exterior Gateway Protokol – EGP). Autonomní systém označuje 16bitové identifikační číslo. Seznam identifikačních čísel udržuje stejná organizace, která se stará o přidělování síťových adres (v Evropě se jedná o organizaci RIPE).

4.3.2 Vnitřní a vnější směrovací protokoly

Protokoly zajišťující směrování uvnitř autonomního systému se označují jako vnitřní směrovací protokoly a můžeme je dále rozdělit na otevřené a firemní. Protokoly, které zajišťují komunikaci mezi autonomními systémy, se označují vnější směrovací protokoly a používají se pro propojení jednotlivých ISP, kde vzájemná redistribuce směrovacích informací se provádí v centrech pro vzájemné propojování. V dnešní době se mezi ISP používá výhradně protokol BGP[1]. Typologii směrovacích protokolů popisuje následující obrázek.



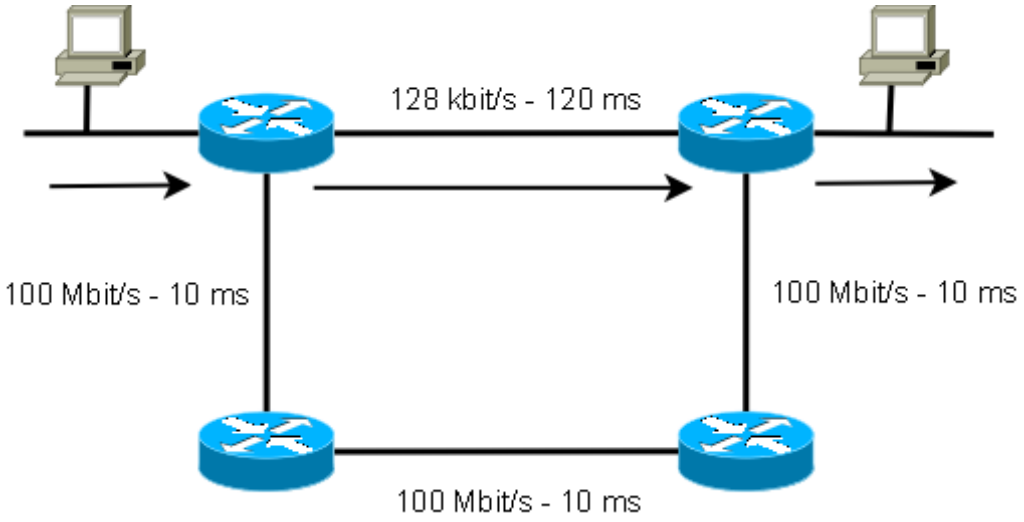
Obrázek 10 – Typologie směrovacích protokolů

4.3.3 Protokol RIP

Protokol RIP (Routing Information Protocol) byl vyvinut firmou Xerox v roce 1981. Je tedy jedním z prvních a tím i nejstarších směrovacích protokolů. Protokol RIP verze jedna je popsán v RFC 1058 a je založen na algoritmu s vektorem vzdáleností. Směrovací protokol RIP pracuje nad transportním protokolem UDP a využívá portu číslo 520. Metrikou u tohoto protokolu je počet směrovačů (hop count) na cestě k cílové stanici. Sítě přímo připojené ke směrovači mají nejnižší metriku a maximální možný počet směrovačů a nejvyšší možná metrika je 15. Vyšší metrika (16) definuje neplatnou cestu. Protokol RIP pracuje se dvěma skupinami účastníků:

- Aktivní – inzerují cesty ostatním směrovačům. Aktivním účastníkem je směrovač, nikoliv stanice v síti.
- Pasivní – pouze naslouchá aktivním účastníkům a aktualizuje si své směrovací tabulky. Pasivním účastníkem je stanice v síti.

Základní směrovací tabulku pro sítě přímo připojené k danému směrovači je nutné nakonfigurovat, nicméně další budování a aktualizace směrovací tabulky je již záležitostí protokolu RIP, který si vyměňuje směrovací informace pouze se svými nejbližšími sousedy. Za nejlepší cestu je považována cesta, která obsahuje nejmenší počet směrovačů (při stejném počtu se bere cesta nalezena jako první v pořadí), ale nic neříká o kvalitě nalezené cesty (kapacita spoje, latence). Nevýhodu tohoto způsobu nalezení nejlepší cesty popisuje následující obrázek [1].



Obrázek 11 – Nevýhoda metriky protokolu RIP, nejkratší cesta

Mezi další nevýhody tohoto dnes již málo využívaného protokolu patří zejména omezení nejdelší možné cesty na 15 směrovačů, což má za následek omezené použití protokolu RIP ve velkých a rozsáhlých sítích. Dále protokol RIP nepodporuje dynamické vyrovnávání zátěže a má poměrně velkou zátěž pro síťový přenos při aktualizaci směrovacích tabulek. Protokol RIP každých 30 sekund rozesílá své směrovací tabulky všemi směry. Problémem může být i stav, kdy v rozmezí 30 sekund čeká směrovač na aktualizaci směrovací tabulky a ke zrušení neplatné cesty dochází až po 180 sekundách, což v porovnání s jinými směrovacími protokoly je v dnešní době podstatným nedostatkem [7] [8].

4.3.4 Protokol RIPv2

Protokol RIP verze 2 je založen na protokolu RIP verze 1 a popisuje jej RFC 2453 [9]. Za nejdůležitější novinky protokolu RIP ve verzi 2 je možné považovat zejména:

- Autentizace vysílacího uzlu vůči ostatním uzlům - zajišťuje autentizaci směrovacích informací původců zpráv, což zabraňuje poškození směrovacích tabulek falešnými cestami odeslanými z neověřených zdrojů.
- Masky podsítí – umožňují využití protokolu RIP verze 2 v prostředí s podsíťovými maskami proměnné délky nebo se směrováním na bázi adresových prefixů.
- Identifikace dalšího přeskoku – zabraňuje zbytečným přeskokům.
- Skupinové adresování – využívá se situace, kde několik různých cílů v síti musí obdržet stejné informace. Není nutné generovat několik zpráv pro každý cíl samostatně, ale je možné je odeslat současně více cílům.

Nevýhoda v omezení maximálního počtu přeskoků na 15 zůstala v protokolu verze 2 nezměněna. Pokud je tedy v cestě více jak 15 směrovačů, je nutné použití jiného směrovacího protokolu.

4.3.5 Protokol OSPF

Podobně jako protokol RIP i protokol OSPF prošel určitým vývojem, kdy původní verze OSPF, označovaná jako OSPF verze 1 popsaná v RFC 1131, byla nahrazena zdokonalenou verzí popsanou v RFC 1247 a byla vzhledem k podstatnému zlepšení označena jako OSPF verze 2. Aktuální verze OSPF je popsána v dokumentu RFC 2328 [10]. Protokol OSPF používá algoritmus stavu spojů, který umožňuje poměrně rychlou konvergenci sítě reagující na změny topologie sítě (výpadky spojů, výpadky směrovačů) a nevede ke směrovacím smyčkám. Protokol OSPF pracuje přímo nad IP a používá port č. 89. Metrika protokolu OSPF je označována jako cena (cost) spoje a zahrnuje propustnost spoje, nákladů na spoj, odezvu apod. Cenu je nutné konfigurovat správcem sítě a vztahuje se k výstupnímu rozhraní směrovače. Vzhledem k tomu, že je možné přiřadit rozhraním sousedních směrovačů různou cenu, není nutné používat stejnou cestu v obou směrech. OSPF umožňuje správci sítě seskupit dohromady sítě v jednom autonomním systému do oblasti (area), do které pak náleží celý síťový segment. Směrovač tedy může ležet uvnitř oblasti, kdy je označován jako vnitřní směrovač, nebo může ležet na hranici mezi několika

oblastmi a je označován jako hraniční směrovač oblasti. Oblasti jsou označovány jako area0, area1 apod., kdy area0 je zvláštní oblast nazývaná jako páteřní a tato funguje jako transportní síť mezi ostatními oblastmi. Páteřní oblast by měla sousedit se všemi ostatními oblastmi a měla by být souvislá. Pokud tomu tak není, je možné přes ostatní oblasti vytvořit virtuální spoj. Každý směrovač má tolik topologických databází, do kolika oblastí je připojen. Oblasti rozlišujeme na tranzitní (přenáší datové toky, které nejsou určeny pro tuto oblast a ani v ní nevznikly) a oblasti listové (stub), které jsou charakteru opačného. Směrovač uvnitř oblasti neví nic o ostatních oblastech, jen ví, jaké sítě jsou dostupné přes hraniční směrovač oblasti. Pro OSPF je podstatné rozlišení typu sítě:

- Dvoubodové sítě – směrovače jsou sousedy, používají protokol o informování o své existenci a zasílají si směrovací informace.
- Rozlehlé sítě – vytváří se soubor dvoubodových spojů, mezi pověřenými směrovači je vytvořena manuální statická konfigurace.
- Lokální sítě – propojení lokální sítě více směrovači s okolím.

Pro udržování sousedských vztahů mezi směrovači a pro výměnu směrovacích informací se používá protokol OSPF Hello, který slouží také pro výběr pověřeného směrovače a jeho zálohy na základě identifikátoru jednotlivých směrovačů. Směrovací informace OSPF se posílají v paketech o stavu spojů (Link State Packet – LSP), ve kterých se popisuje lokální stav směrovače nebo sítě. Přehled zpráv OSPF popisuje následující tabulka.

Tabulka 7 – Přehled zpráv OSPF

Typ	Zpráva	Funkce
1	Hello	Ověření sousedů
2	Database Description	Sumarizace obsahu databáze
3	Link State Request	Žádost o topologickou databázi
4	Link State Update	Aktualizace databáze
5	Link State ACK	Potvrzení

4.3.6 Implementace protokolu OSPF

Pro implementaci protokolu OSPF je možné použít směrovací softwarovou sadu Quagga, která implementuje OSPF protokol, RIP protokol i BGP protokol pro platformu Unix. Démon quagga lze konfigurovat pomocí CLI, a to např. tak, že po přihlášení ke konkrétnímu směrovači a zadání příkazu *telnet localhost ospfd* se provádí konfigurace OSPF protokolu. Konfigurace protokolu je uložena v souboru, který je standardně umístěn v */etc/quagga/ospfd.conf* [11].

Ukázka konfiguračního souboru protokolu OSPF:

```
!
! Zebra configuration saved from vty
!   2009/09/09 18:17:37
!
hostname prag1-ospfd
password xxxxx
enable password xxxxx
log file /var/log/quagga/ospfd.log informational
log monitor warnings
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface eth3
 ip ospf cost 10
!
interface eth4
```

Cena spoje

```
interface eth4.5
 ip ospf cost 20
!
interface eth4.607
!
interface eth5
!
interface lo
```

Označení směrovače - identifikace

```
router ospf
 ospf router-id 80.251.240.1
 redistribute kernel route-map STATICKE_ROUTY
```

Na rozhraní eth0, eth1 a eth5 nebude navázána OSPF adjacence

```
 passive-interface eth0
 passive-interface eth1
 passive-interface eth5
```

OSPF bude aktivní na zbývajících rozhraních

```
network 80.251.241.16/30 area 0.0.0.0
network 80.251.241.172/30 area 0.0.0.5
network 80.251.241.224/29 area 0.0.0.9
network 80.251.242.224/29 area 0.0.0.0
network 80.251.247.64/28 area 0.0.0.4
 default-information originate always metric-type 1
!
access-list ADSL_CUST permit 80.251.255.128/25
access-list ADSL_CUST permit 80.251.252.0/25
!
route-map STATICKE_ROUTY permit 20
 match ip address CTC_RADIUS
!
line vty
!
```

4.3.7 Protokol BGP

Protokol BGP prošel několika etapami vývoje. V dnešní době je jedinou možnou použitelnou verze BGP 4, která je z roku 1994. Protokol BGP používá pro výměnu informací mezi směrovači protokol TCP a port 179, jedná se tedy o spolehlivou transportní službu. BGP je v současnosti jediný externí směrovací protokol v sítích IP. Protokol BGP je poměrně složitý a nezvládnutá konfigurace může mít za následek škody obrovského rozsahu[1]. O výpadku sítě Internet způsobeným vadnou konfigurací protokolu BGP na platformě RouterBoard informoval Zbyněk Pospíchal na serveru www.etron.cz, s jehož souhlasem si autor dovoluje na tomto místě článek ocitovat[1][12].

„*Malý český ISP způsobil světový kolaps*

Sobota, 21 Únor 2009 15:20

K čemu došlo (lidskými slovy): *jeden regionální český poskytovatel internetu špatně nakonfiguroval routing, což se stalo špatným napsáním jednoho čísla. To znamenalo, že jako optimální se do internetu propagovala nesmyslně dlouhá trasa a to počtem až 100 000 požadavků za vteřinu. To pro řadu starších routerů znamenalo něco jako přetečení bufferu, zařízení nebyla schopna odbavovat normální provoz a chyba se šířila dále. Chyba se*

[1] Poznámka autora: Hloubovou analýzu zpracoval Renesys na

http://www.renesys.com/blog/2009/02/the-flap-heard-around-the-worl.shtml

projevila v řadě regionů, prakticky ale ne v Česku. ISP chybu rychle odstranil a během hodiny oživly všechny postižené sítě. Jedno memento ale zůstalo: jak může malá chybička u regionálního poskytovatele internetu zkolabovat provoz na polovině internetu? Inu, může.

Malý ISP z jihovýchodní Moravy konfiguroval propoj ke svému druhému (záložnímu) poskytovateli tranzitní konektivity. Každá síť je v Internetu reprezentována svým číslem autonomního systému, což bývala jen dvoubajtová, dnes už to však může být i čtyřbajtová hodnota unikátní pro každou síť, kterou dále používá směrovací protokol BGP a to v zásadě jen ke dvěma věcem - k nalezení nejvýhodnější cesty a k zamezení vzniku směrovacích smyček. Celý princip funguje tak, že pro každý prefix (samostatně směrovaný blok IP adres) existuje ve směrovacích tabulkách samostatná položka, obsahující řetězec se seznamem autonomních systémů, přes které k danému prefixu vede cesta (AS-path). Nyní uvedu příklad, jak takové cesty mohou vypadat, vybírá se obvykle podle nejkratší AS-path (to nemusí být vždy pravidlem, lze router jistým množstvím aplikovaného násilí přesvědčit, že může použít i jinou cestu, ale to není pro další text tohoto článku podstatné):

```
 Number of BGP Routes matching display condition : 5
 Status codes: s suppressed, d damped, h history, * valid, > best, i
internal
 Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop        Metric LocPrf Weight Path
 *>  169.232.0.0/16     137.164.130.61  1      100    0      11164 2152
52 i
 *i  169.232.0.0/16     137.164.130.57  20     100    0      11164 2152
52 i
 *i  169.232.0.0/16     137.164.130.53  20     100    0      11164 2152
52 i
 *   169.232.0.0/16     213.248.98.93   48     70     0      1299 3356
2152 2152 52 i
 *   169.232.0.0/16     64.214.121.169  49     70     0      3549 209
2152 2152 52 i
      Last update to IP routing table: 5d13h42m4s, 1 path(s) installed:
```

V druhém sloupci zleva vidíme prefix, zcela vpravo pak AS-path, nejlepší vybraná cesta je označena znakem > zcela vlevo hned za hvězdičkou. V posledních dvou řádcích pak vidíme, že se nám číslo AS 2152 opakuje. Co to znamená? Jde o tzv. prepend, tedy umělou penalizaci (znevýhodnění) dané cesty. A právě o prepend jde v tomto příběhu především.

Onen ISP chtěl svůj druhý upstream právě takto znevýhodnit. To je celkem běžná záležitost, kterou vidíte i v předchozím příkladu. Problém však byl v tom, že jako u všeho, existuje nějaký limit pro délku AS-path a za ten se všeobecně považuje 255 položek. To není žádný zásadní limit, protože jen velmi málo cest v současném Internetu obsahuje více čísel

autonomních systémů než 6, a AS-path s více než 15 položkami je naprostou raritou (a ne, v tomto případě se neočekává, že by se na tento limit do budoucna naráželo jako na překážku dalšího rozvoje, protože obecný trend je, že se průměrná délka AS-path v celosvětové směrovací tabulce postupem času mírně zkracuje).

Co se tedy stalo? Ano, správně, dotyčný ISP svou cestu skutečně znevýhodnil, a to poměrně zásadním způsobem. Není zcela jasné, jakým konkrétním způsobem toho dosáhl, avšak provedl jsem vlastní šetření a na jeho základě zjistil, že použitou platformou, na které k uvedenému problému došlo, je pravděpodobně MikroTik RouterBoard, tedy zařízení primárně určené pro poněkud jiné nasazení než je ASBR (Autonomous System Border Router) a o implementaci BGP na této platformě se vyprávějí legendy (ano, základní věci tam fungují poměrně spolehlivě, vím). Jistý nejmenovaný konkurenční výrobce, jehož boxy jsou pro podobné nasazení přece jen o něco málo vhodnější, podporuje syntaxi typu "set as-path prepend last-as N", kde N je počet, kolikrát se poslední AS v cestě zopakuje a může nabývat hodnoty 1 - 10. Naprosto stačí, aby výrobce nějaké minoritní směrovací platformy kontrolu této hodnoty do svého zařízení nezadal, zmatená obsluha tam namísto počtu, kolikrát se má číslo AS zopakovat, omylem napíše číslo svého autonomního systému a problém je, pokud AS dotyčného ISP zrovna nemá nějaké prominentní nízké číslo, na světě.

Paradoxně problém nepostihl každého - v ČR nebyl tento problém téměř ani zaznamenán, v ČR dochází u operátorů k celkem pravidelným upgradům hardware i software a zapomenuté infrastruktury není mnoho. Podobná situace platí u velkých operátorů i jinde ve světě, avšak na regionálních a místních sítích v mnoha zemích světa, zejména tam, kde se používají poměrně staré řady operačních systémů pro směrovače, problém nastal. Kupříkladu starší verze Cisco IOS reagovaly tak, že po přijetí takové cesty rozpojily BGP relaci, po které taková cesta přišla. To není zásadní problém, relace se po chvíli znovu spojí, avšak pokud se hned zase rozpojí kvůli přijetí vadné AS-path, už to zásadní problém je. Jevu, kdy se nám relace stále dokola spojuje a rozpojuje, říkáme flap - ano, existují nástroje, jak se s ním vyrovnat, avšak pokud dotyčné síti neflapuje jeden upstream, ale všechny (což byla právě tato situace), nejsou nám tyto nástroje nic platné a nastává problém - dotyčný operátor je bez tranzitní konektivity.

Podrobnější analýzu, jako obvykle, udělal Renesys, disponující nástroji pro hloubkovou

analýzu stavu směrovacích tabulek. Uvedli také mapy nejhůře postižených zemí, významně postižena byla téměř celá západní Evropa (kde nejhůře dopadla Belgie a Španělsko), z nových členů EU pak chyba dopadla velmi tvrdě na Lotyšsko (paradoxně domovská země zařízení MikroTik RouterBoard) a do Pákistánu dorazila tvrdá odplata za únos Youtube, každopádně postiženo bylo i mnoho sítí v takových zemích, jako jsou USA, Čína nebo Egypt; mezi země, které byly naopak postiženy málo nebo téměř vůbec, patří kromě ČR ještě Maďarsko, Chorvatsko, Srbsko, Litva, Turecko, Indie, JAR, Chile nebo Argentina. Celý problém pak trval zhruba hodinu, než dotyčného ISP jeho poskytovatel záložní tranzitní konektivity dočasně odpojil.

Není to však první případ, kdy se něco podobného stalo, je však první, který měl až takhle tvrdé následky. Předchozí podobné případy způsobily ISP z různých zemí (BiH, Bulharsko, Indonesie, Polsko, USA), avšak délka jejich AS-path se hodnotě 255 vždy pouze blížila, nikdy jej však nepřekročila. To se podařilo až tento týden operátorovi z malého městečka nedaleko hranic se Slovenskem.“

Charakter protokolu BGP je v podobě vektoru cest, kdy se jedná o posloupnost autonomních systémů od zdroje k cíli. Neposílá celé směrovací tabulky, ale pouze dílčí změny, což představuje menší nároky na množství přenášených dat. Podporuje autentizaci směrovacích informací, CIDR a agregaci cest. Při připojení nového směrovače si BGP vymění se sousedy celé směrovací tabulky a následně BGP pouze aktualizuje změny ve směrovacích tabulkách. Důležité jsou i zprávy, které udržují a navazují vztah se sousedskými směrovači. Tyto zprávy typu OPEN a KEEPALIVE si musí obě strany vyměnit, než si začnou vyměňovat směrovací informace. Všechny typy zpráv popisuje následující tabulka[1].

Tabulka 8 – Typy zpráv BGP

Kód	Typ	Popis
1	OPEN	Zahájení komunikace
2	UPDATE	Inzerování nebo odstranění cest
3	NOTIFICATION	Odezva na nesprávnou zprávu
4	KEEPALIVE	Aktivní testování dostupnosti partnera

4.3.8 Implementace protokolu BGP

Podobně jako u protokolu OSPF je možné pro implementaci BGP použít směrovací softwarovou sadu Quagga. Konfiguraci démonu quagga provádíme obdobně zadáním příkazu *telnet localhost bgpd* z konkrétního BGP směrovače. Konfigurace je následně uložena v */etc/quagga/bgpd.conf*.

Ukázka konfiguračního souboru protokolu BGP:

```
!
! Zebra configuration saved from vty
!   2011/03/16 10:28:54
!
hostname prag1-dat-bgpd
password xxxxx
enable password xxxxx
log file /var/log/quagga/bgpd.log informational
```

Označení AS, ID BGP směrovače a sítě, dostupné za tímto směrovačem

```
router bgp 39235
 bgp router-id 80.251.240.1
 network 77.95.192.0/21
 network 80.251.240.0/20
```

Konfigurace jednoho souseda s AS 702

```
neighbor 62.40.67.149 remote-as 702
 neighbor 62.40.67.149 description UUnet
 neighbor 62.40.67.149 soft-reconfiguration inbound
 neighbor 62.40.67.149 route-map UUNET_OUT out
 neighbor 62.40.67.149 filter-list NIC-NEPUSTIT in
 neighbor 62.40.67.149 filter-list NIC-NEPUSTIT out
!
ip prefix-list NIX-OUT seq 10 permit 80.251.240.0/20
ip prefix-list NIX-OUT seq 15 permit 77.95.192.0/21
ip prefix-list NIX-OUT seq 20 deny any
ip prefix-list OUT seq 10 permit 80.251.240.0/20
ip prefix-list OUT seq 15 permit 77.95.192.0/21
ip prefix-list OUT seq 20 deny any
!
route-map UUNET_OUT permit 10
 match ip address prefix-list OUT
!
line vty
!
```

5 QUALITY OF SERVICE

Kvalita služby je pojem velmi často skloňovaný v tématech zabývajících se datovou komunikací a datovou komunikací v síti Internet obzvláště. Dochází k velkému rozvoji služeb, kdy jejich kvalita je do velké míry závislá na kvalitativních charakteristikách komunikace přes datovou síť. Pro úspěšné provozování tohoto typu služby je nutné, aby síť poskytovala a dokázala zajistit určitou kvalitu služby po celou dobu provozu. Kvalita je definována parametry, jejichž typ a hodnoty se mohou lišit dle konkrétní požadované služby. Rozdílný bude požadavek na kvalitu služby pro přenos hlasu, případně on line videa a prohlížení webových stránek či posílání e-mailových zpráv. Účelem QoS je tedy definovat maximální nebo minimální šířku pásma pro určitý typ dat, definovat provoz, který je prioritní před jiným typem provozu (resp. určit pořadí odbavování jednotlivých typů provozu). QoS má za úkol zajistit uživatelům dostatečnou šířku pásma, ztrátovost paketů, odezvu a další. QoS se používá zejména v bezdrátových sítích v kombinaci s FUP, protože přenosová kapacita těchto sítí je značně omezená.

5.1 Parametry QoS

Kvalita služby představuje kombinaci několika parametrů, které je nutné dodržet pro zachování požadované kvality služby. Za základní parametry pro udržení kvality služby můžeme určit následující dílčí síťové parametry:

- Ztrátovost paketů – packet loss - tímto parametrem definujeme, kolik procent paketů nedorazí od svého odesílatele k cílovému příjemci. Nejčastějším důvodem je přetížení sítě, ať už se jedná o přetížení přenosové trasy či zahlcení směrovače, přepínače nebo jiného aktivního prvku. Aplikace, které neprobíhají v reálném čase (WWW, FTP), používají transportní protokol TCP, který se do jisté míry dokáže se ztrátovostí paketů vyrovnat, ovšem aplikace pracující v reálném čase (VoIP) naopak používají nespolehlivý transportní protokol UDP, který nepracuje s mechanismem pro opětovné vyslání paketů, které nedorazily do svého cíle. V případě, že není použit mechanismus pro identifikaci vhodných paketů, může dojít ke zničení vhodných paketů s nejvyšší prioritou, například z důvodu přetečení fronty.

- Zpoždění – latency - je doba, kterou paket potřebuje k tomu, aby překonal vzdálenost v síti mezi odesílatelem a příjemcem daného paketu. Výsledné zpoždění je součtem dílčích zpoždění způsobených kódováním a serializací (příprava paketu pro přenos médiem), zpožděním při přenosu (závislé na přenosové vzdálenosti), zpožděním ve frontě na odbavení a zpožděním při přepínání v síti. První dvě uvedená dílčí zpoždění jsou statické hodnoty, další dvě se dynamicky mění. Celkové zpoždění má největší dopad na hlasovou komunikaci, kdy zpoždění větší než 150 ms vytváří velmi nepříznivé hovorové prostředí jak pro mluvčího, tak pro posluchače. Kvalita hlasu tím ovšem není narušena, pouze je komunikace nepříjemná.
- Kolísání zpoždění – jitter – je způsobeno zpožděním při serializaci paketů a rozdílech v délkách front v souvislosti se zahlcením sítě. Při komunikaci se předpokládá, že pakety dorazí od zdroje k cíli ve stejném pořadí, v jakém byly odeslány. Jitter může způsobit, že tento předpoklad bude porušený, a obzvláště při hlasové komunikaci je toto velký problém. Kolísání zpoždění se řeší pomocí vyrovnávací paměti přímo ve VoIP zařízeních, které dokážou tento problém eliminovat do odchylky ve zpoždění v hodnotě 20-50 ms.

5.2 Metody QoS

V dnešní době se v sítích používají tři základní typy mechanismů Quality of Service. Nejjednodušším je metoda Best-effort services (metoda největší snahy), kdy se prakticky žádný QoS neuplatňuje a metoda se snaží každý paket co nejrychleji doručit do cíle. Metoda Differentiated services (DiffServ) rozděluje pakety do kategorií, kdy kategorie je zaznamenána do hlavičky paketů a s paketem se pak zachází podle předdefinovaných parametrů. Třetí metodou je metoda Integrated services (IntServ), která řeší QoS tak, že pro daný datový přenos vyhradí pro aplikaci požadované zdroje v síti. Nevýhodou je nepřetržité signalizování.

6 FAIR USER POLICY

Fair User Policy (FUP) je opatření technického charakteru, které má do jisté míry zajistit všem uživatelům odpovídající a srovnatelnou kvalitu poskytované služby. Aplikuje se zejména v sítích, které mají poměrně malé přenosové pásmo, jako jsou sítě GSM, WI-FI sítě, ADLS připojení apod. Princip fungování FUP je v tom, že uživatelé sítě, kteří zatěžují síť nadměrně velkým provozem (např. velkým objemem stahovaných dat), jsou omezeni ve své prioritě (je jim snížena maximální přenosová rychlost, jejich pakety jsou upozaděny oproti ostatním uživatelům), a tím se uvolní kapacita přenosové sítě pro ostatní uživatele. Nejčastěji se FUP používá v případech, kdy je uživatelům nabízena sdílená služba, která předpokládá určitý způsob chování uživatele a FUP se aplikuje při porušení těchto zásad a pravidel.

6.1 Realizace FUP

Aby aplikace FUP byla smysluplná pro provozovatele sítě a nebyla odrazující pro uživatele sítě, je dobré nejprve provést analýzu chování uživatelů sítě v čase a tomu přizpůsobit chování FUP. Neexistuje obecné pravidlo nastavení FUP, a proto se nastavení FUP liší podle konkrétní sítě či provozovatele datové sítě. Obecně je možné tvrdit, že 60-70% z celkového provozu datové sítě vygeneruje 10% uživatelů. Provozovatel sítě v reálném čase (nebo alespoň v malých pravidelných intervalech) sleduje celkové zatížení sítě a chování jednotlivých uživatelů. Na základě chování konkrétního uživatele aplikuje pravidla FUP v několika možných směrech:

- Omezení efektivní přenosové rychlosti při překročení určeného množství dat za určité, předem definované období. Možností nastavení tohoto pravidla je možné vymyslet téměř neomezené množství, např. v závislosti za sledované časové období. Technicky je realizováno omezením rychlosti na přenosové infrastruktuře.
- Deprioritizace celkového datového toku – pakety uživatelů, kteří akceptují a dodržují pravidla stanovené metodikou FUP, jsou přednostně odbavovány.
- Deprioritizace specifického datového toku – provede se identifikace datového toku uživatele nebo skupiny uživatelů a určitá část (specifická aplikace či druh provozu) je znevýhodněna oproti ostatním uživatelům či jinému typu aplikace. Jedná se o poměrně náročný a sofistikovaný způsob řešení FUP, kdy data je nejprve nutné

klasifikovat a pak jsou na ně aplikována definovaná pravidla (typicky peer-to-peer sítě dostávají největší omezení).

- Platba za nadlimitně přenesená data – tato technika nevyžaduje složitější technické řešení (stačí pouze počítat veškerý datový tok každého uživatele), ale v dnešní době se již téměř nepoužívá nebo jen minimálně.

6.2 Aplikace FUP

Valná část poskytovatelů internetového připojení a provozovatelů datových sítí tvrdí, že žádná pravidla pro FUP neaplikuje. Základní znalost přenosových technologií ovšem toto tvrzení vyvrací. Například bezdrátové sítě (oblíbené WI-FI). Nejvíce používané pásmo 5,4 až 5,7GHz umožňuje využívat celkem 11 kanálů, kde každý má šířku 20 MHz. Dle přenosových podmínek, míry rušení atd., můžeme v průměru určit přenosovou kapacitu jednoho přístupového bodu na cca 20-25 Mbps. Při pohledu do ceníků ISP se běžně nabízí linky s přenosovou rychlostí kolem 10 Mbps, což znamená, že po připojení dvou až tří klientů je kapacita přístupového bodu vyčerpána. Následně musí provozovatel aplikovat agregaci či nějakou jinou formu FUPu. Nejčastějším a nejjednodušším způsobem řízení paketového provozu na síti je pakety, které se do definované fronty nevlezou, zahodit. Podrobněji je toto popsáno např. zde [19]. Jiná situace je v sítích mobilních operátorů. Tady operátoři vesměs přiznávají různá pravidla aplikace FUP, ale je nutné zmínit, že mobilní sítě nejsou primárně určeny k přenosu dat ve velkém objemu. Každý z provozovatelů mobilní sítě se k FUP staví různě, nicméně všichni přiznávají aplikaci pravidel FUP na datové přenosy ve své síti (ať už u všech datových mobilních tarifů, či pouze u některých).

II. PRAKTICKÁ ČÁST

7 PŘÍPADOVÁ STUDIE I - HOTSPOT

První případová studie popisuje konfiguraci RouterBoard jako veřejného přístupového bodu. Funkci Hotspot je možné konfigurovat na libovolné rozhraní (jak bezdrátové, tak metalické), kdy pak takto nakonfigurovaný RouterBoard může sloužit jako hotspot brána pro lokální síť.

Ve všech případových studiích je uvedena a popsána konfigurace jak pomocí GUI Winbox, tak pomocí CLI (například při použití protokolu SSH).

7.1 Podmínky pro konfiguraci

Veřejný přístupový bod se provozuje zpravidla na bezdrátovém rozhraní. Toto musí pracovat v módu AP (access point) a tedy nutnou podmínkou je použití minimálně RouterOS Level 4. Konfigurace je poměrně jednoduchá, prováděna intuitivně průvodcem za předpokladu výchozí konfigurace, která může být například přístupový bod WI-FI sítě, připojený metalickým rozhraním k páteřní přenosové trase, s nastaveným DHCP serverem na rádiovém rozhraní a s překladem adres mezi vnitřní sítí (rozsah adres přidělený na bezdrátovém rozhraní) a WAN sítí (IP adresa přidělena na metalickém ethernetovém rozhraní – kompletní IP konfigurace včetně masky a výchozí brány).

7.2 Postup a návrh řešení

Úkolem bylo vytvořit funkční konfiguraci brány Hotspot pro lokální síť LAN, kdy brána bude jedním metalickým portem připojena do WAN sítě (připojení do Internetu) a na druhém metalickém portu bude provedena konfigurace Hostpot brány. Z toho je patrné, že požadavek na hardware je poměrně malý a můžeme volit libovolný typ RouterBoardu, který disponuje alespoň dvěma metalickými porty (tedy RB 433 a další). Dle přenosových rychlostí připojených sítí pak volíme variantu s porty o rychlosti 100Mbps nebo 1000Mbps. Pro konfiguraci je nutnost instalace a aktivace balíčku *hotspot* v menu System/Packages.

Příkaz CLI pro aktivaci balíčku hotspot:

[admin@Hotspot] > *system package enable hotspot*

[admin@Hotspot] > *system reboot*

7.3 Konfigurace

Jak již bylo výše uvedeno, je konfigurace brány Hotspot, po provedené předchozí konfiguraci základních parametrů, jednoduchá. Pomocí následujících příkazů CLI lze definovat základní parametry, ze kterých se bude vycházet pro konfiguraci brány Hotspot.

Příkazy CLI pro základní nastavení RouterBoardu:

[admin@Hotspot] > *system identity set name=Hotspot*

[admin@Hotspot] > *interface ethernet set ether1 name=ether1-WAN*

[admin@Hotspot] > *interface ethernet set ether2 name=ether2-HotSpot*

[admin@Hotspot] > *ip address add address=192.168.1.151/24 interface=ether1-WAN*

[admin@Hotspot] > *ip address add address=10.0.0.1/24 interface=ether2-HotSpot*

[admin@Hotspot] > *ip route add gateway=192.168.1.100*

[admin@Hotspot] > *ip firewall nat add chain=srcnat src-address=10.0.0.0/24 action=src-nat to-addresses=192.168.1.151*

[admin@Hotspot] > *ip dns set primary-dns=192.168.1.100 allow-remote-requests=yes*

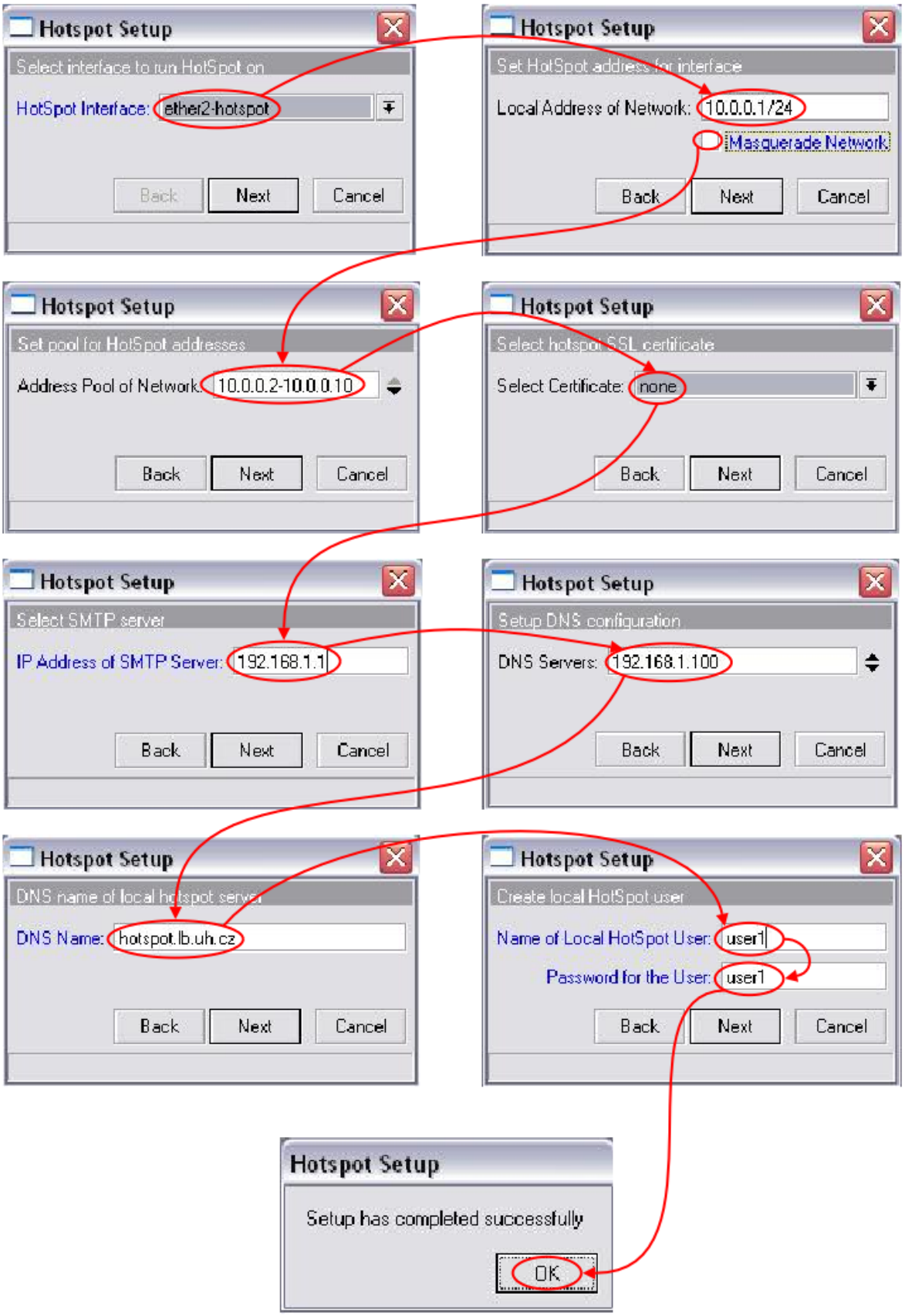
[admin@Hotspot] > *ip pool add name=dhcp_pooll ranges=10.0.0.2-10.0.0.200*

[admin@Hotspot] > *ip dhcp-server add name=dhcp1 interface=ether2-HotSpot address-pool=dhcp_pooll*

[admin@Hotspot] > *ip dhcp-server enable dhcp1*

[admin@Hotspot] > *ip dhcp-server network add address=10.0.0.0/24 gateway=10.0.0.1 dns-server=192.168.1.151*

Po konfiguraci základního nastavení je možné spustit průvodce konfigurací. V menu IP/Hotspot je spuštěn průvodce konfigurací (tlačítko Hotspot Setup). Průvodce konfigurací brány Hotspot popisuje následující obrázek.



Obrázek 12 – Konfigurace Hotspot pomocí průvodce

V prvním kroku se definuje rozhraní, na kterém má běžet brána Hotspot a následně IP adresa pro toto rozhraní. Pokud je již vytvořen překlad adres pravidlem typu src-nat, není již potřeba aktivovat volbu Masqurade Network. V dalším kroku je definován rozsah adres, které budou přidělovány DHCP serverem na rozhraní pro Hotspot. V komunikaci prostřednictvím brány Hotspot je možné použít certifikát SSL, který umožňuje zabezpečit komunikaci mezi klientem připojeným k Hotspot bráně a bránou. Dále se definuje IP adresa SMTP serveru, který budou užívat uživatelé k emailové komunikaci. Následně je definována adresa DNS serveru a DNS jméno pro bránu Hotspot. V posledním kroku se vytvoří uživatelské jméno a heslo pro přihlášení a tím je provedena autentizace uživatele a umožněn mu provoz.

Konfiguraci brány Hotspot pomocí příkazů pro CLI:

[admin@ Hotspot] > *ip hotspot setup ;*

Select interface to run HotSpot on

hotspot interface: *ether2-hotspot*

Set HotSpot address for interface

local address of network: *10.0.0.1/24*

masquerade network: *no*

Set pool for HotSpot addresses

address pool of network: *10.0.0.2-10.0.0.10*

Select hotspot SSL certificate

select certificate: *none*

Select SMTP server

ip address of smtp server: *192.168.1.1*

Setup DNS configuration

dns servers: *192.168.1.100*

DNS name of local hotspot server

dns name: *hotspot.lb.uh.cz*

Create local hotspot user

name of local hotspot user: *user1*

password for the user: *user*

Po provedení konfigurace je vygenerována řada pravidel v záložce Firewall, která zajišťují správnou funkci brány Hotspot, jako jsou přesměrování při autentizaci, přesměrování, pokud autentizace neproběhne správně atd.

7.3.1 Úprava přihlašovací stránky

Stránku, prostřednictvím které je prováděna autentizace uživatele, je možné upravit dle potřeby.

Obrázek 13 - Brána Hotspot

Po konfiguraci brány je přihlašovací stránka login.html uložena v adresáři Hotspot a je dostupná pomocí funkce drag and drop v menu Files v GUI Winbox nebo pomocí protokolu FTP. Přihlašovací stránku je nutné po úpravě umístit na původní místo se stejným jménem.

7.3.2 Další nastavení brány Hotspot

Brána Hotspot disponuje množstvím dalších parametrů, které lze upravovat a tím měnit chování brány, jako jsou:

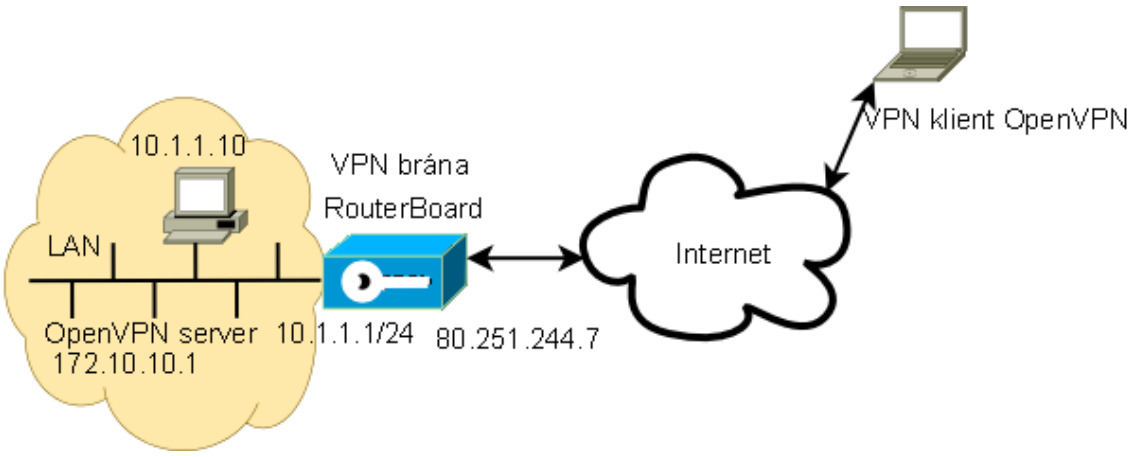
- Omezení maximální přenosové rychlosti pro celou bránu nebo pro daného uživatele.
- Autentizace uživatelů oproti radius serveru.
- Volně dostupné stránky bez přihlášení (walled garden).
- Maximální počet aktivních uživatelů pod jedním uživatelským jménem.
- Čas, po kterém je nutné se znovu autentizovat vůči bráně.
- Omezení pro uživatele dle množství přenesených dat a další.

8 PŘÍPADOVÁ STUDIE II - VIRTUÁLNÍ PRIVÁTNÍ SÍŤ

Platforma RouterBoard s MikrotikRouterOS podporuje realizaci VPN sítě na bázi IPsec a je realizována prostřednictvím balíčku *security*. RouterBoard je možné použít jako VPN bránu pro připojení klientů do LAN (VPN typu remote access), ale i jako VPN bránu pro propojení lokálních sítí (VPN typu LAN-to-LAN). Rozdíl mezi oběma typy VPN jsou popsány v kapitole 3.3 této práce. Na platformě MikrotikRouterOS je k dispozici protokol AH (Authentication Header), který zajišťuje autentizaci obsahu datagramu (ověřuje integritu zprávy) a pro zajištění šifrování přenášených dat je použit protokol ESP (Encapsulating Security Payload), který podporuje i vlastní autentizační systém.

8.1 VPN typu remote access

Pro realizaci virtuální privátní sítě typu remote access je použito brány VPN RouterBoard a jako klienta je možné využít některého z dostupných softwarů pro OS Windows. Platforma RouterBoard nepodporuje VPN typu remote access na IPsec, jako je tomu například u technologie Cisco, kdy je možné použití klienta například od společnosti SHREW Soft, který je dostupný na stránkách společnosti http://www.shrew.net/download a jehož konfigurace není složitá [13]. RouterBoard podporuje nativní Windows L2TP over IPsec klienty. Platforma RouterBoard pro realizaci VPN brány podporuje technologii OpenVPN [14], jak popisuje tato případová studie. Topologie VPN typu remote access (vzdálený přístup) je patrná z následujícího obrázku.



Obrázek 14 – VPN typu vzdálený přístup

Pro realizaci VPN brány je nutné aktivovat balíček *ppp*. Výchozí konfigurace VPN brány je definována pomocí příkazů CLI:

[libor@OpenVPN] > *system package enable ppp*

[libor@OpenVPN] > *ip address add address=10.1.1.1/24 interface=ether2 disabled=no*

[libor@OpenVPN] > *ip address add address=80.251.244.7/25 interface=ether1 disabled=no*

[libor@OpenVPN] > *ip route add dst-address=0.0.0.0/0 gateway=80.251.244.126*

[libor@OpenVPN] > *ip firewall nat add chain=srcnat src-address=10.1.1.0/24 action=src-nat to-addresses=80.251.244.7 comment="Preklad adres – NAT"*

Po provedení základní konfigurace je možné přistoupit ke konfiguraci OpenVPN. OpenVPN pracuje s certifikáty SSL, které je nutné předem připravit k dalšímu použití při realizaci VPN na RouterBoardu. Je třeba vytvořit certifikát certifikační autority, certifikát pro server a k tomu patřičný serverový klíč. Pro tuto aplikaci již certifikační autorita vydala certifikát, a tudíž v práci popis vytvoření certifikátu není obsažen a toto téma je mimo rozsah této práce[2]. Všechny tři je nutné pomocí protokolu FTP nebo funkce drag and drop přesunout do *Files*. V menu System/Certficates je nutné certifikáty i klíč importovat v daném pořadí: certifikát certifikační autority, certifikát serveru a jako poslední klíč k serveru. Úspěšně provedený import certifikátu a klíče popisuje následující obrázek.

Obrázek 15 - Importované certifikáty

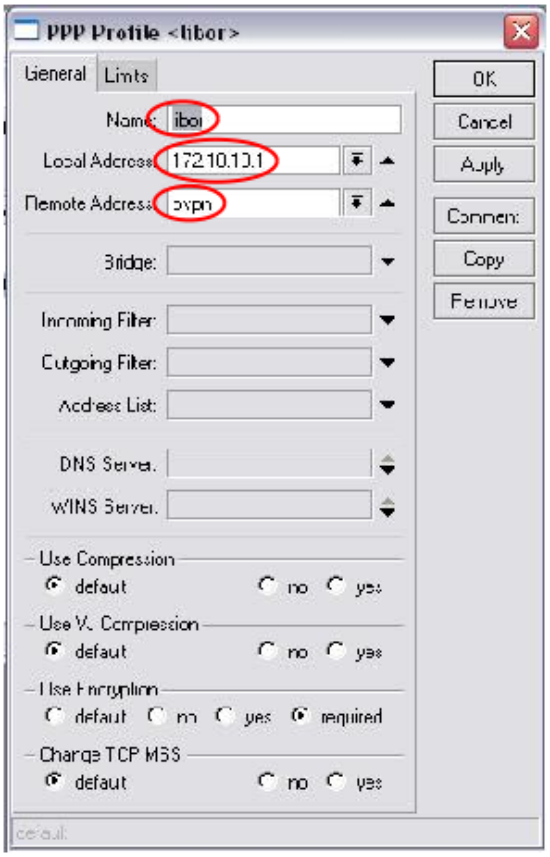
Jako první se definuje rozsah IP adres, které budou přidělovány klientům VPN. Rozsah adres se definuje v záložce IP/Pool.

[2] Pro přehlednost práce uvádí autor jeden z možných odkazů na podrobný popis jak vygenerovat vlastní certifikát: http://www.root.cz/clanky/jak-na-openssl/

Příkaz CLI pro konfiguraci rozsahu adres pro OpenVPN:

[libor@OpenVPN] > *ip pool add name=ovpn ranges=172.10.10.100-172.10.10.200*

PPP profil v záložce Profiles je používán k definování výchozí hodnoty pro přístup k evidenci uživatelů v záložce Secrets. Profil definuje lokální IP adresu, rozsah adres pro klienty OpenVPN a další parametry (např. zda bude provoz šifrovaný či nikoli). Konfigurace profilu je popsána následujícím obrázkem.



Obrázek 16 – PPP profil uživatelů

Příkaz CLI:

[libor@OpenVPN] > *ppp profile add name=libor local-address=172.10.10.1 remote-address=ovpn use-encryption=required*

Záložka Secret představuje databázi uživatelů pro přístup k PPP s profilem uživatele, přidělený každému uživateli. Definuje se jméno uživatele pro autentizaci, jeho heslo, profil, který je přiřazen pro konkrétního daného uživatele, a službu, kterou může definovaný uživatel používat.

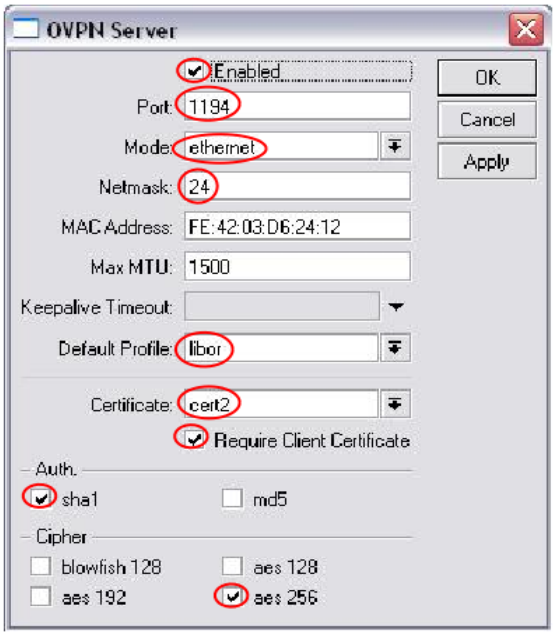
Příkaz CLI:

[libor@OpenVPN] > *ppp secret add name=libor password=blaha profile=libor service=ovpn disabled=no caller-id=“*“

Konfigurace uživatelů je patrná z následujícího obrázku.

Obrázek 17 – Konfigurace uživatele

V posledním kroku je definován OpenVPN server. V záložce Interface je pod tlačítkem OVPN Server dostupná konfigurace VPN Serveru. Zaškrtávací pole Enabled aktivuje funkci OVPN Serveru a v poli port je definováno číslo portu pro server. V poli mode je zvolena volba Ethernet a do pole Netmask je doplněna hodnota síťové masky pro rozsah IP adres, které přiděluje server klientům OpenVPN. Dále je nutné definovat výchozí profil a serverový certifikát (importovaný jako druhý v pořadí). Na závěr se definuje algoritmus pro autentizaci a šifrovací klíč.



Obrázek 18 – OpenVPN server

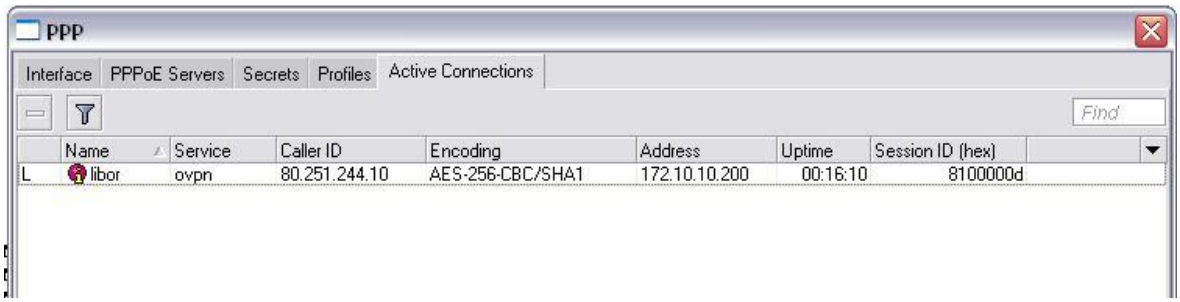
Klientská konfigurace je uložena v podadresáři config v souboru s příponou *.ovpn. Níže uvedený výpis konfiguračního souboru client_config.ovpn koresponduje s výše popsanou konfigurací serveru:

Výpis souboru client_config.ovpn:

```
tls-client
dev tap
proto tcp-client
resolv-retry infinite
nobind
ca cacert_lohcek.pem
verb 1
auth-nocache
remote 80.251.244.7 1194
cert Libor.Blaha.crt
key Libor.Blaha.key
port 1194
ping-timer-rem
persist-tun
persist-key
cipher AES-256-CBC
auth SHA1
pull
auth-user-pass
route-up "route add 10.1.1.0 mask 255.255.255.0 172.10.10.1"
```

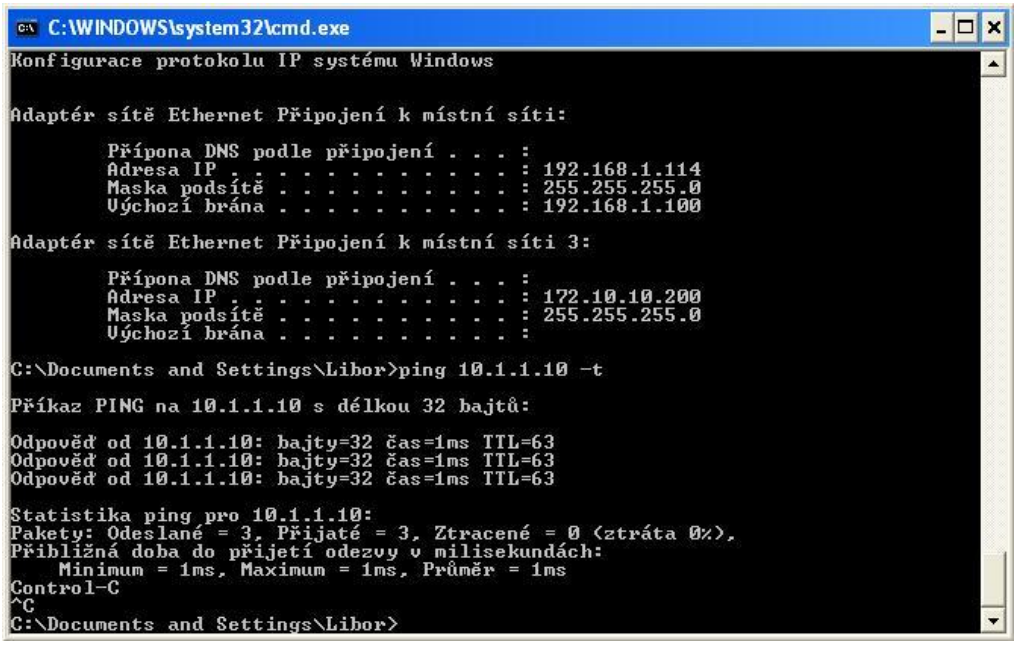
Do adresáře, kde je instalován OpenVPN klient, je nutné nahrát certifikát certifikační autority (shodný s prvním certifikátem importovaným do RouterBoardu) a dále klientský certifikát a klientský klíč (např. Libor.Blaha.crt a Libor.Blaha.key). Po úspěšné konfiguraci

OpenVPN je v záložce Active Connections seznam připojených uživatelů, jak popisuje následující obrázek.



Obrázek 19 – Aktivní uživatelé OpenVPN

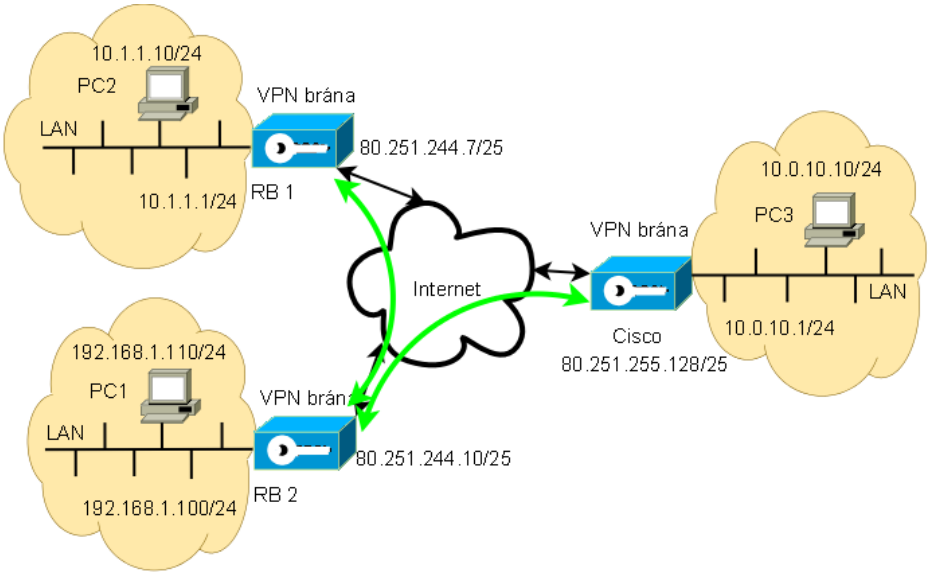
Ověření úspěšného navázání VPN spojení je na klientské straně signalizováno ikonou OpenVPN klienta, která je v barvě zelené (při odpojení je ikona barvy červené a během navazování komunikace je barvy žluté). Z klientské stanice ověříme dostupnost stanic ve vnitřní síti (za bránou VPN), např. příkazem ping.



Obrázek 20 – Ověření VPN spojení

8.2 VPN typu LAN-to-LAN

V případové studii je popsána konfigurace virtuální privátní sítě typu LAN-to-LAN za použití RouterBoard jako brány do lokální sítě LAN na levé straně a brány Cisco na straně pravé, jak je patrné z následujícího obrázku.



Obrázek 21 – VPN typu LAN-to-LAN

VPN spojení bylo navázáno mezi zařízeními RB 1 a RB 2 a mezi zařízeními RB 2 a Cisco 800. Konfigurace VPN brány Cisco:

```
Current configuration : 2863 bytes
!
! Last configuration change at 07:00:16 UTC Tue Apr 5 2011 by dat
! NVRAM config last updated at 07:02:08 UTC Tue Apr 5 2011 by dat
!
version 12.4
no service pad
service timestamps debug uptime
service timestamps log uptime
service password-encryption
!
boot-start-marker
boot-end-marker
!
memory-size iomem 5
aaa new-model
!
aaa session-id common
!
resource policy
!
ip cef
!
crypto isakmp policy 1
  encr 3des
  hash md5
  authentication pre-share
  group 2
crypto isakmp key sp1ssk4 address 80.251.244.10 no-xauth
!
!
crypto ipsec transform-set RB esp-3des esp-md5-hmac
!
crypto map VPN 10 ipsec-isakmp
  set peer 80.251.244.10
  set transform-set RB
```

```
  set pfs group2
  match address 100
!
interface Ethernet0
  ip address 10.0.10.1 255.255.255.0
  ip nat inside
  ip virtual-reassembly
  ip tcp adjust-mss 1412
  ip policy route-map clear-df
  no ip mroute-cache
!
interface FastEthernet1
  duplex auto
  speed auto
!
interface Dialer1
  ip address negotiated
  ip mtu 1452
  ip nat outside
  ip virtual-reassembly
  encapsulation ppp
  dialer pool 1
  dialer-group 1
  ppp authentication pap callin
  ppp pap sent-username ddat password 7 030A5A1D071D
  ppp ipcp dns request
  crypto map VPN
!
ip route 0.0.0.0 0.0.0.0 Dialer1
!
ip nat inside source list NAT interface Dialer1 overload
!
control-plane
!
line vty 0 4
  access-class 25 in
  exec-timeout 120 0
  length 0
  transport preferred ssh
  transport input ssh
  transport output ssh
!
scheduler max-task-time 5000
!
end
```

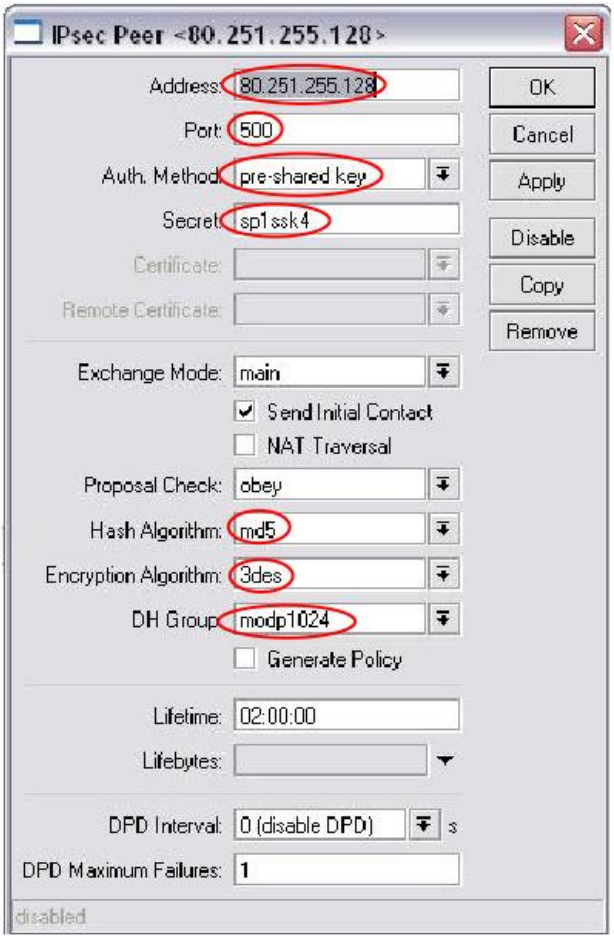
Výchozí konfigurace brány označené RB2 byla provedena příkazy pro CLI. Bylo nutné přidělit IP adresu jednotlivým rozhraním, definovat výchozí bránu a nastavit překlad adres (NAT):

[liborll@PS-1] > *ip address add interface=ether1-wan address=80.251.244.11/25 disabled=no*

[liborll@PS-1] > *ip route add disabled=no distance=1 dst-address=0.0.0.0/0 gateway=80.251.244.126*

[liborll@PS-1] > *ip firewall nat add action=src-nat chain=srcnat disabled=no src-address=192.168.1.0/24 to-addresses=80.251.244.10 comment="Preklad adres – NAT"*

Konfigurace brány VPN se provádí v menu IP/Ipsec, kde je nejprve definována protistrana VPN tunelu. Dále je nutné definovat IP adresu a port druhé strany, předsdílený klíč, na základě kterého je provedena autentizace. Další možnou variantou řešení autentizace je využití RSA signatury s použitím dvojice RSA certifikátů. V poli Secret je pak samotný klíč. Dále se definuje hashovací algoritmus (SHA – Secure Hash Algoritmus je silnější, ale pomalejší než MD5). Důležité je definovat šifrovací algoritmus a DH (Diffie-Hellman) skupinu, která následně určuje z počátečního sdíleného klíče hodnoty neveřejných klíčů vyměňovaných během navázaného spojení (DH skupina určuje tzv. parametr, který je následně použit pro vytvoření šifrovaného kanálu). Pole DPD Maximum Failures udává maximální možný počet chyb za definovaný čas v poli DPD Interval, kdy při překročení tohoto počtu je protistrana označena jako nedostupná. Konfiguraci parametrů protistrany popisuje následující obrázek.

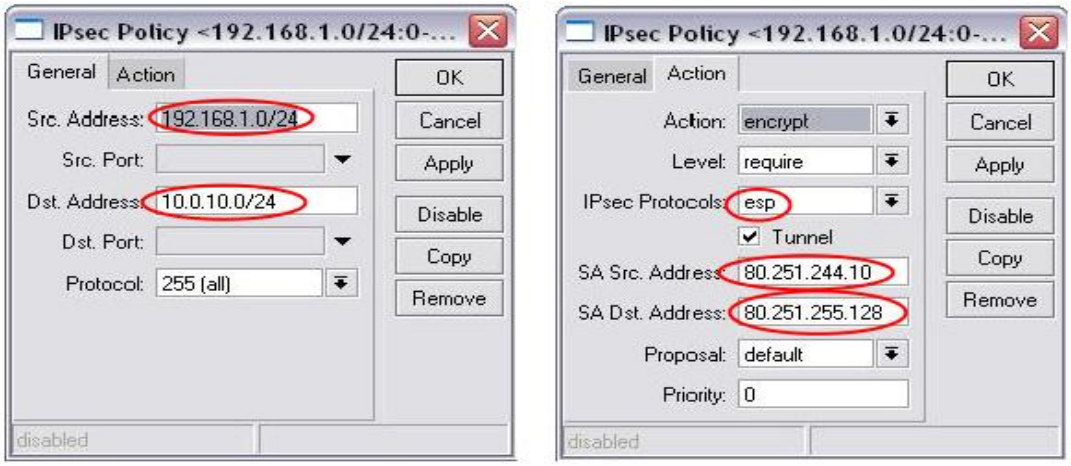


Obrázek 22 – IPsec Peer

Příkaz CLI:

[liborll@PS-1] > *ip ipsec peer add address=80.251.255.128/32:500 auth-method=pre-shared-key dh-group=modp1024 disabled=no dpd-interval=disable-dpd dpd-maximum-failures=1 enc-algorithm=3des exchange-mode=main generate-policy=no hash-algorithm=md5 lifebytes=0 lifetime=2h nat-traversal=no proposal-check=obey secret=sp1ssk4 send-initial-contact=yes*

V první záložce Policies se definují politiky nastavení zabezpečení, které se budou aplikovat na pakety VPN. V záložce Action v poli Action se definuje, co se stane s paketem, který je uzavřený v politice (zašifruje, nezmění, zahodí). Definují se rozsahy zdrojových a cílových adres (záložka General), dále šifrovací protokol tunelu, zda bude použito tunelového módu a zdrojová a cílová adresa SA. Pro každý VPN tunel je nutné definovat vlastní záznam. Nastavení politiky popisuje následující obrázek.



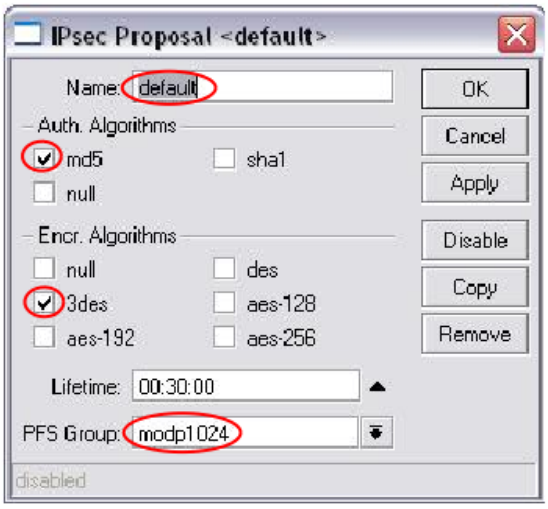
Obrázek 23 – IPsec politiky

Příkaz CLI:

[liborll@PS-1] > *ip ipsec policy add action=encrypt disabled=no dst-address=10.0.10.0/24:any ipsec-protocols=esp level=require priority=0 proposal=default protocol=all sa-dst-address=80.251.255.128 sa-src-address=80.251.244.10 src-address=192.168.1.0/24:any tunnel=yes*

V záložce Proposals je definováno, jakým způsobem bude provedena autentizace a jakým způsobem, resp. jaký algoritmus, bude použit pro šifrování dat v kanálu VPN. K dispozici pro autentizaci je algoritmus MD5 nebo SHA1, případně není autentizace řešena vůbec.

Šifrovací algoritmus je možné volit mezi DES, 3DES, AES-128, AES-192, AES-256. Volby autentizace a šifrování popisuje následující obrázek.



Obrázek 24 – IP sec Proposal

Příkaz CLI:

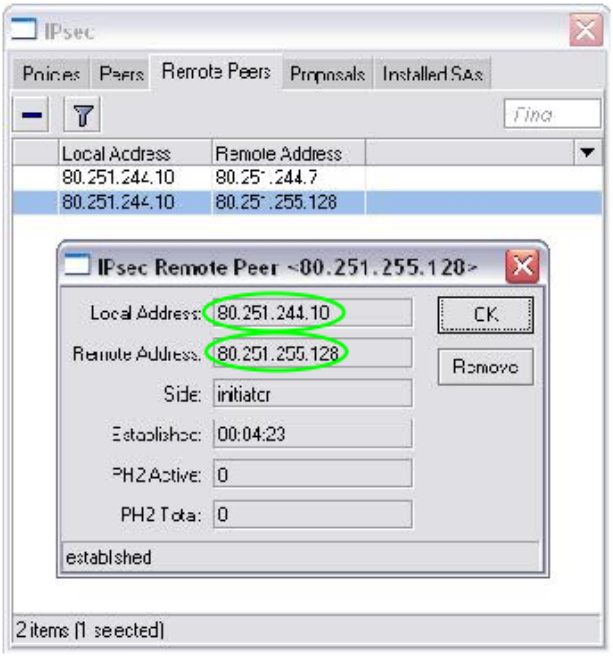
[liborll@PS-1] > *ip ipsec proposal add auth-algorithms=md5 enc-algorithms=3des name=default lifetime=30 pfs-group=modp1024*

V posledním kroku konfigurace VPN je nutné dodefinovat pravidlo v sekci IP/Firewall/NAT, které zajistí správné směrování paketů do VPN a ne mimo VPN, kam je směrován všechen ostatní provoz. V tomto pravidle je definován rozsah zdrojových adres, rozsah cílových adres a typ akce, která se s těmito pakety provede.

Příkaz CLI:

[liborll@PS-1] >*ip firewall nat add chain=srcnat src-address=192.168.1.0/24 dst-address=10.0.10.0/24 action=accept*

Po provedení této konfigurace a po vygenerování datového požadavku do VPN (např. ping na zařízení v síti za druhou VPN branou) dojde k navázání VPN tunelu během cca 2 až 3 sekund (testováno oproti Ciscu), což je možné ověřit v záložce Remote Peers, kde jsou IP adresy obou VPN bran a jsou k dispozici další informace o navázaném VPN spojení (např. doba trvání VPN spojení a další).



Obrázek 25 – IPsec, ověření funkčnosti

IPsec definuje tzv. Security Association (SA). Pro každý směr komunikace existuje samostatná SA, která definuje, jaký SPI (Security Parametr Index) je danému toku přiřazen, který klíč a algoritmus se použije pro šifrování. Po navázání VPN spojení jsou tyto asociace dostupné v záložce Installed SAs. Ověření funkčnosti VPN spojení navázané v síti uvedené na obrázku 21 je možné provést např. příkazem ping z PC1 na IP adresu PC3.

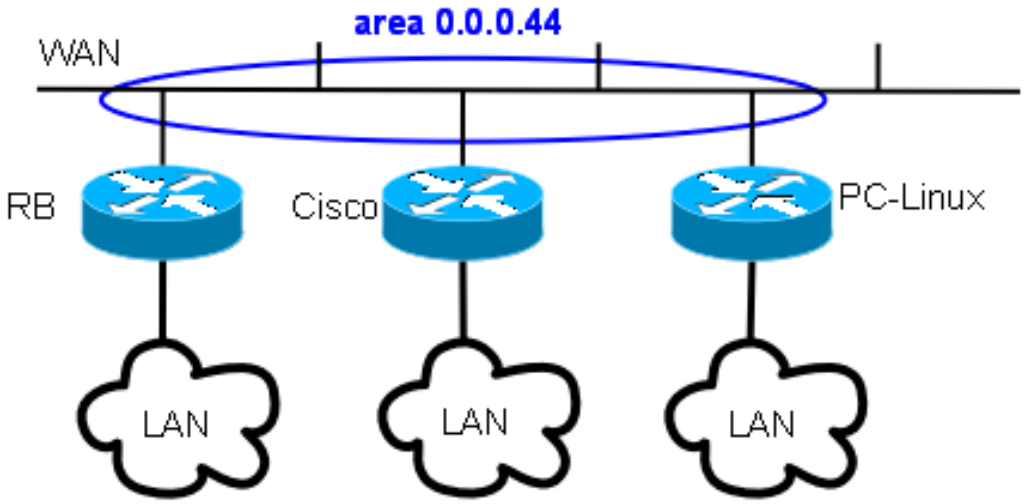
Obrázek 26 – Ověření VPN spojení

8.3 Vazba na IPv6

Vzhledem k tomu, že IPsec je integrální součástí IPv6 standardu, jsou všechny bezpečnostní mechanizmy popsané pro IPv4 dostupné i pro IPv6. U mechanizmů, které zajišťují bezpečnost na transportní (SSL/TLS, OpenVPN) nebo vyšší vrstvě, nemá použitý protokol třetí vrstvy vliv.

9 PŘÍPADOVÁ STUDIE III - SMĚROVAČ

Tato případová studie popisuje konfigurace RouterBoardu jako směrovače s dynamickým směrovacím protokolem OSPF. RouterBoard je spolu s dalšími dvěma směrovači (jeden na platformě Linux, druhým je Cisco) zapojen do sítě a je mezi nimi vytvořena jedna OSPF oblast (area), jak popisuje následující obrázek.



Obrázek 27 – Topologie sítě OSPF

9.1 Výchozí konfigurace směrovačů

Pro konfiguraci dynamického směrování je nutné provést základní IP konfiguraci jednotlivých rozhraní směrovačů, kterými má být směrovač připojen do sítě.

RouterBoard:

IP adresa: 80.251.244.7/25

Příkaz CLI pro konfiguraci výše uvedeného parametru:

ip address add address=80.251.244.7/24 interface=ether1

PC:

IP adresa: 80.251.244.9/25

Příkaz CLI pro konfiguraci výše uvedeného parametru:

ip addr add 80.251.244.9/25 dev eth0

Cisco:

IP adresa: 80.251.244.8/25

Příkaz CLI pro konfiguraci výše uvedeného parametru:

interface Ethernet 0/0

ip address 80.251.244.8 255.255.255.128

Po provedení této základní konfigurace je možné přistoupit ke konfiguraci dynamického směrování. Pro konfiguraci OSPF bylo použito Mikrotik RouterOS ve verzi 4.17, software Quagga v. 0.98.3 a Cisco 2610 Version 11.3(10)T.

9.2 Konfigurace parametrů OSPF

V následujících dvou odstavcích je popsáno nastavení OSPF pro software quagga na platformě Linux a konfigurace OSPF na směrovači Cisco pro výše uvedený případ. Oblast (area) je označena 0.0.0.44, je použita jednoduchá autentizace mezi sousedy a klíč pro autentizaci je 12345. OSPF konfigurace na linuxovém směrovači je nakonfigurována na rozhraní eth0 a na Cisco směrovači na rozhraní Ethernet0/0.

9.2.1 Konfigurace quagga na Linuxu

Výpis konfigurační souboru `/etc/quagga/ospfd.conf`.

```
! Zebra configuration saved from vty
!   2011/03/31 16:13:25
hostname rt-lb-ospfd
password 1234
enable password 1234
log file /var/log/quagga/ospfd.log informational
log monitor warnings
interface dummy0
interface eth0
 ip ospf authentication-key 12345
interface eth1
interface lo
router ospf
 ospf router-id 80.251.244.9
 compatible rfc1583
 network 80.251.244.0/25 area 0.0.0.44
```

```
 area 0.0.0.44 authentication
line vty
 exec-timeout 0 0
```

9.2.2 Konfigurace OSPF na směrovači Cisco

Výpis konfigurace na směrovači Cisco příkazem `show running-config`:

```
Current configuration:
version 11.3
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
hostname cisco-lb
process-max-time 200
interface Ethernet0/0
 ip address 80.251.244.8 255.255.255.128
 ip ospf authentication-key 12345
interface Serial0/0
 no ip address
 shutdown
router ospf 100
 network 80.251.244.8 0.0.0.0 area 0.0.0.44
 area 0.0.0.44 authentication
ip classless
ip route 0.0.0.0 0.0.0.0 80.251.244.126
line con 0
line aux 0
line vty 0 4
 login
no scheduler allocate
end
```

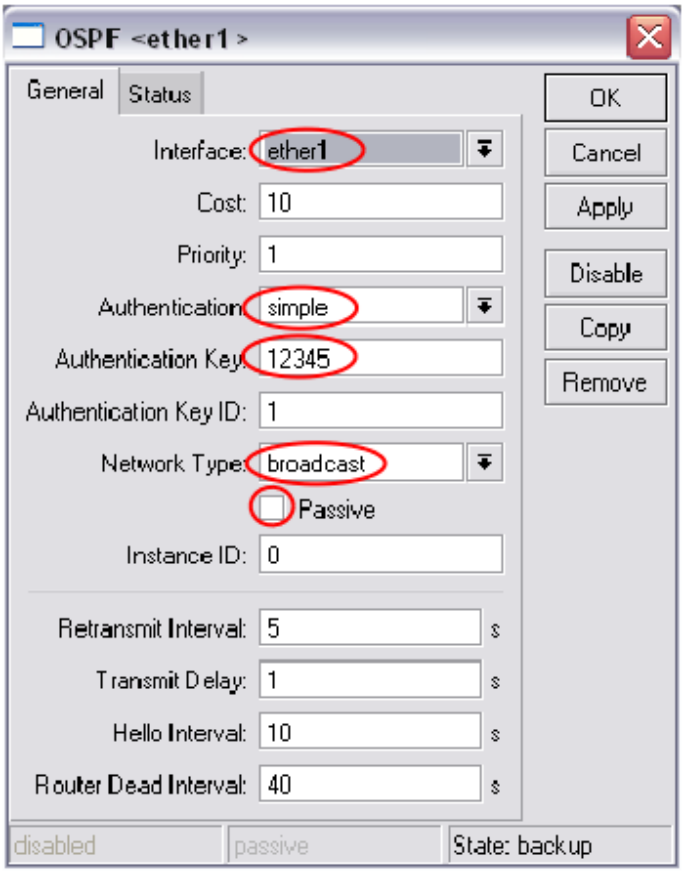
9.2.3 Konfigurace OSPF na RouterBoardu

Pro funkci dynamického směrování na platformě RouterBoard je nutné instalovat a aktivovat balíček *routing*. V menu Routing/OSPF v záložce Interface je nutné definovat rozhraní, na kterém bude probíhat komunikace v rámci OSPF. Zde se definují parametry, např. cena cesty (cost), typ autentizace, autentizační klíč, typ sítě, zda se jedná o interface passive a zejména časové parametry (jak často se rozesílají Hello pakety a další).

Konfigurace pomocí CLI:

[libor@OSPF] > *routing ospf interface add interface=ether1 cost=10 authentication=simple authentication-key=12345 network-type=broadcast passive=no*

Konfigurace rozhraní pomocí GUI Winbox popisuje následující obrázek.



Obrázek 28 – Konfigurace rozhraní OSPF

V záložce Interface se definuje, které rozhraní bude sloužit pro komunikaci se sousedy v OSPF síti. Cost určuje cenu linky, která je přes toto rozhraní připojena k sousednímu směrovači. V poli Authentication se zvolí typ autentizace mezi OSPF směrovači, a to buď bez autentizace, nebo jednoduchý typ autentizace na základě klíče definovaného v poli Authentication Key, nebo autentizace pomocí hashovací funkce algoritmem MD5. Protokol OSPF se prostřednictvím autentizace původce zprávy OSPF dokáže chránit proti útokům, kdy dochází k podvržení falešných směrovacích informací. Pole Network type definuje, jakým způsobem jsou zprávy OSPF rozesílány do sítě. Může se jednat o typ broadcast, kdy zprávy mají charakter všesměrového vysílaní, dále typ NBMA (Non-Broadcast MultiAccess), kdy se k vytvoření sousedských vztahů využívá manuální statické

konfigurace mezi pověřenými směrovači, nebo typu point-to-point, kdy se jedná pouze o dva směrovače, které si mezi sebou vyměňují OSPF zprávy. Tlačítko Passive určuje, zda interface je či není zapojen do OSPF komunikace, není-li zapojen, nerozesílá ani nepřijímá pakety protokolu OSPF na tomto rozhraní.

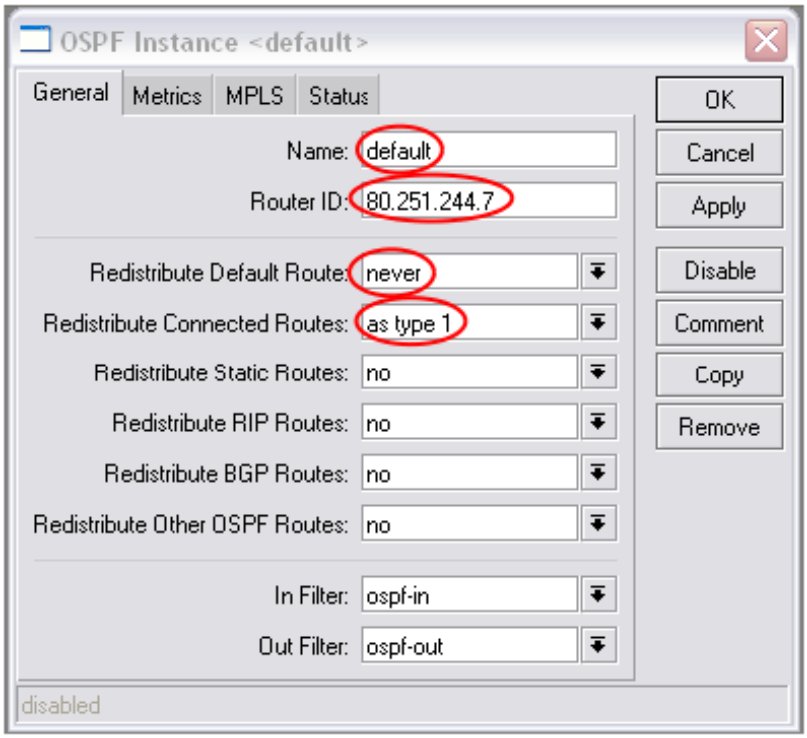
V další záložce Instances se vytváří instance pro konkrétní daný OSPF směrovač. Zároveň může běžet více jak jedna instance OSPF procesu. Pole Name definuje jméno (označení) instance, RouterID identifikuje směrovače (např. IP adresa směrovače) a definují se parametry, jestli jsou distribuovány routy a pokud ano, tak z jakých zdrojů (default, statické, BGP). OSPF podporuje dva typy určování metrik:

- Type 1 – OSPF metrika je součtem interní ceny (cost) OSPF a externích nákladů na cestu.
- Type 2 – OSPF metrika je rovna pouze ceně externí cesty.

Konfigurace pomocí CLI:

[libor@OSPF] > *routing ospf instance add name=ospf router-id=80.251.244.7 distribute-default=never redistribute-connected=as-type-1*

Konfigurace instance pomocí GUI Winbox popisuje následující obrázek.



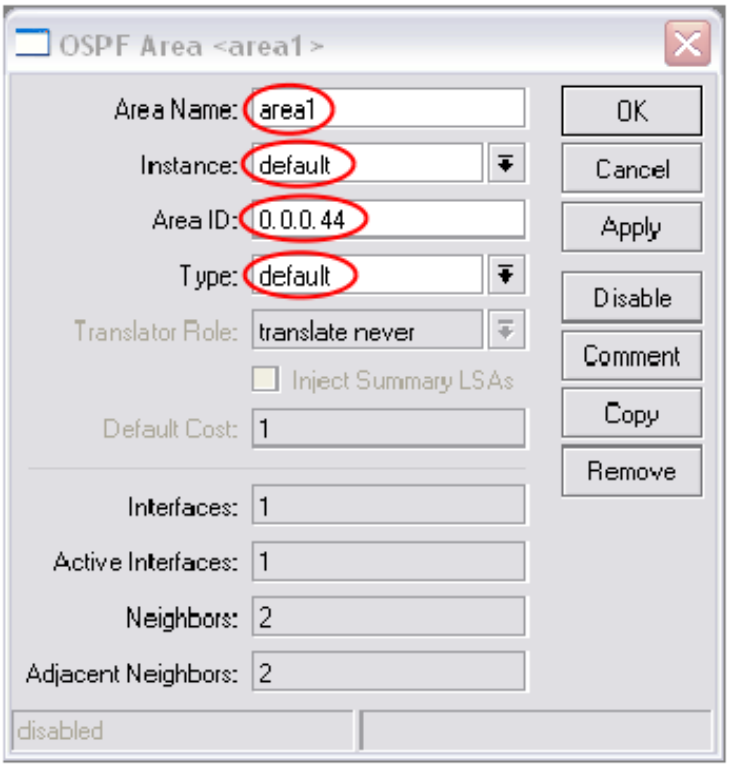
Obrázek 29 – Konfigurace instance OSPF

V další záložce Networks se definuje oblast (area) a k ní síť, do které je přiřazena. Z toho je patrné, že nejprve je nutné definovat oblast (areu) v záložce Areas. Area je definována svým jménem, instancí, ve které je vytvořena, identifikátorem oblasti ID a typem oblasti. Popis jednotlivých typů oblastí je uveden v následující části, která popisuje konfiguraci oblasti stub.

Konfigurace pomocí CLI:

[libor@OSPF] > *routing ospf area add name=area2 instance=default area-id=0.0.0.44 type=default*

Konfigurace oblasti pomocí GUI Winbox popisuje následující obrázek.



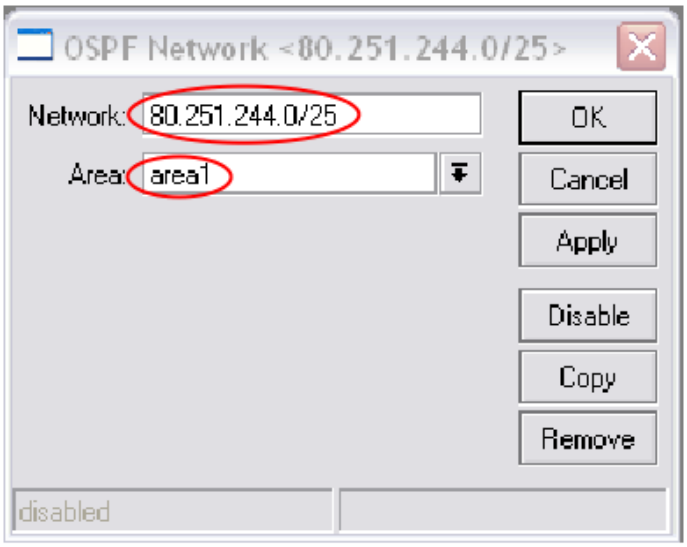
Obrázek 30 – Konfigurace oblasti v OSPF

Nyní je možné vrátit se ke konfiguraci sítě v záložce Networks. Pokud má být protokol OSPF spuštěn, je nutné definovat sítě, na kterých běží OSPF a související oblasti pro každou z těchto sítí, tzn. určit oblast, do které je přiřazena ta která IP síť.

Konfigurace pomocí CLI:

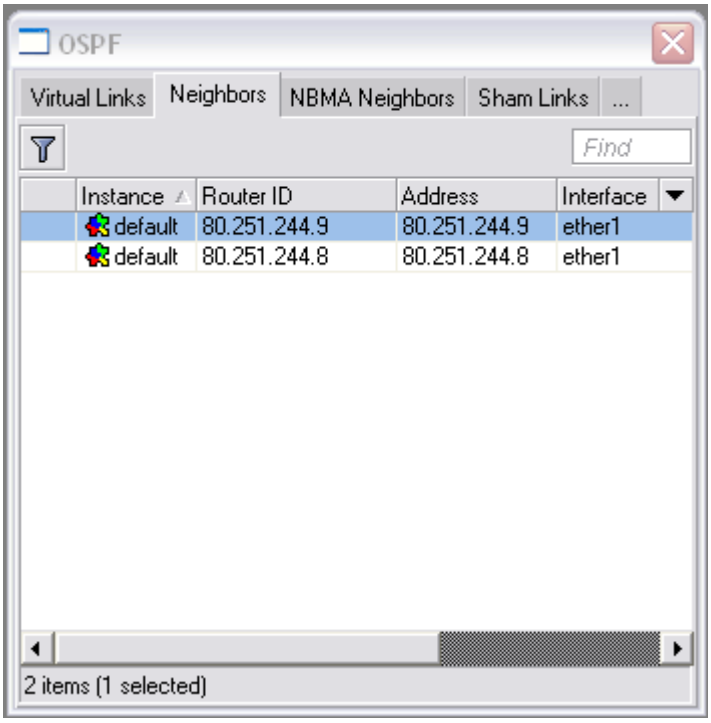
[libor@OSPF] > *routing ospf network add area=area1 network=80.251.244.0/25*

Konfigurace oblasti pomocí GUI Winbox popisuje následující obrázek.



Obrázek 31 - Konfigurace IP sítě a oblasti

Nyní je navázána komunikace OSPF mezi směrovači, což je možné ověřit v záložce Neighbors, kde jsou v tabulce uvedeny směrovače, se kterými je aktuálně navázána OSPF komunikace, jak uvádí následující obrázek.

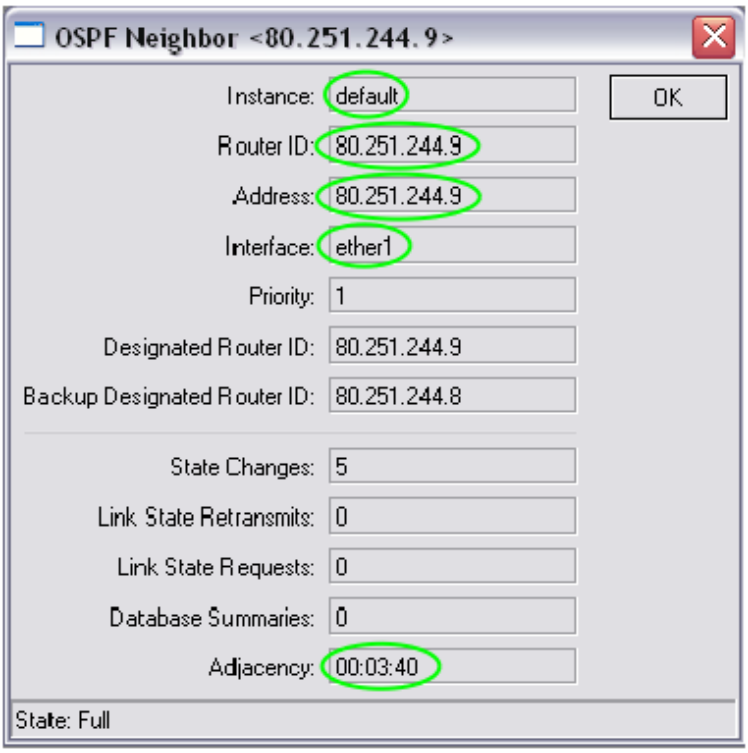


Obrázek 32 – Neighbors

Konfigurace pomocí CLI:

[libor@OSPF] > *routing ospf neighbor print*

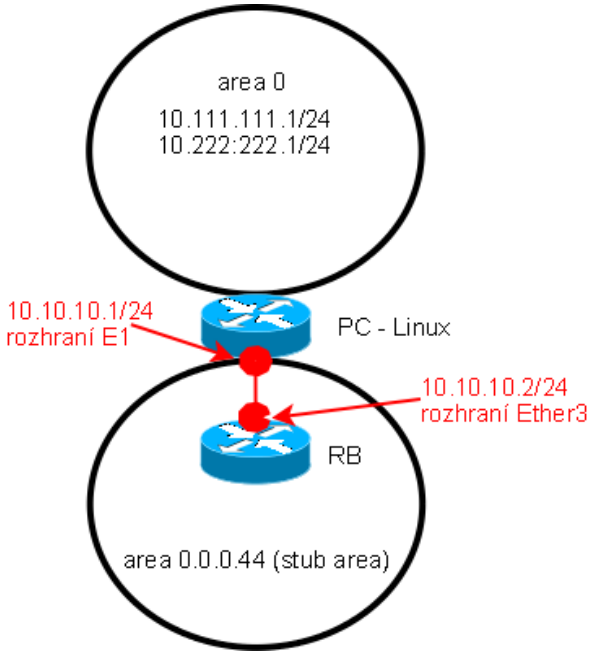
Detailní konfigurace konkrétního směrovače je patrná z obrázku 33. V poli Adjacency je uvedený čas, jak dlouho je již komunikace navázána.



Obrázek 33 – Stav směrovače po navázání OSPF spojení

9.3 Konfigurace OSPF – STUB AREA

Stub area je takovým typem oblasti, kdy veškerý datový tok v této oblasti začíná i končí a neprochází přes tuto oblast dále. Do oblasti typu stub se neoznamují přilehlé sítě, ani přes tuto oblast neprochází. Přes oblast typu stub není možné definovat virtuální link a nesmí být v této oblasti žádný ASBR směrovač (směrovač, který je umístěn na hranici autonomního systému mezi OSPF směrováním a např. RIP směrováním). Oblasti typu stub snižují velikost topologické databáze a tím šetří množství paměti směrovače. Pokud je směrovač nakonfigurován uvnitř stub oblasti, pak automaticky inzeruje výchozí trasy. Topologie sítě s využitím stub oblasti je patrná z následujícího obrázku.



Obrázek 34 – OSPF s oblastí typu stub

Konfigurace quagga s využitím stub oblasti na směrovači PC-Linux:

```
! Zebra configuration saved from vty
!   2011/03/31 23:23:02
hostname rt-lb-ospfd
password 1234
enable password 1234
interface dummy0
interface dummy1
interface eth0
interface eth1
interface lo
router ospf
 ospf router-id 10.10.10.1
 compatible rfc1583
 network 10.10.10.0/24 area 0.0.0.44
 network 10.111.111.0/24 area 0.0.0.0
 network 10.222.222.0/24 area 0.0.0.0
 area 0.0.0.44 stub no-summary
line vty
 exec-timeout 0 0
```

Konfigurace OSPF s využitím stub oblasti na směrovači RouterBoard:

[libor@OSPF] *>/routing ospf instance set default comment="" disabled=no distribute-default=never in-filter=ospf-in metric-bgp=20 metric-connected=20 metric-default=1*

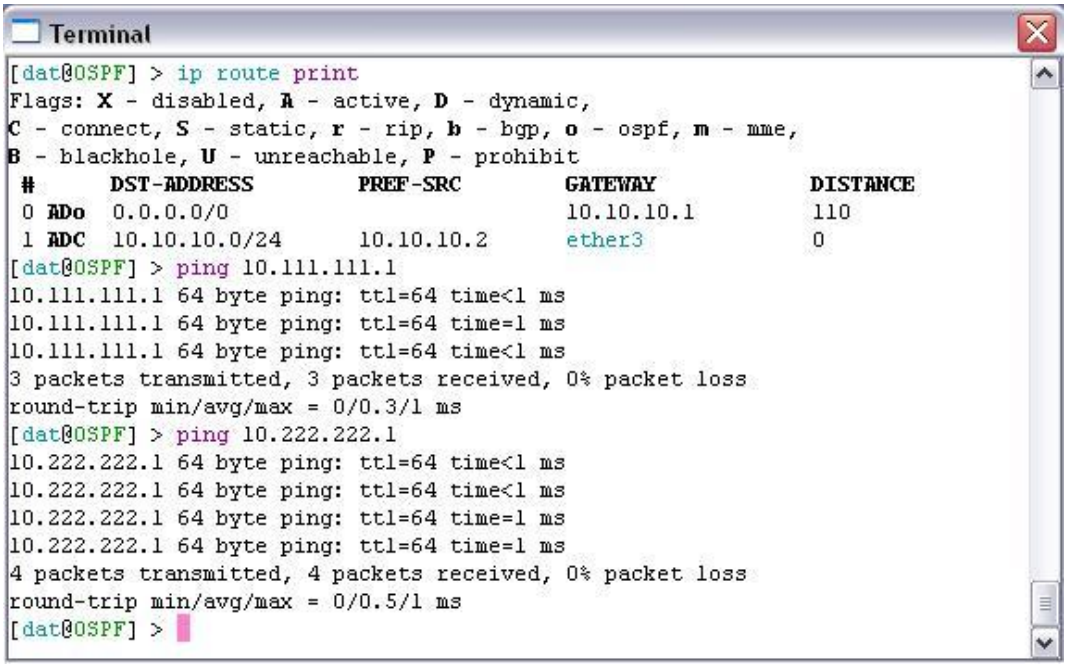
metric-other-ospf=auto metric-rip=20 metric-static=20 name=default out-filter=ospf-out redistribute-bgp=no redistribute-connected=as-type-1 redistribute-other-ospf=no redistribute-rip=no redistribute-static=no router-id=80.251.244.7

[libor@OSPF] > */routing ospf area set backbone area-id=0.0.0.0 comment="" disabled=no instance=default name=backbone type=default add area-id=0.0.0.44 comment="" default-cost=1 disabled=no inject-summary-lsas=yes instance=default name=area1 type=stub*

[libor@OSPF] > */routing ospf interface add authentication=none authentication-key=12345 authentication-key-id=1 comment="" cost=10 dead-interval=40s disabled=no hello-interval=10s instance-id=0 interface=ether1 network-type=broadcast passive=no priority=1 retransmit-interval=5s transmit-delay=1s use-bfd=no*

[libor@OSPF] > */routing ospf network add area=area1 comment="" disabled=no network=10.10.10.0/24*

Po takto realizované konfiguraci je po zadání příkazu *ip route print* na směrovači RouterBoard vypsána směrovací tabulka, ze které je patrné, že konfigurace výchozí brány je provedena dynamicky, je aktivní a generována na základě OSPF směrování. Taktéž je možné provést ověření dostupnosti sítí umístěných v oblasti 0, jak je uvedeno na následujícím obrázku.



Obrázek 35 – Ověření OSPF s oblastí typu stub

9.4 Komunikace OSPF

Na následujících řádcích je zachycena komunikace na rozhraní E1 směrovače dle předchozí topologie po aktivaci protokolu OSPF na rozhraní E1 směrovače. Nejprve směrovač 10.10.10.2 vysílá Hello paket (každých 10 sekund), směrovač 10.10.10.1 neodpovídá (není aktivní protokol OSPF). V čase 130.013771 sekund je aktivován protokol OSPF na směrovači 10.10.10.1 a vysílá Hello paket. V čase 130.081189 odpovídá směrovač 10.10.10.2. Následuje sumarizace obsahu databáze obou směrovačů a v čase 130.082495 žádá směrovač 10.10.10.2 o topologickou databázi směrovač 10.10.10.1 a ten provádí její aktualizaci. V čase 130.083279 si oba směrovače vymění role a žádost odesílá směrovač 10.10.10.1 a směrovač 10.10.10.2 odpovídá zprávou Link State Update. Směrovač 10.10.10.1 potvrzuje v čase 130.261337 přijetí aktualizace databáze a v čase 130.623819 odesílá potvrzovací paket do sítě ostatním směrovačům. Následuje potvrzení směrovače 10.10.10.2 v čase 136.101250 a tím je komunikace OSPF navázána a následuje udržovací komunikace prostřednictvím OSPF Hello paketů do té doby, než dojde ke změně v konfiguraci či nastavení (např. přidání sítě, která je dostupná přes směrovač) některého ze směrovačů.

110.061123 10.10.10.2 -> 224.0.0.5 OSPF Hello Packet

120.071177 10.10.10.2 -> 224.0.0.5 OSPF Hello Packet

130.013771 10.10.10.1 -> 224.0.0.5 OSPF Hello Packet

130.081189 10.10.10.2 -> 224.0.0.5 OSPF Hello Packet

130.081480 10.10.10.1 -> 10.10.10.2 OSPF DB Descr.

130.081949 10.10.10.2 -> 10.10.10.1 OSPF DB Descr.

130.082032 10.10.10.1 -> 10.10.10.2 OSPF DB Descr.

130.082495 10.10.10.2 -> 10.10.10.1 OSPF LS Request

130.082549 10.10.10.1 -> 224.0.0.5 OSPF LS Update

130.083207 10.10.10.2 -> 10.10.10.1 OSPF DB Descr.

130.083264 10.10.10.1 -> 10.10.10.2 OSPF DB Descr.

130.083279 10.10.10.1 -> 10.10.10.2 OSPF LS Request

130.083447 10.10.10.2 -> 224.0.0.5 OSPF LS Update

130.084308 10.10.10.2 -> 10.10.10.1 OSPF LS Update

130.261220 10.10.10.2 -> 224.0.0.5 OSPF LS Update

130.261337 10.10.10.1 -> 10.10.10.2 OSPF LS Acknowledge

130.623819 10.10.10.1 -> 224.0.0.5 OSPF LS Acknowledge

131.083902 10.10.10.1 -> 224.0.0.5 OSPF Hello Packet

131.084156 10.10.10.2 -> 224.0.0.5 OSPF LS Acknowledge

135.094345 10.10.10.1 -> 224.0.0.5 OSPF LS Update

136.101250 10.10.10.2 -> 224.0.0.5 OSPF LS Acknowledge

140.091299 10.10.10.2 -> 224.0.0.5 OSPF Hello Packet

150.101320 10.10.10.2 -> 224.0.0.5 OSPF Hello Packet

151.086029 10.10.10.1 -> 224.0.0.5 OSPF Hello Packet

9.5 Vazba na IPv6

Protokol RIP je zastaralý jednoduchý směrovací protokol dynamického směrování a v této základní variantě IPv6 nepodporuje. Verze podporující IPv6 je označována jako RIPng a je definována v RFC 2080 [15]. Bohužel, veškeré jeho omezující vlastnosti jsou i v této verzi ponechány. Protokol OSPFv3 plně podporuje směrování v sítích IPv6. Je definován v RFC 5340 [16]. Podpora IPv6 v externím směrovacím protokolu BGP4+ je popsána v definici RFC 4271 [17].

10 PŘÍPADOVÁ STUDIE IV – UŽITÍ QOS A FUP

V této případové studii je popsáno, jakým způsobem lze zajistit na platformě RouterBoard realizaci služeb QoS. Platforma RouterBoard umožňuje řídit (prioritizovat) a omezovat provoz pomocí front (Queue). Fronty je možné použít k těmto typům aplikací:

- Omezení rychlosti přenosu dat u zvolených IP adres, adresných rozsahů, protokolů nebo portů.
- Omezení rychlosti přenosu dat z a do peer-to-peer sítí.
- Upřednostňovat definované paketové toky před jinými.
- Používat datové limity v závislosti na časových intervalech.
- Rozdělovat přenosovou rychlost spravedlivě mezi uživatele, nebo v závislosti na zatížení kanálu.

Implementace front na Mikrotik RouterOS je založena na HTB (Hierarchical Token Bucket), které umožňuje vytvářet hierarchické struktury front a určovat vztah mezi nimi. Existují dva možné způsoby, jak lze fronty v Mikrotik RouterOS konfigurovat:

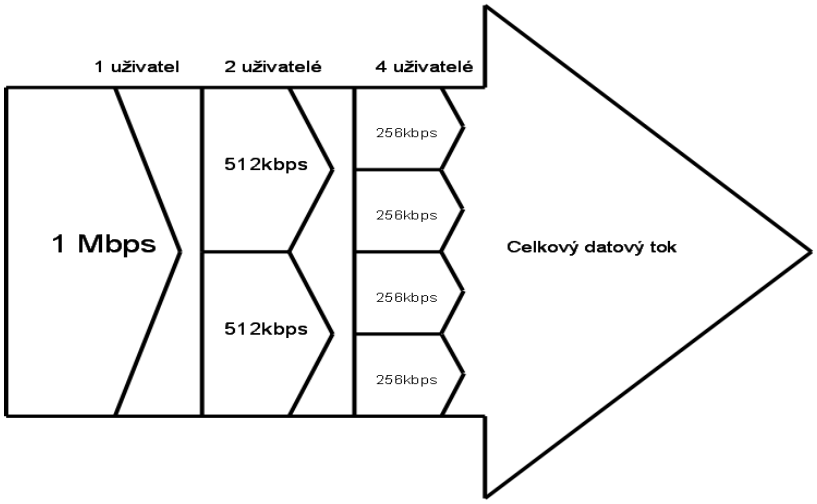
- Jednoduchá fronta – Simple Queue – používají se pro jednoduché konfigurace, např. pro omezení provozu jednoho klienta, omezení P2P provozu atd.
- Stromová struktura front – Queue Tree – pro implementaci pokročilých úloh, jako je globální politika atd.

V Mikrotik RouterOS je definováno několik typů front, které určují chování a nakládání s pakety, např. při přetížení velkým datovým tokem:

- PFIFO, BFIFO – založeno na algoritmu FIFO (First In First Out), P=packet, B=byte. Definuje se maximální počet paketů, které lze ve frontě držet.
- RED (Random Early Drop) – řídí průměrnou velikost fronty a při přetížení začne pakety náhodně zahazovat.
- SFQ (Stochastic Fairness Queuing) – jednoznačné určení datového toku je realizováno čtyřmi možnostmi (zdrojová adresa, cílová adresa, zdrojový port, cílový port). Nezajistí spravedlivé dělení mezi jednotlivé uživatele, ale např. mezi jednotlivými spojeními. Rozděluje provoz do více FIFO front, do kterých rozděluje jednotlivá TCP i UDP spojení.

- PCQ (Per Connection Queuing) – podobně jako SFQ, ale má další možnosti identifikace určení datového toku. PCQ bylo zavedeno s cílem optimalizovat velké QoS systémy a zajistit dělení datového toku spravedlivě mezi jednotlivé klienty (např. IP klienty).

Při použití front typu strom je nutné zajistit v prvé řadě značení paketů v datovém toku. Případová studie popisuje konfiguraci fronty, která zajišťuje prioritu paketů, které generuje provoz VoIP zařízení a rozděluje volnou kapacitu spravedlivě mezi všechny aktivní uživatele. Způsob dělení datového toku s využitím SFQ mezi jednotlivé uživatele popisuje následující obrázek.



Obrázek 36 – Dělení pásma pomocí SFQ

Případová studie je řešena na RB 450, kde interface ether1 je nastaven jako rozhraní pro připojení do Internetu a zbylá čtyři rozhraní jsou využita pro připojení stanic ve vnitřní síti. Maximální přenosová rychlost do Internetu je omezena na 1 Mbps. Dále je definována IP adresa SIP serveru, ke kterému jsou připojeny VoIP zařízení z vnitřní sítě (80.251.241.225). V menu IP/Firewall/Mangle jsou definována pravidla pro značkování paketů. Je třeba označit pakety, které mají vazbu na VoIP (cílová nebo zdrojová IP adresa je IP adresa SIP serveru). Pravidlo je definováno pro každý směr zvlášť (uplink/downlink). Další dvě pravidla označí veškerý provoz mimo VoIP komunikaci.

Příkazy CLI pro značkování provozu:

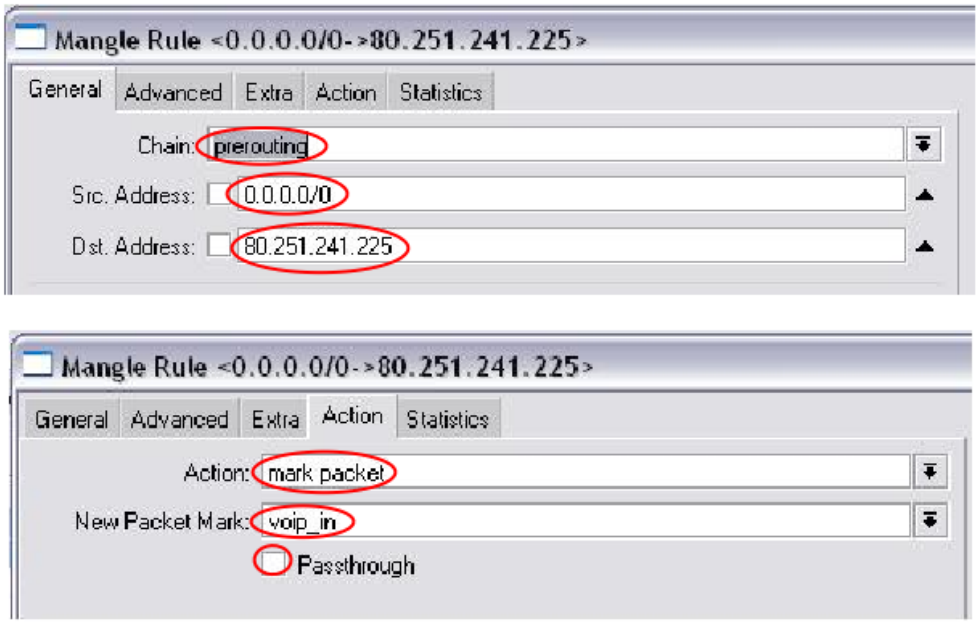
[liborll@PS-1] > *ip firewall mangle add action=mark-packet chain=prerouting comment="VoIP z vnitrni site" disabled=no dst-address=80.251.241.225 new-packet-mark=voip_in passthrough=no src-address=0.0.0.0/0*

[liborll@PS-1] > *ip firewall mangle add action=mark-packet chain=prerouting comment="VoIP do vnitrni site" disabled=no dst-address=0.0.0.0/0 new-packet-mark=voip_out passthrough=no src-address=80.251.241.225*

[liborll@PS-1] > *ip firewall mangle add action=mark-packet chain=prerouting comment="Zbytek provozu mimo VoIP z vnitrni site" disabled=no dst-address=!80.251.241.225 in-interface=ether1-wan new-packet-mark=zbytek_in passthrough=no src-address=0.0.0.0/0*

[liborll@PS-1] > *ip firewall mangle add action=mark-packet chain=postrouting comment="Zbytek provozu mimo VoIP do vnitrni site" disabled=no dst-address=0.0.0.0/0 new-packet-mark=zbytek_out out-interface=ether1-wan passthrough=no src-address=!80.251.241.225*

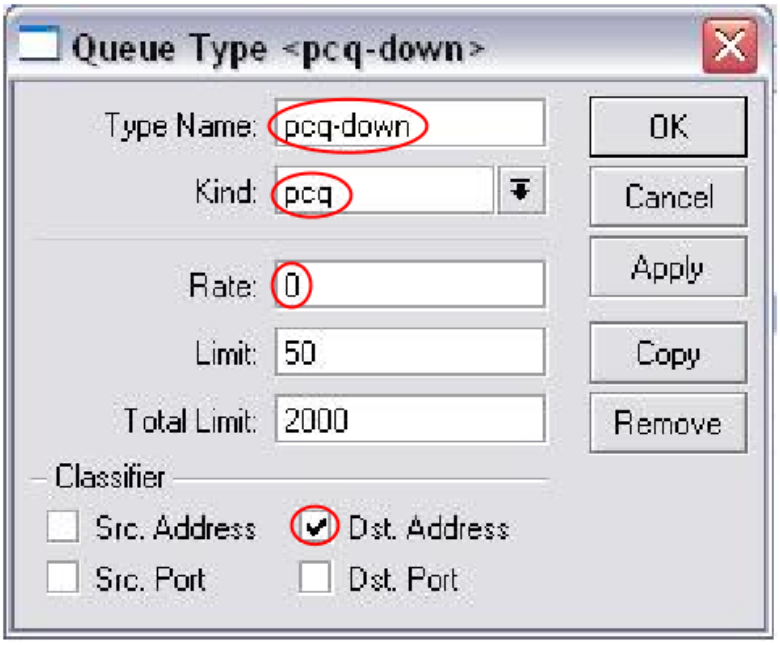
Konfiguraci pravidla pro označení paketu VoIP komunikace směrem z vnitřní sítě pomocí GUI Winbox popisuje následující obrázek.



Obrázek 37 – Značkování paketů

V menu IP/Firewall v záložce Mangle je vytvořeno pravidlo, které zajistí označení všech paketů (mark_packet) s libovolnou zdrojovou adresou (Src. Adddres – 0.0.0.0/0) a cílovou adresou 80.251.241.225 značkou voip_in. Volba Passthrough určuje, zda má paket být po označení předán k prověření dalšímu pravidlu či nikoliv. V dalším kroku je nutné definovat typ front, které budou použity v konfiguraci stromu front. Z výše uvedeného je patrné, že pro řešení popsané situace je třeba definovat dva typy front, a to frontu typu PCQ pro

downlink a frontu typu PCQ pro uplink. V menu Queue v záložce Queue Types je definována fronta pcq_down a pcq_up. Vytvoření fronty pcq_down popisuje následující obrázek.



Obrázek 38 – Definice fronty

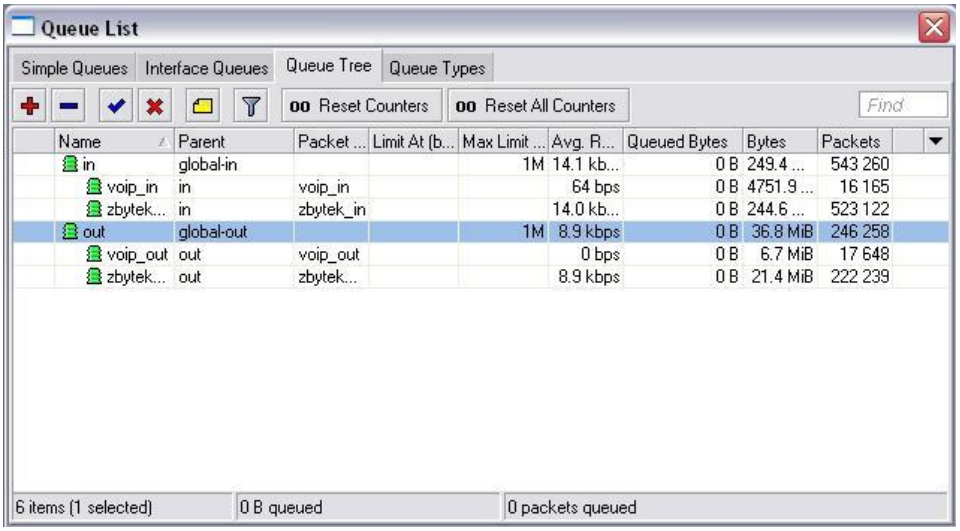
V poli Type Name je definováno jméno fronty, pole Kind definuje, jaký druh fronty bude použit. Pokud je v poli Rate uvedena jiná hodnota než 0, pak tato hodnota udává hodnotu maximálního datového toku pro jednu frontu (v této konfiguraci pro jednu IP stanici). V sekci Classifiers je definováno, jakým parametrem bude klasifikován provoz. Pro frontu downlink je to cílová adresa, pro frontu uplink je to zdrojová adresa. Pole Limit a Total Limit definují velikost fronty v paketech pro jeden tok a součet všech toků.

Příkaz CLI:

[liborll@PS-1] > *queue type add kind=pcq name=pcq-down pcq-classifier=dst-address pcq-limit=50 pcq-rate=0 pcq-total-limit=2000*

[liborll@PS-1] > *queue type add kind=pcq name=pcq-up pcq-classifier=src-address pcq-limit=50 pcq-rate=0 pcq-total-limit=2000*

V záložce Queue Tree lze definovat strom front. Definují se dvě globální fronty, jedna pro uplink, druhá pro downlink a v každé takto definované rodičovské frontě se vytvoří dvě fronty, kde jedna bude pro VoIP provoz a druhá pro všechen ostatní provoz. Výsledný stav po konfiguraci kompletního stromu front je patrný z následujícího obrázku.



Obrázek 39 – Strom front

Při konfiguraci stromu front je nutné nejdříve definovat obě rodičovské fronty jak pro provoz z vnitřní sítě, tak do vnitřní sítě (fronta in a out) s definovaným maximálním limitem datového toku pro každý směr. Příkazy pro CLI pro definici rodičovských front.

[liborll@PS-1] > *queue tree add burst-limit=0 burst-threshold=0 burst-time=0s disabled=no limit-at=0 max-limit=1M name=in parent=global-in priority=1*

[liborll@PS-1] > *queue tree add burst-limit=0 burst-threshold=0 burst-time=0s disabled=no limit-at=0 max-limit=1M name=out parent=global-out priority=1*

V každé z obou rodičovských front (in, out) je definována fronta pro provoz VoIP a pro ostatní zbylý provoz.

Příkaz CLI:

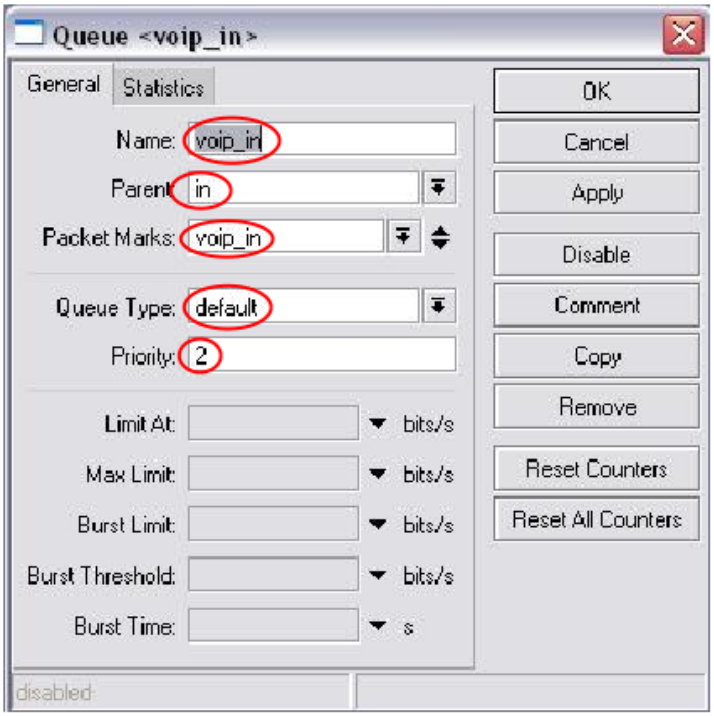
[liborll@PS-1] > *queue tree add burst-limit=0 burst-threshold=0 burst-time=0s disabled=no limit-at=0 max-limit=0 name=voip_in packet-mark=voip_in parent=in priority=2 queue=default*

[liborll@PS-1] > *queue tree add burst-limit=0 burst-threshold=0 burst-time=0s disabled=no limit-at=0 max-limit=0 name=zbytek_in packet-mark=zbytek_in parent=in priority=8 queue=pcq-down*

[liborll@PS-1] > *queue tree add burst-limit=0 burst-threshold=0 burst-time=0s disabled=no limit-at=0 max-limit=0 name=voip_out packet-mark=voip_out parent=out priority=2 queue=default*

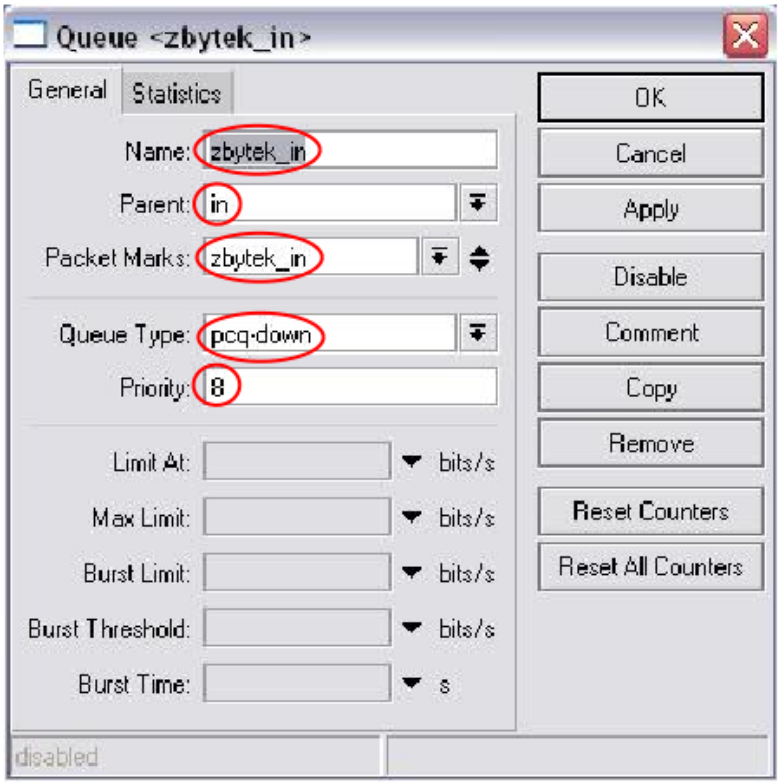
[liborll@PS-1] > *queue tree add burst-limit=0 burst-threshold=0 burst-time=0s disabled=no limit-at=0 max-limit=0 name=zbytek_out packet-mark=zbytek_out parent=out priority=8 queue=pcq-up*

Vytvoření fronty pro VoIP ve směru do vnitřní sítě (voip_in) popisuje následující obrázek.



Obrázek 40 – Fronta typu default

V poli Name je definován název fronty. Pole Parent definuje nadřazenou rodičovskou frontu (v tomto případě již vytvořenou frontu in). Pole Packet Marks určuje, které pakety (resp. jak označené) budou do této fronty zahrnuty. Typ fronty je uveden v poli Queue Type (jedná se pouze o provoz VoIP, lze tedy použít výchozí konfiguraci default, kde se jedná o klasickou frontu FIFO). Pole Priority udává, s jakou prioritou budou pakety z této fronty odbavovány (čím nižší hodnota, tím větší priorita). Provozu VoIP je třeba určit vyšší priority než ostatnímu provozu (v tomto případě je definována pro VoIP hodnota 2 a pro zbylý provoz je to hodnota 8). Fronta definovaná v každém směru pro jiný provoz než VoIP musí mít definovaný typ fronty pcq_down pro frontu in a pcq_up pro frontu out z důvodu spravedlivého dělení datového toku pro všechny uživatele. Vytvoření fronty ve směru do vnitřní sítě pro jiné než VoIP pakety je patrné z následujícího obrázku.



Obrázek 41 – Fronta typu pcq

Tímto je strom front definován a řízení toku sítě odpovídá zadaným požadavkům. Prioritně je odbavován provoz směřující do a ze SIP serveru (hlasová komunikace VoIP). Zbylé volné pásmo je spravedlivě děleno mezi všechny ostatní uživatele (stanice v síti).

ZÁVĚR

Cílem práce bylo popsat konfiguraci platformy RouterBoard s Mikrotik RouterOS realizující bránu veřejného přístupového bodu pro připojení do sítě Internet, dynamického směrovače s konfigurací protokolu OSPF a brány do virtuální privátní sítě. Závěrečná část popisuje použití platformy RouterBoard pro řízení datového provozu. Diplomová práce krok po kroku popisuje konkrétní, výše uvedené případy využití platformy RouterBoard a lze ji využít jako návodu pro realizaci uvedených řešení.

Všechny body zadání diplomové práce jsou naplněny jak v teoretické, tak i v praktické části. Veškeré realizované případové studie byly na platformě RouterBoard otestovány a ověřeny jako funkční. Dynamické směrování protokolem OSPF v rámci autonomního systému vykazovalo standardní chování a došlo k navázání OSPF komunikace mezi platformou RouterBoard a OSPF realizací na platformě Cisco i mezi platformou RouterBoard a instalací software quagga na OS Linux. Způsob konfigurace protokolu OSPF je na platformě RouterBoard v některých částech poněkud odlišná od konfigurace např. v softwaru quagga a dle mého názoru je konfigurace rozdělena do zbytečně velkého množství malých konfiguračních bloků, což bylo možná jedním z důvodů vadné konfigurace BGP směrovače (realizovaného na platformě RouterBoard), který způsobil nedostupnost velké části sítě Internet, jak je popsáno v této práci. Při realizaci brány IPsec virtuální privátní sítě typu LAN-to-LAN jsem problém v konfiguraci nezaznamenal. Bohužel není možné použít platformu RouterBoard jako IPsec bránu pro virtuální privátní síť typu remote access. Tento požadavek je možné řešit buď využitím IPsec/L2TP nebo pomocí OpenVPN, kde se mi ovšem konfigurace jeví v porovnání s konkurenční platformou složitá a ne úplně přehledná. I přes tento postřeh se realizace brány OpenVPN pro VPN typu remote access zdařila a přes implementační složitosti fungovala bezproblémově. V realizaci VPN brány typu remote access na platformě RouterBoard vidím široké uplatnění vzhledem k nízkým nákladům na pořízení a minimální energetické náročnosti. Konfigurace veřejně přístupného bodu (Hotspot) je ve své základní konfiguraci jednoduchá, nicméně dokáže uspokojit požadavky i náročných uživatelů a lze definovat velké množství omezujících a upřesňujících parametrů. Jako zařízení, které má řídit provoz sítě, lze platformu RouterBoard poměrně úspěšně využít v malých a středních aplikacích. Poslední případová studie popisuje nastavení s prvky QoS i FUP a s ohledem na hardwarovou konfiguraci ji lze dle konkrétního požadavku kombinovat a rozšiřovat.

Vzhledem k realizovaným případovým studiím spatřuji široké uplatnění platformy RouterBoard zejména v domácích a menších až středních podnikových sítích. Zařízení dokáže splnit i náročnější požadavky, které při správě a řízení takto velkých sítí mohou nastat.

ZÁVĚR V ANGLIČTINĚ

The goal of my work is to describe a configuration of Mikrotik RouterOS for following purposes: public hotspot for Internet access, OSPF router, VPN gateway, traffic management. The master thesis deals with fore mentioned topics in details and exhibits them in real-world scenarios, therefore, it could be used as an instruction for implementations very well.

All of requirements have been fulfilled in practical and theoretical parts. Described case studies have been tested and confirmed as working. It has been verified that Mikrotik's OSPF implementation is interoperable with OSPF enabled Cisco routers and with Zebra/Quagga routing suite. No deviations from standards were observed in Mikrotik's OSPF implementation, however, the user interface for routing configuration generally is very different from Cisco IOS or Zebra/Quagga CLI and is not well arranged. The last mentioned drawback is most likely one of reasons of BGP misconfiguration that caused unreachability of a part of Internet as cited in the thesis. Routerboard running RouterOS acts as a site-to-site IPsec VPN gateway well. I didn't encounter any problems regarding interoperability during my tests. However, it's not possible to use an equipment running RouterOS as a gateway for IPsec remote clients. Only IPsec over L2TP (Windows native) and OpenVPN remote clients are supported. OpenVPN implementations has been successfully tested. I have to mention again that user interface for configuration is complicated and not well arranged. For traffic management purposes, a Routerboard running RouterOS suits best for small to mid implementations. It supports all necessary features for traffic policing and shaping as described along with hardware requirements in the last case study which deals with QoS and FUP topics.

For all described purposes and with regard to case studies, I consider Routerboard platform most suitable for home, single or small offices or small businesses and ISPs. The equipment is very flexible and features can be enabled or disabled according the intended usage.

SEZNAM POUŽITÉ LITERATURY

[1] PUŽMANOVÁ, Rita. *TCP/IP v kostce*. České Budějovice : KOOP, 2004. 608 s. ISBN 80-7232-236-2.

[2] *MikroTik Routers and Wireless* [online]. 2011 [cit. 2011-01-26]. Dostupný z WWW: <http://www.mikrotik.com/>.

[3] *Routerboard.com* [online]. 2011 [cit. 2011-01-26]. Dostupný z WWW: <http://www.routerboard.com/>.

[4] *MikroTik Wiki* [online]. 2011 [cit. 2011-03-21]. Category : Manual - MikroTik Wiki. Dostupné z WWW: <http://wiki.mikrotik.com/wiki/Category:Manual>.

[5] *RFC 2764* [online]. 2011 [cit. 2011-03-11]. Dostupný z WWW: <http://www.ietf.org/rfc/rfc2764.txt>.

[6] *RFC 2401* [online]. 2011 [cit. 2011-02-16]. Dostupný z WWW: <http://www.ietf.org/rfc/rfc2401.txt>.

[7] SPORTACK, Mark. *Směrování v sítích IP*.Brno : Computer Press, a.s., 2004. ISBN 80-251-0127-4. s. 351.

[8] *RFC 1058* [online]. 2011 [cit. 2011-03-13]. Dostupný z WWW: <http://www.ietf.org/rfc/rfc1058.txt>.

[9] *RFC 2453* [online]. 2011 [cit. 2011-03-13]. Dostupný z WWW: <http://www.ietf.org/rfc/rfc2453.txt>.

[10] *RFC 2328* [online]. 2011 [cit. 2011-03-13]. Dostupný z WWW: <http://www.ietf.org/rfc/rfc2328.txt>.

[11] *Quagga Software Routing Suite* [online]. 2011 [cit. 2011-03-15]. Dostupný z WWW: <http://www.quagga.net/>.

[12] POSPÍCHAL, Zbyněk. *Www.etron.cz* [online]. 21.2.2009 [cit. 2011-03-31]. Malý český ISP způsobil světový kolaps. Dostupné z WWW: <http://www.etron.cz/index.php?option=com_content&view=article&id=67:maly-esky-isp-zpsobil-svtovy-kolaps&catid=15:tiskove-zpravy>.

[13] *Shrew Soft Inc* [online]. 2009 [cit. 2011-04-03]. Shrew Soft Inc : Download. Dostupné z WWW: <http://www.shrew.net/download>.

[14] *OpenVPN* [online]. 2011 [cit. 2011-04-17]. OpenVPN-Open Source VPN. Dostupné z WWW: <http://openvpn.net/>.

[15] *RFC 2080* [online]. 2011 [cit. 2011-03-13]. Dostupný z WWW: <http://www.ietf.org/rfc/rfc2080.txt>.

[16] *RFC 5340* [online]. 2011 [cit. 2011-03-13]. Dostupný z WWW: <http://www.ietf.org/rfc/rfc5340.txt>.

[17] *RFC 4271* [online]. 2011 [cit. 2011-03-13]. Dostupný z WWW: <http://www.ietf.org/rfc/rfc4271.txt>.

[18] PUŽMANOVÁ, Rita. *Bezpečnost bezdrátové komunikace : Jak zabezpečit Wi-Fi, bluetooth, GPRS či 3G*. [s.l.] : [s.n.], 2005. 184 s. ISBN 80-251-0791-4.

[19] HOLÍK, Aleš. *Kontrola rychlosti přenosu dat*. Zlín, 2007. 55 s. Bakalářská práce. UTB Zlín, FAI.

[20] KRČÁL, Martin. *Citace.com* [online]. 2011 [cit. 2011-04-04]. Citace 2.0 - vše o citování literatury a dokumentů (http://www.citace.com). Dostupné z WWW: <http://www.citace.com/generator.php>.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

3DES	Triple DES.
AES	Advanced Encryption Standard
AH	Authentication Header
AS	Autonomous Systém
ASBR	Autonomous System Boundary Router
ATM	Asynchronous Transfer Mode
BFIFO	Byte First In First Out
BGP	Border Gateway Protokol
CLI	Command Line Interface
CIDR	Classless Inter-Domain Routing
DES	Data Encryption Standard
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Systém
EGP	Exterior Gateway Protokol
ESP	Encapsulating Security Payload
FTP	File Transfer Protocol
FUP	Fair User Policy
GPS	Global Positioning System
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
HTTPS	Secure Hypertext Transfer Protocol
IGP	Interior Gateway Protocol
IGRP	Interior Gateway Routing Protocol
IP	Internet Protocol

IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
IPsec	Internet Protokol Security.
ISP	Internet Service Provider
L2TP	Layer 2 Tunneling Protocol
LAN	Local Area Network.
LSA	Link State Advertisements
MD5	Message Digest
MPLS	Multiprotocol Label Switching
NAT	Network Address Translation
NTP	Network Time Protocol
OSPF	Open Shortest Path First
OpenVPN	Open Virtual Private Network.
PFIFO	Packet First In First Out
PPP	Point to Point Protocol
PPPoE	Point to Point Protocol over Ethernet
PPTP	Point to Point Tunneling Protocol
QoS	Quality Of Service
RED	Random Early Drop
RFC	Request For Comments
RIP	Routing Information Protocol
SHA	Secure Hash Algorithm
SMTP	Simple Mail Transfer Protocol
SNTP	Simple Network Time Protocol
SPI	Security Parameter Index

SPF	Shortest Path First
SSH	Secure Shell
TFTP	Trivial File Transfer Protocol
VLAN	Virtual LAN
VPN	Virtual Private Network
WAN	Wide Area Network.

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

SEZNAM PŘÍLOH

PŘÍLOHA P I: HARDWAROVÁ SPECIFIKACE

RB411

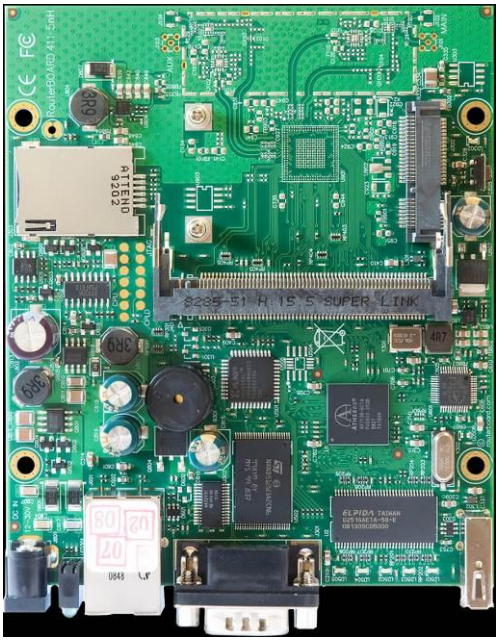
Detail	
Produkt	RB/411
CPU	300MHz
RAM	32MB
Architektura	MIPS-BE
LAN port	1
MiniPCI	1
USB	0
Paměťová karta	0
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Monitor napětí	Ne
RouterOS Licence	Level3

RB411AR

Detail	
Produkt	RB/411AR
CPU	300MHz
RAM	64MB
Architektura	MIPS-BE
LAN port	1
MiniPCI	1
Integrovaná karta wifi	1
Wifi standard	802.11 b/g
USB	0
Paměťová karta	0
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Monitor napětí	Ano
RouterOS Licence	Level4

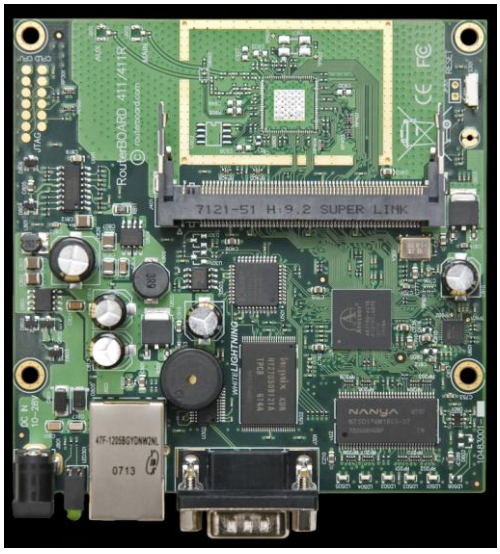
RB411U

Detail	
Produkt	RB/411U
CPU speed	300MHz
RAM	32MB
Architektura	MIPS-BE
LAN port	1
MiniPCI	1
miniPCI-e	1
Integrovaná karta wifi	0
USB	1
Paměťová karta	0
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Monitor napětí	Ano
RouterOS Licence	Level4



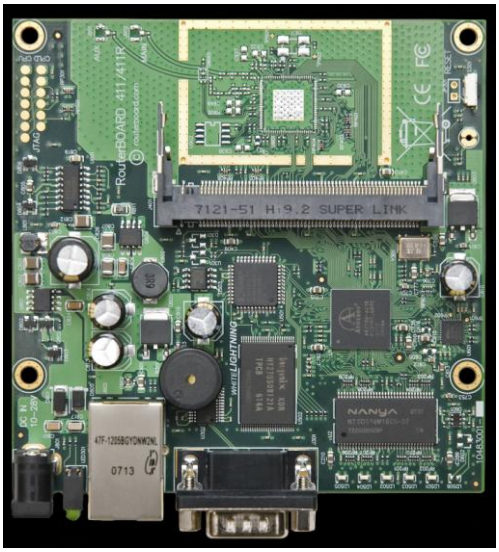
RB411AH

Detail	
Produkt	RB/411AH
CPU	680MHz
RAM	64MB
Architektura	MIPS-BE
LAN port	1
MiniPCI	1
Integrovaná karta wifi	0
USB	0
Paměťová karta	0
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Monitor napětí	Ano
RouterOS Licence	Level4



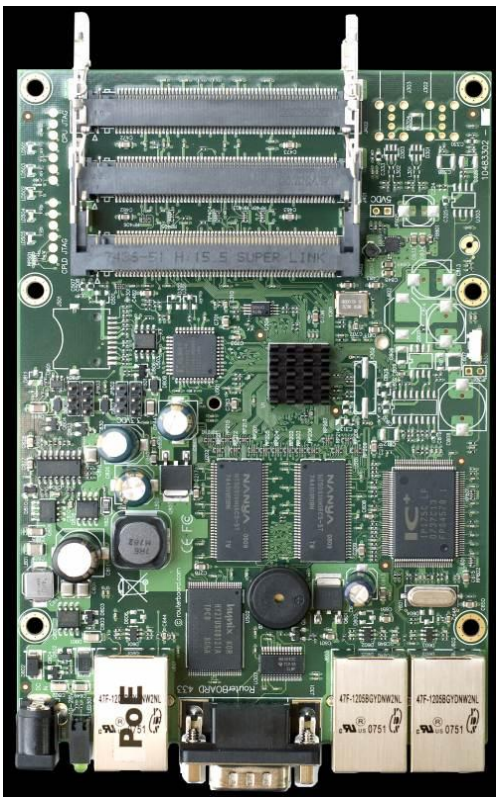
RB411UAHR

Detail	
Produkt	RB/411UAHR
CPU	680Mhz
RAM	64MB
Architektura	MIPS-BE
LAN port	1
Gigabit	0
MiniPCI	1
miniPCI-e	1 (USB/3G)
Integrovaná karta wifi	1
Wifi standard	802.11 b/g
USB	1
Paměťová karta	0
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Rozměr	105x105mm
RouterOS Licence	Level4



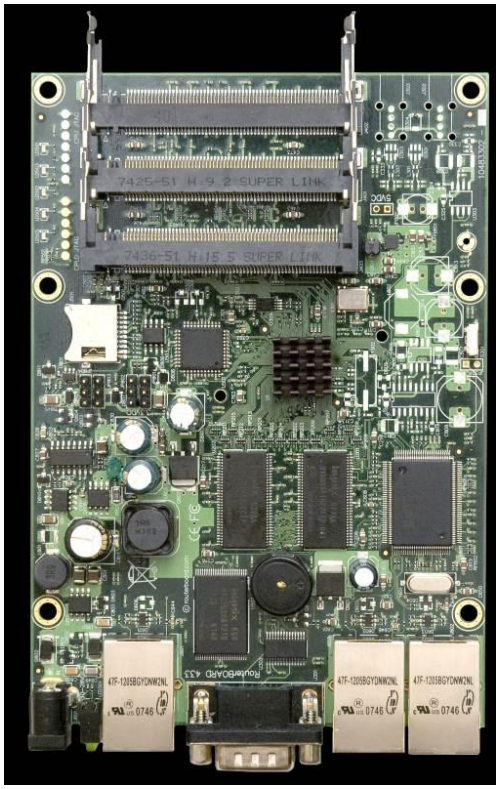
RB433

Detail	
Produkt	RB/433
CPU	300MHz
RAM	64MB
Architektura	MIPS-BE
LAN port	3
MiniPCI	3
Integrovaná karta wifi	0
USB	0
Paměťová karta	0
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Monitor napětí	Ne
RouterOS Licence	Level4



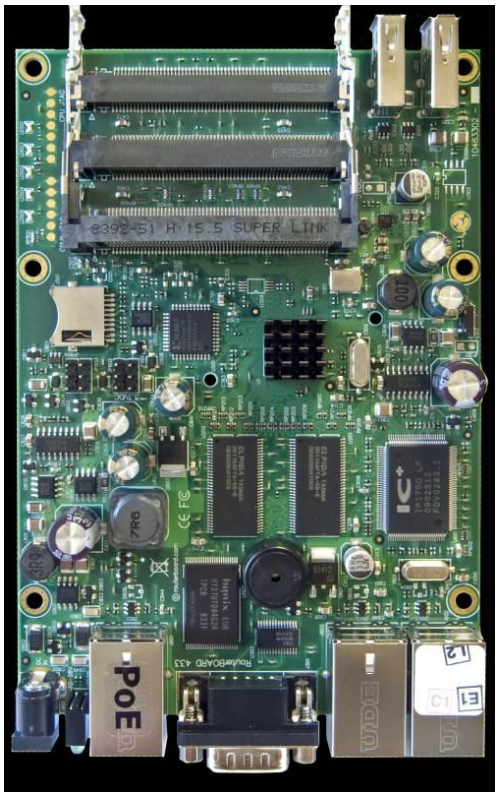
RB433AH

Detail	
Produkt	RB/433AH
CPU	680MHz
RAM	128MB
Architektura	MIPS-BE
LAN port	3
MiniPCI	3
Integrovaná karta wifi	0
USB	0
Paměťová karta	1
Paměťová karta typ	microSD
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Monitor napětí	Ano
RouterOS Licence	Level5



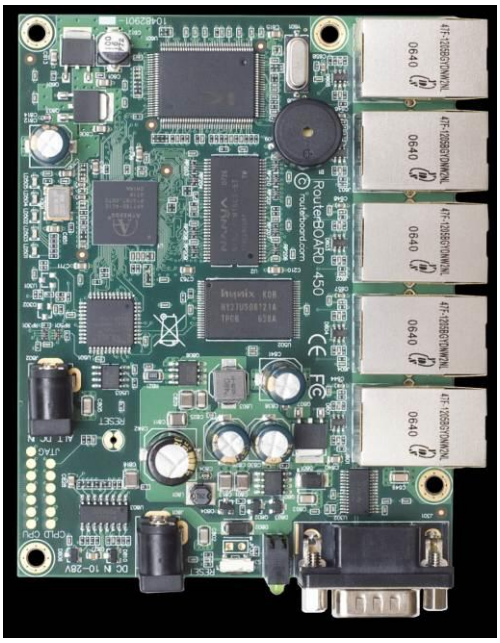
RB433UAH

Detail	
Produkt	RB/433UAH
CPU	680MHz
RAM	128MB
Architektura	MIPS-BE
LAN port	3
MiniPCI	3
Integrovaná karta wifi	0
USB	2
Paměťová karta	1
Paměťová karta typ	microSD
Napájecí konektor	10..28V
Podpora 802.3af	Ne
PoE	10..28V
Monitor napětí	Ano
Rozměr	105x150mm, 140g
RouterOS Licence	Level5



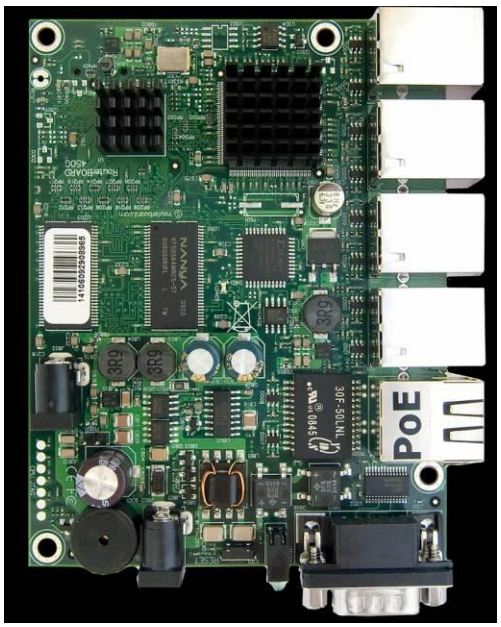
RB450

Detail	
Produkt	RB/450
CPU	300MHz
RAM	32MB
Architektura	MIPS-BE
LAN port	5
MiniPCI	0
Integrovaná karta wifi	0
USB	0
Paměťová karta	0
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Monitor napětí	Ne
RouterOS Licence	Level5



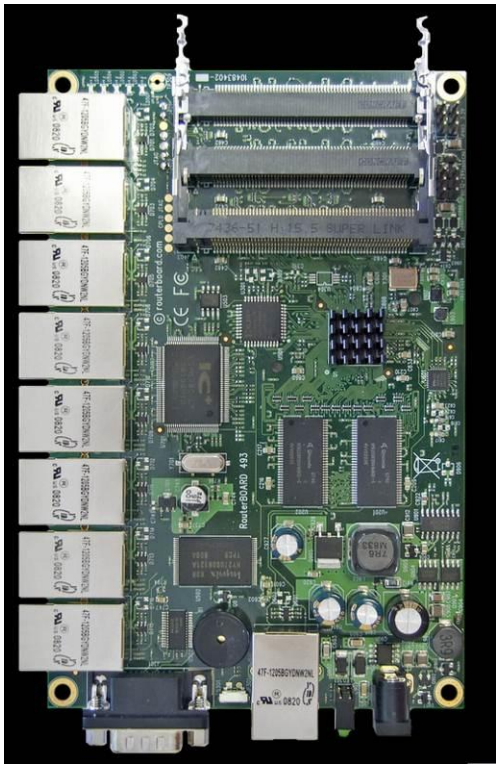
RB450G

Detail	
Produkt	RB/450G
CPU	680MHz
RAM	256MB
Architektura	MIPS-BE
LAN port	5
Gigabit	Ano
MiniPCI	0
Integrovaná karta wifi	0
USB	0
Paměťová karta	1
Paměťová karta typ	microSD
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Monitor napětí	Ano
RouterOS Licence	Level5



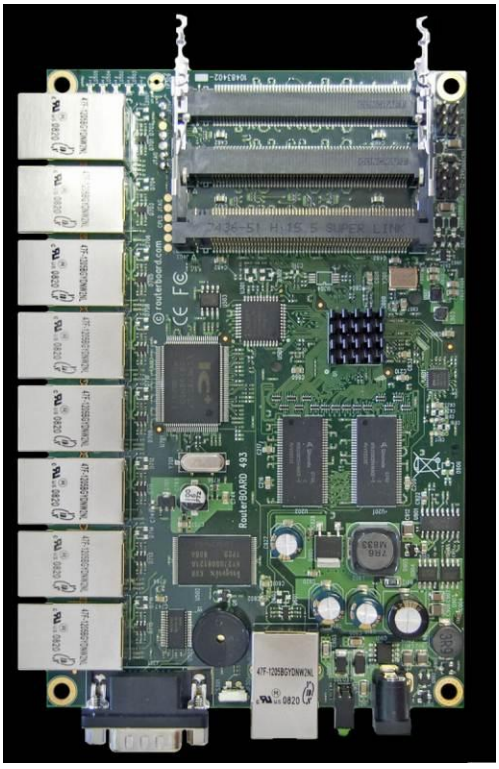
RB493

Detail	
Produkt	RB/493
CPU speed	300MHz
RAM	64MB
Architektura	MIPS-BE
LAN port	9
MiniPCI	3
Integrovaná karta wifi	0
USB	0
Paměťová karta	0
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Monitor napětí	Ne
RouterOS Licence	Level4



RB493AH

Detail	
Produkt	RB/493AH
CPU speed	680MHz
RAM	128MB
Architektura	MIPS-BE
LAN port	9
MiniPCI	3
Integrovaná karta wifi	0
USB	0
Paměťová karta	0
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Monitor napětí	Ne
RouterOS Licence	Level5

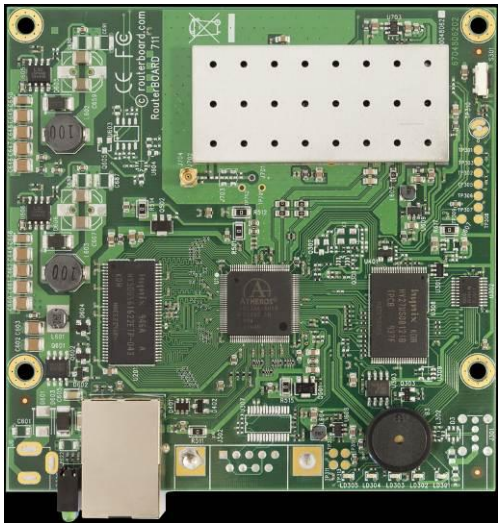


RB493G

Detail	
Produkt	RB/493G
CPU	680Mhz
RAM	256MB
Architektura	MIPS-BE
LAN port	9
Gigabit	Ano
MiniPCI	3
USB	1
Napájecí konektor	10..28v
Podpora 802.3af	Ne
PoE	10..28V DC
Monitor napětí	Ano
Senzor teploty	Ano
Rozměry	105mm x 160mm
RouterOS Licence	L5

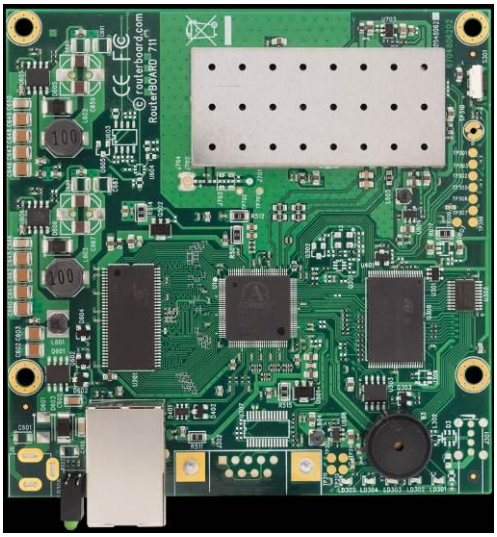
RB711-5Hn-M

Detail	
Produkt	RB711-5Hn-M
CPU	400Mhz
RAM	32MB
Architektura	MIPS-BE
LAN port	1
MiniPCI	0
Integrovaná karta wifi	1, MMCX
Wifi standard	802.11an
USB	0
Paměťová karta	0
Napájecí konektor	Ne
Podpora 802.3af	0
PoE	10..28V DC
Monitor napětí	0
RouterOS Licence	L3



RB711-5Hn-U

Detail	
Produkt	RB711-5Hn-U
CPU	400Mhz
RAM	32MB
Architektura	MIPS-BE
LAN port	1
Gigabit	0
MiniPCI	0
Integrovaná karta wifi	1, uFl
Wifi standard	802.11an
USB	0
Paměťová karta	0
Napájecí konektor	Ne
PoE	10..28V DC
RouterOS Licence	L3

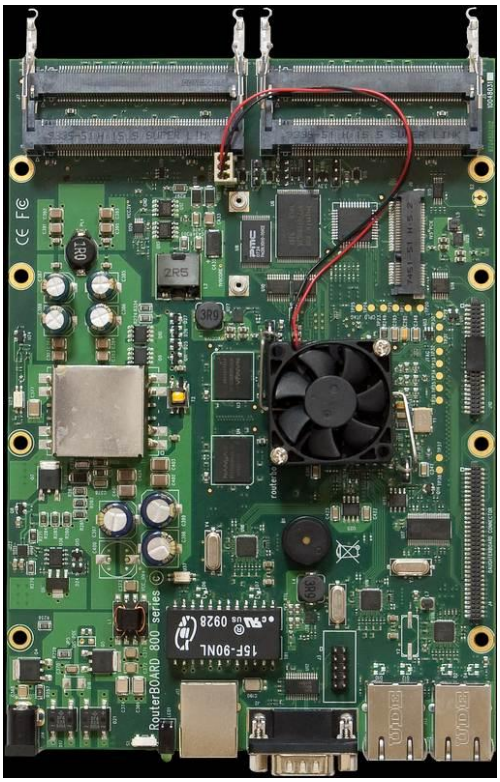


RB711A-5Hn-M

Detail	
Produkt	RB711A-5Hn-M
CPU	400Mhz
RAM	64MB
Architektura	MIPS-BE
LAN port	1
Gigabit	0
MiniPCI	0
Integrovaná karta wifi	1, MMCX
Wifi standards	802.11an
USB	0
Paměťová karta	0
Napájecí konektor	Ne
Podpora 802.3af	0
PoE	10..28V DC
Monitor napětí	0
RouterOS Licence	L4

RB800

Detail	
Produkt	RB/800
CPU	800MHz
RAM	256MB
Architektura	PPC
LAN port	3
Gigabit	Ano
MiniPCI	4
Integrovaná karta wifi	0
USB	0
Paměťová karta	1
Paměťová karta typ	CF
Napájecí konektor	10-56V DC
Podpora 802.3af	Ano
PoE	36-56V DC
Monitor napětí	Ano
Rozměry	14cmx20cm
RouterOS Licence	Level 6



SXT5HnD

Detail	
Produkt	RB/SXT
CPU	400MHz
Vysílací výkon	26dBm
Zisk antény	16dBi (+/- 2)
Architektura	MIPS-BE
LAN port	1
Gigabit	Ne
MiniPCI	0
Integrovaná karta wifi	Ano
Wifi standard	802.11 a/n
USB	1
Napájecí konektor	Ne
Podpora 802.3af	Ne
PoE	Ano
RouterOS Licence	Level3

RB250GS

Detail	
Produkt	RB/250GS
Architektura	RISC
LAN port	5
Gigabit	Ano
MiniPCI	0
miniPCI-e	0
Integrovaná karta wifi	0
USB	0
Paměťová karta	0
Napájecí konektor	9-28V DC
PoE	Ano
Rozměry	113x89x28mm

RB750

Detail	
Produkt	RB/750
RAM	32MB
Architektura	MIPS-BE
LAN port	5
Gigabit	Ne
MiniPCI	0
Integrovaná karta wifi	0
USB	0
Paměťová karta	0
Napájecí konektor	10-28V
Podpora 802.3af	Ne
PoE	10-28V
Monitor napětí	Ne
RouterOS Licence	Level4

RBN750G

Detail	
Produkt	RB/750G
CPU	680MHz
RAM	32MB
Architektura	MIPS-BE
LAN port	5
Gigabit	Ano
MiniPCI	0
Integrovaná karta wifi	0
USB	0
Paměťová karta	0
Napájecí konektor	9-28V
Podpora 802.3af	No
PoE	9-28V
Monitor napětí	No
Rozměry	113x89x28mm
RouterOS Licence	Level4

RB1100

Detail	
Produkt	RB/1100
CPU	800MHz
RAM	512MB
Architektura	PPC
LAN port	13
Gigabit	Ano
MiniPCI	0
miniPCI-e	0
Integrovaná karta wifi	0
Paměťová karta	microSD
Napájecí konektor	12-24VDC
PoE	12-24VDC
Rozměry	1U skříň: 45x75x440mm
RouterOS Licence	Level6