VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE

Brno, 2018 Pavel Zavadil

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ
DEPARTMENT OF TELECOMMUNICATIONS

MANAGEMENT SÍTĚ S PRVKY MIKROTIK
NETWORK MANAGEMENT USING MIKROTIK

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — **Pavel Zavadil**
AUTHOR

VEDOUCÍ PRÁCE — **doc. Ing. Jaroslav Koton, Ph.D.**
SUPERVISOR

BRNO 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní obor **Teleinformatika**
Ústav telekomunikací

Student: Pavel Zavadil ***ID:*** 133118
Ročník: 3 ***Akademický rok:*** 2017/18

NÁZEV TÉMATU:

Management sítě s prvky Mikrotik

POKYNY PRO VYPRACOVÁNÍ:

Popište stávající nástroje pro automatické hodnocení stavu sítě, detekci výpadků, vytváření záložních cest atd. Použití takových mechanismů aplikujte na síťové prvky Mikrotik při realizaci LAN sítí. Navrhněte scénáře a události při provozu sítě a tyto prakticky ověřte s cílem zachování komunikace mezi vybranými koncovými stanicemi, přičemž využijte i pokročilého nastavení síťových prvků skripty.

DOPORUČENÁ LITERATURA:

[1] A.S. Tanenbaum, D.J. Wetherall: Computer networks, Pearson, 2010, ISBN: 978-0132126953.

[2] Scripts, Mikrotik documentation [online]. 2016 [cit. 2016-09-12]. Dostupné z: http://wiki.mikrotik.com/wiki/Scripts

Termín zadání: 18.6.2018 ***Termín odevzdání:*** 15.8.2018

Vedoucí práce: doc. Ing. Jaroslav Koton, Ph.D.
Konzultant:

prof. Ing. Jiří Mišurec, CSc.
předseda oborové rady

ABSTRAKT

Cílem této práce je popis nástrojů hodnocení stavu sítě, dále popis síťových komponent Litevské firmy Mikrotik a ukázka scénářů výpadku sítě s její následnou rekonfigurací pro zachování komunikace.

KLÍČOVÁ SLOVA

Mikrotik, Wifi, LAN, WinBox, routování, RouterBoard, RouterOS, skripty

ABSTRACT

The aim of this work is to describe the tools of evaluation of the state of the network, the description of network components of Lithuanian company Mikrotik and the demonstration of network failure scenarios with its subsequent reconfiguration to maintain communication.

KEYWORDS

Mikrotik, Wifi, LAN, WinBox, routing, RouterBoard, RouterOS, scripts

ZAVADIL, Pavel *Management sítě s prvky MikroTik*: bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2018. 49 s. Vedoucí práce byl doc. Ing. Jaroslav Koton, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Management sítě s prvky MikroTik“ jsem vypracoval(a) samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor(ka) uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil(a) autorská práva třetích osob, zejména jsem nezasáhl(a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(a) následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno ....

podpis autora(-ky)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu doc. Ing. Jaroslavu Kotonovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno ....

podpis autora(-ky)

Faculty of Electrical Engineering
and Communication
Brno University of Technology
Purkynova 118, CZ-61200 Brno
Czech Republic
http://www.six.feec.vutbr.cz

PODĚKOVÁNÍ

Výzkum popsaný v této bakalářské práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

Brno . . . . . .

. . . . . .
podpis autora(-ky)

EVROPSKÁ UNIE
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ
INVESTICE DO VAŠÍ BUDOUCNOSTI

OP Výzkum a vývoj
pro inovace

OBSAH

SEZNAM OBRÁZKŮ

SEZNAM VÝPISŮ

ÚVOD

Tato bakalářská práce se zabývá možnostmi hodnocení stavu sítě, diagnostikou, automatického vytvoření náhradních přenosových cest a rekonfigurací sítě za pomoci využití již fungujících prvků na technologii MikroTik.

V současné době vstupují datové sítě do běžného života většiny z nás. Když jen málo zúžíme pojem datové sítě, dostaneme jejího nejznámějšího zástupce - celosvětovou síť internet. Většina lidí, firem nebo rovnou států si již asi nedokáže představit život bez něj. Přes internet posíláme peníze, bavíme se s přáteli, ovládáme vzdáleně domácnosti, sledujeme televizi a tak dál. Možností je téměř nekonečno. Tím ale vznikají požadavky na kvalitu a stabilitu připojení k této síti.

Když pomineme samotné zabezpečení k této síti, stabilitu připojení ovlivňuje více faktorů. Jedním z nich je výpadek přenosové cesty. Ať už se jedná o externí vlivy nebo přímo poruchu zařízení, oprava je jen málokdy možná vzdáleně a doba výpadku je časově i finančně náročná. Proto je vhodné tyto přenosové cesty zálohovat. Záložní cesty většinou nemusí mít parametry hlavního spoje (rychlost, latence), na překlenutí doby opravy většinou stačí i méně kvalitní spoj.

1 SÍTĚ

Pojmem datová síť se rozumí jakákoliv síť, která přenáší digitální data. Tato práce je zaměřena na datové počítačové sítě. Ty se dělí na několik typů podle velikosti - PAN, LAN, MAN a WAN. PAN je typ nejmenší sítě, obvykle použitá pro propojení dvou zařízení pomocí technologie Bluetooth nebo infračerveného portu. LAN je síť spojující uzly v rámci jedné, nebo několika budov. MAN je sít v rámci města nebo velké rozlehlé firmy. WAN je síť v rámci státu, případně i kontinentu. [1]

1.1 Local Area Network

Názvem LAN neboli Local Area Network se označuje počítačová síť, kterou tvoří síťové prvky a koncové stanice (např. počítače, telefony atd.) umístěné na geograficky malé ploše. Obvykle se jako LAN označují sítě v rodinných domech, firmách a školách. Prakticky je to poslední podsíť umístěná nejblíž ke koncovým zařízením. LAN síť je obvykle tvořena kombinací jednoho až tří typů fyzických rozhraní:

- metalické – propojení pomocí strukturované kabeláže,
- bezdrátové – technologie WiFi používaná v nelicencovaném pásmu 2,4GHz, 5GHz,
- optické – pomocí optických kabelů.

Při návrhu LAN sítě je třeba brát v úvahu spoustu kritérií. Jedním z nejdůležitějších aspektů u sítě je zajistit dostupnost široké škále uživatelů - u kabelových vedení je tento aspekt zcela jasný. Avšak i u bezdrátových sítí je potřeba myslet dopředu a dobře naplánovat umístění všech AP pro dostupnost dostatečně silného signálu. Dalším důležitým bodem je hardware sítě. Při návrhu je potřeba brát v potaz na každoročně se zvyšující přenos dat i na potřebou nízké latence. Podle toho bychom měli vybírat daný hardware a také dané přenosové medium. Je dobré myslet dopředu, jak se může síť vyvíjet a podle toho správně dimenzovat daný hardware. To vše proto, aby se předešlo jeho časté výměně už při menší změně v síti.

Ve firemní sféře nebo ve školách či úřadech již nemá smysl používat UTP kabel CAT 5, protože jeho vlastnosti nejsou vhodné pro gigabitový přenos. Vhodnější je použít novější CAT 5e nebo rovnou CAT 6 který již umožňuje větší šířku pásma. To samé platí pro bezdrátové přenosy – pokud je třeba použít pro přenos mezi budovami technologii WiFi, pásmo 2,4GHz je především ve městech tak zarušené, že se k tomuto účelu vůbec nehodí. Mnohem lepší je použít pásmo 5GHz, ale i to pomalu začíná být přeplněné. Při návrhu fyzického rozložení je dobré počítat i se záložními spoji pro případ výpadku, nebo poruchy. [3], [1]

2 NÁSTROJE PRO SPRÁVU SÍTĚ

Pro správu sítě lze nástroje rozdělit podle místa, ze kterého se spouští a podle informací, které tyto nástroje poskytnou.

2.1 Nástroje obsažené v zařízení

Většina síťových zařízení obsahuje nástroje pro diagnostiku sítě. Množství těchto nástrojů se odvíjí podle kvality a ceny zařízení. Levné domácí routery obvykle obsahují jen minimum interních nástrojů, naopak zařízení od renomovaných firem (jako je Cisco nebo MikroTik) nabízejí celou škálu těchto nástrojů. Přehled běžných nástrojů:

- ukládání log souboru na vzdálený server - díky této funkci je možné přečíst logovací soubor i z jiného místa, než jen pouze z daného zařízení (toto je vhodné v případě kompletního výpadku zařízení). V logovacím souboru se může objevit počáteční degradace přístroje, pokud například napájecí zdroj začne ztrácet své parametry (degradace zdroje),
- nástroje pracující v reálném čase - jsou to nástroje jako ping a tracert - viz dále,
- nástroje pro test linky - tyto nástroje se obvykle vyskytují v xDSL a kabelových modemech,
- notifikační nástroje - některá zařízení dokáží odeslat informační email, nebo SMS na zadané číslo s informacemi dle výběru,
- dohledové nástroje - tyto nástroje mohou kontrolovat dostupnost zadané IP adresy. V případě její nedostupnosti pak provedou určenou akci. Často je to změna trasy a odeslání informace o chybě,
- nástroje pro změnu směrování - tuto práci obvykle zastanou routovací protokoly, které v případě výpadku cesty přeroutují síť na jinou.

Pokud se nejedná o výpadek konektivity k zařízení, nebo výpadek napájení, samotné zařízení je prvek který výpadek registruje jako první.

2.2 Webové nástroje

Pro monitorování a nastavení základních parametrů síťových prvků se většinou používají standardizované protokoly SNMP (Simple Network Management Protocol) nebo WMI (Windows Management Instrumentation). Ty jsou podporovány monitorovacími programy. SNMP je jednoduchý protokol, využívaný ke zjišťování stavu síťových komponent a k nastavování hodnot na těchto komponentách. Podporuje

jej řada zařízení jako aktivní síťové prvky, tiskárny, čidla a podobně. Hodnoty lze jednoduše zaznamenávat a pak dále zpracovávat za pomoci velké škály různých monitorovacích programů. Pro svou správnou funkci vyžaduje dvě části - první je správce a druhá je agent. Protokol pracuje v dvou režimech činnosti:

- správce si sám žádá agenta o zjišťované hodnoty,
- agent sám v definovaných situacích zasílá správci dané hodnoty.

SNMP protokol používá pro svou komunikaci protokol UDP, díky kterému je velice rychlý. Standardně se pro komunikaci využívají porty 161 (na straně agenta) a 162 (na straně serveru). Při vyslání dotazu využívá klient dynamický port a na něj se mu poté vrací odpověď. V praxi je pro každý z dotazů použit jiný dynamický port.

Podobně jako SNMP i WMI dokáže sbírat data o zařízeních a nastavovat je. Navíc umí spravovat vzdálené počítače. Ale na rozdíl od SNMP je WMI proprietární protokol firmy Microsoft s uzavřeným kódem. Služba zahrnuje úložiště objektů, které jsou databází definic objektů. [6] [7]

2.2.1 Nagios

Nagios je flexibilní open source nástroj vyvíjený již od roku 1996, který umožňuje detekovat a případně opravit problémy po jejich detekci. Tímto lze eliminovat výpadky dříve, než se dostanou ke koncovým uživatelům. Pomocí nástroje Nagios lze:

- monitorovat celou síťovou infrastrukturu,
- automaticky opravit problémy, když jsou detekovány,
- posílat upozornění pomocí e-mailu nebo SMS,
- logování výpadků, různých událostí v síti, výstrah,
- další funkce.

Nagios jako takový se skládá z hlavního programu a dále k němu závislých externích pluginů, takže je velmi dobře rozšířitelný o další funkce. Monitorování probíhá v cyklech. Typicky se monitorují následující parametry sítě:

- odezva (ping),
- množství přenesených dat,
- poštovní služby,
- webové služby,
- využití hardwaru,
- logování do systému.

Potřebné informace nástroj získává ze sítě za pomoci výše uvedených protokolů. [8]

2.2.2 Icinga

Icinga je fork (alternativně vyvíjená větev) nástroje Nagios, takže konfigurační soubory a pluginy je možné využívat v obou nástrojích. Rozdíl nástrojů je patrný především ve vzhledu uživatelského rozhraní. Jádro tohoto systému je vyvíjeno paralelně s jádrem systému Nagios. Proto se také spousta záplat z Nagiosu aplikuje i na Icingu a naopak záplaty vyvinuté v Icinga jsou backportovány na Nagios. [9]

2.2.3 DuDe

Nástroj přímo vyvíjený společností MikroTik pro svá zařízení. Tento nástroj automaticky detekuje zařízení pomocí protokolu MNDP (MikroTik Neighbour Discovery Protokol), jež vysílá UDP Broadcast na portu 5678. Nástroj si poté sám uspořádá a zařadí detekovaná zařízení. Po oskenování sítě následně umožňuje rozkreslit rozložení sítě a poté monitorovat celou síť. Dále zvládá zasílat upozornění v případě výskytu problémů. Jedná se o volně šiřitelný nástroj, určený pouze pro zařízení MikroTik. Seznam hlavních funkcí nástroje DuDe je následující:

- automatické nalezení sítě a jejího rozvržení,
- zjištění typu nebo značky zařízení,
- zahrnuje SVG ikony pro zařízení a podporuje vlastní ikony a pozadí,
- umožňuje kreslit vlastní mapy a přidávat vlastní zařízení,
- podporuje SNMP, ICMP, DNS a monitorování TCP pro zařízení, které jej podporují,
- individuální monitoring a grafy Link využití,
- přímý přístup k nástrojům dálkového ovládání pro správu zařízení.

Program je spustitelný v Linux Wine prostředí, MacOS Darwine a Windows. [10]

2.2.4 Cacti

Další z řady monitorovacích programů, který je se zaměřuje převážně na grafy. Nástroj má širokou škálu možností monitorování. Data je možné sbírat pomocí standardizovaných protokolů nebo vlastních skriptů. Program má přehledné grafické uživatelské rozhraní, podporuje pluginy a šablony vzhledu. Díky široké uživatelské podpoře existuje mnoho různých rozšíření. Mezi podporované platformy patří:

- Unix/Linux systémy,
- systémy Microsoft Windows,
- Novell,
- Cisco IOS,
- MikroTik.

Další vlastnosti Cacti:

- udržuje seznam monitorovaných zařízení, jejich dostupnost a umí upozorňovat na jejich výpadky elektronickou poštou,
- umožňuje vizualizaci dat do do grafů a sestav s nimiž je možné dále pracovat,
- lze využít import a export šablon (xml), export naměřených dat,
- pomocí Cacti si lze snadno vytvářet vlastní zdroje dat, šablony grafů, šablony celých zařízení,
- umožňuje nastavit práva pro uživatele, třeba jen pro prohlížení některých grafů,
- pomocí pluginů lze přidat mnoho dalších funkcí.

[11]

2.3 Nástroje pracující v reálném čase

Pro diagnostiku sítě v reálném čase se mohou využít programy většinou běžně obsažené v operačním systému. Pomocí nich lze ve většině případů určit příčinu výpadku či nedostupnosti sítě. Následující programy jsou obsaženy v operačním systému Windows (v jiných operačních systémech jsou obvykle obsaženy programy poskytující obdobnou funkcionalitu).

- Příkaz ping - nástroj funguje tak, že vyšle dotaz na zadanou adresu a počká na odpověď. Výstupem je čas odezvy od testované adresy. Syntaxe příkazu je ping <název nebo adresa testovaného uzlu> a případně další parametry. Například ping 192.168.1.1 -t bude nepřetržitě testovat zadanou adresu.
- Příkaz ipconfig - nástroj slouží k zobrazení aktuálního nastavení síťových adaptérů. Obvykle se používá pro kontrolu nastavené nebo přiřazené IP adresy, masky, brány, jmenného serveru a MAC adresy. Příklad ipconfig -all zobrazí podrobné nastavení všech adaptérů.
- Příkaz tracert - slouží k výpisu trasy mezi použitým zařízením a cílovým uzlem. Ve výpisu je možné také identifikovat uzly jež blokují nebo filtrují ICMP zprávy. Pomocí tohoto nástroje lze tedy odhalit nefunkční zařízení na trase v síti. Výstupem je IP adresa uzlu případně i doménové jméno. Pokud uzel negeneruje nebo filtruje ICMP odezvu, program zobrazí pouze hvězdičky. Příklad tracert 77.75.79.39 zobrazí trasu mezi zařízením, na kterém běží a serverem stránek www.seznam.cz.
- Příkaz nslookup - tento nástroj se dotazuje DNS serveru na záznamy o IP adrese. Pomocí něj lze otestovat, zda-li je DNS server správně nastavený a funguje. Příklad nslookup www.seznam.cz vrátí IP adresu 77.75.79.39 a IP adresu použitého DNS serveru. [13] [14]

3 ZAŘÍZENÍ MIKROTIK

Firma MikroTik vznikla v Lotyšsku v roce 1996, kde začala vyvíjet směrovače a bezdrátové systémy pro poskytovatele internetového připojení. V dnešní době má na trhu vybudované poměrně silné jméno, protože nabízí kvalitní výkonné zařízení s velkou podporou (jak firemní, tak komunitní) za příznivou cenu. [2], [5]

3.1 Systém RouterOS

Systém vznikl v roce 1997 po zkušenostech s možnostmi tehdejšího PC hardwaru v průmyslových aplikacích a směrovacích systémech. Na základě těchto zkušeností vytvořila firma MikroTik svůj vlastní universální operační systém, který je postaven na Linuxovém jádře. Jedná se o je stabilní, flexibilní a dobře ovladatelný operační systém.

RouterOS lze ovládat vícero způsoby - Telnet, SSH, webové rozhraní a asi nejvyužívanější program pro platformu Windows WinBox. Ten nabízí přehledně zpracovaném grafickém rozhraní a velké množství nastavení. Pomocí nástroje WinBox lze provést kompletní nastavení přístroje. [2], [5]

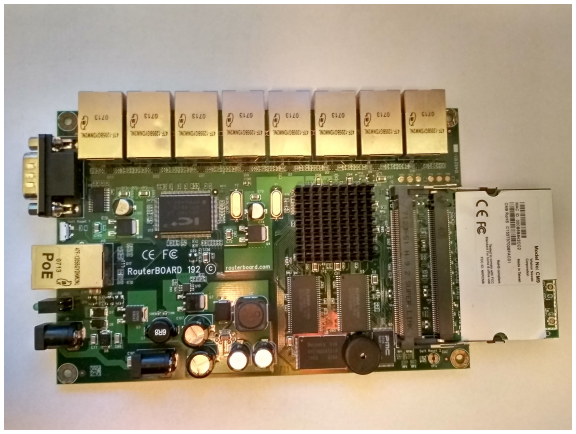
3.2 Zařízení RouterBOARD

V roce 2002 se MikroTik zaměřil také na vývoj vlastního hardwaru, který nazval jednoduše RouterBOARD. Výrobce nabízí několik řad zařízení, která se liší cenou, výbavou a cílovou skupinou uživatelů. Tato zařízení dokáží fungovat dlouhé roky bez vnějšího zásahu, nejslabším článkem bývá napájecí zdroj a elektrolytické kondenzátory umístěné přímo na desce RouterBoardu (vlastní zkušenost). Výčet známějších řad:

- RB1xx - nejstarší modelová řada. Verze určené pro bezdrátovou síť už se nevyrábí, víceportové verze bez primární podpory WiFi sítí se objevují jako switche,
- RB3xx - starší řada RouterBoardů se třemi ethernet porty a třemi miniPCI sloty,
- RB4xx - novější řada s jedním ethernet portem a jedním miniPCI slotem obsahující 300MHz procesor a 32MB pamětí RAM,
- RB5xx - slot pro CF karty, 2 miniPCI sloty, 3 ethernet porty, 400MHz procesor, 64MB pamětí RAM, podpora rozšiřující desky,
- RB6xx - dva sloty na CF karty, 3 gigabitové ethernet porty, 400MHz procesor, 64MB pamětí RAM,

- RB9xx - řada s integrovanou bezdrátovou kartou a jedním ethernet portem (v některých verzích gigabitovým). Tato řada je také v modelové řadě SXT,
- RB10xx - model bez podpory WiFi, zato se silným procesorem a 4 gigabitovými ethernet porty. Používá se jako vysokorychlostní router.

[15]



Obr. 3.1: Ukázka RouterBOARDu RB192

V příloze se nachází ukázka skriptu pro nastavení čistého RouterBoardu jako klienta. Přímo ve skriptu jsou použity komentáře pro jeho lepší čitelnost.[5]:

3.3 Nástroje pro automatizaci správy sítě

Operační systém RouterOS obsahuje velké množství nástrojů pro konfiguraci, testování a údržbu sítě. Některé z nich, použité v rámci této bakalářské práce, jsou zde představeny:

3.3.1 Nástroj Netwatch

Netwatch je nástroj, který v zadaném intervalu kontroluje dostupnost zadané IP adresy. Pokud se tato IP adresa stane na zadanou dobu nedostupná, spustí skript pro nedostupnost IP adresy. Při opětovné dostupnosti spustí skript pro dostupnost.

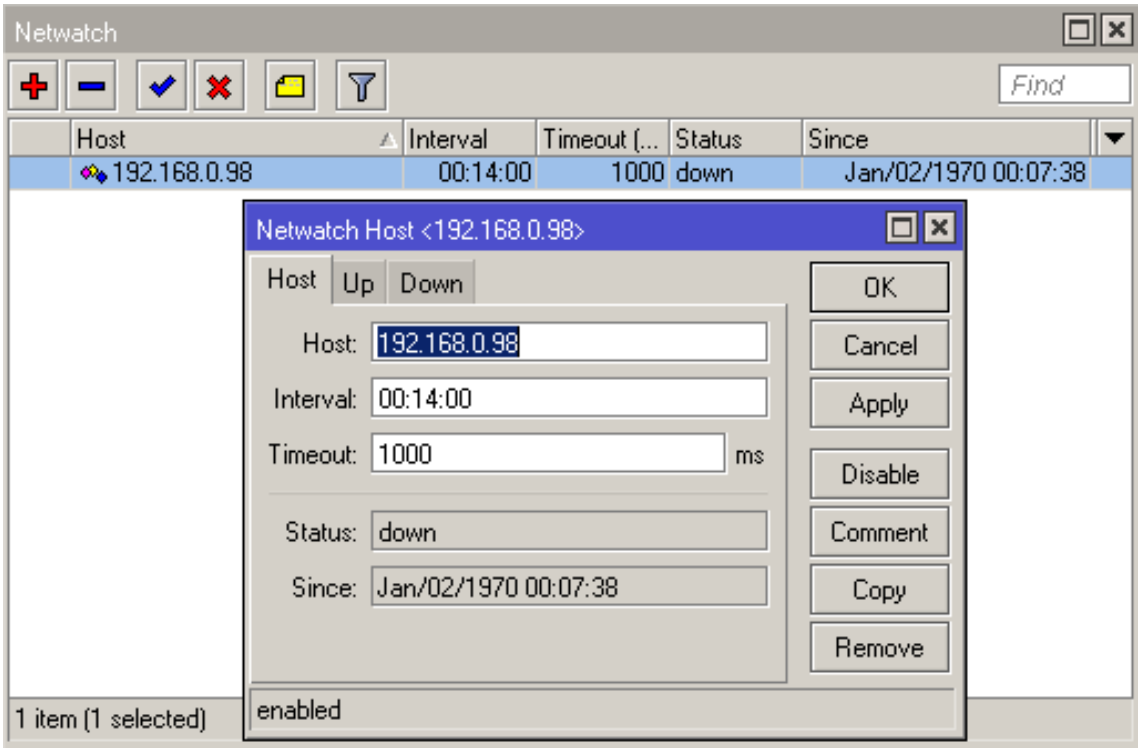
Skript lze zapsat přímo do pole pro skript a nebo se na něj lze odkázat do databáze vlastních skriptů uložených v RouterBoardu (na obrázku 3.2 je ukázka nastavení kontrolované IP adresy, intervalu a času výpadku).

Z konzole je možné zadat skript následovně :

Výpis 3.1: Skript pro Netwatch

```
tool netwatch add host =192.168.0.98 interval =00:00:14
timeout =0.5 down-script = script_5g_down
up-script = script_5g_up
```

[2]



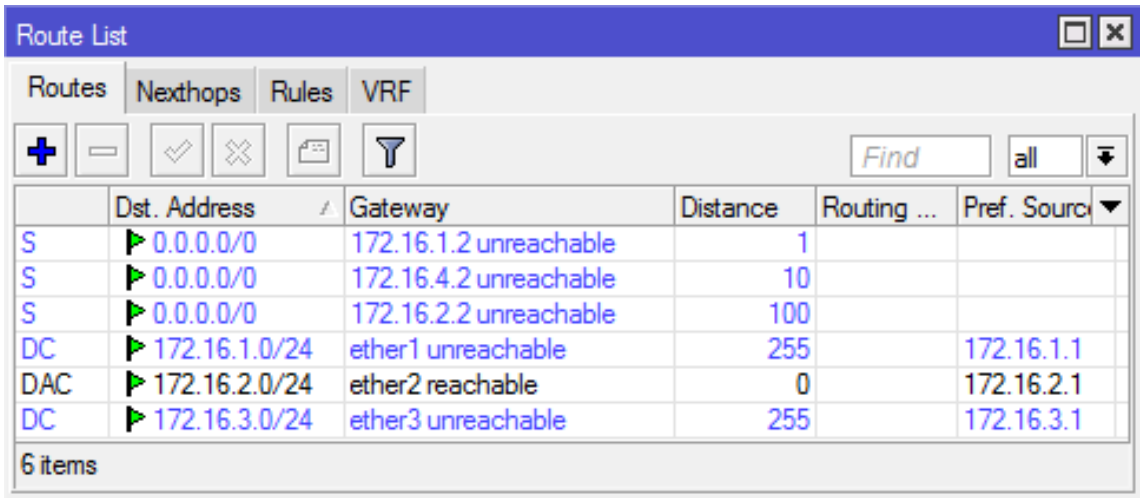
Obr. 3.2: Služba Netwatch

3.3.2 Statické směrování

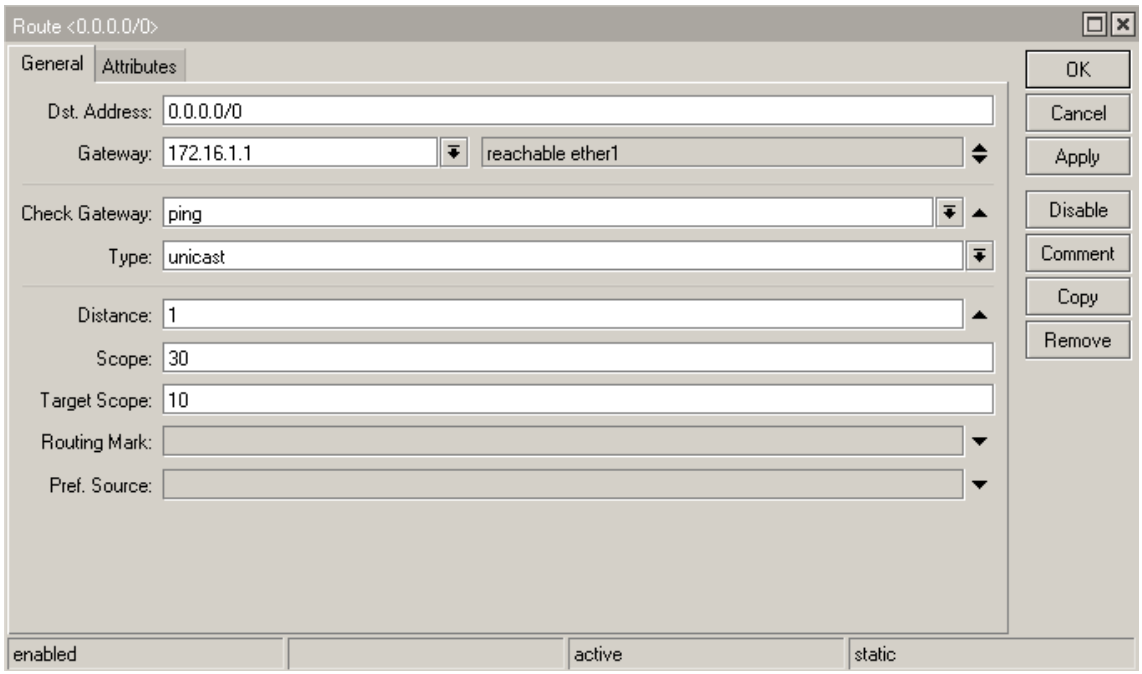
Manuální zadání směrování se provádí v okně Route List na záložce Route (obr. 3.3). V okně zadání/nastavení (obr. 3.4) zadáváme cílovou adresu (Dst. Address), kdy se pro všechny zdrojové adresy použije adresa 0.0.0.0/0, dále bránu (Gateway), na kterou se mají cílové adresy odeslat a vzdálenost (Distance), podle které RouterBoard určuje prioritu cesty. Čím nižší číslo, tím vyšší priorita. Nakonec je vhodné nastavit kontrolu dostupnosti cesty (Check Gateway). [2]

3.3.3 Nástroj protokolu OSPF

OSPF protokol je představitel tzv. Link State protokolu. Každý směrovač se pokusí navázat spojení se sousedními směrovači pomocí tzv. Hello packetů a následně se

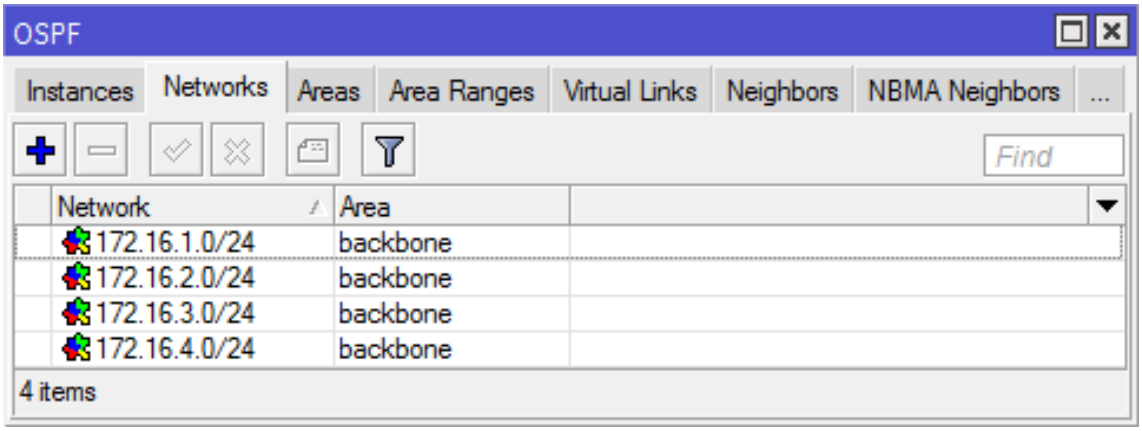


Obr. 3.3: Výpis směrovací tabulky

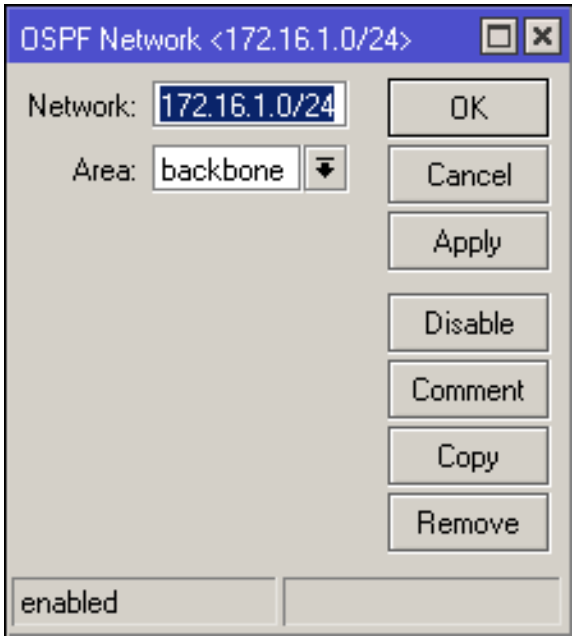


Obr. 3.4: Nastavení statického směrování

dohodnou na komunikaci. V té si směrovače předají mapu sítě. Ve výsledku pak každý směrovač obsahuje celou mapu sítě, nad kterou pak pomocí Shortest Path First (SPF) algoritmu provádí výpočty pro určení nejlepší cesty. Při těchto výpočtech se určí nejkratší cesty a smyčky. Metrika těchto cest je označovaná jako cena. Čím má cesta menší cenu, tím je více preferována. Na obrázku 3.5 je zobrazena karta nastavení sítí, přidání se provádí pomocí tlačítka + v dialogovém okně zobrazeném na obrázku 3.6. [4] [2]



Obr. 3.5: Zadané sítě protokolu OSPF

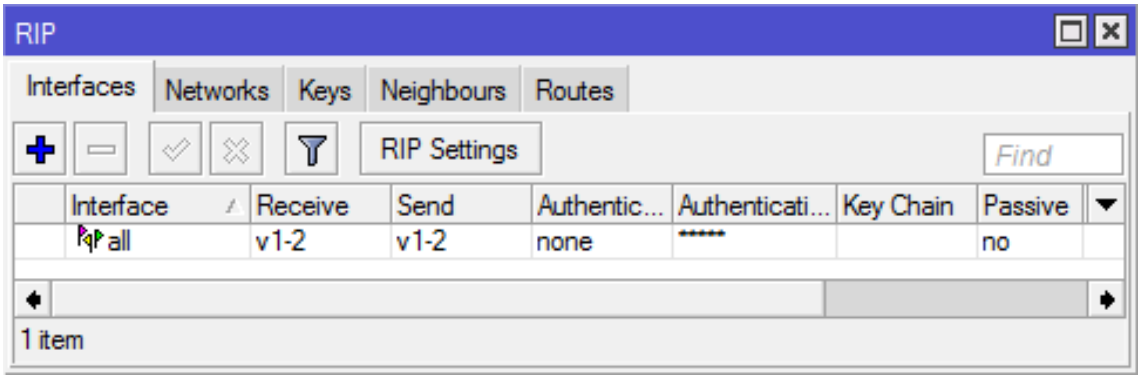


Obr. 3.6: Nastavení okolních sítí v OSPF

3.3.4 Nástroj protokolu RIP

Směrovací internetový protokol (Routing Internet Protocol - RIP) byl vyvinut v 80. letech a byl speciálně navržen pro zpracování přenosů v malých nebo středních sítích. Tento protokol spadá do kategorie typu distance-vector (vektor vzdálenosti), tedy určuje vhodnou trasu podle počtu skoků k cílové síti, nejvýše však 15 skoků. Toto omezení slouží jako ochrana proti smyčkám. Směrovače s aktivním protokolem RIP si posílají každých 30 sekund aktualizaci směrovací tabulky. Nevýhodou protokolu RIP je jeho dlouhá konvergence. [16]

Při nastavování protokolu RIP v systémech MikroTik je třeba vybrat na záložce interfaces (obr. 3.7) rozhraní, které má používat RIP, případně lze použít volba všech rozhraní (obr. 3.8). Vlastní nastavení sítí je na záložce Networks (obr. 3.9) pomocí tlačítka + v dialogovém okně (obr. 3.10). [2]



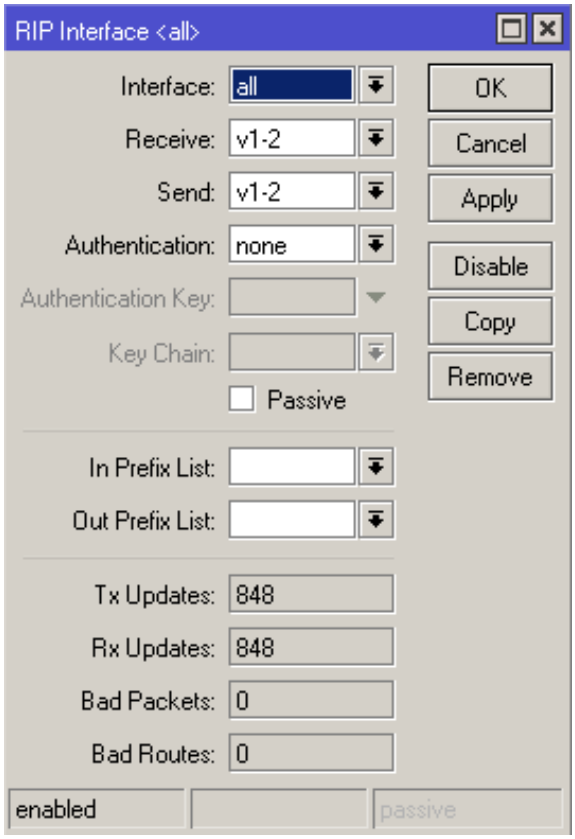
Obr. 3.7: Rozhraní používající RIP

3.3.5 Nástroj Email

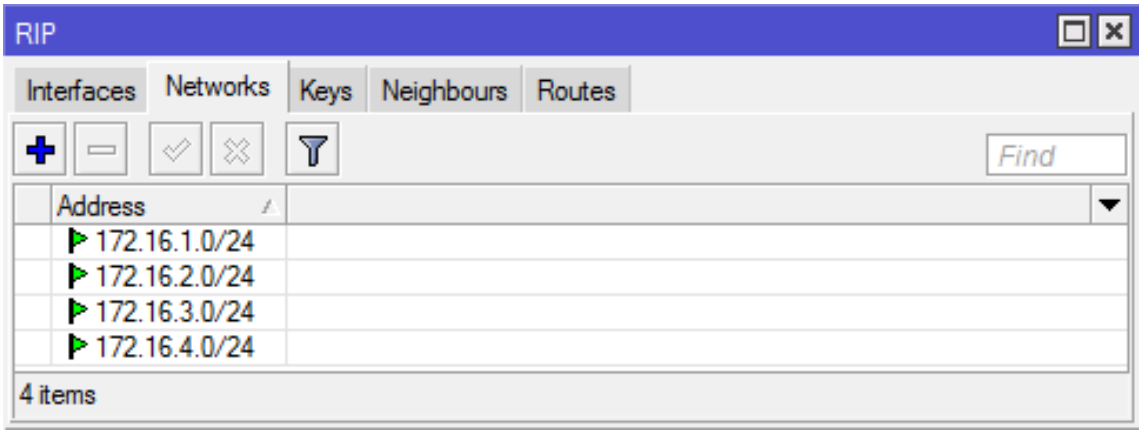
Nástroj email umožňuje odeslání emailu přímo z RouterBoardu (ukázka nastavení emailu obr. 3.11). Tato služba je vhodná pro informování správce sítě v případě nenadálé situace. Častěji se však využívá v příkazové řádce nebo jako součást skriptu. [2]

Výpis 3.2: Skript pro Email

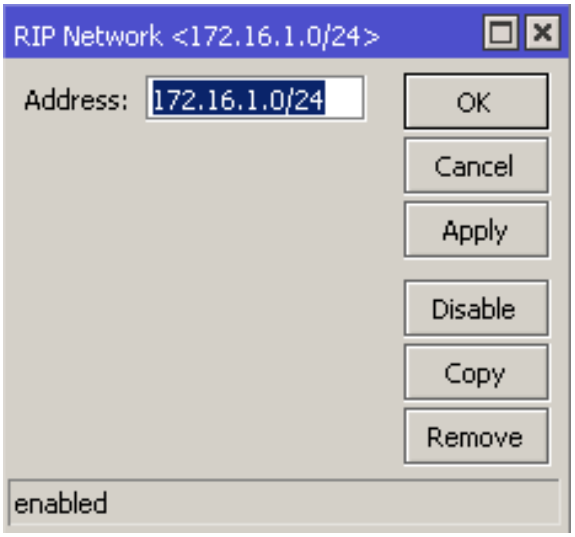
```
/tool e-mail send body="Casova␣znacka:␣($[/system␣clock␣get␣date␣]
$[/system␣clock␣get␣time␣])" from=check.mikrotik@gmail.com
password=cm_nG.com port=587 server=108.177.15.109
subject=network_status to=pzavak@gmail.com
user=check.mikrotik@gmail.com start-tls=yes
```



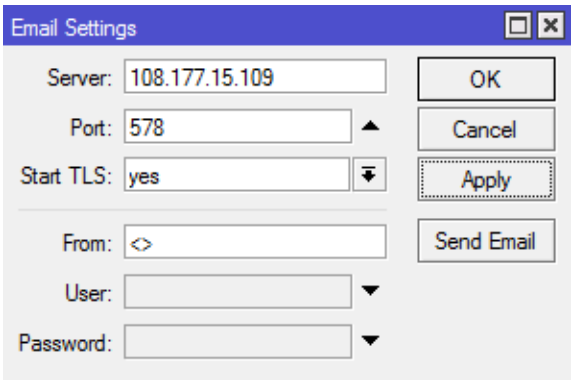
Obr. 3.8: Nastavení rozhraní pro RIP



Obr. 3.9: Zadané sítě protokolu RIP



Obr. 3.10: Nastavení okolních sítí v RIP



Obr. 3.11: Nástroj Email v Mikrotiku

4 SCÉNÁŘE A UDÁLOSTI PŘI PROVOZU

V této části budou prezentovány modelové situace z pozice menšího poskytovatele internetového připojení.

4.1 Síť se záložním spojem

První situace popisuje menší obec (cca 150 domů) se dvěma stanicemi z nichž každá pokrývá zhruba polovinu obce. K první je přivedena konektivita z páteřní sítě. Druhá stanice je k první připojena bezdrátovým pojítkem v nelicencovaném pásmu 5 GHz. V případě výpadku tohoto spoje skript automaticky přepne WiFi kartu na druhé stanici na příjem ze sekundárního spoje určeného pro klienty a zároveň odešle email s upozorněním, že nastala uvedená situace. Tím je zachována alespoň částečná konektivita.

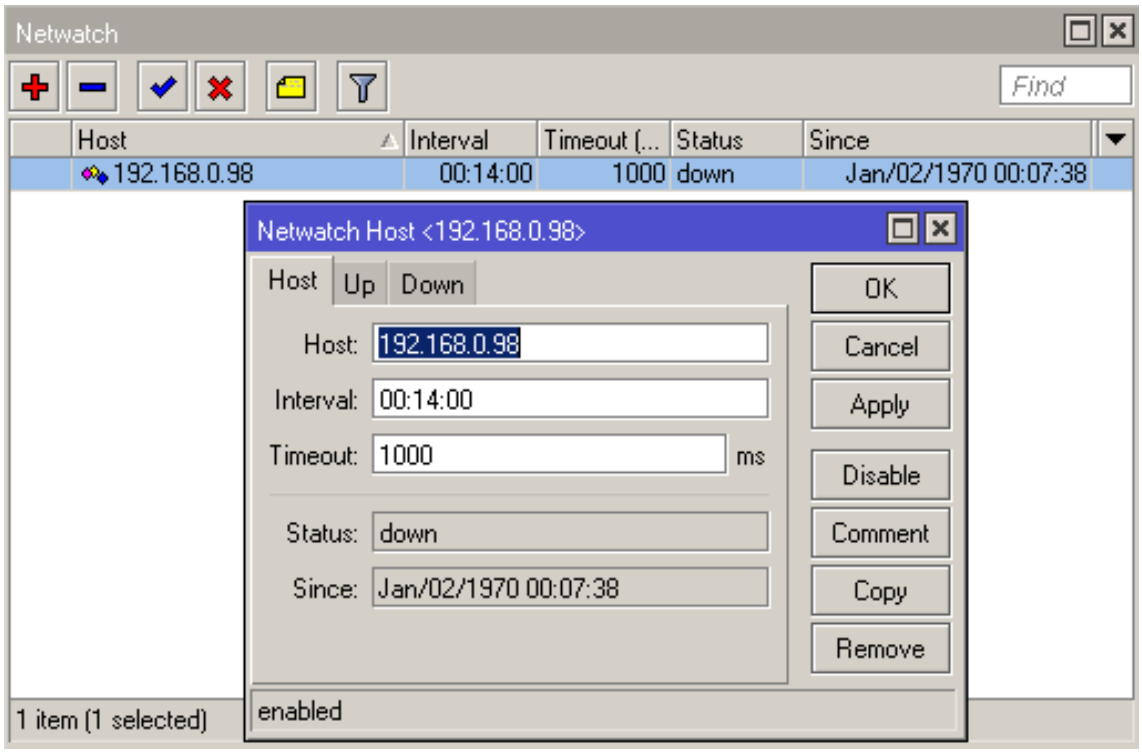
Scénář je tedy varianta zálohy bezdrátového spoje mezi dvěma sítěmi stejného rozsahu dalším přidaným bezdrátovým spojem. V případě výpadku primární cesty se spoj přepne na již existující záložní trasu, přes kterou funguje do doby, než dojde k obnově primární cesty. (obr. 4.3)

O samotný proces přepínání se stará služba Netwatch, která periodicky monitoruje zadanou adresu (v tomto případě protější stranu). V případě výpadku delším než je definovaná doba se spustí námi zadaný skript. Po obnově spojení je spuštěn jiný skript. Oba skripty jsou zadané přímo ve službě Netwatch, nebo mohou být uložené v databázi skriptů a ve službě Netwatch na ně stačí dát odkaz.

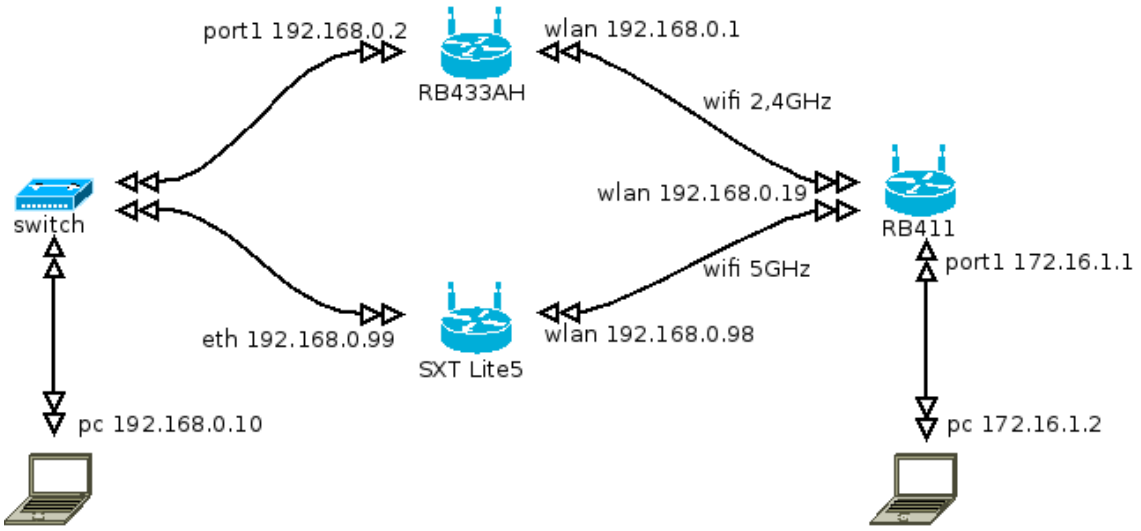
Služba Netwatch, spuštěná na RB411 kontroluje dostupnost IP adresy WLAN portu na zařízení SXT Lite. Mezi těmito dvěma zařízeními bylo bezdrátové spojení WiFi pracující na frekvenci 5 GHz. Při výpadku tohoto spoje služba Netwatch spustí skript (script 5g down), který změní parametry WiFi karty tak, aby se připojila na podružné AP první stanice na frekvenci 2,4 GHz. Při opětovné dostupnosti primárního spoje Netwatch spustil druhý skript (script 5g up), který přepnul WiFi kartu u RB411 zpět na 5 GHz spoj. Schéma zapojení je na obrázku 4.2 a diagram na obrázku 4.3. Oba skripty i skript pro nastavení Netwatche (script pro zprovozneni netwatch) jsou k nalezení v příloze A.

Při testování služby nastal výpadek, který ovlivnil koncové uživatele. Tento výpadek trval v prvním případě 10 až 20 sekund, druhém případě pak 10 sekund. Při každém spuštění skriptu byl odeslán informační email (viz obr. 4.4).

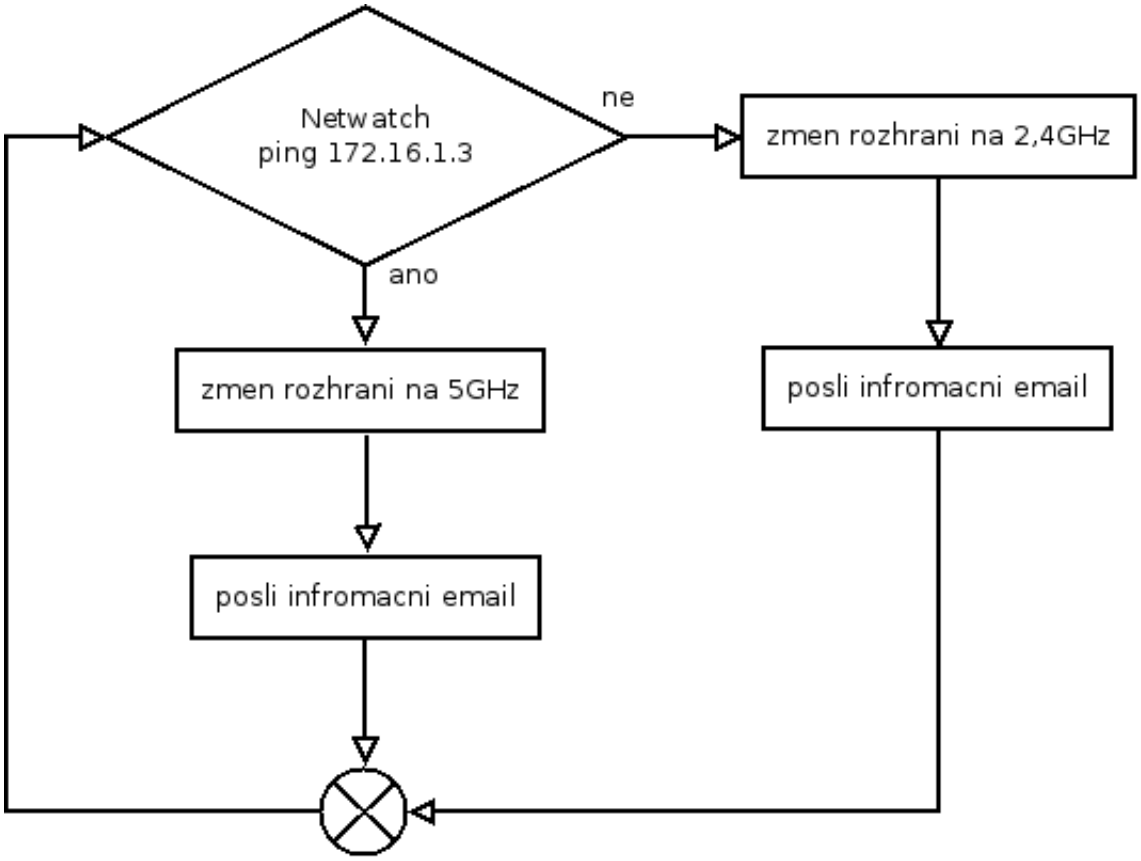
Při odladování bylo potřeba zvednout interval, kdy Netwatch testuje zadanou IP adresu, na delší dobu (konkrétně 14 sekund) pro případ opakujícího se krátkodobého výpadku spoje. Při spuštění skriptu, který přepínal mezi 2,4 GHz spojem a 5 GHz



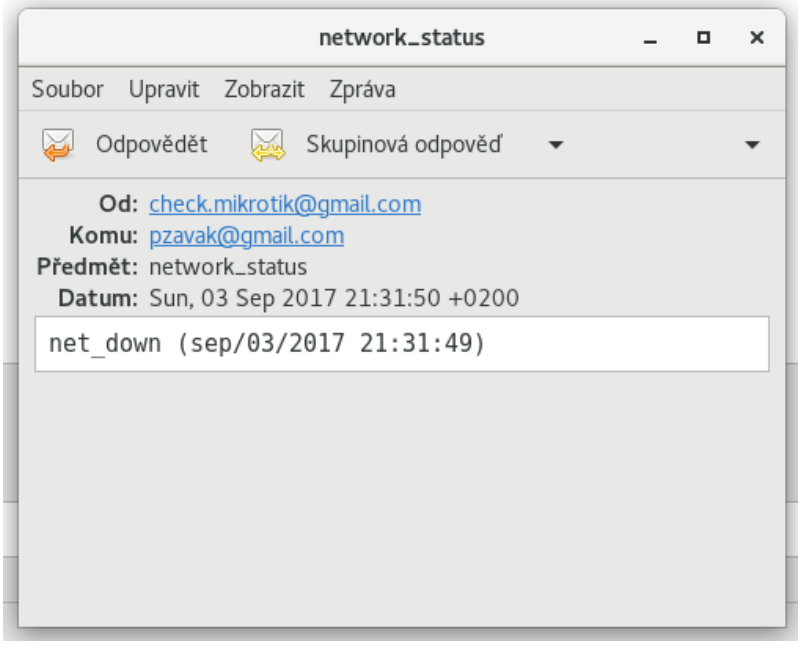
Obr. 4.1: Služba Netwatch



Obr. 4.2: Diagram sítě



Obr. 4.3: Vývojový diagram přepínání provozu



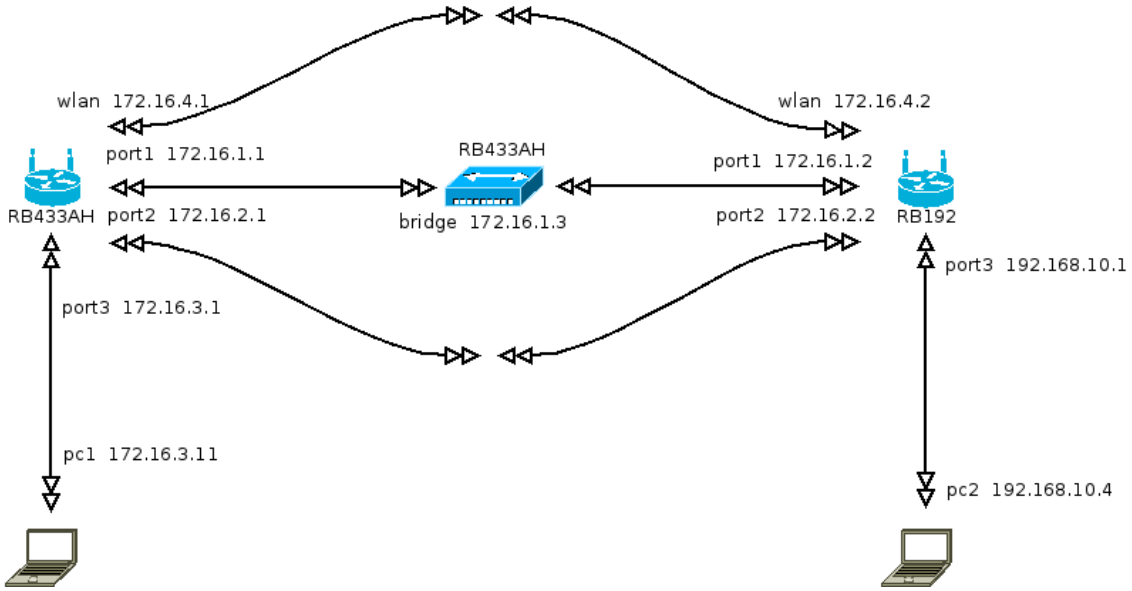
Obr. 4.4: Ukázka automaticky poslaného informačního e-mailu

spojem, nastávala situace, kdy se prodloužila doba spojení AP s protilehlou stanicí. V případě, že byl interval krátký, začal Netwatch testovat spoj před ustálením situace což vyústilo v cyklické přepojování s dopadem na koncové uživatele. Při zvolených 14 sekundách se již tento problém neobjevoval (nastavená doba je i s menší rezervou).

4.2 Použití směrování

V druhé situaci je popsána společnost, která má dvě budovy, jež jsou od sebe umístěny ve větší vzdálenosti. Každá budova má svoji síť a mezi budovami jsou tři různé spoje - metalický spoj, bezdrátový spoj a spoj přes externího poskytovatele připojení. Primárně se chce využít metalický spoj, který je vlastněn přímo společností. Bezdrátový spoj slouží jako záloha, je pomalejší s větší latencí. Spoj přes externího poskytovatele slouží jako poslední možnost pro zajištění konektivity.

Cílem tedy je přepínání transportních cest podle jejich dostupnosti a zároveň podle priority těchto cest.



Obr. 4.5: Diagram sítě

V modelové simulaci (obr. 4.5) proti sobě stojí dvě zařízení - RB433, které představuje zástupce první sítě a zařízení RB192, představující síť druhou. Mezi nimi jsou vytvořeny 3 linky - primární spoj je tvořen kabelem z portu č. 1 do portu č. 1, sekundární spoj je tvořen bezdrátovou sítí a poslední spojení je opět tvořeno kabelem mezi porty č. 3. Do prvního zařízení (RB433AH), do portu č.3, je přímo přímo připojen PC pro testování. Do druhého zařízení (RB192), do portu č. 3, je připojen druhý PC pro testování.

Na samotných zařízení jsou nastaveny IP adresy portů následovně:
RB433:

Rozhraní	**IP adresa**
eth 1	172.16.1.2/24
eth 2	172.16.2.2/24
wlan 1	172.16.4.2/24
eth 3	192.168.10.1/24

RB192:

Rozhraní	**IP adresa**
eth 1	172.16.1.1/24
eth 2	172.16.2.1/24
eth 3	172.16.3.1/24
wlan 1	172.16.4.1/24

4.2.1 Použití RIP protokolu

V menu protokolu RIP na kartě Interfaces jsou vybrána rozhraní, na které bude aplikován RIP. Dále jsou na kartu Networks postupně nadefinovány všechny sítě, se kterými RIP pracuje. RIP si následně vymění směrovací tabulku s ostatními směrovači, kde je také aktivován. Výsledný stav je zobrazen na obrázku 4.6.

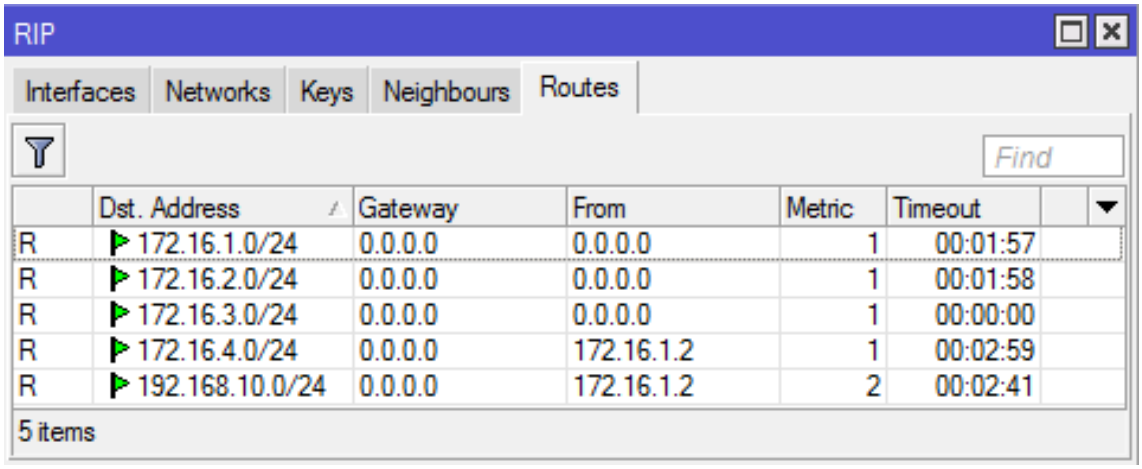
Reakční doba na výpadek spojení se pohybovala od jedné do dvou minut.

Nevýhodou tohoto řešení byla nemožnost ovlivnění priority cesty - směrovač si sám volil, kterou trasu použít. V případě propojení přes bezdrátovou síť udržoval protokol RIP spojení i přes silnou degradaci spoje. A v mém zobrazovaném příkladě si po výpadku primární cesty zvolil rovnou záložní cestu přes eth2. Obrázek číslo 4.7 zobrazuje grafický průběh na časové ose. Modré peaky mající rychlost nad 100 kbps ukazují přenos a pod 100 kbps výpadek.

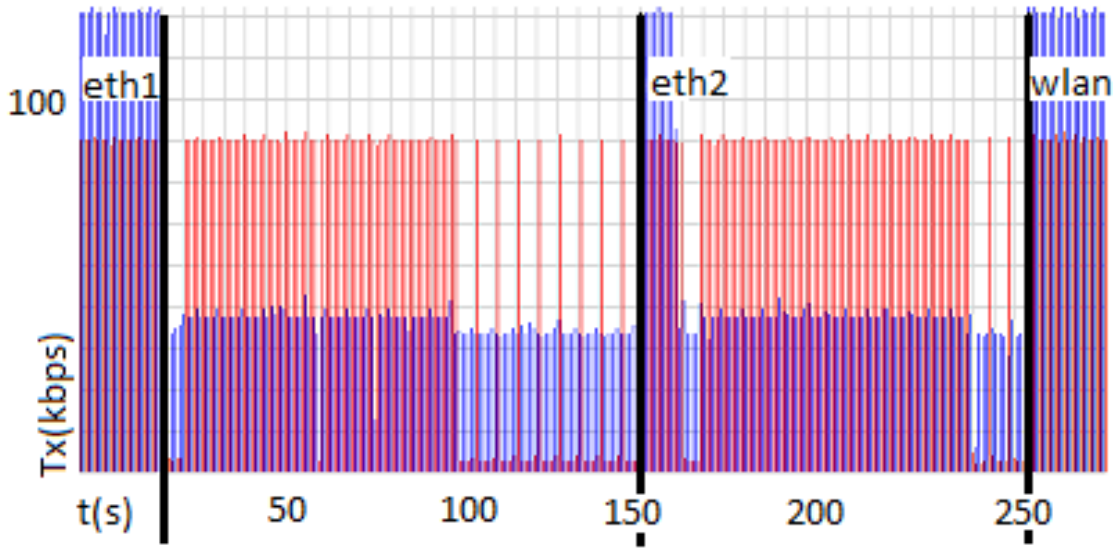
4.2.2 Použití OSPF protokolu

Při tomto nastavení jsou statické routy neaktivní. V menu protokolu OSPF na kartě Networks jsou nadefinována všechna rozhraní, která jsou připojená k RouterBoardu. Následně se směrovač zkontaktuje se sousedními směrovači a předají si mapu sítě (obr. 4.8). Výsledkem je seznam aktivních cílových rozhraní (obr. 4.9).

Nastavení OSPF protokolu obsahuje nejméně kroků a je nejrychlejší na nastavení ze všech zde uvedených. Při testování byla reakční doba na výpadek spojení přibližně 5 sekund. Na obrázku 4.10 jdou vidět jak krátké byly výpadky při odpojení aktivní

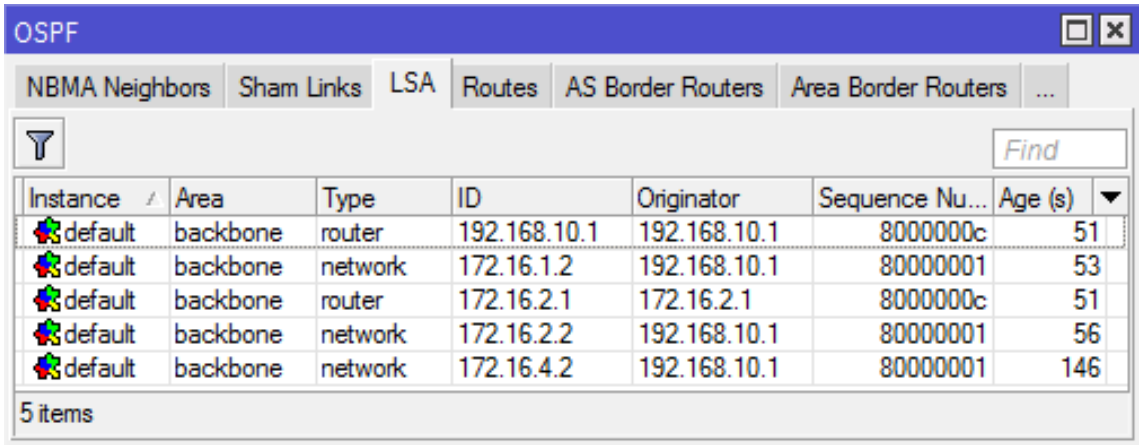


Obr. 4.6: Vygenerované routy

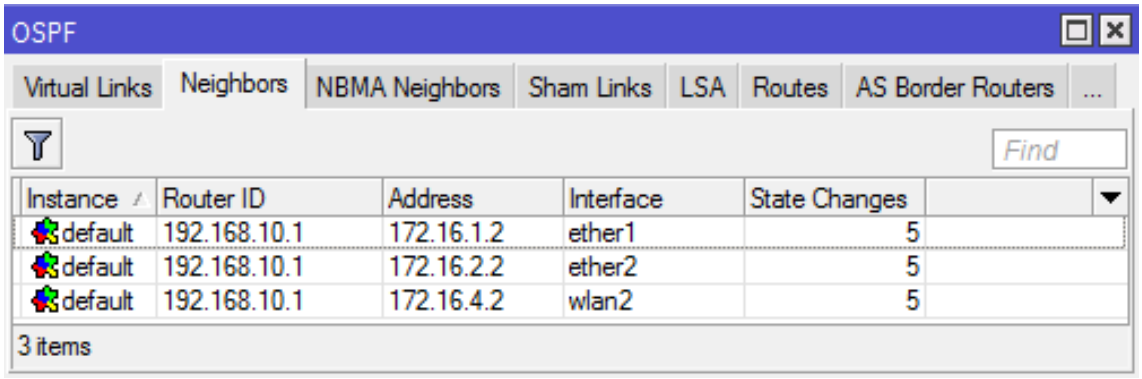


Obr. 4.7: Průběh testování RIP protokolu

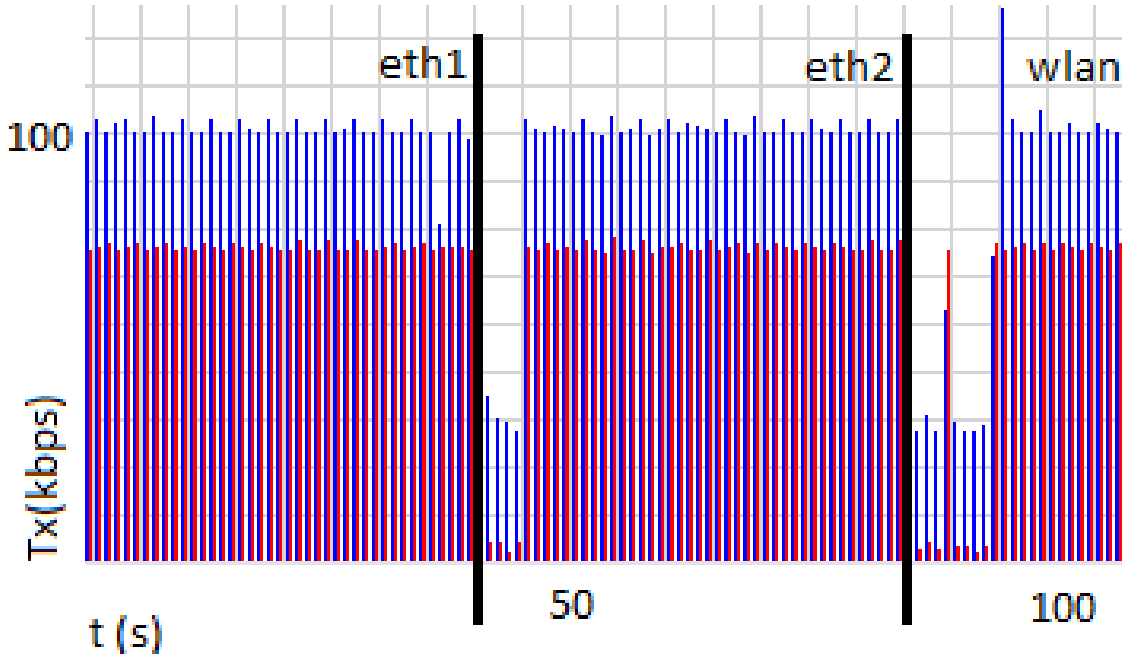
linky. Stejně jako v případě RIP nebylo možné ovlivňovat metriky cest, a proto po obnově trasy nepřepnul zpět na preferovanou cestu.



Obr. 4.8: Mapa sítě



Obr. 4.9: Rozhraní s cílovou cestou



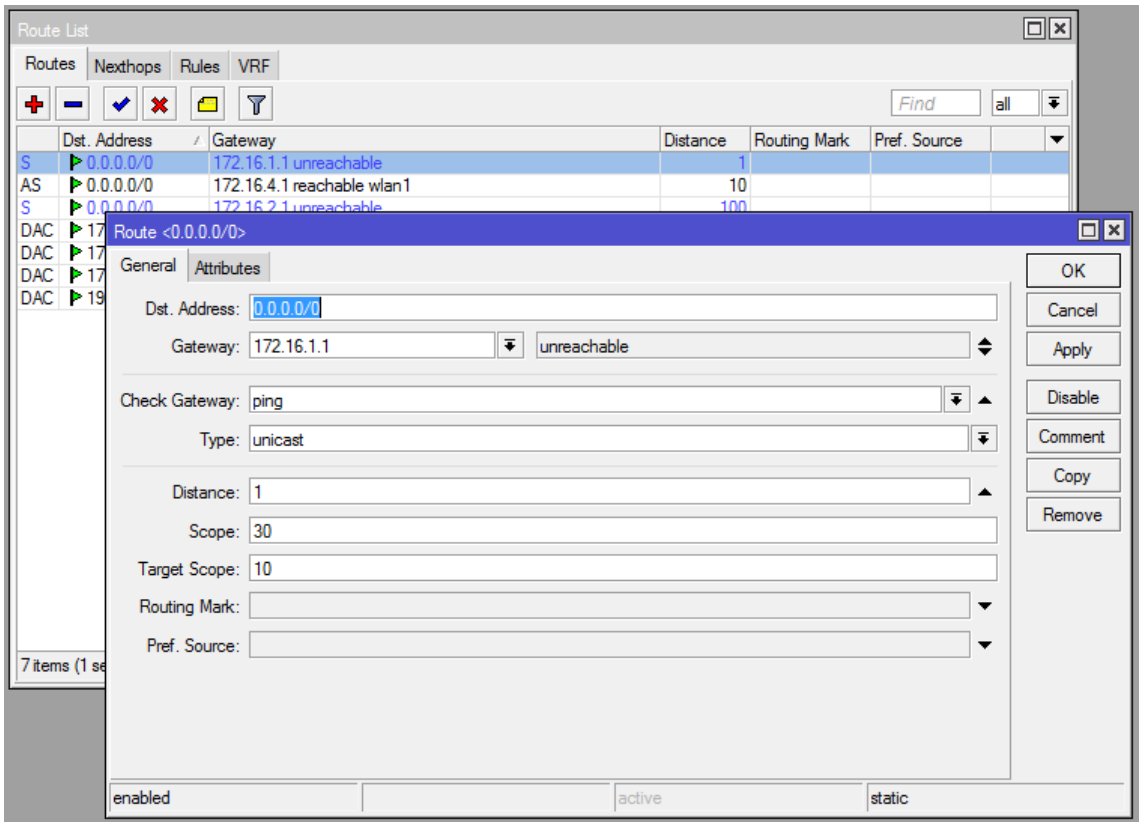
Obr. 4.10: Průběh testování OSPF protokolu

4.2.3 Síť se statickým směrováním

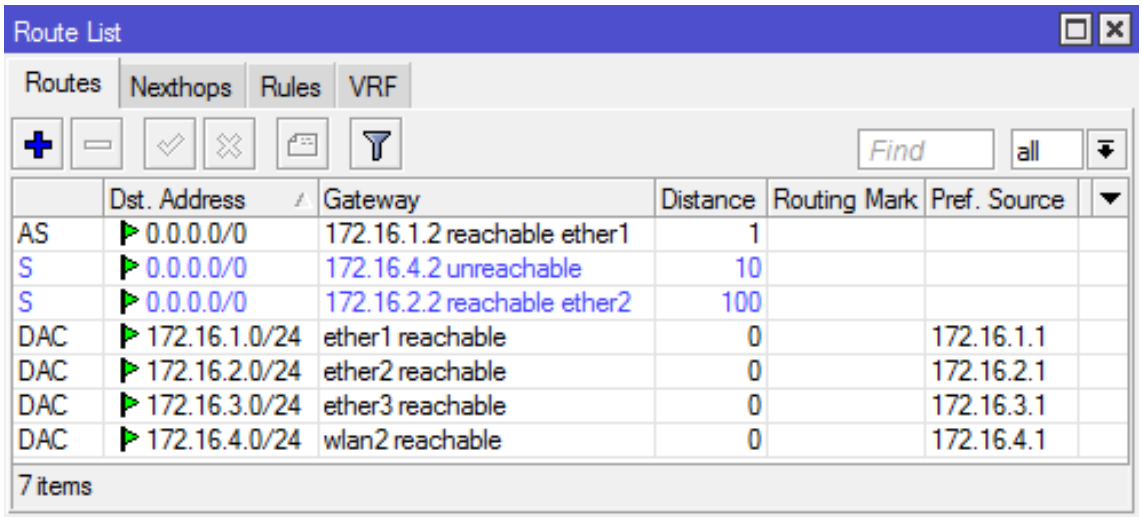
Pomocí služby Route (obr. 4.11) jsou vytvořeny tři cesty vždy pro celou síť (obr. 4.12), každá z nich má nastavenou jinou prioritu pomocí parametru Distance. Toto nastavení bylo dostačující a funkční ihned zpočátku testování.

V případě statického směrování se nevyskytly žádné problémy. Při odpojení primární linky přešlo plynule spojení na sekundární linku, i když ta měla (uměle vytvořené) výrazně horší parametry, než terciální linka přes LAN kabel. Až po zarušení linky do meze, kdy se přenos rozpadl, se přepnulo směrování na poslední linku. V případě obnovení spojení se cesta posunula zpět na tuto linku.

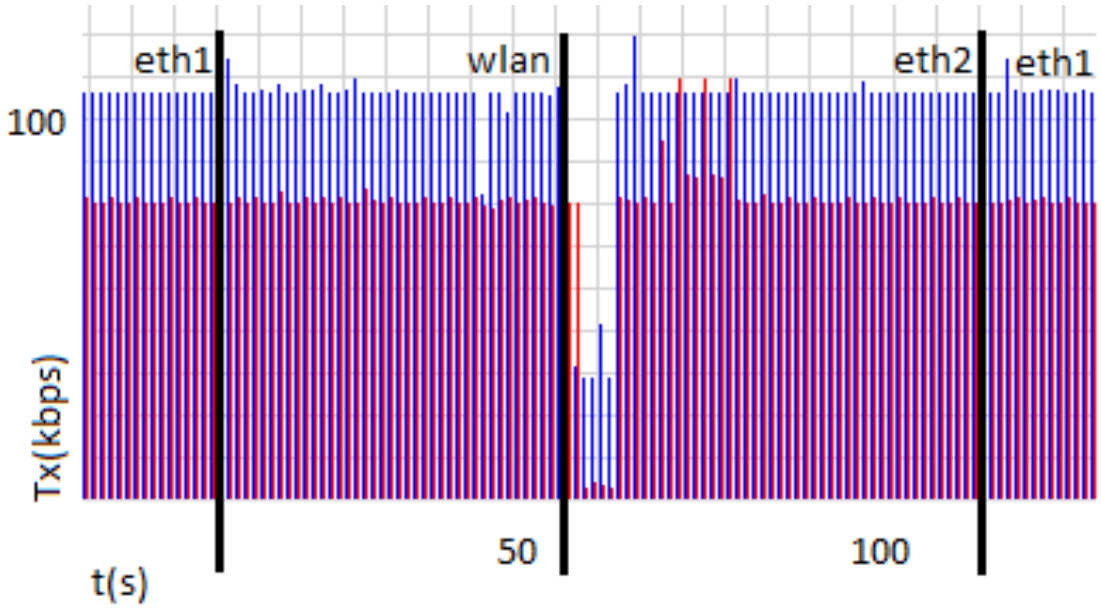
Dále byla testována situace, při které se mělo zobrazit, zda-li RouterBOARD pozná výpadek LAN podle odpojeného portu. Do primární cesty byl proto vložen ještě další RouterBOARD, u kterého byly nastaveny dva porty do bridge. Směrovač v cestě se choval jako standardní switch. Při deaktivaci bridge, zůstaly porty fyzicky aktivní. V té chvíli spojení vypadlo, ale cesta byla stále aktivní. U statických cest byla v nastavení aplikovaná možnost kontroly brány této cesty (Check Gateway). S tímto nastavením přepínání cest začalo fungovat opět korektně. Interval odezvy je interně nastaven na 10 sekund a nedá se změnit. Tato hodnota, podle příspěvků uživatelů MikroTiku na oficiálních stránkách výrobce, není příliš vyhovující, avšak firma MikroTik uvedla, že v dohledné době změnu této hodnoty nechystá. [12]



Obr. 4.11: Služba Route



Obr. 4.12: Nastavení směrování



Obr. 4.13: Průběh testování při použití statických rout

4.2.4 Síť se směrováním pomocí skriptu

Nemožnost ovlivnění ceny aktivní routy stavem a přenosovou rychlostí linky dává prostor k použití vlastních skriptů pro změnu routovací tabulky a kontrolujících parametry spoje, kde především u bezdrátové linky je potřeba kontrolovat její stav. Jako nejvhodnější se jeví kontrola rychlosti spoje. Ta se spouští v cyklických intervalech v době, kdy Netwatch nastaví aktivní routu na sekundární linku.

Celý proces probíhá tak, že při nedostupnosti primární linky Netwatch spustí skript (PrimaryDown), který spustí další skript (CheckSpeed), který zkontroluje přenosovou rychlost sekundární linky a podle ní nastaví ceny rout ve směrovací tabulce a zároveň jej nastaví do plánovače, který jej každých pět minut znovu spustí. V případě obnovení dostupnosti primární linky služba Netwatch spustí skript (PrimryUp), který zkontroluje a smaže úkol v plánovači a nastaví nejnižší cenu routy na primární linku (obr. 4.14 a obr. 4.15).

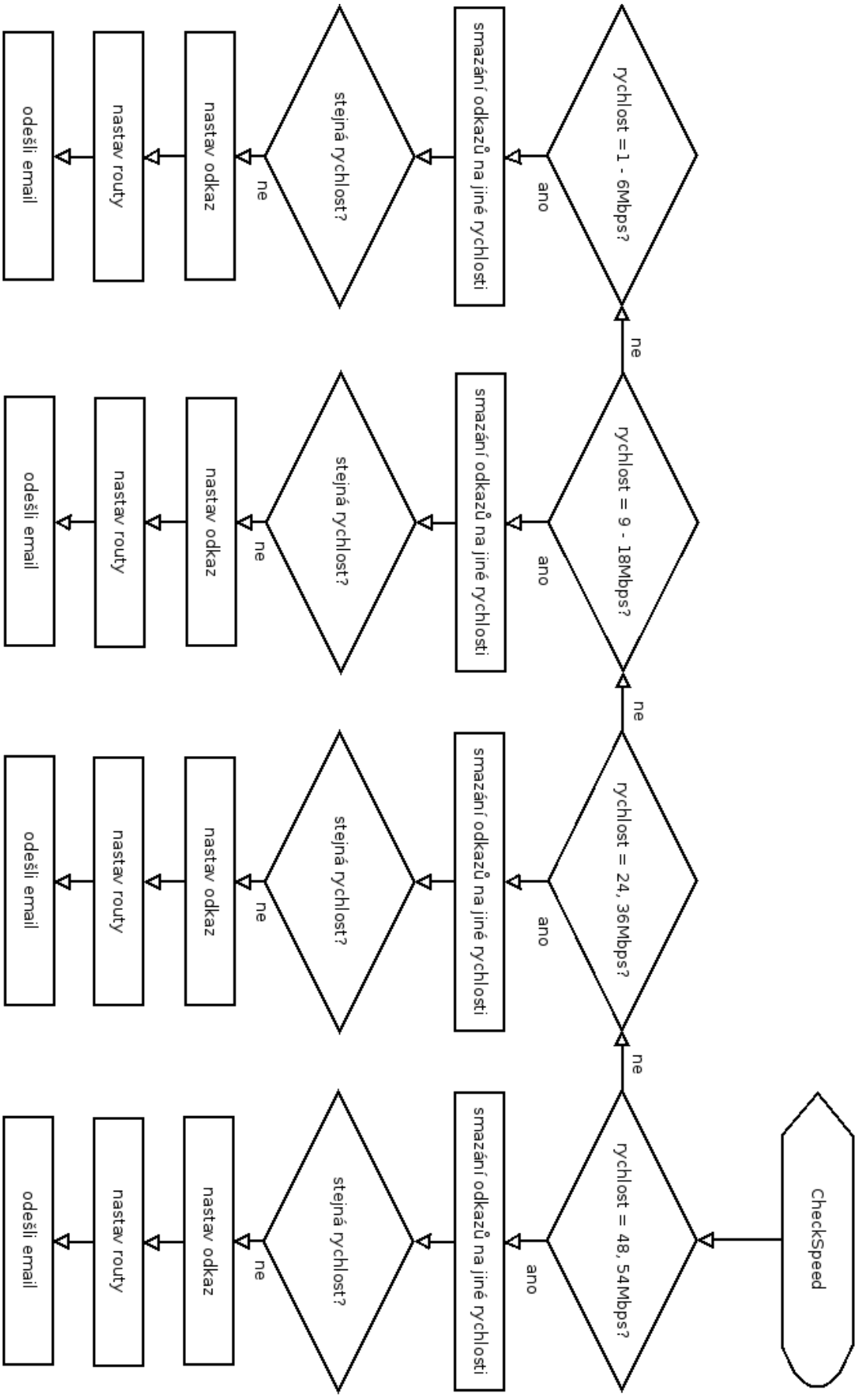
Samotný popis skriptu CheckSpeed na kontrolu rychlosti spoje je následující:

- v první fázi si načte hodnotu rychlosti spoje a uloží si ji do proměnné. Hodnota se dá načíst pouze jako proměnná typu string.
- v další fázi je tato hodnota porovnána a podle rychlosti rozdělena do čtyřech různých vrstev. V každé vrstvě je vždy kontrolována existence záznamu z předchozí kontroly. Pokud je nalezen záznam o tom, že v předchozí kontrole se skript pohyboval v jiné vrstvě rychlosti, je tento záznam smazán. Dále se kontroluje výskyt záznamu o tom, zda-li se při předchozí kontrole skript pohyboval

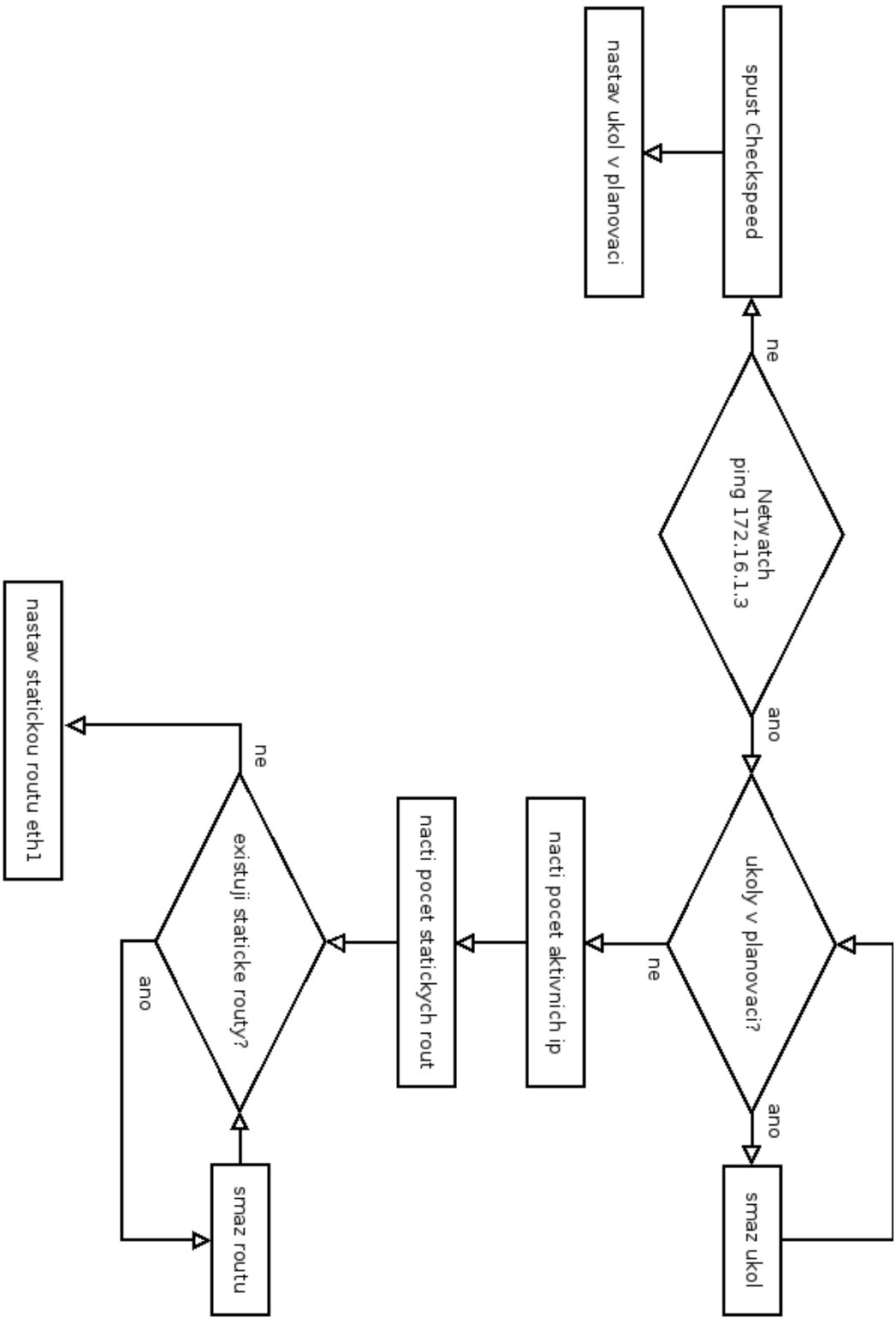
ve stejné vrstvě jako je aktuální. Pokud se najde, neprovede se nic. Pokud tento záznam nalezen není, je vytvořen a je spuštěn podskript (FullSpeed, HalfSpeed, QuarterSpeed or SlowSpeed) pro danou rychlostní vrstvu. Záznam je tvořen souborem uloženým do paměti RouterBoardu a to z důvodu, že RouterBoard si neumí udržet hodnotu v globální proměnné po skončení skriptu. Tento krok eliminuje časté přenastavování síťové karty RouterBoardu, pří kterém se generují krátké výpadky spoje.

- samotné podskripty spouštěné podle rychlosti linky obsahují část, která vyhledá počet manuálních routovacích záznamů v routovací tabulce a smaže je. Poté jsou nastaveny nové routy s danými cenami. Nakonec je poslán informační email na adresu správce sítě.

K neustálému znovunastavení rout před pouhým zneaktivněním nepoužívaných rout bylo přistoupeno z důvodu jistějšího a jednoduššího nastavení těchto hodnot. Časová náročnost je u obou způsobů shodná. Všechny zmíněné skripty se nacházejí v příloze A.



Obr. 4.14: Vývojový diagram skriptu část 1.



Obr. 4.15: Vývojový diagram skriptu část 2.

4.3 Omezení výkonu AP

V další modelové situaci byla snaha o snížení vysílacího výkonu AP.

V prvním předpokladu šlo o automatické snížení vysílacího výkonu AP tak, aby klientské stanice stále měli kvalitní a plný příjem. Pouze beacon by byl vysílán plným výkonem, aby jej zachytili noví klienti, vyhledávající danou síť. Bohužel toto nastavení zařízení nepodporují v základních možnostech nastavení. Jedinou možností je přímá úprava firmware zařízení, což ale může způsobit různé potíže. Vysílaný výkon lze nastavovat pouze pro celé rozhraní a změna výkonu navíc způsobuje krátké výpadky bezdrátové sítě.

První situace byla pozměněna na scénář, kdy by se vysílací výkon AP snížil podle klienta s nejnižším signálem při zachování přenosové rychlosti (parametr Antenna Gain). Samotné vyhledání tohoto klienta je lehký úkol, rychlost přenosu je však ovlivněna více parametry než jen sílou signálu a proto se nedá jednoznačně říci, jak moc může být vysílací výkon zeslaben. Postupným krokováním opět dochází ke krátkým výpadkům při změně nastavení bezdrátové karty, proto tato varianta není vhodná.

V poslední situaci byla snaha o úpravu vysílacích výkonů na určitých rychlostech linky v závislosti na klientovi s nejnižší přenosovou rychlostí (parametr Tx power). I tato varianta skončila na nejednoznačnosti závislosti rychlosti na síle vysílacího signálu.

5 ZÁVĚR

Při konzultaci s kolegy, kteří běžně pracují na dané technologii jsem se dozvěděl, že při ostrém nasazování těchto záložních spojů často došli do bodu, kdy se dané zařízení začalo chovat jinak, než čekali. Například i mnou výše zmíněná chyba, kdy vypadne logické spojení na portu, ale elektricky je stále připojen (například do protějšího switche), routovací tabulka stále zůstávala na chybné routě. Routa musela být ošetřena na zkoušku dostupnosti. Jeden z pokusů bylo i chování při vysoké zátěži na jedné z cest a její následná degradace (vysoká odezva, nízká propustnost). Bohužel ani v jednom případě nebyl brán na tuhle situaci zřetel a spojení na vybrané cestě probíhalo až do doby rozpadnutí se této cesty.

Práce v příkazovém řádku RouterBOARDu je celkem intuitivní, příjemná a logická. Nápověda se dá vyvolat téměř v každém kroku pomocí otazníku nebo klávesy TAB a nabídne možné následující příkazy a jejich syntaxi.

Při práci jsem narazil na dva bugy v programu. První se týkal RB192 (firmware 2.9), kdy mi na příkaz zobrazení počtu záznamů v tabulce *print count-only* (například */ip address print count-only* - zobrazující počet zadaných adres rozhraní) bach vždy vracel hodnotu -1. Tato chyba se projevovala jen v této verzy, novější již fungovala v pořádku. Druhý se týkal použití příkazu *get* (například */system scheduler get value-name=name number=0* - zobrazující jméno naplánovaného úkolu). Při testování skriptu s tímto příkazem vše v příkazové řádce fungovalo bez problémů, ovšem při spuštění skriptu z rozhraní Winbox tento příkaz nevracel žádnou hodnotu. Řešením nakonec bylo zařadit před tento příkaz příkaz *print* (/system scheduler print), který vypisuje veškeré informace. Dále jsem měl snahu na zkoušku zprovoznit dohledový systém DuDe, bohužel jsem natrefil na zařízení, které nepodporují instalaci DuDe serveru (SXT Lite5, RB433AH, RB192, LHG 5 ani wAP G-60ad), který se přidává jako rozšiřující balíček přímo do RouterBoardu.

LITERATURA

[1] Mgr. Radek Hoszowski: *Datové sítě* [online]. Učební text, poslední aktualizace 01. 12. 2012 [cit. 19. 05. 2018]. Dostupné z URL: <http://www.sslch.cz/files/163/13-datove-site-u.pdf>.

[2] MikroTik: *MikroTik Wiki* [online]. 2017, poslední aktualizace 28. 11. 2017 [cit. 19. 05. 2018]. Dostupné z URL: <https://wiki.mikrotik.com/>.

[3] Vysoká škola ekonomická v Praze: *Návrh malé až střední počítačové sítě LAN* [online]. Bakalářská práce Praha: 2010, poslední aktualizace 29. 06. 2010 [cit. 19. 05. 2018]. Dostupné z URL: <http://info.sks.cz/www/zavprace/soubory/68510.pdf>.

[4] Jiří Grygarek: *Směrovací protokol OSPF* [online]. poslední aktualizace 2016 [cit. 19. 05. 2018]. Dostupné z URL: <http://www.cs.vsb.cz/grygarek/SPS/lect/OSPF/ospf.html>.

[5] VUT v Brně: *Systém detekce útoku pro sítě s platformou MikroTik* [online]. Bakalářská práce Brno: 2016, poslední aktualizace 01. 06. 2016 [cit. 19. 05. 2018]. Dostupné z URL: <https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=130144>.

[6] www.svetsiti.cz: *Další vývoj protokolu SNMP.* [online]. Článek, poslední aktualizace 15. 06. 2000 [cit. 19. 05. 2018]. Dostupné z URL: <http://www.svetsiti.cz/clanek.asp?cid=Dalsi-vyvoj-protokolu-SNMP-1562000>.

[7] www.microsoft.com: *Přehled služby Windows Management Instrumentation (WMI).* [online]. Článek, poslední aktualizace 2016 [cit. 19. 05. 2018]. Dostupné z URL: <https://technet.microsoft.com/cs-cz/library/dn265977.asp>.

[8] www.nagios.org: *Nagios – The Industry Standard In IT Infrastructure Monitoring.* [online]. poslední aktualizace 2016 [cit. 19. 05. 2018]. Dostupné z URL: <https://www.nagios.org>.

[9] Michael Friedrich: *Icinga vs Nagios – a developer's comparison* [online]. poslední aktualizace 03 11 2011 [cit. 19. 05. 2018]. Dostupné z URL: <https://www.icinga.com/2011/11/03/icinga-vs-nagios-a-developers-comparison/>.

[10] MikroTik: *The Dude* [online]. poslední aktualizace 2015 [cit. 19. 05. 2018]. Dostupné z URL: <https://www.mikrotik.com/thedude>.

[11] Petr Macek: *Cacti: vše důležité v jednom monitoru* [online]. poslední aktualizace 31 03 2009 [cit. 19. 05. 2018]. Dostupné z URL: <https://www.root.cz/clanky/cacti-vse-dulezite-v-jednom-monitoru/>.

[12] Mikrotik forum: *enhance "check-gateway"feature - use arbitrary check IP* [online]. poslední aktualizace 25. 05. 2017 [cit. 19. 05. 2018]. Dostupné z URL: <https://forum.mikrotik.com/viewtopic.php?t=81083>.

[13] Petr Bouška: *Řádkové příkazy Windows* [online]. poslední aktualizace 22 12 2009 [cit. 19. 05. 2018]. Dostupné z URL: <https://www.samuraj-cz.com/clanek/radkove-prikazy-windows/>.

[14] Zbyněk Pospíchal: *Jak správně číst výpisy z traceroute* [online]. poslední aktualizace 16 01 2009 [cit. 19. 05. 2018]. Dostupné z URL: <https://www.lupa.cz/clanky/jak-spravne-cist-vypisy-z-traceroute/>.

[15] Pavel Herrmann: *MikroTik* [online]. poslední aktualizace 28 02 2008 [cit. 19. 05. 2018]. Dostupné z URL: <https://www.eduroam.cz/cs/spravce/ap/mikrotik>.

[16] Ramalan Harifa: *Difference between RIP and OSPF* [online]. poslední aktualizace 14 08 2017 [cit. 19. 05. 2018]. Dostupné z URL: <http://www.differencebetween.net/technology/internet/difference-between-rip-and-ospf/>.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

LAN	místní síť – Local Area Network
MAN	metropolitní síť – Metropolitan Area Networkk
WAN	rozsáhlá síť – Wide Area Network
UTP	nestíněná kroucená dvoulinka – Unshielded Twisted Pair
WiFi	bezdrátová síť – Wireless LAN
Hz	kmitočet – Herz
AP	přístupový bod – Access point
CAT 5	kategorie 5 – Category 5 cable
IBM	International Business Machines Corporation
ATT	American Telephone and Telegraph Corporation
PC	osobní počítač – personal computer
Telnet	teletype network
SSH	Secure Shell
IP	internetový protokol – Internet Protocol
WLAN	bezdrátová lokální síť – Wireless local area network
RIP	Routing Information Protocol
OSFP	Open Shortest Path First
ICMP	Internet Control Message Protocol
SNMP	Simple Network Management Protocol
WMI	Windows Management Instrumentation
SMS	Short Message Service
SVG	Scalable Vector Graphics
XML	eXtensible Markup Language
DNS	Domain Name System

A PŘÍLOHA - POUŽITÉ SKRIPTY

Výpis A.1: skript PrimaryDown

```
#skript PrimaryDown

/system script run CheckSpeed;
/system scheduler add name=check_wlan interval=300 on-event=CheckSpeed;
```

Výpis A.2: skript PrimaryUp

```
#skript PrimaryUp

#smaz scheduler
:global pocetUkolu [/system scheduler print count-only ];
if ($pocetUkolu>0) do={
system scheduler print;
if ([system scheduler get value-name=name number=0]="CheckSpeed")
do={/system scheduler remove CheckSpeed;
}

#skript pocet aktivnich adres
:global pocetAdres [/ip address print count-only ];
:global aktivAdr [/ip address print count-only ];
:for pocetAdres from=$aktivAdr to=1 do={:if [/ip address get
value-name=disabled ($pocetAdres-1)] do={:set aktivAdr ($aktivAdr-1)}};
:put $aktivAdr;

#pocet manualnich rout
:global routy ([/ip route print count-only]-$aktivAdr);

#odstraneni zaznamu z routovaci tabulky
:while ($routy>0) do={/ip route print ; /ip route remove numbers=0 ;
:set routy ($routy-1)};

#nastav routy
/ip route add dst-address=0.0.0.0/0 gateway=172.16.1.2 distance=20
check-gateway=ping;
```

Výpis A.3: skript CheckSpeed

```
#skript CheckSpeed

#nacti_hodnoty
/interface wireless registration-table print stats

#uloz_do_sn_rychlost_prenosu_klienta_0
:global sn [interface wireless registration-table get value-name=tx-rate 0];

#rozdeleni_podle_rychlosti
#plna
if ($sn="54Mbps" or $sn="48Mbps") do={

if ([:len [/file find name=stavM.txt]] > 0) do={/file remove stavM.txt};
if ([:len [/file find name=stavL.txt]] > 0) do={/file remove stavL.txt};
if ([:len [/file find name=stavQ.txt]] > 0) do={/file remove stavQ.txt};

if ([:len [/file find name=stavF.txt]] > 0) do={} else={/file print
file=stavF.txt; /system script run FullSpeed};

};

#polovicni
if ($sn="36Mbps" or $sn="24Mbps") do={

if ([:len [/file find name=stavF.txt]] > 0) do={/file remove stavF.txt};
if ([:len [/file find name=stavL.txt]] > 0) do={/file remove stavL.txt};
if ([:len [/file find name=stavQ.txt]] > 0) do={/file remove stavQ.txt};

if ([:len [/file find name=stavM.txt]] > 0) do={} else={/file print
file=stavM.txt; /system script run HalfSpeed};

};

#ctvrtinova
if ($sn="18Mbps" or $sn="12Mbps" or $sn="11Mbps" or $sn="9Mbps") do={

if ([:len [/file find name=stavF.txt]] > 0) do={/file remove stavF.txt};
if ([:len [/file find name=stavL.txt]] > 0) do={/file remove stavL.txt};
if ([:len [/file find name=stavM.txt]] > 0) do={/file remove stavM.txt};

if ([:len [/file find name=stavQ.txt]] > 0) do={} else={/file print
```

```
file=stavQ.txt; /system script run QuarterSpeed};

};

#nizka
if ($sn="6Mbps" or $sn="5.5Mbps" or $sn="2Mbps" or $sn="1Mbps") do={

if ([:len [/file find name=stavF.txt]] > 0) do={/file remove stavF.txt};
if ([:len [/file find name=stavM.txt]] > 0) do={/file remove stavM.txt};
if ([:len [/file find name=stavQ.txt]] > 0) do={/file remove stavQ.txt};

if ([:len [/file find name=stavL.txt]] > 0) do={} else={/file print
file=stavL.txt; /system script run SlowSpeed};

};
```

Výpis A.4: skript FullSpeed

```
#skript FullSpeed

#skript pocet aktivnich adres
:global pocetAdres [/ip address print count-only ];
:global aktivAdr [/ip address print count-only ];
:for pocetAdres from=$aktivAdr to=1 do={
:if [/ip address get value-name=disabled ($pocetAdres-1)] do={:set
aktivAdr ($aktivAdr-1)}}; :put $aktivAdr;

#pocet manualnich rout
:global routy ([/ip route print count-only]-$aktivAdr);

#odstraneni zaznamu z routovaci tabulky
:while ($routy>0) do={/ip route print ; /ip route remove numbers=0 ;
:set routy ($routy-1)};

#nastav routy
/ip route add dst-address=0.0.0.0/0 gateway=172.16.4.2 distance=20
check-gateway=ping;
/ip route add dst-address=0.0.0.0/0 gateway=172.16.2.2 distance=30
check-gateway=ping;
```

```
/tool e-mail send body="Spoj␣ma␣plnou␣rychlost␣($[/system␣clock␣get␣date␣]
␣$[/system␣clock␣get␣time␣])" from=check.mikrotik@gmail.com
password=cm_nG.com port=587 server=108.177.15.109 subject=network_status
to=pzavak@gmail.com user=check.mikrotik@gmail.com start-tls=yes
```

Výpis A.5: skript HalfSpeed

```
#skript HalfSpeed

#skript pocet aktivnich adres
:global pocetAdres [/ip address print count-only ];
:global aktivAdr [/ip address print count-only ];
:for pocetAdres from=$aktivAdr to=1 do={
:if [/ip address get value-name=disabled ($pocetAdres-1)] do={:set
aktivAdr ($aktivAdr-1)}}; :put $aktivAdr;

#pocet manualnich rout
:global routy ([/ip route print count-only]-$aktivAdr);

#odstraneni zaznamu z routovaci tabulky
:while ($routy>0) do={/ip route print ; /ip route remove numbers=0 ;
:set routy ($routy-1)};

#nastav routy
/ip route add dst-address=0.0.0.0/0 gateway=172.16.4.2 distance=20
check-gateway=ping;
/ip route add dst-address=0.0.0.0/0 gateway=172.16.2.2 distance=30
check-gateway=ping;

/tool e-mail send body="Spoj␣ma␣polovicni␣rychlost␣($[/system␣clock␣get␣date␣]
␣$[/system␣clock␣get␣time␣])" from=check.mikrotik@gmail.com
password=cm_nG.com port=587 server=108.177.15.109 subject=network_status
to=pzavak@gmail.com user=check.mikrotik@gmail.com start-tls=yes
```

Výpis A.6: skript QuarterSpeed

```
#skript QuarterSpeed

#skript pocet aktivnich adres
```

```
:global pocetAdres [/ip address print count-only ];
:global aktivAdr [/ip address print count-only ];
:for pocetAdres from=$aktivAdr to=1 do={
:if [/ip address get value-name=disabled ($pocetAdres-1)] do={:set
aktivAdr ($aktivAdr-1)}}; :put $aktivAdr;

#pocet manualnich rout
:global routy ([/ip route print count-only]-$aktivAdr);

#odstraneni zaznamu z routovaci tabulky
:while ($routy>0) do={/ip route print ; /ip route remove numbers=0 ;
:set routy ($routy-1)};

#nastav routy
/ip route add dst-address=0.0.0.0/0 gateway=172.16.4.2 distance=29
check-gateway=ping;
/ip route add dst-address=0.0.0.0/0 gateway=172.16.2.2 distance=30
check-gateway=ping;

/tool e-mail send body="Spoj␣ma␣ctvrtinovou␣rychlost␣($[/system␣clock␣get␣date␣]
␣$[/system␣clock␣get␣time␣])" from=check.mikrotik@gmail.com
password=cm_nG.com port=587 server=108.177.15.109 subject=network_status
to=pzavak@gmail.com user=check.mikrotik@gmail.com start-tls=yes
```

Výpis A.7: skript SlowSpeed

```
#skript SlowSpeed

#skript pocet aktivnich adres
:global pocetAdres [/ip address print count-only ];
:global aktivAdr [/ip address print count-only ];
:for pocetAdres from=$aktivAdr to=1 do={
:if [/ip address get value-name=disabled ($pocetAdres-1)] do={:set
aktivAdr ($aktivAdr-1)}}; :put $aktivAdr;

#pocet manualnich rout
:global routy ([/ip route print count-only]-$aktivAdr);

#odstraneni zaznamu z routovaci tabulky
:while ($routy>0) do={/ip route print ; /ip route remove numbers=0 ;
:set routy ($routy-1)};
```

```
#nastav routy
/ip route add dst-address=0.0.0.0/0 gateway=172.16.4.2 distance=30
check-gateway=ping;
/ip route add dst-address=0.0.0.0/0 gateway=172.16.2.2 distance=20
check-gateway=ping;


/tool e-mail send body="Spoj␣ma␣nizkou␣rychlost␣-␣prenastaven␣na␣zalozni
($[/system␣clock␣get␣date␣]␣$[/system␣clock␣get␣time␣])"
from=check.mikrotik@gmail.com password=cm_nG.com port=587
server=108.177.15.109 subject=network_status to=pzavak@gmail.com
user=check.mikrotik@gmail.com start-tls=yes
```

Výpis A.8: skript pro zprovozneni netwatch

```
#script_pro_zprovozneni_netwatch

tool netwatch add host=192.168.0.98 interval=00:00:14
timeout=0.5 down-script=script_5g_down up-script=script_5g_up
```

Výpis A.9: skript 5g up

```
#script_5g_up

interface wireless set mode=station country="czech␣republic"
wireless-protocol=any numbers=0 band=5ghz-a
ssid=MikroTik security-profile=default
/tool e-mail send body="net_5g_up␣($[/system␣clock␣get␣date␣]
$[/system␣clock␣get␣time␣])" from=check.mikrotik@gmail.com
password=cm_nG.com port=587 server=108.177.15.109
subject=network_status to=pzavak@gmail.com
user=check.mikrotik@gmail.com start-tls=yes
```

Výpis A.10: skript 5g down

```
#script_5g_down

interface wireless set mode=station country="czech␣republic"
wireless-protocol=any numbers=0 band=2ghz-b/g
ssid=UPC3318785 security-profile=wpa2
```

```
/tool e-mail send body="net_5g_down␣($[/system␣clock␣get␣date␣]
$[/system␣clock␣get␣time␣])" from=check.mikrotik@gmail.com
password=cm_nG.com port=587 server=108.177.15.109
subject=network_status to=pzavak@gmail.com
user=check.mikrotik@gmail.com start-tls=yes
```