

Univerzita Hradec Králové

Fakulta informatiky a managementu

DIPLOMOVÁ PRÁCE

2017

Matěj Drdla

Univerzita Hradec Králové

Fakulta informatiky a managementu

Katedra informačních technologií

**Využití a implementace softwarového
směrovače na platformě Mini PC**

DIPLOMOVÁ PRÁCE

Autor: Matěj Drdla

Studijní obor: Aplikovaná informatika

Vedoucí práce: Mgr. Josef Horálek, Ph.D.

Hradec Králové

říjen 2017

PROHLÁŠENÍ

Prohlašuji, že jsem diplomovou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 14. října 2017

Matěj Drdla

PODĚKOVÁNÍ

Na tomto místě bych rád vyjádřil poděkování za převzetí vedení mé práce, cenné náměty, připomínky a komentáře Mgr. Josefu Horálkovi, Ph.D. Dále bych chtěl poděkovat za podporu své přítelkyni.

ANOTACE

Práce se zabývá komplexní implementací softwarových směrovačů BIRD a Quagga na dvou různých platformách Mini PC – HummingBoard a Raspberry Pi a to za účelem jejich integrace do Internetu věcí. Výzkum se postupně zaměřuje na analýzu a výběr vhodných hardwarových platforem a relevantních operačních systémů. Následně je realizováno jak statické, tak i dynamické směrování při použití směrovacích protokolů RIP a OSPF. Dále je provedeno základní testování funkčnosti formou komparativní analýzy vzniklých kombinací hardwarových platforem, operačních systémů, softwarových démonů a směrovacích protokolů. Na závěr práce je rovněž poukázáno na jedno z potenciálních využití softwarového směrovače v rámci realizace dálkového ovládání chytrého domu.

ANNOTATION

Title: Implementation and application of software routers on Mini PC platform

This thesis deals with the complex implementation of the BIRD and Quagga software routers on two different Mini PC platforms – HummingBoard and Raspberry Pi for their integration into the Internet of Things. This research focuses on the analysis and selection of suitable hardware platforms and relevant operating systems. Subsequently, both static and dynamic routing is performed using the RIP and OSPF routing protocols. In addition, basic functionality testing is performed in the form of comparative analysis of combinations of hardware platforms, operating systems, software daemons, and routing protocols. Finally, one of the potential uses of the software routers in the framework of the smart house remote control is pointed out.

OBSAH

	ÚVOD	1
1	REŠERŠE STÁVAJÍCÍCH ŘEŠENÍ	4
1.1	TURRIS OMNIA	4
1.1.1	CZ.NIC	4
1.1.2	PARAMETRY	5
2	POPIS ZÁKLADNÍ PROBLEMATIKY	13
2.1	SOFTWAREVÉ SMĚROVAČE	14
2.1.1	QUAGGA	15
2.1.2	BIRD	17
3	ANALÝZA HARDWARE	20
3.1	HUMMINGBOARD	20
3.1.1	SOLIDRUN LIMITED	20
3.1.2	KATEGORIZACE PRODUKTŮ SPOLEČNOSTI SOLIDRUN	22
3.1.3	VOLBA KONKRÉTNÍHO PRODUKTU	23
3.2	RASPBERRY PI	30
3.2.1	RASPBERRY PI FOUNDATION	31
3.2.2	VERZE RASPBERRY PI	31
3.2.3	VOLBA KONKRÉTNÍHO RASPBERRY PI	33
3.3	EXTERNÍ SÍŤOVÁ ROZHRANÍ	36
4	ANALÝZA SOFTWARE	38
4.1	DEBIAN GNU/LINUX	38
4.1.1	ZÁKLADNÍ CHARAKTERISTIKA	38
4.2	RASPBIAN	39
4.2.1	ZÁKLADNÍ CHARAKTERISTIKA	39
4.3	PŘEDÁVÁNÍ	40
5	NÁVRH MODELOVÉHO ŘEŠENÍ	41

6	IMPLEMENTACE	46
6.1	HARDWARE	46
6.2	OPERAČNÍ SYSTÉM	46
6.3	SOFTWAREVÝ DÉMON	48
6.3.1	QUAGGA	48
6.3.2	BIRD	51
7	KRITICKÉ ZHODNOCENÍ NAVRŽENÉHO ŘEŠENÍ	56
7.1	ZÁKLADNÍ TESTOVÁNÍ FUNKČNOSTI	57
7.1.1	OBECNÉ VLASTNOSTI	57
7.1.2	RYCHLOST SMĚROVÁNÍ	59
7.1.3	SOFTWAREVÉ NÁROKY	64
7.1.4	ROZŠÍŘENOST	65
7.2	VYUŽITÍ SMĚROVAČE V INTERNETU VĚCÍ	66
8	ZÁVĚR	69
9	POUŽITÁ LITERATURA	71
10	SEZNAM OBRÁZKŮ	79
11	SEZNAM TABULEK	80
12	SEZNAM GRAFŮ	81
13	PŘÍLOHY	82

Úvod

Čtenář této práce bude pravděpodobně již obeznámen s termínem Internet věcí (IoT). Jen pro úplnost zde zmíníme jednu z možných definic. Výrobce mikroprocesorů Intel Corporation ve zdroji [1] popisuje IoT jako evoluci mobilních, domácích a embedded aplikací. Ty jsou propojeny s Internetem, integrují větší výpočetní schopnosti a využívají datovou analýzu pro získávání smysluplných informací. Zdroj [1] uvádí, že tisíce milionů zařízení bude připojeno k Internetu a další budou následovat. Tato zařízení se mohou stát inteligentním systémem systémů sdílejících a analyzujících data skrze cloudové úložiště. To vše může přeměnit naši každodenní činnost, naše životy a náš svět nespočetným množstvím způsobů. Názornou ukázkou „chytrého světa“ lze nalézt v přílohách této práce (příloha A, Smart World)

Dané myšlenky jsou si vědomy i přední světové organizace. Z akademických řad jmenujme například Massachusetts Institute of Technology, který uvádí ve zdroji [2], že do roku 2020 bude 50 miliard zařízení připojených k Internetu. Výrobce síťových prvků, Cisco Systems, Inc. zmiňuje, že ve stejném roce bude 46% komunikace realizováno čistě mezi zařízeními (M2M) [3]. Společnost IBM dále uvádí, že přibližně 13 miliard zařízení je již připojeno a v roce 2020 jich bude 29 miliard [4].

Společnost Gartner, Inc. zabývající se výzkumem a poradenstvím v oblasti informačních technologií, uvedla v lednu 2017 ve zdroji [5], že na konci tohoto roku bude 8,4 miliard zařízení připojeno k IoT. V porovnání s rokem 2016 se dle zmiňovaného výzkumu jedná o navýšení o 31% koncových zařízení. Enormní vzestup využití IoT má navíc dle této výzkumné organizace dále růst a v roce 2020 dosáhnout 20,4 miliard koncových zařízení. Celková částka v roce 2017 za jednotlivá zařízení a služby je odhadovaná na 2 biliony amerických dolarů (USD).

Ať už jsou prognózy týkající se vývoje IoT jakékoliv, je zřejmé, že se jedná o trend s velkým potenciálem.

Cílem práce je navrhnout a implementovat vybrané softwarové (SW) směrovače na platformě Mini PC, kterou lze využít právě v rámci IoT.

Nejprve je provedena rešerše stávajících řešení, která se důkladně zabývá směrovačem Turris Omnia realizovaným sdružením CZ.NIC. I přesto, že je v rešerši vyzdvížena celá řada vysoce zajímavých parametrů, je poukázáno i na několik potencionálních nedostatků, na základě kterých je odvozen závěr, že především nenaroční uživatelé ocení méně výkonné, avšak daleko levnější řešení. Právě z tohoto důvodu jsme rozhodli o realizaci vlastního směrovače využívajícího pouze „svobodného“ (open-source) SW a to jak operačního systému (OS), tak i SW směrovacího démona.

Dále jsou velmi stručně popsány základy problematiky směrování úzce související s vybranými principy SW směrovačů.

Následuje podrobná analýza vhodných hardwarových (HW) platform, jejich výrobců, jednotlivých produktů těchto výrobců a případných potřebných externích periférií. Mimo jiné je zkoumáno nasazení SW směrovačů na dvou odlišných platformách. První je relativně neznámý, avšak na parametry atraktivní HummingBoard pocházející z dílen izraelského výrobce SolidRun Ltd., druhé je potom všeobecně známé britské Raspberry Pi.

Následně jsou vybrány a stručně popsány jedině, výrobci konkrétních Mini PC oficiálně doporučené, relevantní OS včetně jejich nutně potřebného nastavení (forwarding). Konkrétně se jedná v případě izraelského řešení o OS Debian GNU/Linux a OS Raspbian v případě Raspberry Pi.

Poté je představen návrh komplexního řešení a konkrétní postup k jeho realizaci a to především s důrazem na implementaci a konfiguraci zkoumaných SW démonů. Ke každému směrovacímu démonu je zmíněna jeho základní charakteristika, možnosti jeho implementace, spouštění a několik názorných způsobů konfigurace umožňujících realizaci směrování při využití různých směrovacích protokolů.

V závěru práce je provedeno kritické zhodnocení jednotlivých zkoumaných variant na základě vybraných testů výchozí funkčnosti. Tyto testy se zabývají popisem a komparativní analýzou vytyčených parametrů, kterými jsou obecné vlastnosti směrovacího démona, rychlost jeho směrování, SW nároky a tzv. míra rozšířenosti zkoumaného SW.

Práce se tedy zabývá implementací dvou SW směrovačů (BIRD a Quagga) na dvou platformách Mini PC (HummingBoard a Raspberry Pi) využívajících dva OS (Debian GNU/Linux a Raspbian).

Poznámka

Na závěr úvodní kapitoly bychom rádi poznamenali, že tato práce volně vychází z poznatků získaných výzkumem, jež byl veden při zpracovávání bakalářské práce Komparativní analýza softwarových směrovačů na platformě GNU/Linux [6].

1 Rešerše stávajících řešení

Po důkladné analýze použitých zdrojů v rámci našeho předchozího výzkumu SW směrovačů jsme došli k závěru, že v daném období neexistovala nabídka žádné komerční komplexní síťové platformy, jejíž využití by nebylo čistě jen v rámci počítačových sítí, ale i v IoT. Z tohoto důvodu jsme se rozhodli o vývoj vlastního komplexního směrovače nabízejícího celou řadu, nejen síťově orientovaných, možností.

Absence takového produktu na trhu jsme si nebyli vědomi jen my, ale i zájmové sdružení CZ.NIC, které ve stejném období začalo rovněž vyvíjet svůj vlastní IoT směrovač.

1.1 Turris Omnia

1.1.1 CZ.NIC

Zdroj [7] popisuje toto zájmové sdružení právnických osob jako správce české národní domény CZ. Organizace vznikla v roce 1998 a nyní má 114 členů. CZ.NIC je členem sdružení EURid, spravujícího evropskou doménu EU a dalších obdobně zaměřených mezinárodních společností (CENTR, ccNSO a další) [8]. Tuto činnost zajišťuje prostřednictvím správy domain name serverů. Mimo to se organizace dále zabývá bezpečností viz. aktuálně popisovaný projekt Turris Omnia [9]. Zdroj [8] dále zmiňuje, že sdružení provozuje jak svůj interní bezpečnostní tým CZ.NIC–CSIRT, tak od roku 2011 i národní bezpečnostní tým CSIRT.CZ, který dle zdroje [10] v současné době tvoří skupina 10 členů. Všichni jsou zaměstnanci sdružení CZ.NIC. Oba dva týmy jsou připraveny zasáhnout v případě kybernetického útoku. Zdroj [7] rovněž dodává, že veškerý SW, který organizace vyvíjí, poskytuje jako open–source. V roce 2013 stálo sdružení u vzniku bezpečnostního projektu Fenix [8], zabývajícího se bezpečností virtuálních lokálních počítačových sítí (VLAN). Jako další projekty můžeme jmenovat například systém FRED pro správu registru domén, který kromě správy české domény CZ je dále nasazen pro použití v Argentině [11], Estonsku, Albánii, Costa Rice, Tanzanii a jiných zemích.

Dalším projektem sdružení je internetový směrovací démon BIRD, díky kterému se dorozumí téměř 2/3 propojovacích uzlů tvořící páteř celého Internetu [7].

Odborníci počítačových sítí z laboratoří CZ.NIC založili 12. ledna 2016 crowdfundingovou (forma skupinového financování) kampaň na Internetu za účelem vybrání dostatečného množství finančních zdrojů, potřebných k provedení dalšího výzkumu. Jak je patrné ze zdroje [12], autorům kampaň vyšla, a tak skrze portál Indiegogo vybrali úctyhodných 1,2 milionu USD, čímž docílili 857 % původně požadované částky. Zdroj [9] rovněž dodává, že přispěvatelé byli z celého světa a jejich přibližný počet byl 5 000. Již 22. ledna 2016 zmínil GNU/Linux orientovaný webový portál Root informaci o tom, že nejžádanější přístroj na Indiegogo se stal domácí router [13]. Tým CZ.NIC tak neváhal a ujal se potencionálního volného místa na trhu se svým řešením prvního směrovače komplexně zapouzdřujícího využití jak v IoT, tak rovněž i při směrování. Cena směrovače se k datu 2. června 2017 pohybuje lehce pod hranicí 9 000 Kč (verze s 2GB RAM) a to při použití veškerých technologií pod licencí GNU General Public Licence (GNU/GPL) jako open–source SW.

Vzhledem k jeho využití se domníváme, že tato částka je zbytečně vysoká, a tak jsme se rozhodli zaměřit náš výzkum na tvorbu obdobného zařízení, avšak při vynaložení značně nižších pořizovacích nákladů. S ohledem na stanovení nižší pořizovací ceny je nutné hned zpočátku našeho výzkumu říci, že rovněž předpokládáme nižší výkon námi zvoleného řešení. Z tohoto důvodu předběžně očekáváme nasazení zařízení do méně náročných míst, jakými mohou být například běžné domácnosti či různé rekreační objekty (chalupy, chaty a jiné) obsahující menší počet zařízení pracujících v rámci jedné lokální sítě (LAN).

1.1.2 Parametry

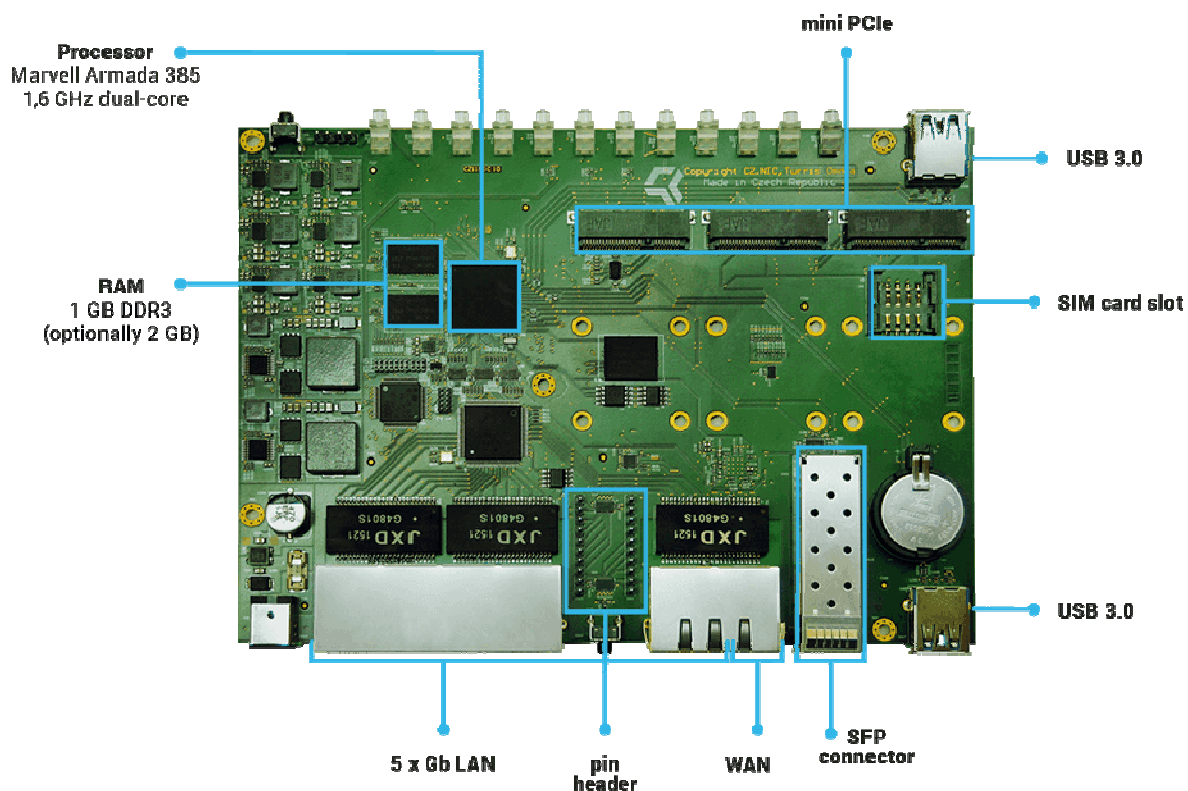
Oficiální zdroj [9] uvádí jednotlivé vlastnosti řešení realizovaného odborníky z CZ.NIC. Jako první ze tří základních prvků, které výrobce uvádí, je využití čistě otevřeného HW se svobodným OS (open–source). Docílení vysokého výkonu je potom

zajištěno využitím výkonného dvoujádrového procesoru ARM na 1,6 GHz s výkonem blízkým standardním PC. Třetím stěžejním bodem je dle výrobce bezpečnost, která je zajištěna díky výchozí konfiguraci, jednoduchému nastavení a automatickým aktualizacím. Další vlastnosti zmíněné na oficiálních webových stránkách produktu jsou potom znázorněny v následující tabulce:

TABULKA 1: OBECNÉ VLASTNOSTI TURRIS OMNIA [9]

Turris Omnia	
CPU	1,6 GHz dvoujádrový ARM
RAM	1 GB DDR3 (volitelně 2GB)
Úložiště	8GB flash
LAN	5x Gbit port
WAN	1x Gbit port
SFP	ANO
USB	2x USB 3.0
Mini PCI Express	2x
mSATA / Mini PCI Express	1x
Wi-Fi (Mini PCIe)	3x3 MIMO 802.11ac
	2x2 MIMO 802.11b/g/n

Výše zmíněný zdroj rovněž dodává, že Turris Omnia má dále slot na SIM kartu, hodiny reálného času zálohované baterií, kryptočip pro bezpečné generování náhodných čísel, stmívatelné RGB diody s GPIO, I²C, SPI a další.



OBRÁZEK 1: ZÁKLADNÍ DESKA SMĚROVAČE TURRIS OMNIA [9]

Automatické aktualizace

Z našeho pohledu velice zajímavou a nadstandardní bezpečnostní vlastností tohoto směrovače je možnost jeho automatických aktualizací. Zcela se ztotožňujeme s myšlenkou uvedenou ve zdroji [9], jež říká, že jedním z největších bezpečnostních problémů v dnešních směrovačích je fakt, že je v nich neaktualizovaný firmware. Přitom nové bezpečnostní hrozby se objevují téměř neustále. Tato myšlenka je vskutku jednoduchá, stačí si jen představit osobní počítač, který nikdo několik let neaktualizuje. Tuto skutečnost se rozhodli tvůrci směrovače změnit zavedením SW podpory v rámci automatických aktualizací systému a to po celou dobu životnosti směrovače. Výrobce rovněž dodává, že zmiňované aktualizace probíhají na pozadí, avšak v případě zájmu je možné nakonfigurovat systém tak, aby uživatele o jednotlivých aktualizacích informoval.

Využití svobodného software

Turris Omnia používá OS založený na GNU/Linux open–source distribuci OpenWrt. Bohužel je nad rámec této práce se tímto OS zabývat podrobněji. Pro více informací vřele doporučujeme zdroj [14]. Uživatelům je k dispozici systém, ke kterému se mohou chovat téměř obdobně jako k jinému GNU/Linuxovému serveru – jednoduše instalovat další SW, měnit základní konfiguraci a tak dále. Volba svobodného OS tak nijak neomezuje uživatele při práci se směrovačem. Navíc všechny zdrojové kódy jsou veřejné. Tvůrci rovněž dodávají, že budou jedině rádi, pokud uživatelé budou dále sdílet svá případná vylepšení skrze webový projekt GitHub.

Vysoce propustná síťová konfigurace

Další zajímavou vlastností je vysoce propustná síťová konfigurace, jelikož vnitřní uspořádání síťového zapojení Turrisu Omnia je mezi domácími routery unikátní. Řešení je docíleno tak, že procesor nabízí rovnou tři gigabitová rozhraní. Z tohoto důvodu má Omnia oddělený WAN port a switch, který se stará o další LAN porty. Tento port je rovněž připojen pomocí dvou gigabitových linek. Díky tomu mohou uživatelé Turris Omnia využívat gigabitovou rychlost v plně duplexním režimu při přeposílání mezi WAN a LAN. Výrobce rovněž uvádí možnost oddělení LAN portu pomocí VLAN. Vystává tak možnost využití vlastní gigabitové kapacity bez omezení provozem na ostatních portech.

Sdílené síťové úložiště

Celkem zajímavou možností je využití směrovače jako sdíleného síťového úložiště (NAS). Výrobce uvádí, že většina směrovačů je vybavena USB portem, kterým lze připojit NAS. Této možnosti však není drtivá většina schopná vzhledem k nedostatku výkonu pro využití plné rychlosti připojeného disku. Jelikož Turris Omnia disponuje dvěma USB 3.0 porty a výkonnou centrální procesorovou jednotkou (CPU), tak nabízí takový výkon, který je srovnatelný s dedikovanými NAS úložišti. Dostáváme se tak do situace, kdy směrovač může nejen vykonávat standardní práci, ale zároveň sloužit jako obsluha NAS.

Virtuální server

Jelikož Turris Omnia nabízí vysoký výkon CPU a RAM, rozhodli se jeho tvůrci integrovat do základu systému přednastavený virtuální server. V moderních serverech se často setkáváme s virtualizačními technologiemi pomáhajícími organizovat nainstalovaný SW. Přidaný virtuální server se chová podobně jako další počítač, připojený do LAN a umožňuje uživatelům směrovače spustit jakoukoliv distribuci OS. Experimentování s novým SW je tak snadné a bezpečné.

Operační paměť

Z důvodu hlavního účelu dnešních domácích směrovačů je zřejmé, proč výrobci těchto zařízení neobohacují směrovače o vysokokapacitní RAM. Nadměrné kapacity by nebylo využito, a proto se většina výrobců snaží především minimalizovat náklady na výrobu. CZ.NIC uvádí ve zdroji [9], že drtivá většina dnešních domácích směrovačů nemá k dispozici dokonce ani 128MB RAM. Pro normální použití směrovače tato kapacita vcelku stačí, avšak při provozu webového serveru DLNA či virtuálního serveru, každý uživatel jistě uvítá více paměti. Z tohoto důvodu se rozhodli výrobci Turris Omnia vytvořit dvě verze, jednu levnější s 1GB RAM a druhou dražší s 2GB RAM. Ať už uživatel pracuje s 1GB či 2GB verzí, lze v obou případech tvrdit, že se dané řešení přibližuje spíše malému serveru než obyčejnému směrovači.

Small form–factor pluggable konektor

Turris Omnia dále disponuje Small form–factor pluggable (SFP) konektorem, o kterém lze najít podrobnější informace ve zdroji [15]. Ten se využívá především v páteřních počítačových sítích, kde jsou jednotlivé uzly současného Internetu takřka bezvýhradně propojeny optickými vlákny. Tato technologie se dnes však často používá i na lokální přípojky a může vést až k uživateli směrovače. Pokud uživatel Turris Omnia patří mezi ty šťastlivce, kteří mají tento typ připojení, Turris Omnia může plně nahradit zařízení, která mu připojení zajišťovala doposud. Stačí vložit správný SFP modul [9].

Záložní připojení

Další vlastností, kterou v rámci projektu Turris Omnia zmíníme, je existence záložního připojení. Tento prvek uvítají ti uživatelé, kteří potřebují Internetové připojení za každé situace (např. v době kdy dojde k výpadku Internetu poskytovaného od dodavatele). CZ.NIC zmiňuje, že uživatel Turris Omnia může pomocí USB nebo Mini PCIe LTE modemu tento problém jednoduše vyřešit. Řešení navíc nabízí slot pro SIM kartu pro případ, že není přímo v modemu. Tímto způsobem lze vytvořit záložní připojení. Uživatel tak může jednoduše nastavit aktivaci záložního připojení v situaci, kdy hlavní připojení není dostupné.

HoneyPots

Turris Omnia disponuje rovněž tzv. Honeypoty. Zdroj [16] je definuje jako pečlivě sledované síťové nástrahy, sloužící k více účelům. Můžeme zmínit například rozptýlení útočníka před útokem na cennější stroje v dané LAN, případně upozornění na nové útoky či jiné formy vykořisťování. Tato technologie umožňuje hloubkové vyšetření protivníků během a po vykořisťování honeypotu. Jedná se tak o vysoce flexibilní bezpečnostní nástroj s různými aplikacemi. Nutno dodat, že se nejedná o opravu žádného problému, ale spíše o prevenci, odhalení či shromažďování informací o podezřelých aktivitách v LAN. Turris Omnia nabízí ověření informací o potencionálních internetových útočnicích. CZ.NIC provozuje dané honeypoty na svých serverech. Ty simulují napadnutelný systém. Uživatel potom může snadno nastavit svůj Turris Omnia tak, aby určitý provoz přesměřoval do honeypotů, fungujících na serverech CZ.NIC. Jelikož se zde všechny akce zaznamenávají, je možné sledovat útočnickovu aktivitu. Toto řešení je navíc naprosto bezpečné, neboť je honeypot realizován na serveru výrobce, nikoliv na uživatelově směrovači [9].

Kryptočip

Existuje celá řada definic termínu entropie (tzv. míry neurčitosti systému) a je nad rámec této práce se jí dále zabývat. Pro případné přiblížení dané problematiky vřele doporučujeme zdroje [17] či [18].

Autoři Turrís Omnia jsou si vědomi bezpečnostního rizika spojeného s generováním pseudonáhodných čísel, založených na nedostatku entropie. Právě z tohoto důvodu se rozhodli do jejich řešení integrovat speciální kryptočip tak, aby generovaná čísla byla skutečně náhodná. Zdroj [9] mimo jiné uvádí, co již bylo mnohokrát ukázáno. Vestavěná zařízení, jako právě domácí směrovače, nemají dostatek zdrojů entropie.

Hodiny reálného času zálohované baterií

Obdobně jako v případě nedostatku entropie vystávají potenciální bezpečnostní hrozby rovněž při absenci informace o přesném aktuálním čase. Při neznalosti této informace není možné správně ověřovat elektronické podpisy, protože mají všechny certifikáty s omezenou délkou platnosti. To platí jak pro HTTPS certifikáty, tak i pro DNSSEC. Z tohoto důvodu výrobci obdařili směrovač čipem hodin reálného času RTC modulem poháněným baterií, díky čemuž je správný čas zachován i po výpadku napájení. Tímto způsobem se lze bránit různým typům útoků, které využívají špatně nastaveného systémového času [9].

Flash paměť

CZ.NIC ve zdroji [9] uvádí, že Turrís Omnia má k dispozici 8 GB flash paměti, přičemž předpokládá, že u běžných domácích routerů je kapacita paměti flash přibližně mezi 4 až 32 MB. Takovýto rozdíl umožňuje uživatelům Turrís Omnia bez problémů nainstalovat značné množství aplikací dle jejich potřeby.

Rozšířitelnost

Směrovač lze využít k mnoha dalším účelům. Toho si je výrobce vědom a proto na svých webových stránkách [9] uvádí možnosti rozšíření řešení. Například lze zmínit DLNA Server s přidáním DVB-T tuneru a sdílet s počítači v síti televizní vysílání. Dále je možné rozšířit řešení přidáním zvukové karty s využitím směrovače jako Internetového rádia či realizace alarmu proti zlodějům při použití webové kamery. V případě narušení uživatelského domu začne směrovač zasílat fotografie na email jeho správce. Tvůrci navíc uvádí, že existuje rozsáhlá dokumentace popisující jednotlivá výše zmíněná řešení. Tato dokumentace je údajně k dispozici všem uživatelům směrovače.

Ostatní

Na rozdíl od většiny směrovačů, které mohou rozšířit svoji kapacitu paměti pouze pomocí USB disku nabízí Turris Omnia rozhraní mSATA, které lze využít pro připojení SSD disku. Toto řešení umožňuje využití směrovače jako opravdového serveru.

I přes primární cíl vytvoření směrovače určeného především pro dosažení opravdu vysokých rychlostí připojení, která je možné realizovat pouze díky fyzickým spojům (drátové připojení či optika) si je výrobce dále vědom i nezbytnosti integrace bezdrátových sítí, v níž CZ.NIC nabízí nejnovější technologie a tak Turris Omnia disponuje dvoupásmovou Wi-Fi s 3x3 MIMO 802.11ac pro rychlé připojení a 2x2 MIMO 802.11b/g/n pro starší nenáročná zařízení [9].

2 Popis základní problematiky

Úvodem kapitoly bychom rádi poznamenali, že je nad rámec této práce zabývat se detailní analýzou jednotlivých termínů úzce souvisejících se směrováním. Z tohoto důvodu je očekáváno, že čtenáři námi prováděného výzkumu mají alespoň základní znalost problematiky počítačových sítí, směrování a OS.

Směrovač

Zdroj [19] popisuje směrovač jako základní stavební prvek propojující jednotlivé LAN do jedné WAN. Další zdroj [20] říká, že směrování umožňuje nalézt „určitou“ cestu k danému cíli, přičemž efektivita vyhledávání je dána použitým algoritmem (nejčastěji BELLMAN–FORDův a DIJSKRův algoritmus). V příloze B lze nalézt znázorněný základní princip práce směrovače na jednotlivých vrstách OSI modelu.

Směrovací tabulka

Jedná se o seznam adres LAN konkrétního směrovače. Existují dvě možnosti tvorby obsahu této tabulky. Obsah lze vytvořit buď staticky (administrátor zadá ručně adresy sousedních LAN) nebo dynamicky (administrátor nakonfiguruje dynamický směrovací protokol) [21].

Operační systém

Jsou ty programové moduly ve výpočetním systému, jež ovládají řízení prostředků, kterými je tento výpočetní systém vybaven. OS poskytuje uživateli základní propojení s počítačem [22]. Primární funkce OS spočívá v tom, že poskytuje podporu pro realizaci počítačových programů [23]. OS je programové vybavení počítače, které je nezbytné pro jeho provoz [24].

Jádro operačního systému

Jádro OS neboli kernel je zcela nenahraditelnou součástí každého OS [23]. V drtivé většině dnešních OS plní funkce přidělování systémových zdrojů, spouštění programů, přidělování času mikroprocesoru programům, které jsou právě spuštěny a podobně.

Operační systém GNU/Linux

Obecně lze říci, že OS GNU/Linux je oproti jiným OS spolehlivější, bezpečnější, zcela ovladatelný, variabilnější a ve většině případů pro všechny uživatele zcela legálně zdarma [6]. Především stabilita OS je pro každý směrovač vysoce důležitým faktorem, protože se všeobecně očekává, že směrovač bude spolehlivě fungovat každý den v týdnu, 24 hodin denně.

Drtivou většinu OS lze získat v distribucích – jedná se o souhrn nejrůznějších aplikací sloužících k práci s počítačem [25].

2.1 Softwarové směrovače

SW směrovač je směrovací platforma, někdy též označovaná jako směrovací démon, určená pro většinu OS typu GNU/Linux. Zdroj [26] popisuje SW směrovače jako uživatelské aplikace běžící mimo jádro OS.

Nyní je vhodné vysvětlit, proč je nezbytné se směrovacími démony vůbec zabývat. Hlavním úkolem je především řízení obsahu směrovací tabulky OS. Znamená to, že OS sám o sobě umí přeposílat pakety cizím stanicím (forwarding) jelikož má jednoduchou směrovací tabulku, avšak nerozumí směrovacím protokolům a další složitější síťové problematice, kterou za něj řeší již zmínění směrovací démoni.

Tyto nástroje tak umožňují správci definovat jednotlivé směrovací protokoly, podle kterých se později OS rozhoduje.

Jelikož je směrovacích řešení více, argumentace výběru konkrétních směrovacích démonů závisí na mnoha faktorech. Mezi nejznámější komerční směrovací platformy patří

IOS od firmy Cisco Systems, Inc. Jako další příklady můžeme jmenovat směrovací platformy BIRD, gated, OpenOSPF, OpenBGP, Quagga, routed a XORP. V rámci našeho výzkumu jsme se rozhodli o komparativní analýzu takových směrovacích platform, které jsou všeobecně distribuované pod open-source licencí GNU/GPL, jejíž znění lze nalézt ve zdroji [27].

Prvním démonem je Quagga, která je v oblasti směrování při využití OS GNU/Linux již zaběhlým standardem. Tento démon má rozsáhlou historii a je všeobecně jedním z neznámějších a nejvyužívanějších GNU/GPL směrovacích projektů svého druhu.

Druhou námi zkoumanou směrovací platformou je relativně mladá, čistě česká směrovací platforma BIRD, která má nejen velké ambice, ale i několik úspěchů při překovávání svého výše popsaného, známějšího konkurenta.

Existuje celá řada dalších směrovacích démonů, jejichž rozbor je daleko nad rámec této práce. V rámci našeho výzkumu se tak budeme nadále zabývat pouze výše zmíněnými směrovacími platformami Quagga a BIRD.

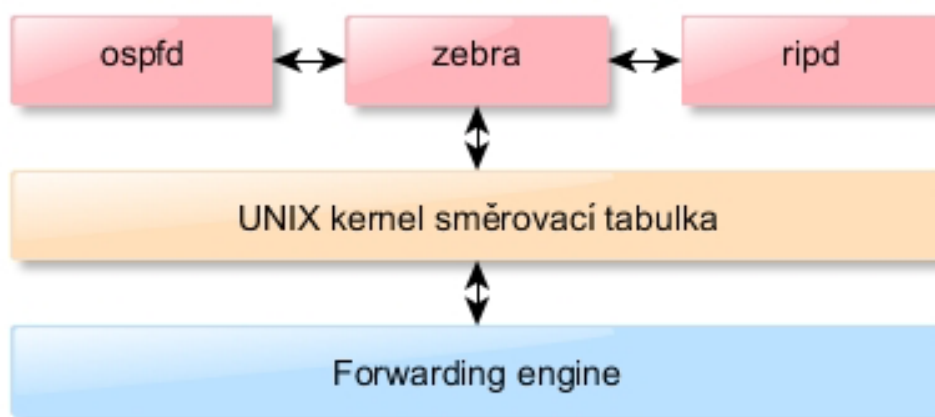
2.1.1 Quagga

Zdroj [28] popisuje platformu Quagga jako démona určeného pro unix-like OS poskytujícího směrovací služby pro TCP/IP sítě.

Tvůrcem tohoto projektu je Kunihiro Ishiguro, který vytvořil Quaggu jako vedlejší větev projektu GNU Zebra. Quagga kromě statického směrování umožňuje implementaci směrovacích protokolů RIPv1, RIPv2, RIPv6, OSPFv2, OSPFv3 a BGPv4+ (zahrnujícího rodinnou adresaci pro multicast a IPv6). Démon dále externě podporuje směrovací protokol IS-IS jak ve verzi pro IPv4, tak i pro IPv6. Quagga mimo jiné nově podporuje Optimized Link State Routing (OLSR) protokol skrze plugin pro bezdrátové směrování, Multiprotocol Label Switching (MPLS) Label Distribution Protocol (LDP), jako derivát z ldpd určeného pro OS OpenBSD a dvousměrovou detekci směrování díky podpoře Bidirectional Forwarding Detection (BFD).

Z výčtu podporovaných směrovacích protokolů je patrné, že tento démon podporuje jak jejich IPv4, tak i IPv6 verze [29]. Díky tomuto zdroji dále víme, že podpora OSPFv3 a IS-IS směrovacích protokolů je v současné době v tzv. beta fázi. Autor zmiňuje, že si je vědom existence nedostatků při implementaci OSPFv3 a IS-IS při použití jejich verzí pro IPv6.

Mezi významné výhody této platformy patří především rozsáhlá komunita uživatelů, díky níž existuje velké množství nejrůznějších relativně snadno získatelných informací vztahujících se k této platformě. Dokonce se dá říci, že hledá-li někdo směrovací GNU/GPL SW pro OS GNU/Linux, potom s největší pravděpodobností narazí při prvním kroku právě na tohoto démona.



OBRÁZEK 2: ARCHITEKTURA SMĚROVACÍ PLATFORMY QUAGGA (ZPRACOVÁNO DLE [28])

Na výše uvedeném obrázku 2, můžeme vidět základní architekturu Quaggy. Platforma se skládá z dílčích démonů specifických pro každý směrovací protokol. Během našeho výzkumu se budeme nadále zabývat jen některými z dílčích démonů tvořících Quaggu jako celek. Těmito dílčími démony jsou Zebra, Ripd a Opsfd.

Dílčí démon Zebra tvoří hlavní řídicí prvek mezi směrovací tabulkou OS a jednotlivými dílčími směrovacími démony orientovanými na konkrétní technologii směrování. To znamená, že tento démon řídí distribuci směrovacích informací mezi jednotlivými dílčími démony a OS. Jedná se tak o nepostradatelnou součást Quaggy.

Zebra dále umožňuje základní konfiguraci jednotlivých síťových rozhraní, statické směrování a další všeobecná základní směrovací pravidla.

Další dílčí démoni jsou potom pojmenováni na základě konkrétního směrovacího protokolu, který umožňují naimplementovat – Ripd (RIP démon) slouží pro konfiguraci směrovacího protokolu RIP a Ospfd (OSPF démon) k nastavení pravidel pro směrovací protokol OSPF.

2.1.2 BIRD

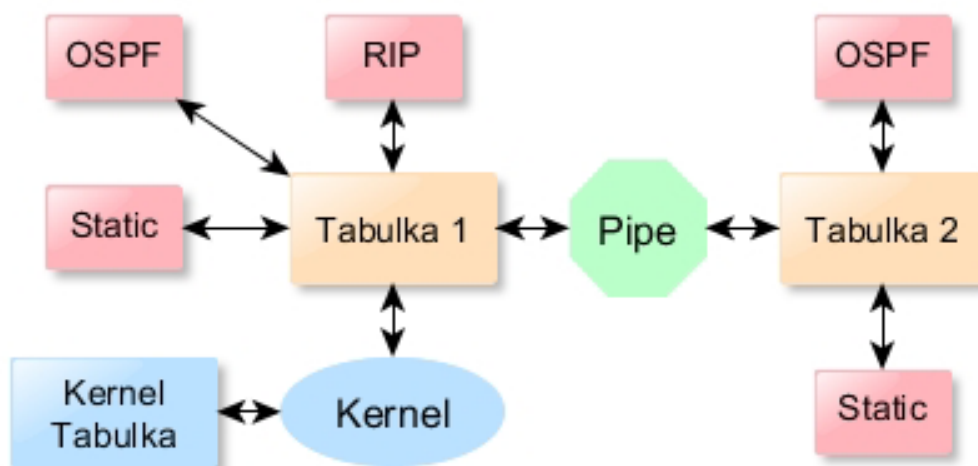
Bird Internet Routing Daemon (BIRD) [30] je další směrovací platforma, kterou se budeme dále zabývat. Jedná se o relativně novou českou směrovací platformu, určenou pro všechny unix-like OS. Počátek projektu se datuje k roku 1999, poté byl však tento projekt uspán a na přelomu let 2008/2009 plně obnoven v rámci laboratoří CZ.NIC [31].

Démon byl vytvořený jako školní projekt na Matematicko-fyzikální fakultně Univerzity Karlovy v Praze. Zdroj [32] dále uvádí, že současný tým, spravující tento SW směrovač tvoří Martin Mareš, Ondřej Zajíček a Ondřej Filip, který je zároveň ředitelem organizace CZ.NIC jak dodává zdroj [32]. Obdobně jako Quagga začal BIRD nově podporovat rovněž MPLS. Toto relativně nové rozšíření má na starost Jan Matějka. Původními členy vývojového týmu byly Libor Forst a Pavel Machek. Poslední z výše uvedených zdrojů dále popisuje tuto platformu jako alternativu k běžně používanému démonu Quagga, který byl již popsán v předchozí kapitole.

Autor platformy uvádí ve zdroji [32], že projekt BIRD je určen především pro plně dynamické směrování. Platforma je primárně vyvíjena pro OS GNU/Linux, avšak také pro další unix-like OS. Obdobně jako Quagga je BIRD distribuován pod licencí GNU/GPL. Zdroj dále uvádí, že BIRD podporuje jak statické, tak i dynamické směrování a to prostřednictvím směrovacích protokolů BGP, RIP, OSPF, BFD a nově i pomocí směrovacího protokolu Babel. BIRD obdobně jako Quagga podporuje jak IPv4, tak i IPv6 směrování.

Podobně jako u platformy Quagga nabízí tato směrovací platforma výpis (avšak nikoliv ovládání) nastavení skrze příkazovou řádku (CLI). Ta má zcela odlišnou syntaxi, která je naprosto odlišná od té, kterou používají přístroje od výrobce Cisco Systems, Inc. (jež tvůrci SW směrovače Quagga napodobují).

Mezi výhody můžeme uvést především programovatelné filtry, uspořádané konfigurační soubory, obsáhlou dokumentaci, nízké paměťové a CPU nároky, přehledný kód a rovněž podporu více nezávislých směrovacích tabulek [30].



OBRÁZEK 3: ARCHITEKTURA SMĚROVACÍ PLATFORMY BIRD [AUTOR]

Na obrázku 3 uvedem výše můžete vidět architekturu tohoto SW démona. Na rozdíl od Quaggy lze v rámci jednoho konfiguračního souboru vytvořit více směrovacích instancí, tedy definovat více druhů dynamického směrování.

Za zmínku jistě stojí „protocol pipe“ umožňující konstruovat spojení mezi více směrovacími tabulkami. Díky němu lze řídit zásady směrování (policy routing), kdy manipulace s jedním paketem nezávisí pouze na jeho cílové adrese, ale také na jeho zdrojové adrese, zdrojovém rozhraní, typu protokolu a dalších podobných parametrech [32].

Jako nevýhody lze potom zmínit absenci podpory multicastu, striktní oddělení IPv4 a IPv6 režimu a již zmíněná odlišná syntaxe konfigurace. Díky zdroji [28] rovněž víme, že BIRDu chybí podpora relativně běžného protokolu RIPng.

3 Analýza hardware

Pro výzkum, který jsme se rozhodli realizovat, je třeba využití takového HW, který splňuje celou řadu podmínek potřebných k implementaci, integraci a konfiguraci jednotlivých kroků realizujících námi vytyčený výsledný produkt – IoT směrovač. Následující kapitola popisuje dva námi zvolené jednočipové počítače, které jsme se rozhodli využít. Existuje celá řada dalších (nejen jednočipových) počítačů, které rovněž vyhovují požadavkům potřebným při tvorbě funkčního domácího směrovače, fungujícího na principu IoT. Množina všech dostupných HW řešení, které lze potencionálně využít, je však tak rozsáhlá, že je nad rámec této práce zmapovat všechny HW, který může být použit. Argumentací toho, proč jsme se rozhodli vybrat právě námi zvolené řešení, lze nalézt v následujících kapitolách, zabývajících se podrobně jednotlivými jednočipovými počítači.

3.1 HummingBoard

Následující kapitola je věnována první z námi zvolených HW platform. Postupně zde popíšeme výrobce a konkrétní typ jednočipového Mini PC, který jsme využili během našeho výzkumu.

3.1.1 SolidRun Limited

Izraelská společnost, založená v roce 2010, je údajně globálním lídrem ve vývoji a výrobě vysoce výkonných, ale zároveň energeticky šetrných systémů na modulech (SoM), jednočipových (SBC) a výrobních počítačů.

SolidRun, Ltd. (dále jen SolidRun) popisuje svá řešení jako inovativní, kompaktní a zapouzdřená, založená na architekturách ARM a x86. Produkty v sobě zahrnují obsáhlou databázi balíčků, ovladačů a podporují majoritní OS.

SolidRun je rovněž hrdým členem komunity volně dostupného SW (open–source SW comunity) a silný promotér open–source principů [33].

Oficiální zdroj rovněž dodává, že misí této společnosti je zjednodušení a zkrácení času vývoje a nasazení řešení, pracujících na principech IoT. Tato řešení dále konkrétně popisuje zdroj [33] jako kombinaci HW, SW, služeb a ekosystému zjednodušujícího vývoj a nasazení aplikací IoT. Globální zákaznickou základnu společnosti zahrnuje celá řada „original equipment manufacturer“ (OEM) produktů a to v nejrůznějších segmentech. Jako příklady užití výrobků této společnosti zdroj [33] uvádí projekty pro inteligentní automobilovou dopravu, letecké drony, projekty chytrých měst (Smart City), projekty domácí automatizace (Smart Home), industriální automatizaci známou též pod označením Průmysl 4.0 (čtvrtá průmyslová revoluce), multimédia, různé prodejní automaty (včetně bankomatů) a přístupové body do počítačových sítí. SolidRun dále podporuje projekty fungující na protokolu Server Message Block a NAS.

Zdroj [33] dále uvádí, že výkonný řídicí tým tvoří pouze čtyři členové, jimiž jsou spoluzakladatel společnosti Rabeeh Khoury, výkonný ředitel Atai Ziv, výkonný předseda Kossay Omary a viceprezident pro operace a nákup Vily Lupo.

Mezi služby a schopnosti společnosti patří projekty jednorázového inženýrství týkající se jednorázových nákladů na výzkum, návrh, vývoj a testování zcela nového a originálního produktu. SolidRun dále nabízí asistenci při celkovém vývoji nové společnosti, která by potencionálně využívala řešení nabízených právě touto izraelskou společností. Vysoké kvality výroby je dosaženo použitím moderně vybavené montážní linky.

Mezi nejznámější klienty společnosti patří společnost Google, Dell a Sony [33]. Řešení SolidRun jsou rovněž využívána u světoznámého výrobce letadel – společnosti Boeing. Jako příklad klientů, zabývajících se telekomunikací, můžeme jmenovat společnosti Ericsson či Airspan. Dále pak nesmíme opomenout společnosti zabývající se řešením chytrého domu jako společnost Homey, pracující na řešení propojení jednotlivých zařízení v bytě či společnost Domadoo, zabývající se automatizací domova. Na závěr můžeme jmenovat společnost Dov-e zabývající se zapojením spotřebitelů prostřednictvím

mobilního telefonu a jako příklad klienta zabývajícího se průmyslovou automatizací můžeme jmenovat například společnost Unitronics.

3.1.2 Kategorizace produktů společnosti SolidRun

Ve svém produktovém katalogu [34] firma nabízí celou řadu nejrůznějších HW řešení. Nadále se budeme zabývat pouze stručným seznámením s jednotlivými produkty. Klasifikaci lze provést na základě dvou odlišných úhlů pohledů – třídění dle tzv. rodiny produktů a rovněž dle typu produktu.

Třídění dle rodin produktů

Jak je znázorněno v příloze C je členění dle rodin jednotlivých produktů rozděleno na tři odlišné skupiny, které se odvíjí od použitého čipu (či jak je uvedeno v produktovém katalogu dle technologického partnera) firmy. Těmito třemi rodinami jsou tzv. NXP Family, Marvell Family a Intel Family. Dle zdroje [33] bude rodina i.MX6 ve velmi brzké době doplněna i o novou variantu i.MX8.

Každá ze tří uvedených rodin je nadále tvořena několika členy (skupinami HW řešení). Tito členové jsou dále specifikováni dle konkrétního použitého HW (nejčastěji přímo dle použitého mikroprocesoru či velikosti kapacity operační paměti).

Členy NXP Family jsou iMX6 SOM, HummingBoard a CuBox–i. Marvell Family je složena z ARMADA SOM, ClearFog SBC, Marvell MacchiatoBIN a SolidNet OCP–8K. Poslední z uvedených, Intel Family, čítá pouze dva členy, jimiž jsou Braswell SOM a SolidPC Q4 SBC.

Tito členové rodin jsou dále nabízeni v jednotlivých verzích, které jsou specifikovány dle konkrétního použitého HW.

Člen iMX6 SOM z NXP Family se nadále dělí dle použitého mikroprocesoru na Single/Dual Core (SOM i1, SOM i2 a SOM i2eX) či Quad Core (SOM i4Pro). HummingBoard lze rozdělit do čtyř verzí, jimiž jsou: HummingBoard Base, HummingBoard Pro, HummingBoard Gate a HummingBoard Edge. SolidRun dále nabízí CuBox–i

ve variantách CuBox-i1, CuBox-i2, CuBox-i2eX, CuBox-i4Pro a CuBox-i 4x4. Jedinou specifikací ARMADA SOM pocházejícího z Marvell Family, je verze SOM A388. ClearFog SBC je tvořen dvěma zástupci – ClearFog Base a ClearFog Pro. Marvell MacchiatoBIN a SolidNet OCP-8K existují pouze v jediné verzi, jejíž název je totožný s názvem člena rodiny. Intel Family obsahuje dva členy, jimiž jsou Braswell SOM a SolidPC Q4 Carrier Board. Na rozdíl od SolidPC Q4 Carrier Board, jež existuje pouze v jediné verzi je jeho kolega Braswell SOM nabízen ve dvou specifikacích, jako SOM IB8000 a SOM IB3710, který je navíc nabízen ve 4GB i 8GB RAM subverzi.

Třídění dle typu produktu

Dělení dle typu je obdobně jako u rodin tvořeno třemi subkategoriemi. Tyto kategorie jsou SOMs & COMs, SBCs & FanlessPCs a Network & Servers. První jmenovaná skupina SOMs & COMs se dále dělí na iMX6 SOM, ARMADA SOM a Braswell SOM. Druhá množina typů produktů SBCs & FanlessPCs je potom tvořena z CuBox-i, HummingBoard, ClearFog, SolidPC Q4 a Marvell MacchiatoBIN. Poslední kategorie Network & Servers čítá pouze jediného člena, jímž je SolidNet OCP-8K Platform. Tito členové nadále existují ve verzích obdobných těm, které byly popsány výše.

3.1.3 Volba konkrétního produktu

Po přezkoumání všech aktuálně nabízených řešení, zmíněných v předchozí kapitole, vyvstává otázka, jaký konkrétní produkt společnosti SolidRun vybrat. Vzhledem k námi stanovenému cíli, se jeví jako ideální jedno z méně výkonných, levnějších řešení, které je však obdařeno dvěma či více, ideálně, GbE (gigabit ethernetovými) porty. Jmenujme například model Marvell MacchiatoBIN, který je obdařen rovnou třemi GbE porty, či libovolné řešení ClearFog obdařené dle verze buď dvěma či dokonce rovnou sedmi GbE porty. Posledním produktem, obsahujícím více než jeden GbE port je SolidPC Q4 Carrier Board. Bohužel k datu našeho výběru jednoho z nabízených řešení, nebyla nabídka společnosti SolidRun tak rozsáhlá, jak je tomu k dnešnímu datu. Z tohoto důvodu

jsme se tehdy rozhodli pro jednoportové, GbE řešení, jímž se stal HummingBoard Gate, spadající do NXP Family. Rádi bychom tímto poznamenali, že je zcela zjevné, že v době, kdy jsme hledali víceportový GbE Mini PC, si byl SolidRun vědom potencionální síly SW směrovačů a jejich budoucího propojování s IoT. Současně si společnost byla vědoma i toho, že nenabízela produkt s více ethernetovými porty, a tak v relativně krátké době po zakoupení jednoportového GbE řešení přišel SolidRun s první nabídkou víceportového GbE produktu. Bohužel pro nás již v době, kdy jsme disponovali zařízením HummingBoard.

HummingBoard Family

Jedná se o malý, výkonný, nízkonákladový Mini PC založený na architektuře ARM. HummingBoard umožňuje vytvářet inovativní řešení IoT, do kterého je primárně určen. HummingBoard je dle výrobce moderní, vysoce flexibilní platforma IoT, která zjednodušuje design a nasazení produktů v něm participujících [33].

SolidRun nabízí v rámci rodiny HummingBoard hned čtyři odlišné verze, jimiž se budeme v této kapitole nadále věnovat.

TABULKA 2: VERZE HUMMINGBOARDU [33]

	HummingBoard Base	HummingBoard Pro	HummingBoard Gate	HummingBoard Edge
SOM Model	NXP–Freescale i.MX6 based Solo to Dual Core SOM	NXP–Freescale i.MX6 based Solo to Dual Core SOM	NXP–Freescale i.MX6 based Solo to Dual Core SOM	NXP–Freescale i.MX6 based Solo to Dual Core SOM
Memory and Storage	Up to 2GB DDR3 uSD	Up to 2GB DDR3 uSD, mSATA*	Up to 2GB DDR3 uSD	Up to 2GB DDR3 uSD, eMMC (8GB), M.2 (2242)*
Connectivity	1xRJ-45** 2xHost USB 2.0	1xRJ-45**, 2xHost USB2.0, 2xHeader USB2.0,mPCIe–half size	1xRJ-45**, 4xUSB2.0,mPCIe with SIM card holder	1xRJ-45**, 4xUSB2.0,mPCIe with SIM card holder
Media	HDMI–Out, SPDIF, MIPI–CSI–2 Camera	HDMI–Out, LVDS, SPDIF, Analog Audio, MIPI–CSI–2 Camera	HDMI–Out, MIPI–CSI–2 and MIPI–DSI, Parallel Camera	HDMI–Out, LVDS, Analog Audio, MIPI–CSI–2 and MIPI–DSI, Parallel Camera (in GPIO header)
I/O	Reset Button, 26 pins GPIO Header	Reset Button, 26 pins GPIO Header, RTC, IR	Reset Button, 36 pins GPIO Header, RTC with Battery, mikroBUS™ click interface	Reset Button, 36 pins GPIO Header, RTC with Battery, IR
Power	5V, uUSB	5V, uUSB	7V–36V, 5.5mm in (Twist and Lock mechanism)	7V–36V, 5.5mm in (Twist and Lock mechanism)
Dimensions	85mmx56mm	85mmx56mm	102mmx69mm	102mmx69mm
Software	GNU/Linux	GNU/Linux	GNU/Linux	GNU/Linux
Environment	No Enclosure	No Enclosure	Metal Enclosure	Metal Enclosure
Price	From 73 USD	From 83 USD	From 83 USD	From 98 USD

HummingBoard Base

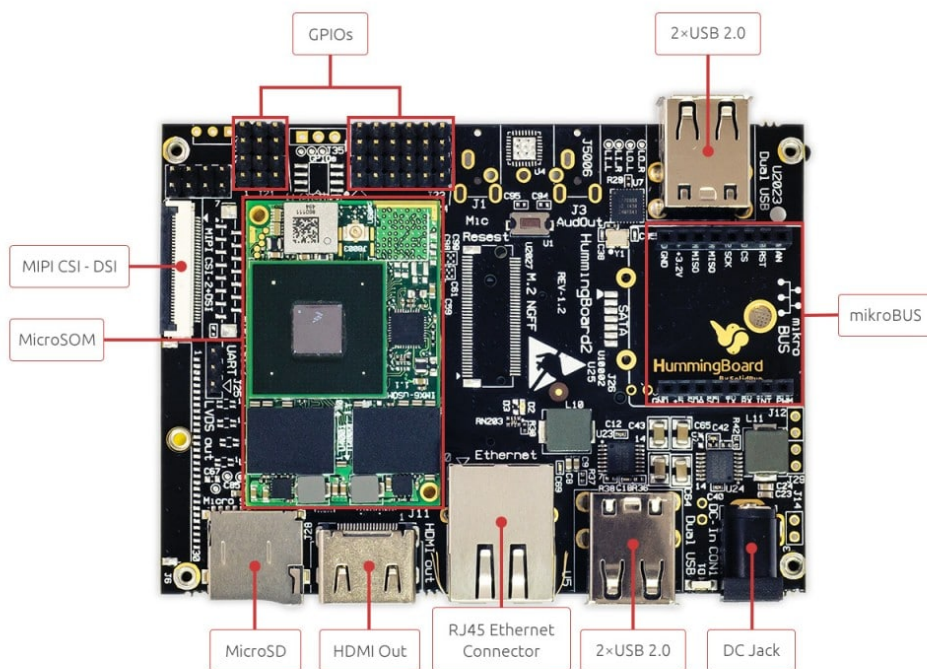
Jedná se o nejlevnější verzi, kdy je uživateli nabídnuto užitečné a ekonomické řešení vhodné jako ideální výchozí bod pro nízkovýpočtové potřeby v rámci jednoho zařízení. Tato verze je primárně zaměřena na využití souborového systému výhradně dostupného skrze slot pro Micro-SD paměťové karty společně s možností vykonávat základní vstupně/výstupní (GPIO) operace, zpracování médií a připojení k síti.

HummingBoard Pro

Verze nazvaná Pro nabízí cennou vývojovou platformu s pokročilými možnostmi (včetně několika způsobů ukládání paměti) v zabudovaném řešení. Na rozdíl od dříve zmíněného řešení Base se výrobce u této verze zaměřil na větší možnosti práce se souborovým systémem, například skrze dostupný mSATA port, či integraci LVDS displayového portu stejně tak jako rozšíření skrze další rozhraní základní desky (Mini PCIe, IR přijímač, konektor pro externí baterii RTC) [33].

HummingBoard Gate

Gate je první SBC zahrnující integrovaný mikroBUSTM socket nabízející potenciálnímu uživateli jednoduchou konfiguraci rozsáhlého množství doplňkových HW modulů (click boardsTM) nabízených partnerem společnosti SolidRun, firmou MikroElektronika. Konektor mikroBusTM je vhodný především pro ty vývojáře, kteří pracují na víceúčelových a modulárních produktech. Zdroje [33] a [35] dále popisují konektor mikroBUSTM a k němu pasující click boardTM jako ideální řešení pro splnění nejrůznějších potřeb uživatelů IoT. Dle zdroje [36], oficiálního výrobce těchto click boardTM modulů je v aktuální době k dispozici více než 347 různých druhů rozšiřujících desek standartizovaných pro použití skrze propojení s mikroBUSTM. Případné další informace jsou k dispozici v rámci specifikace tohoto standardu, který je dostupný v následujícím zdroji [37].



OBRÁZEK 4: HUMMINGBOARD GATE [33]

HummingBoard Edge

Poslední verze, kterou výrobce nabízí je ideální M2M platforma. Edge byl vyvinut tak, aby splňoval potřeby B2B (business to business) trhu jako vysoce výkonné, spolehlivé a inovativní zařízení. Díky tomu je tato nejvýkonnější verze HummingBoardu obdařena zcela novou sadou vlastností. Jmenujme například eMMC (8GB), mPCIe (poloviční nebo plná velikost), širokopsámový vstup 7V–36V, volitelná kovová skříň a další [33].

Specifikace modulového systému

Po zvolení konkrétní verze – HummingBoard Gate je dále třeba přesněji specifikovat, jakým SOM chceme náš Mini PC nechat „osadit“. Výrobce nabízí hned čtyři verze i.MX6 MicroSOM (i1, i2, i2eX a i4Pro). Primární rozdíly mezi jednotlivými verzemi, jsou k dispozici v níže uvedené tabulce (komplexní analýza jednotlivých rozdílů je specifikována podrobněji v příloze D).

TABULKA 3: STRUČNÁ SPECIFIKACE KONKRÉTNÍHO MIKROPROCESORU (PŘEVZATO ZE ZDROJE [33])

	SOM i1	SOM i2	SOM i2eX	SOM i4Pro
System On Chip	i.MX6 Solo	i.MX6 Dual Lite	i.MX6 Dual	i.MX6 Quad
Processor core	Single core ARM A9	Dual core ARM A9	Dual core ARM A9	Quad core ARM A9
Graphics Processing Unit	Vivante GC880	Vivante GC880	Vivante GC2000	Vivante GC2000
3D GPU Support	OpenGL ES1.1/2.0	OpenGL ES1.1/2.0	OpenGL ES1.1/2.0 OpenCL 1.1E	OpenGL ES1.1/2.0 OpenCL 1.1E
Memory	32 bit, 512MB DDR3 @ 800Mbps	64 bit, 1GB DDR3 @ 800Mbps	64 bit, 1GB DDR3 @ 1066Mbps	64 bit, 2GB DDR3 @ 1066Mbps
Supported External Storage	NOR–Flash, eMMC, SD/microSD, PCIe SSD	NOR–Flash, eMMC, SD/microSD, PCIe SSD	NOR–Flash, eMMC, mSATA, SD/microSD, PCIe SSD	NOR–Flash, eMMC, mSATA, SD/microSD, PCIe SSD
Camera interface port	2 Lane CSI	2 Lane CSI	4 Lane CSI	4 Lane CSI
S–ATA	–	–	Gen II, 3Gbps	Gen II, 3Gbps
Price	From 48 USD	From 66 USD	From 81 USD	From 115 USD

Jelikož hodláme náš budoucí směrovač nasadit v domácnosti, kde neočekáváme náročný síťový provoz a spíše než o výkon je naším kritériem nízká pořizovací hodnota, rozhodli jsme se pro MicroSOM i2 (i.MX6 Dual Lite), který SolidRun prodává za 67 USD. Mimo volbu SOM je dále třeba upřesnit, zda má zákazník zájem o pasivní chlazení (5 USD), paměťovou microSD kartu (10 USD), síťový adaptér (10 USD) a pouzdro (25 USD) na celé Mini PC.

Vzhledem k tomu, že má HummingBoard Gate pouze jedno síťové rozhraní RJ–45, víme již předem, že jej bude třeba osadit dalším externím rozhraním RJ–45 a z tohoto důvodu volíme variantu, obsahující většinu výše zmíněných doplňků (pasivní chlazení, microSD katu a síťový adaptér) kromě pouzdra, protože externí síťové rozhraní RJ–45

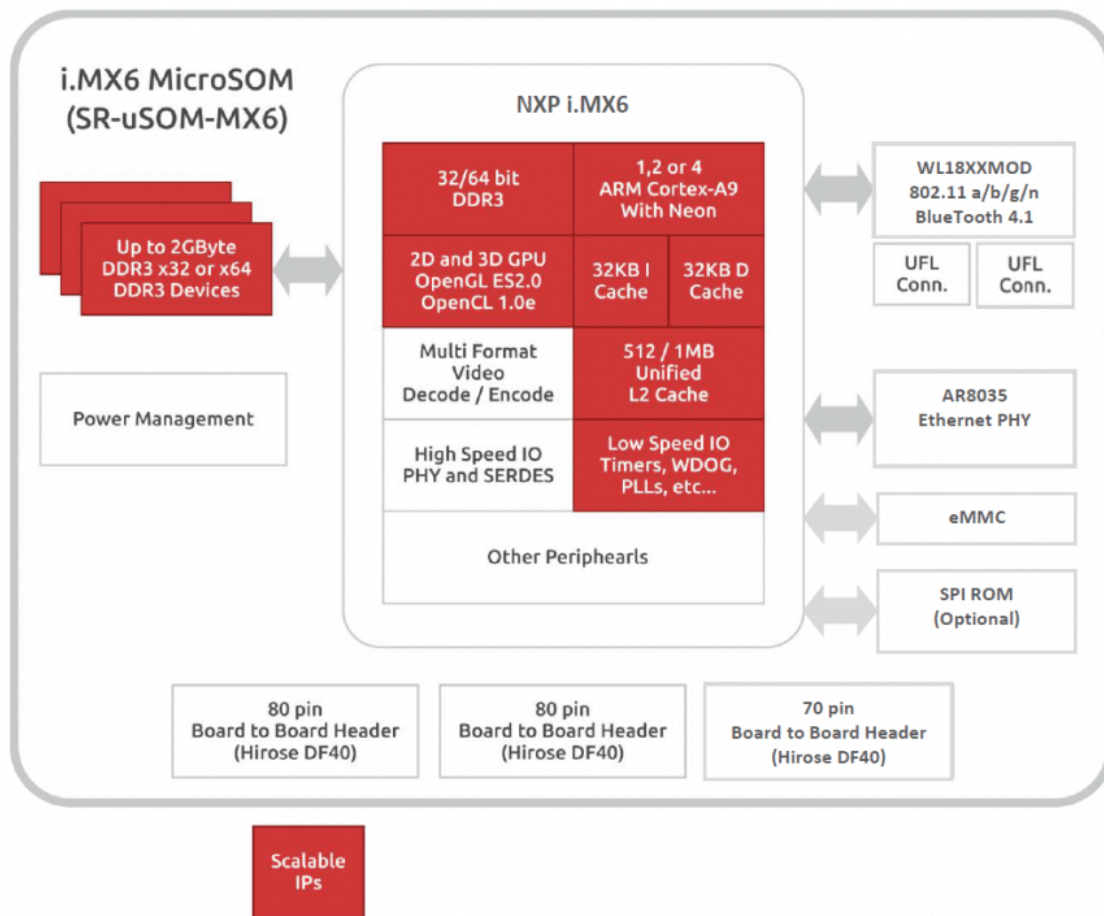
by se do něj již nevešlo. Celkově se tedy dostáváme na cenu 92 USD za kompletní Mini PC (uvedená cena je k 16.6.2017). Dne 11.11.2015, kdy jsme daný Mini PC pořizovali, jsme za stejnou konfiguraci zaplatili 110 USD.

NXP–Freescale i.MX6 MicroSOM i2 Dual Core SOM



OBRÁZEK 5: I2 SOM [38]

Vzhledem k naší volbě tohoto mikroprocesoru je nasnadě uvést alespoň obecný popis tohoto „srdce“ našeho budoucího Mini PC. Zdroj [38] uvádí, že se jedná o řešení rozšiřující kapacitu jednojádrového SOM i1 s dual lite CPU, které je navíc rozšířeno na 1GB DRAM a paměť integrovanou přímo na desce.



OBRÁZEK 6: BLOKOVÉ SCHÉMA I.MX6 MICROSOM [39]

3.2 Raspberry Pi

V této kapitole se budeme věnovat druhé z námi zvolených HW platform – Raspberry Pi. S ohledem na všeobecnou oblíbenost tohoto Mini PC se domníváme, že čtenář této práce již HW produkt zná, a proto zmíníme v rámci kapitoly pouze základní informace. V případě, že čtenář není Raspberry Pi známo, doporučujeme mu alespoň shlédnutí krátkého oficiálního videa dostupného ve zdroji [40] popisujícího velice stručně toto Mini PC. V rámci naší práce zde postupně zmíníme obecné informace o výrobci, jednotlivých v aktuální době dostupných typech Raspberry Pi a konkrétním modelu Raspberry Pi 1 Model B, který jsme využili v rámci našeho výzkumu.

3.2.1 Raspberry Pi Foundation

Zdroj [41] popisuje tuto nadaci jako skupinu lidí pracujících na tom, aby se moc digitálního světa dostala do rukou lidí po celém světě. Primárním cílem nadace je údajně docílení toho, aby stále více a více lidí rozumělo a naučilo se formovat náš neustále se rozvíjející digitální svět, umělo řešit problémy spjaté s digitálním světem a zároveň aby lidé byli více připraveni pro práci v budoucnosti.

K dosažení vytyčeného cíle nabízí nadace levné, vysoce výkonné počítače, které používají lidé k učení, řešení problémů a zábavě. Nadace rovněž poskytuje informační a vzdělávací služby, pomáhající více lidem přistupovat k výpočetní a digitální tvorbě. To vše prostřednictvím volně dostupných zdrojů.

Od uvedení prvního produktu nadace v únoru roku 2012 se do roku 2016 prodalo 12 miliónů počítačů Raspberry Pi. Nadace rovněž pomohla vytvořit vysoce početnou globální komunitu digitálních tvůrců a pedagogů, přičemž veškeré profity získané z nabízených HW řešení byly údajně použity k naplnění vzdělávacích cílů. Nadace je mimo jiné podporována celou řadou velkorysých partnerů a dárců, sdílejících stejný cíl [42]. Nadaci tvoří dle zdroje [41] 64 členů, přičemž díky zdroji [42] víme, že mezi význačné patří předseda David Cleavelly, výkonný ředitel Philip Colligan a obchodní ředitel Eben Upton.

3.2.2 Verze Raspberry Pi

Narozdíl od společnosti SolidRun je portfolio nabízených produktů nadace Raspberry Pi daleko skromnější. Nadace za 5 let své existence vydala celkem 4 modely odlišujících se vzájemně v celé řadě parametrů a mezi sebou dle jednotlivých generací.

Model A

Raspberry Pi Model A+ je na oficiálních stránkách [41] propagován především jako nízkospotřební varianta. Ve srovnání se starší verzí A disponuje Model A+ více GPIO piny, modulem pro Micro SD karty (na rozdíl od předchozího podporujícího větší SD karty), lepším výstupem zvuku a svojí velikostí. Nízké spotřeby bylo docíleno nahrazením

lineárních regulátorů přepínacími. Tím se tvůrcům podařilo snížit spotřebu o přibližně 0,5 až 1W.

Výrobce doporučuje tento model především do zabudovaných systémů a projektů, které jsou citlivé na spotřebu energie, přičemž nepotřebují žádné ethernetové rozhraní či několik USB portů.

Přibližná cena Modelu A+ by dle výrobce měla být okolo 20 USD [41].

Model B

V současné době pomyslnou vlajkovou loď nadace tvoří právě 3. generace tohoto modelu zvaná Raspberry 3 Model B nahrazující starší 2. generaci. Ve srovnání s ní disponuje 64 bitovým čtyřjádrovým mikroprocesorem Broadcom BCM2837 typu ARM Cortex-A53 pracujícím na frekvenci 1,2GHz, 802.11n bezdrátovým síťovým rozhraním, bluetooth 4.1 s technologií nízké spotřeby energie. Třetí generace má obdobně jako druhá k dispozici 1GB RAM, 4 USB porty, 40 GPIO pinů, full HDMI port, ethernetové rozhraní, kombinovaný 3,5mm audio jack, CSI rozhraní pro kameru a DSI port pro displej, slot pro paměťovou kartu Micro SD a VideoCore 4 3D grafické jádro [41].

Cena 2. i 3. generace tohoto modelu by se dle výrobce měla pohybovat kolem 35 USD [41].

Zero

Nejmenší, nejtenčí a nejdostupnější ze všech Raspberry Pi [43]. Rozměry tohoto modelu jsou pouhých 65mm na délku, 30mm na šířku a 5mm na výšku, přičemž cena standardní verze bez Wi-Fi se pohybuje kolem 5 USD. Za účelem nízké spotřeby disponuje Zero pouze jednojádrovým ARM mikroprocesorem, totožným s tím, který je použit v Modelu B+, pracujícím na 1 GHz frekvenci. Vzhledem k miniaturním rozměrům musela být omezena rovněž i kapacita operační paměti a to na 512 MB.

Nejnovější nabízená verze se nazývá Raspberry Pi Zero W a rozšiřuje o rok starší verzi zvanou Raspberry Pi Zero o zabudovaný Bluetooth a Wi-Fi modul [44].

Compute Module

Zdroj [45] informuje, že nadace představila třetí generaci tohoto modelu nazvanou jednoduše Compute Model 3 (CM3). Tato verze je dostupná ve dvou variantách – lite (CM3L) a standard (CM3). Společným rysem je použití nového systému na čipu (SoC), který má být údajně desetkrát výkonnější než první verze CM1. Nová verze je rovněž obdařena dvojnásobnou kapacitou paměti RAM a je obdobně jako její předchůdce navržena tak, aby pasovala do standardního slotu SO-DIMM, z čehož vyplývá, že CM3 byl vyroben za stejným účelem jako původní verze – za účelem poskytnout zákazníkům možnost vytvoření vlastní HW platformy za pomoci propojení skrze speciální základní desky. Nadace dále nabízí jednu takovou základní desku pod názvem Compute Module IO Board V3 (CMIO3).

Standardní verze je založena na SoC z Modelu B 3. generace vybavené 4 jádrovým ARM-Cortex-A53 BCM2837 pracujícím na frekvenci 1,2 GHz. Dále je k dispozici 1GB RAM a 4 GB eMMC paměti.

Verze Lite postrádá eMMC paměť, z čehož vyplývá, že tato paměť musí být Mini PC poskytnuta externě, například skrze výše zmíněný přípojný modul CMIO3.

Na závěr bychom rádi dodali, že očekávaná cena nového CM3 je přibližně 30 USD.

3.2.3 Volba konkrétního Raspberry Pi

Jelikož nabídka Raspberry Pi Foundation není tak rozsáhlá a variabilní jako v případě společnosti SolidRun, byl výběr Mini PC vcelku jednoduchý.

TABULKA 4: ROZDÍLY MEZI MODELY RASPBERRY PI [41]

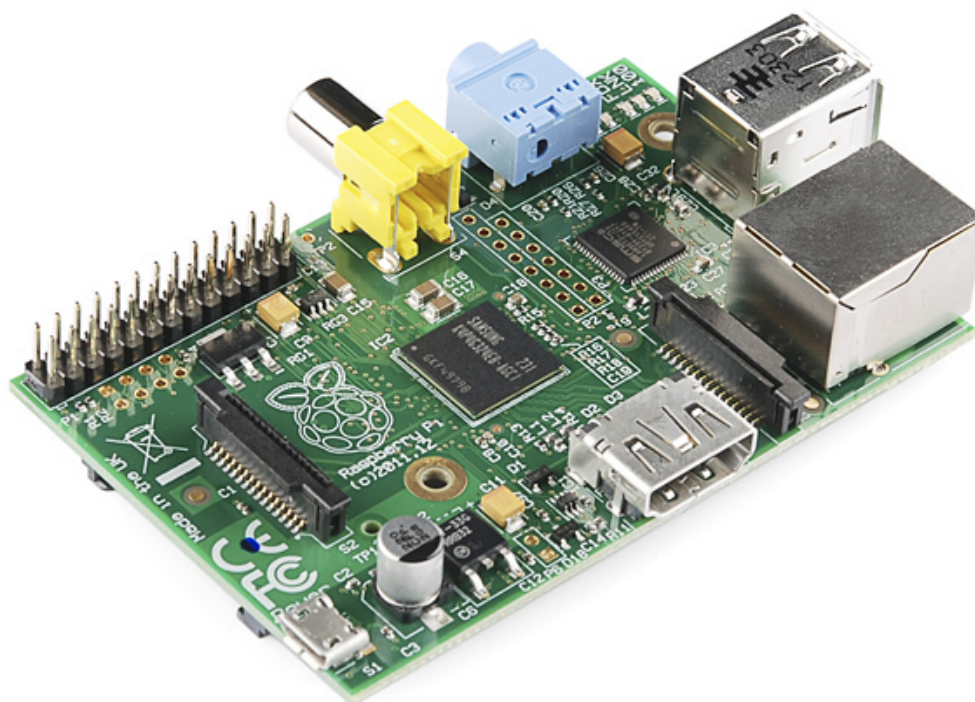
	Model A+	Model B+	Model 2 B	Model 3B	Zero	Zero W
Systém na čipu	BCM2835	BCM2835	BCM2836/ BCM 2837	BCM2837	BCM2835	BCM2835
Speed	700 Mhz	700 Mhz	900 Mhz	1 200 Mhz	1 200 Mhz	1 000 Mhz
RAM	512MB	512MB	1GB	1GB	512MB	512MB
USB	1	4	4	4	1	1
Ethernet	Ne	Ano	Ano	Ano	Ne	Ne
Wi-Fi	Ne	Ne	Ne	Ano	Ne	Ano
Cena	20 USD	25 USD	35 USD	35 USD	5 USD	10 USD

Díky miniaturním rozměrům modelu Zero (W) či nutnost vlastnění přípojného modulu k Raspberry Pi Compute Module (3/3L) by byla v případě tvorby IoT směrovače spíše na škodu než k užítku, rozhodli jsme se pro postupnou eliminaci výše uvedených modelů. Obdobně jsme eliminovali i Model A(+) určený výrobcem především pro nasazení v nízkospotřebních řešeních, která nepotřebují ethernetový port či více USB rozhraní.

Na základě předchozí úvahy je zřejmé, že námi zvolený typ je jedna z generací Raspberry Pi Modelu B. Vystávající otázkou tedy je, kterou konkrétní verzi tohoto modelu jsme se rozhodli použít v rámci našeho výzkumu. Jelikož každá z generací Modelu B disponuje ethernetovým portem a více USB rozhraními (viz. tabulka 4) nebyli jsme nijak omezeni při výběru konkrétní subverze. Nadále vzhledem k odlišnému výkonu jednotlivých generací souvisejících ruku v ruce s pořizovací cenou jsme se rozhodli pro nejlevnější možnou variantu, která byla k dispozici. Prozkoumáním realizace funkčnosti IoT směrovače na nejméně výkonné verzi očekáváme z toho vyplývající funkčnost i na více výkonných modelech.

Raspberry Pi Model B

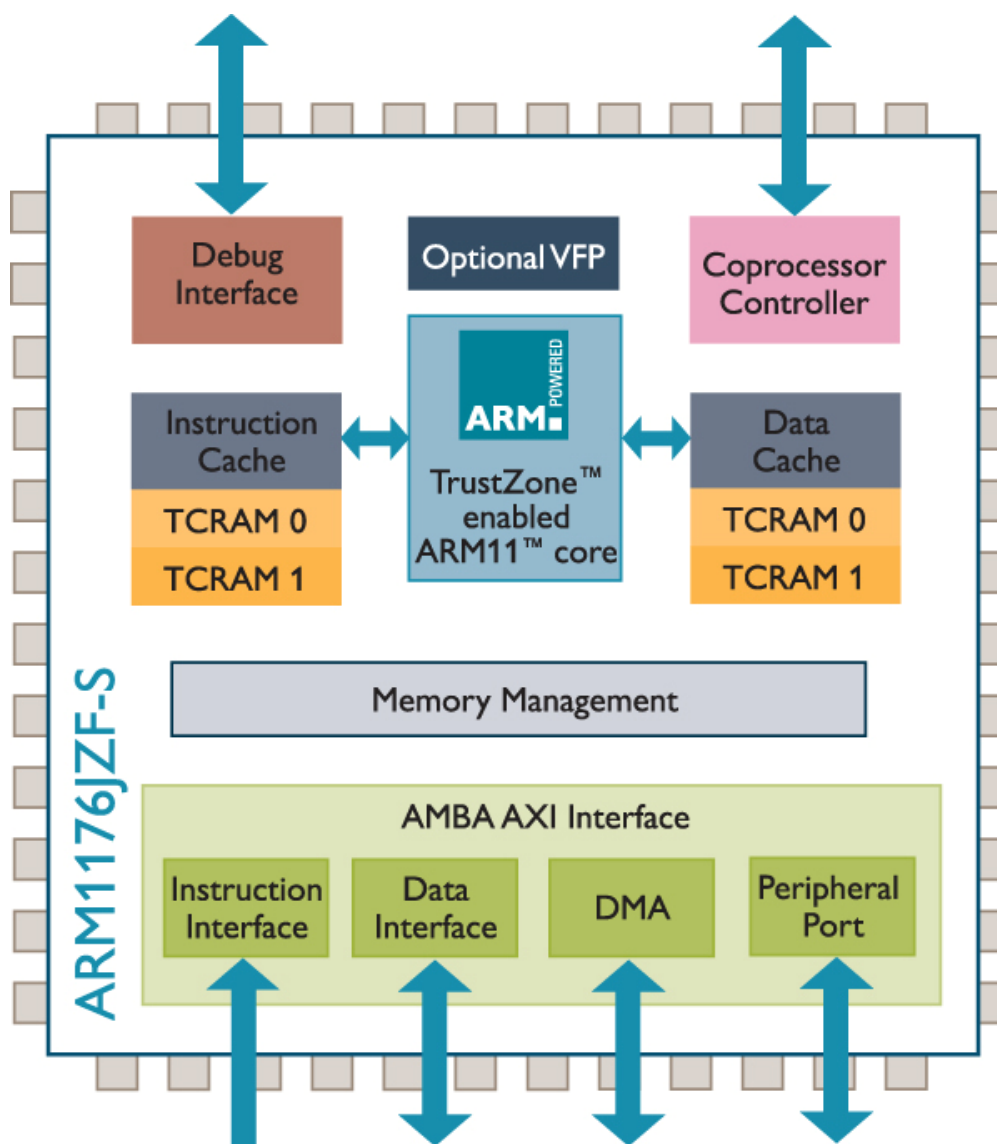
Konkrétně Raspberry Pi 1 Model B je výrobcem již nevyráběný a nepodporovaný model první generace, z něhož byl následně vyvinut aktuálně stále dostupný a podporovaný Raspberry Pi 2 Model B. Běžného čtenáře nyní může zarazit, proč jsme se rozhodli zvolit si právě tento, již nepodporovaný a nedostupný model. V rámci našeho výzkumu jsme měli možnost zapůjčení přístroje z Katedry informačních technologií při UHK/FIM. Dalším argumentem je očekávání fungování na výkonnějších a novějších Mini PC, jestliže námi vytyčené řešení bude fungovat i na méně výkonném a zastaralém HW řešení.



OBRÁZEK 7: RASPBERRY PI 1 MODEL B [46]

Díky zdrojům [41],[46] a [47] víme, že tato verze Mini PC je osazena Broadcom BCM2835 SoC obsahujícím jednojádrový mikroprocesor ARM1176JZF-S implementující architekturu ARM11 v6, pracující na frekvenci 700 Mhz. Model dále disponuje 26 GPIO piny, slotem na SD kartu, 512MB RAM, dvěma USB 2.0 porty, pouze 10/100 ethernetovým

rozhraním a dalšími, vzhledem k našemu výzkumu, nepodstatnými rozhraními. Poslední podstatnou informací je, že tento Mini PC podporuje OS GNU/Linux.



OBRÁZEK 8: BLOKOVÉ SCHÉMA MIKROPROCESORU ARM1176JZF-S [48]

3.3 Externí síťová rozhraní

Mimo samotné Mini PC je dále třeba externích síťových rozhraní rozšiřujících daná HW řešení o další ethernetové porty. V případě Raspberry Pi nemáme na výběr a musíme využít jednoho z mnoha veřejně dostupných adaptérů umožňujících převod z USB do RJ-

45. V případě HummingBoard Gate máme na výběr ze dvou možností a to rozšíření Mini PC o síťové rozhraní skrze mPCIe sběrnici či stejně jako v případě Raspberry PI skrze USB. Zde bychom rádi poznamenali, že volba rozšíření Mini PC HummingBoard skrze mPCIe sběrnici se může jevit jako vhodnější řešení, jelikož do ní lze zapojit i GbE síťové karty umožňující rychlejší komunikaci, avšak vzhledem k vnitřní sběrnici i.MX6 je tato rychlost limitována do maximální výše 470 Mbps [49]. S ohledem k tomuto omezení, univerzálnosti našeho řešení do budoucna (drtivá většina PC disponuje USB portem), nákladům potřebným k vynaložení pořízení adaptérů jsme se nakonec rozhodli pro USB 2.0 Fast Ethernet Adapter (výrobní číslo EU-4208) vyrobený společností Edimax.



OBRÁZEK 9: EXTERNÍ SÍŤOVÝ ADAPTÉR EDIMAX [50]

Adaptér vyhovuje normám IEEE 802.3/802.3u, splňuje specifikace rozhraní USB 2.0 (včetně zpětné kompatibility s USB 1.1). Podporuje funkci automatického vyjednávání (auto-negotiation) o rychlosti 10/100 Mbps. Uvnitř adaptéru je integrována miniaturní vyrovnávací paměť SRAM (20kB RX a 20kB TX) [50]. Zároveň podporuje nízkou spotřebu v klidovém režimu či nízkých provozních podmínkách (600 mW při plném zatížení, 46 mW při vzdáleném probuzení a 0,7 mW bez vzdáleného probuzení). Dále nepotřebuje externí napájecí adaptér a předně podporuje námi potřebný OS GNU/Linux.

4 Analýza software

Úvodem kapitoly je vhodné zargumentovat, z jakého důvodu jsme se rozhodli pro využití níže zmiňovaných OS.

V případě výběru OS pro HummingBoard Gate je odpověď na tuto otázku k dispozici v následujícím zdroji [51] popisujícím, že mezi jediné, společností SolidRun oficiálně podporované OS, patří OS Android, OS Debian a OS Yocto. Android je dnešní veřejnosti všeobecně známý především díky svému masivnímu použití v chytrých telefonech, ovládaných skrze dotykový displej. OS Yocto je naopak primárně vyvíjen pro nasazení v embedded systémech [52]. Na rozdíl od dvou předchozích kandidátů je Debian GNU/Linux vysoce univerzální OS a právě proto jsme se ho rozhodli použít během našeho výzkumu.

V případě volby OS pro Raspberry Pi 1 Model B jsme zvolili nasazení výrobcem jediné podporované distribuce OS GNU/Linux jímž je OS Raspbian [41].

4.1 Debian GNU/Linux

4.1.1 Základní charakteristika

Zdroje [53] a [54] popisují tento svobodný OS jako sadu základních programů a pomůcek, které k provozu potřebuje každý počítač. Základem OS je jádro, přičemž Debian GNU/Linux v současnosti používá jádro GNU/Linux, avšak ve vývoji je i podpora pro jádro GNU/Hurd. Na rozdíl od jiných distribucí OS GNU/Linux se Debian pyšní jistou tradicí, jelikož existuje již od srpna roku 1993 [55].

Neustále se rozšiřující sada základních programů tohoto OS je značně obsáhlá, jelikož v aktuální době čítá již přes 50 000 balíčků, přičemž díky našemu dřívějšímu výzkumu evidujeme nárůst o přibližně 13 500 balíčků mezi roky 2014 a 2017 a díky zdroji [55] víme, že nárůst mezi roky 2009 a 2017 je o přibližných 30 000 balíčků. Na základě

těchto údajů je zřejmé, že popularizace této distribuce se neustále navyšuje, jelikož více než polovina balíčků byla vyvinuta až ve 3. třetině života tohoto projektu.

Distribuce dále podporuje obsáhlou množinu architektur. Jmenujme například ARMHF, i386, MIPS či x86_64 a spousty dalších [56]. Přibližně každé dva roky potom vychází nová verze [55].

Debian vždy udržuje alespoň tři verze: stabilní, testovací a nestabilní [57]. Pro potřeby našeho výzkumu je podstatné, že za nejnovější oficiálně vydanou distribuci je považována verze stabilní, kterou se zcela nově v červnu 2017 stal Debian 9 s kódovým označením Stretch [58].

Na závěr této kapitoly bychom rádi poznamenali, že během našeho výzkumu, prováděného v průběhu jara roku 2017, jsme používali Debian verze 8.7 – Jessie, který byl v daném okamžiku považován za aktuální oficiální verzi.

4.2 Raspbian

4.2.1 Základní charakteristika

Raspbian je jediný, výrobcem Raspberry Pi, oficiálně podporovaný svobodný OS vycházející z OS Debian GNU/Linux. Jedná se o optimalizovaný soubor základních programů a utilit umožňující práci na veškerých produktech nadace Raspberry Pi [41] a [59].

Jelikož je Raspbian stvořen čistě pro práci s Raspberry Pi, podporuje na rozdíl od OS Debian GNU/Linux pouze jedinou architekturu již je architektura ARMHF [60].

Datum vzniku se datuje k létu roku 2012 [60] a skutečnou zajímavostí je, že tvůrci tohoto OS jsou pouze fanoušky Raspberry Pi a nejedná se tedy o nikoho, kdo by byl nějak přidružen k výrobcu daného HW. V současné době je k dispozici přes 35 000 balíčků [59].

Výrobce Mini PC potom doporučuje ke stažení novější Raspbian vycházející z OS Debian GNU/Linux Jessie [41]. Na oficiálních stránkách OS Debian GNU/Linux je pro

majitele 1. generace produktů Raspberry Pi všeobecně doporučeno nainstalovat raději OS Raspbian [61].

4.3 Předávání

S ohledem k námi zvolenému výběru plně komplexních OS, není potřeba se nadále zaobírat žádnou složitější konfigurací kromě nastavení tzv. předávání (forwarding).

Jedná se o povolení předávání IP paketů, které jsou určeny pro jiná koncová zařízení v počítačové síti.

Ve standardních situacích je forwarding vypnutý, takže OS konkrétního směrovače „zahodí“ každou zprávu, která není určena právě jemu, jakožto koncové stanici.

Při zapnutém předávání se naopak OS chová ke zprávám, určeným pro někoho jiného daleko vlídněji, jelikož konkrétní zprávu určenou pro jiný HW v počítačové síti „nezahodí“, ale zpracuje a předá dále konkrétní směrovací platformě, která již sama určí, co se s danou zprávou nadále bude dít.

Zapnutí forwardingu lze docílit různými způsoby. V rámci našeho výzkumu jsme na obou ze zkoumaných OS využili následujícího příkazu.

```
#echo "1" > /proc/sys/net/ipv4/conf/all/forwarding
```

Analogicky pouze stačí přepsat hodnotu „0“ v textovém souboru umístěném v adresářové struktuře **/proc/sys/net/ipv4/conf/all/forwarding**.

5 Návrh modelového řešení

V rámci této kapitoly bychom rádi shrnuli a především vzájemně „provázali“ jednotlivé poznatky popsané v předchozích kapitolách. To vše za účelem implementace vlastního směrovače. Ideálním prostředím, pro které směrovač navrhujeme, může být například domácnost či různá zařízení rodinně–rekreačního charakteru (chalupa či chata).

Tvorba vlastního IoT směrovače by se dala obecně shrnout do několika zásadních kroků, které se Vám nyní pokusíme nastínit.

Všeobecně platí, že prvním krokem při tvorbě vlastního IoT směrovače je volba takové HW platformy, která disponuje alespoň dvěma (či více) síťovými rozhraními či případně existuje možnost daný stroj o tato rozhraní rozšířit. Mimo jiné je vhodné zvolit dostatečně kvalitní a výkonný HW s ohledem na prostředí, ve kterém bude daný směrovač nasazen (dle očekávaného síťového provozu).

Máme-li k dispozici potřebný HW je na řadě zvolit správný řídicí systém. Ten bude mít na starost kompletní vykonávání jednotlivých síťových úkonů, správu nejen síťových, ale i dalších ovladačů, správu směrovací tabulky a zajišťování fungování konkrétních směrovacích protokolů. Obdobně jako v případě HW je třeba při výběru OS postupovat pečlivě a opatrně. Základní otázkou je, zda-li administrátor navrhovaného řešení zvažuje nasazení směrovače v nenáročných podmínkách (např. rekreační domek) či v plně automatizovaném prostředí vysoce náročném na síťovou komunikaci (např. průmyslová továrna).

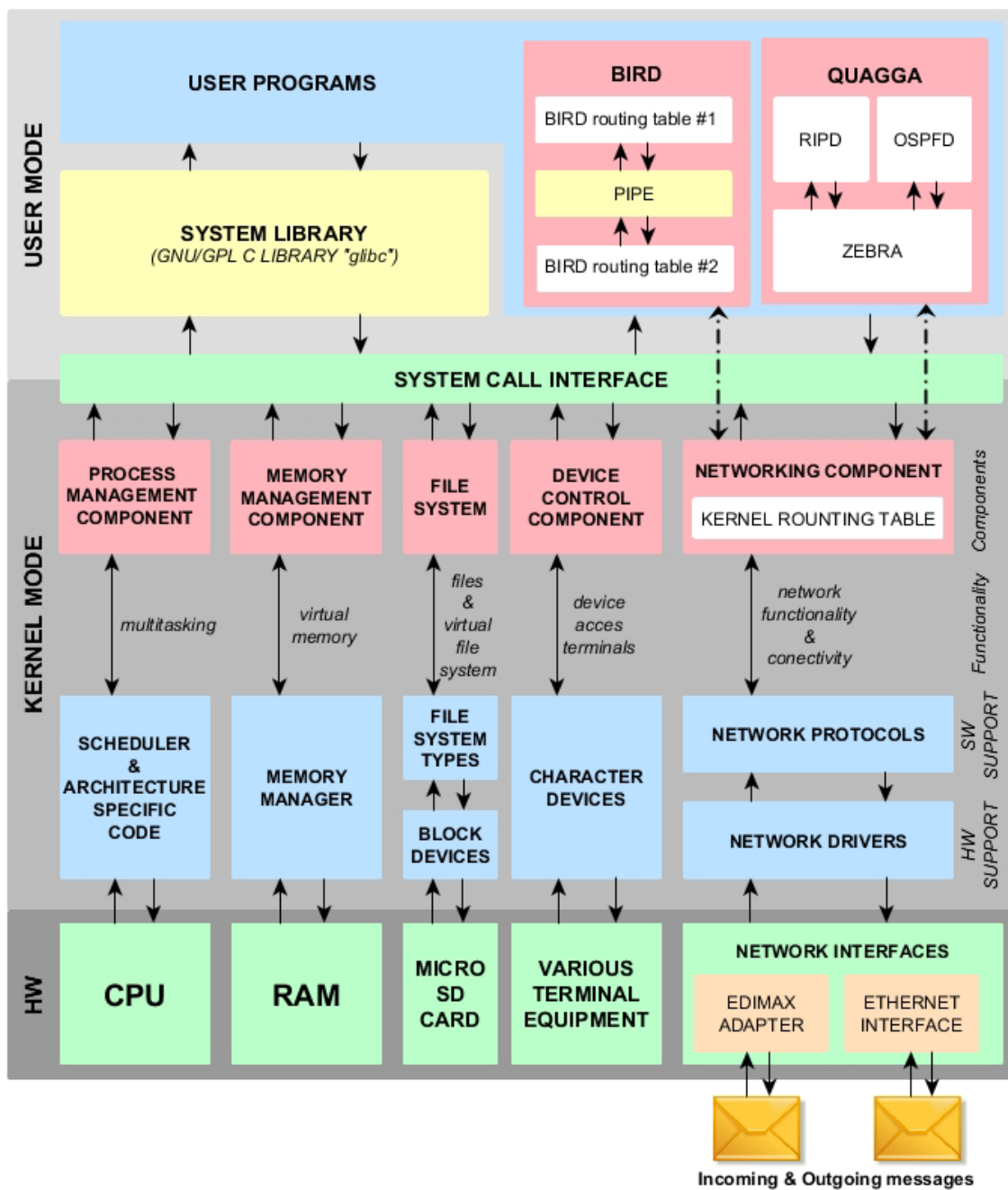
Po zvolení vhodného HW a kvalitního OS je na řadě volba samotné řídicí platformy, skrze kterou bude správce směrovače řídit fungování celého procesu směrování.

V rámci našeho výzkumu, jsme se rozhodli nejen o implementaci popisovaného řešení, ale i o vzájemnou analýzu. Konkrétně jsme použili dvě rozdílné HW platformy spravované dvěma částečně rozdílnými OS, avšak při využití dvou stejných řídicích platforem – SW démonů Quagga a BIRD.

K demonstraci toho, že SW směrovače lze využít téměř na libovolných zařízeních, jsme se rozhodli o srovnání dvou různých Mini PC. Prvním, ze kterého se stal téměř standard, je Raspberry Pi (Raspberry Pi (1) Model B). Druhou HW platformou je naopak relativně neznámý, papírově o mnoho výkonnější přístroj HummingBoard i2 Gate. Obě tato řešení disponují již od výroby pouze jedním síťovým portem, takže jsme je byli nuceni rozšířit o další ethernetová rozhraní. V rámci tohoto kroku jsme využili externího síťového adapteru od společnosti Edimax.

Po zvolení vhodného HW je na řadě volba odpovídajícího SW. Hned z několika důvodů jsme se rozhodli pro OS typu GNU/Linux. Prvním důvodem je všeobecný požadavek na stabilitu fungování řídicího systému použitého ve směrovači. Uživatelé zkrátka očekávají fungování směrovače nepřetržitě. Mimo jiné výrobci námi zvoleného HW oficiálně doporučují využití OS typu GNU/Linux a to konkrétně OS GNU/Debian v případě platformy HummingBoard i2 Gate a OS Raspbian v případě Raspberry Pi (1) Model B.

Již od začátku jsme plánovali provést náš výzkum za pomoci použití SW směrovacích démonů BIRD a Quagga. Jejich funkce je z hlediska fungování směrovače naprosto zásadní, jelikož se jedná o centrální řídicí prvek celého našeho řešení.



OBRÁZEK 10: ARCHITEKTURA MODELOVÉHO ŘEŠENÍ [AUTOR]

Na výše uvedeném obrázku 10 můžete vidět komplexní architekturu námi navrhnutého řešení.

V následující části bychom rádi vysvětlili funkčnost jednotlivých modulů zobrazených na obrázku 10.

V první řadě je nutné si uvědomit, že celková architektura se skládá ze tří základních prostředí, jimiž jsou oblast HW, oblast jádra OS (znázorněný jako KERNEL MODE) a oblast uživatelských aplikací OS (USER MODE).

Dále je vhodné si uvědomit, že celý proces „síťování“ musí být zpracováván OS zvláště především, protože většina síťových operací je asynchronních. Je tedy nutné jednotlivé zprávy identifikovat, uspořádat a zpracovat ještě předtím, než jsou předány konkrétnímu procesu. Toto pověření za doručení zprávy mezi síťovým rozhraním a uživatelskou aplikací má na starost OS fungující právě v oblasti KERNEL MODE. Mimo jiné musí OS kontrolovat vykonávání procesů a to na základě jejich „síťového chování“. Navíc všechny problémy týkající se směrování či IP adres jsou rovněž implementovány v jádře OS [62].

Budeme-li nyní postupovat obdobně jako příchozí zpráva, dorazíme nejprve do síťového rozhraní (v našem případě buď ethernetový port integrovaný na základní desce Mini PC či síťový adaptér Edimax).

Jelikož je síťový port čistě HW záležitostí, je třeba na prvním místě zpracovat příchozí zprávu (aktuálně ve formě bitů) a převést ji do vyššími vrstvami akceptovatelné formy. Převod mají na starost (síťové) ovladače. Konkrétní název síťového ovladače použitého v OS GNU/Linux lze všeobecně nalézt díky příkazu:

```
#lspci
```

Po přeložení zprávy a vzájemném porovnání adresy příjemce s vlastní adresou následuje vyhodnocení, zda se jedná o zprávu určenou přímo pro námi spravovaný stroj či pro někoho jiného. Je-li zpráva určená aplikaci existující v prostoru USER MODE, začne se směrovač chovat obdobně jako koncová stanice – zpracovávat a „vybalovat“ příchozí paket.

Je-li ale zpráva určena jiné stanici, nejprve směrovač ověří, zda-li má zapnuté přeposílání (forwarding). V případě, že je vypnuté, neváhá a ihned zahazuje příchozí zprávu.

V případě, že je forwarding zapnutý, začne směrovač vykonávat směrovací činnost a to na základě směrovacího protokolu definovaného v modulu Network Protocols (obrázek 10). Tento modul nemůže standardně fungovat sám o sobě, ale potřebuje přístup do směrovací tabulky jádra OS (KERNEL ROUTING TABLE umístěné v bloku NETWORK COMPONENTS).

Směrovací tabulka OS potom obsahuje záznam specifikující, kam má být daná zpráva přeposlána. Tento obsah lze tvořit právě díky použití SW směrovacích démonů a to buď staticky (statické směrování) či dynamicky (dynamické směrování).

SW směrovací démoni však na rozdíl od všech výše zmíněných komponent fungují v prostoru uživatelských aplikací (USER MODE), tudíž s nimi může přímo manipulovat i správce zařízení. Jedná se tedy o zásadní řídicí prvek nepřímo definující obsah směrovací tabulky jádra OS, specifikujícího použitý směrovací protokol (skrze blok NETWORK PROTOCOLS) to vše skrze SYSTEM CALL INTERFACE umístěné mezi USER MODE a KERNEL MODE prostory. Jádro OS následně nastaví novou adresu příjemce zprávy na základě nakonfigurovaných parametrů a odešle paket zpět ovladačům a ty konkrétnímu rozhraní. Touto cestou opustí nová zpráva náš směrovač.

6 Implementace

6.1 Hardware

V rámci implementace SW směrovače jsme se rozhodli pro použití dvou HW platforem (HummingBoard Gate a Raspberry Pi (1) Model B). Dále bylo nezbytné tato zařízení rozšířit o další ethernetový port (externí síťový adaptér Edimax). Ve výsledku jsme tak získali dvě zařízení disponující dvěma přístupovými komunikačními body.

Rádi bychom poznamenali, že kromě rozšíření Mini PC o další síťová rozhraní je při tvorbě domácího IoT SW směrovače vhodné disponovat libovolným monitorem určeným pro PC (s konektorem HDMI), klávesnicí (případně i myší), paměťovou kartou (v případě HummingBoard i2 Gate Micro SD a v případě Raspberry Pi (1) Model B SD s dostatečnou kapacitou paměti), příslušnými zdroji napětí a v případě Raspberry Pi (1) Model B i USB hubem, rozšiřujícím toto HW řešení o více USB portů potřebných pro připojení všech výše zmíněných periférií.

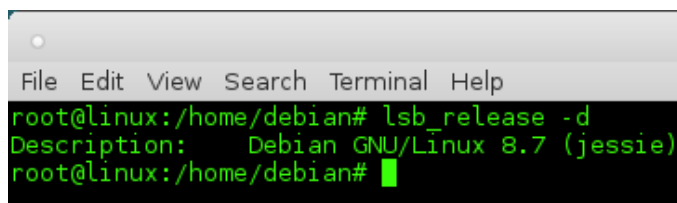
6.2 Operační systém

Po provedení výběru jednotlivých HW komponent a jejich vzájemném propojení je dále na řadě zvolení si vhodného SW. V předchozí kapitole týkající se návrhu řešení je jasně specifikováno, jakými možnostmi by měl takový OS disponovat.

Jelikož výrobci obou dvou námi zvolených Mini PC oficiálně doporučují použití OS na bázi GNU/Linux, rozhodli jsme se pro použití právě jeho.

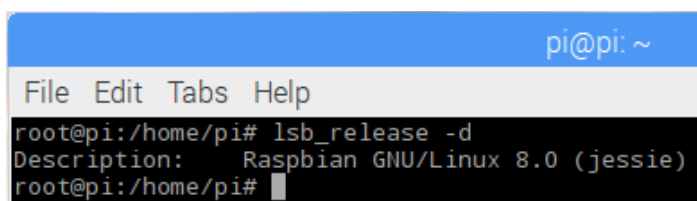
V případě HummingBoard i2 Gate se tedy jedná o OS GNU/Debian a v případě Raspberry Pi (1) Model B o derivát OS GNU/Debian – OS Raspbian.

Konkrétní verze jednotlivých OS, které jsme během našeho výzkumu použili, můžete vidět na níže uvedených obrázcích 11 a 12.



```
File Edit View Search Terminal Help
root@linux:/home/debian# lsb_release -d
Description:    Debian GNU/Linux 8.7 (jessie)
root@linux:/home/debian#
```

OBRÁZEK 5: KONKRÉTNÍ VERZE OPERAČNÍHO SYSTÉMU GNU/DEBIAN



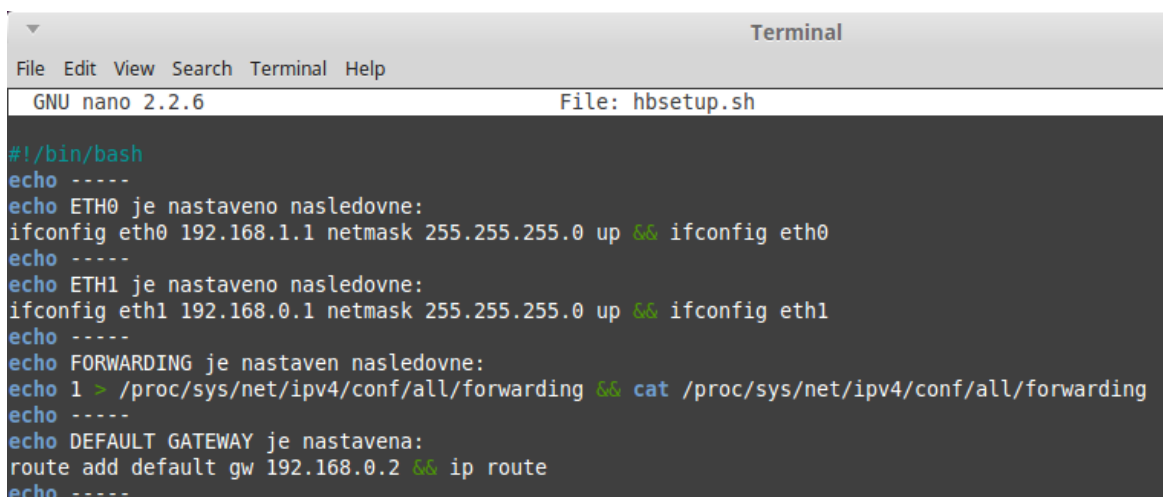
```
pi@pi: ~
File Edit Tabs Help
root@pi:/home/pi# lsb_release -d
Description:    Raspbian GNU/Linux 8.0 (jessie)
root@pi:/home/pi#
```

OBRÁZEK 6: KONKRÉTNÍ VERZE OPERAČNÍHO SYSTÉMU RASPBIAN

Jak je patrné z výše uvedených obrázků 11 a 12, výpis konkrétní verze OS lze získat díky příkazu:

#lsb_release -d

Po zvolení OS je třeba jejich nastavení. To bylo v rámci našeho výzkumu realizováno prostřednictvím níže uvedeného skriptu



```
Terminal
File Edit View Search Terminal Help
GNU nano 2.2.6 File: hbsetup.sh
#!/bin/bash
echo -----
echo ETH0 je nastaveno nasledovne:
ifconfig eth0 192.168.1.1 netmask 255.255.255.0 up && ifconfig eth0
echo -----
echo ETH1 je nastaveno nasledovne:
ifconfig eth1 192.168.0.1 netmask 255.255.255.0 up && ifconfig eth1
echo -----
echo FORWARDING je nastaven nasledovne:
echo 1 > /proc/sys/net/ipv4/conf/all/forwarding && cat /proc/sys/net/ipv4/conf/all/forwarding
echo -----
echo DEFAULT GATEWAY je nastavena:
route add default gw 192.168.0.2 && ip route
echo -----
```

OBRÁZEK 7: SKRIPT KONFIGURACE OPERAČNÍHO SYSTÉMU

6.3 Softwarový démon

Na úvod této kapitoly bychom rádi poznamenali, že před samotnou prací s konkrétními směrovacími platformami je vhodné disponovat vhodným textovým editorem (např. nano) umožňujícími úpravu jednotlivých konfiguračních souborů.

6.3.1 Quagga

Implementace

Existuje více způsobů umožňujících implementaci tohoto SW směrovacího démona. Jednou z možností, kterou jsme při implementaci Quaggy využili, je použití CLI OS, kdy po zadání následujícího příkazu je vyhledán v repozitářích OS daný balíček, který je následně nainstalován do OS.

```
#apt-get install quagga
```

Po úspěšně dokončené instalaci je veškerá implementace směrovacího démonu Quagga dokončena.

V obou námi zkoumaných OS lze nalézt následující adresářovou strukturu **/etc/quagga/**, která obsahuje veškeré potřebné konfigurační soubory tohoto démona.

Konfigurace

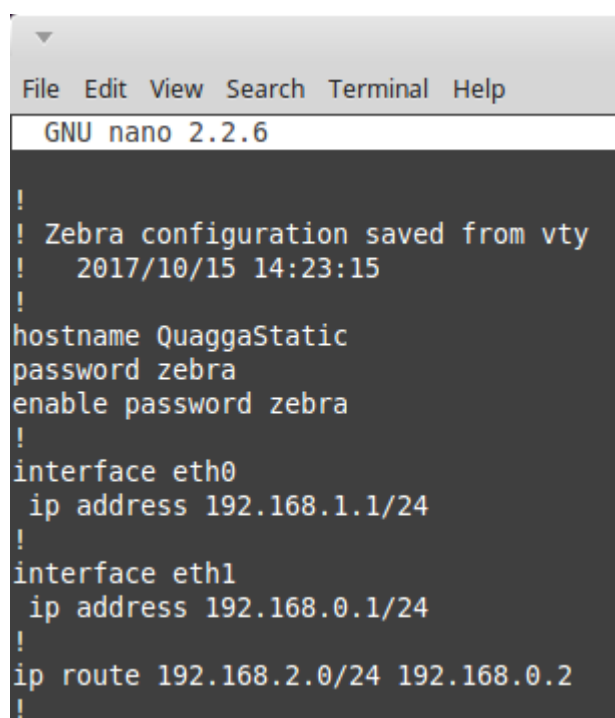
Existují dva způsoby jak lze Quaggu nakonfigurovat. V rámci našeho výzkumu jsme využili textového editoru nano, díky němuž jsme vytvořili konkrétní konfigurační soubory.

Druhým způsobem, jak lze tohoto směrovacího démona nastavit, je využití interaktivní CLI, kterou Quagga disponuje. Díky aplikaci telnet se lze připojit k CLI umožňující konfiguraci konkrétního subdémona. Stačí pouze nadefinovat adresu a název dílčího démona Quaggy, ke kterému se má telnet připojit (viz. následující příkaz):

```
#telnet localhost zebra
```

Každý z těchto přístupů má své klady i zápory. Konfigurace s využitím textového editoru působí na první pohled velmi přehledně a jasně, avšak dovolí lehce udělat syntaktickou chybu. Naopak konfigurace prostřednictvím CLI sice nedovolí uživateli napsat syntakticky nekorektní příkazy, avšak v jistých aspektech může tato konfigurace působit nepřehledně. Především snaží-li se správce směrovače nakonfigurovat dynamické směrování, při čemž musí postupně nastavit jednotlivé dílčí demony (např. konfigurace směrovacího protokolu OSPF vyžaduje konfiguraci CLI jak dílčího démona Zebra, tak i dalšího subdémona OSPFD).

Na následujících obrázcích můžete vidět konfigurace, které byly využity během našeho výzkumu.



```
File Edit View Search Terminal Help
GNU nano 2.2.6
!
! Zebra configuration saved from vty
!   2017/10/15 14:23:15
!
hostname QuaggaStatic
password zebra
enable password zebra
!
interface eth0
  ip address 192.168.1.1/24
!
interface eth1
  ip address 192.168.0.1/24
!
ip route 192.168.2.0/24 192.168.0.2
!
```

OBRÁZEK 8: OBSAH KONFIGURAČNÍHO SOUBORU ZEBRA.CONF

Na výše uvedeném obrázku 14 můžete vidět konfigurační soubor démona Zebra. V tomto případě se jedná o konfiguraci statického směrování.


```
File Edit View Search Terminal Help
GNU nano 2.2.6
!
! Zebra configuration saved from vty
!   2017/10/15 14:25:10
!
hostname QuaggaRIP
password zebra
enable password zebra
!
router rip
  version 2
  network 192.168.0.0/24
  network 192.168.1.0/24
  network 192.168.2.0/24
!
```

OBRÁZEK 9: OBSAH KONFIGURAČNÍHO SOUBORU RIPD.CONF

Obrázek 15 znázorňuje konfigurační soubor dílčího démona RIPD, který musí být spustěn společně s démonem Zebra. Oba dva tyto dílčí démoni poté umožňují dynamické směrování pomocí směrovacího protokolu RIPv2.

```
File Edit View Search Terminal Help
GNU nano 2.2.6
!
! Zebra configuration saved from vty
!   2017/10/15 14:27:16
!
hostname Quagga0SPF
password zebra
enable password zebra
!
router ospf
  network 192.168.0.0/24 area 0
  network 192.168.1.0/24 area 0
  network 192.168.2.0/24 area 0
!
```

OBRÁZEK 10: OBSAH KONFIGURAČNÍHO SOUBORU OSPFD.CONF

Obrázek 16, obdobně jako obrázek 15, znázorňuje konfigurační soubor dalšího dílčího démona OSPFD, který společně se spuštěním démona Zebra umožňuje dynamické směrování pomocí směrovacího protokolu OSPF.

Spouštění

Před samotným spuštěním SW směrovače je nutné nadefinovat, jaké konkrétní subdémony si přeje správce směrovače spustit. Stačí upravit konfigurační soubor **/etc/quagga/daemons.conf**, který je umístěn přímo v hlavní složce tohoto SW démona.

Po nastavení konfiguračního souboru stačí dále zadat do příkazového řádku OS následující příkaz a vyčkat na jeho reakci, ten obratem vypíše úspěšnost spuštění jednotlivých dílčích démonů.

```
#/etc/init.d/quagga start
```

Závěrem je vhodné říci, že vypínání Quaggy probíhá obdobně jako její spuštění. Namísto klíčového slova **start** je však nutné využít slova **stop**.

6.3.2 BIRD

Implementace

BIRD lze naimplementovat obdobně jako v případě směrovacího démona Quagga. V rámci našeho výzkumu jsme využili následujícího příkazu zadaného přímo do konzole OS:

```
#apt-get install bird
```

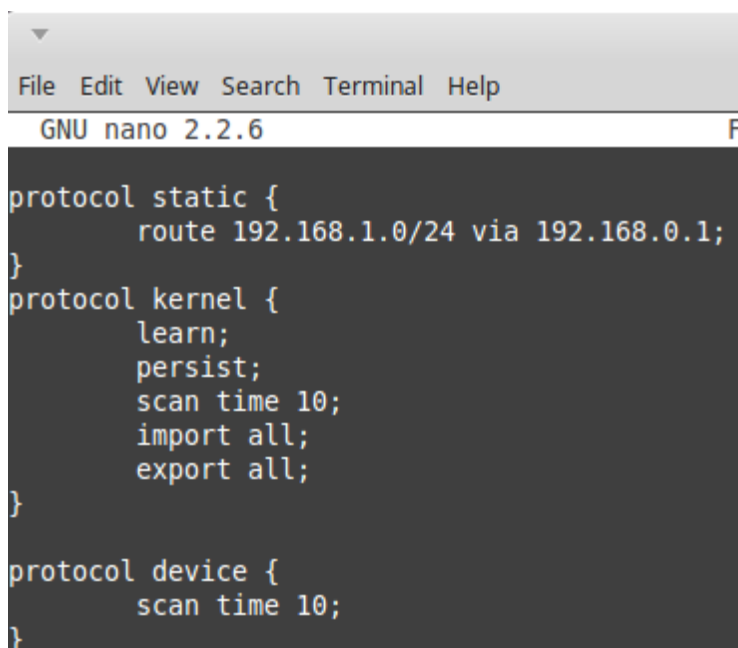
Po úspěšném nainstalování této směrovací platformy je možné nalézt ukázkové konfigurační soubory v adresáři **/etc/** nebo **/etc/bird/**.

Konfigurace

Na rozdíl od Quaggy existuje pouze jeden jediný způsob, jak lze nakonfigurovat tohoto SW démona a je k němu třeba textového editoru.

Výhodou potom je, že na rozdíl od směrovacího démona Quagga není celkový konfigurační kód rozptýlen mezi více konfiguračních souborů. Další výhodou je, že na rozdíl od Quaggy může správce SW démona BIRD použít libovolný název pro konkrétní konfigurační soubor, který může rovněž umístit kamkoliv na disku. Znamená to, že stačí vytvořit libovolně pojmenovaný textový soubor s příponou *conf*.

Na obrázcích níže můžete postupně vidět několik konfiguračních souborů, se kterými jsme se během našeho výzkumu zabývali.



```
File Edit View Search Terminal Help
GNU nano 2.2.6 F
protocol static {
    route 192.168.1.0/24 via 192.168.0.1;
}
protocol kernel {
    learn;
    persist;
    scan time 10;
    import all;
    export all;
}
protocol device {
    scan time 10;
}
```

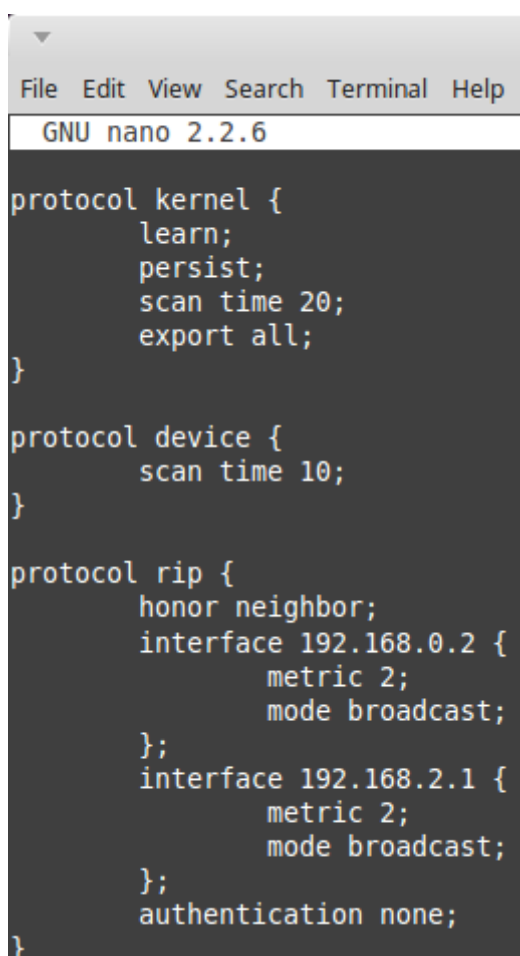
OBRÁZEK 11: OBSAH KONFIGURAČNÍHO SOUBORU REALIZUJÍCÍHO STATICKÉ SMĚROVÁNÍ

Jak můžete vidět na obrázku 17, základ konfiguračního souboru tvoří tři základní textové bloky *protocol static*, *protocol kernel* a *protocol device*.

Pravidla pro řízení správy směrovací tabulky OS jsou nadefinována v protocol kernel. Tento blok popisuje řízení informací mezi směrovací tabulkou démona a směrovací tabulkou OS.

Blok protocol device určuje periodu prozkoumávání stavů jednotlivých síťových rozhraní směrovače.

Jak již název napovídá, protocol static slouží k definování konkrétních pravidel statického směrování.



```
File Edit View Search Terminal Help
GNU nano 2.2.6

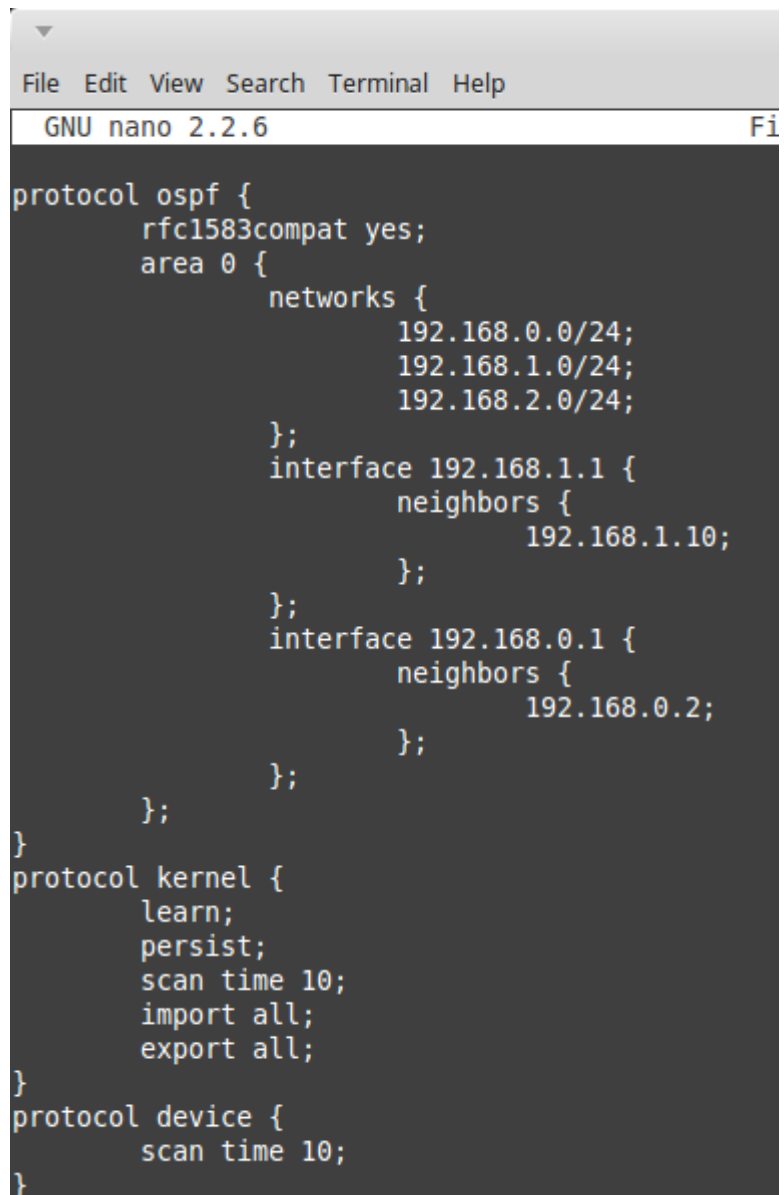
protocol kernel {
    learn;
    persist;
    scan time 20;
    export all;
}

protocol device {
    scan time 10;
}

protocol rip {
    honor neighbor;
    interface 192.168.0.2 {
        metric 2;
        mode broadcast;
    };
    interface 192.168.2.1 {
        metric 2;
        mode broadcast;
    };
    authentication none;
}
```

OBRÁZEK 12: OBSAH KONFIGURAČNÍHO SOUBORU SPECIFIKUJÍCÍ SMĚROVACÍ PROTOKOL RIP

Obrázek 18 zobrazuje konfigurační souboru protokolu RIP. Bloky kódu kernel a device jsme si již popsali u ukázky konfigurace statického směrování. Blok kódu protocol rip slouží k definici dynamického směrování pomocí protokolu RIP.

A screenshot of a terminal window running GNU nano 2.2.6. The window title bar shows 'File Edit View Search Terminal Help' and 'GNU nano 2.2.6'. The terminal content displays the following OSPF configuration code:

```
protocol ospf {
    rfc1583compat yes;
    area 0 {
        networks {
            192.168.0.0/24;
            192.168.1.0/24;
            192.168.2.0/24;
        };
        interface 192.168.1.1 {
            neighbors {
                192.168.1.10;
            };
        };
        interface 192.168.0.1 {
            neighbors {
                192.168.0.2;
            };
        };
    };
}
protocol kernel {
    learn;
    persist;
    scan time 10;
    import all;
    export all;
}
protocol device {
    scan time 10;
}
```

OBRÁZEK 19: OBSAH KONFIGURAČNÍHO SOUBORU SPECIFIKUJÍCÍ SMĚROVACÍ PROTOKOL OSPF

Na obrázku 19 můžete vidět konfiguraci směrovacího protokolu OSPF.

Spouštění

Máme-li nadefinována všechna potřebná směrovací pravidla, je dále na řadě spuštění samotného démona. To lze provést následujícím příkazem:

```
#bird
```

Tento příkaz standardně aktivuje démona a načte konfiguraci uloženou v souboru **bird.conf** umístěném v domácím adresáři směrovací platformy.

Přejeme-li si, aby byl zapnut démon a načten konfiguraci z konkrétního konfiguračního souboru, je třeba přidat parametr „-c“ specifikující cestu k danému souboru jako například níže.

```
#bird -c /etc/bird/birdospf.conf
```

Najde-li se správný konfigurační soubor, neobsahující syntaktickou chybu, zahlásí nám OS úspěšné spuštění démona. To, že BIRD běží, si můžeme ověřit zadáním příkazu spouštějícího jeho CLI.

```
#birdc
```

Závěrem této kapitoly je vhodné zmínit, že vypínání tohoto démona probíhá zcela odlišným způsobem.

```
>down
```

Výše znázorněný příkaz je třeba zadat do CLI démona (nikoliv do konzole OS).

7 Kritické zhodnocení navrženého řešení

V průběhu realizace námi popisovaného řešení při použití HW platformy Raspberry Pi nedošlo k žádnému problému. Veškeré instalace, konfigurace či implementace jednotlivých HW i SW komponent probíhaly bez složitějších potíží přesně tak, jak jsme očekávali.

Naopak při použití HW platformy HummingBoard jsme narazili na problém s částečně nekompatibilní paměťovou micro SD kartou, která standardně slouží jako paměťové médium pro uložení OS. Výrobce do dnešní doby neuvádí žádný seznam jím standardně podporovaných paměťových micro SD karet. Postupně jsme se tak pokoušeli o instalaci celé řady jak oficiálně podporovaných, tak i oficiálně nepodporovaných GNU/Linux distribucí. Jmenovitě se jednalo o OS Fedora, OpenSUSE, Debian (Wheezy), Arch linux, ARMBian, RedSleeve a Slackware, přičemž veškerá naše snaha končila pokaždé stejně – neúspěchem a to i přesto, že se instalace určitých OS jevila naprosto v pořádku – OS nakonec korektně „nenaběhly“. Po této negativní zkušenosti jsme neváhali a zaregistrovali se do oficiálního uživatelského fóra společnosti SolidRun. Až zde se nám podařilo nalézt informace o problémech týkajících se teoretické nekompatibility paměťového média. Po výměně micro SD karty se nám již podařilo nainstalovat OS GNU/Debian Jessie.

Mimo výše uvedený problém probíhala kompletní implementace jednotlivých SW směrovačů dle předem očekávaného postupu popsaného v kapitole 7 Implementace.

Na závěr této části bychom rádi dodali, že kromě výše uvedených síťových adaptérů jsme rovněž vyzkoušeli použití beztrátového síťového adaptéru DWL-G122 od společnosti D-link.

7.1 Základní testování funkčnosti

Domníváme se, že i přesto, že nebylo primárním účelem této práce komparativně analyzovat jednotlivé SW směrovače, tak samotným procesem porováním otestujeme základní funkčnost a odpovíme na otázku, které z dvou námi implementovaných řešení bychom upřednostnili. Za tímto účelem vznikla tato kapitola popisující jednotlivé výhody a nevýhody námi implementovaných řešení.

Dále bychom také rádi poznamenali, že parametrů pro analýzu může být celá řada a je bohužel nad rámec této práce zabývat se veškerými možnostmi vzájemného srovnání. Z tohoto důvodu jsme byli nuceni abstrahovat jen ty vlastnosti, které se jeví jako podstatné.

Mimo konkrétní, níže zmíněné vlastnosti se nám z našeho úhlu pohledu jeví jako velmi důležité prostředí, ve kterém hodláme náš směrovač nasadit a používat. Očekáváme, že námi navrhnutý SW směrovač bude nasazen spíše v síťově nenáročných prostředích, kde nejsou nároky na směrovač nijak extrémní.

7.1.1 Obecné vlastnosti

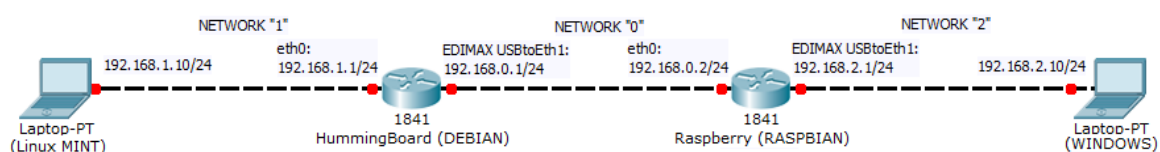
Prvním ze vzájemných porovnáání je komparativní analýza tzv. obecných vlastností. Tímto parametrem jsme se v rámci našeho výzkumu snažili zanalyzovat jakými, především síťovými vlastnostmi jednotlivé SW směrovače disponují. Vlastnosti vyjadřují nejen jaké síťové protokoly jsou podporovány, ale například i licence, pod kterými jsou démoni distribuováni či možnosti debugingu, seznam podporovaných jader OS GNU/Linux a další zajímavé vlastnosti.

TABULKA 5: OBECNÉ SROVNÁNÍ SMĚROVACÍCH PLATFORMEM

		BIRD	QUAGGA
IP Protokol	IPv4	Ano	Ano
	IPv6	Ano	Ano
Statické směrování		Ano	Ano
Dynamické směrovací protokoly	RIPv1	Ne	Ano
	RIPv2	Ano	Ano
	RIPng	Ne	Ano
	OSPFv2	Ano	Ano
	OSPFv3	Ano	Ano (<i>Beta verze</i>)
	IS-IS	Ne	Ano (<i>Beta verze</i>)
	BGPv4	Ano	Ano
	OLSR	Ne	Ano (<i>Externě</i>)
	Babel	Ano	Ne
Licence		GNU/GPL	GNU/GPL
Možnost debugingu		Komplexní	Komplexní
Podpora linuxu		Od kernelu 2.0	Od kernelu 2.4
Multicast		Ne	Ne
Interaktivní CLI		Ano (<i>jen pro výpis informací</i>)	Ano (<i>syntaxe podobná IOS</i>)
Další zajímavé vlastnosti a protokoly		MPLS, BFD	MPLS, BFD

7.1.2 Rychlost směrování

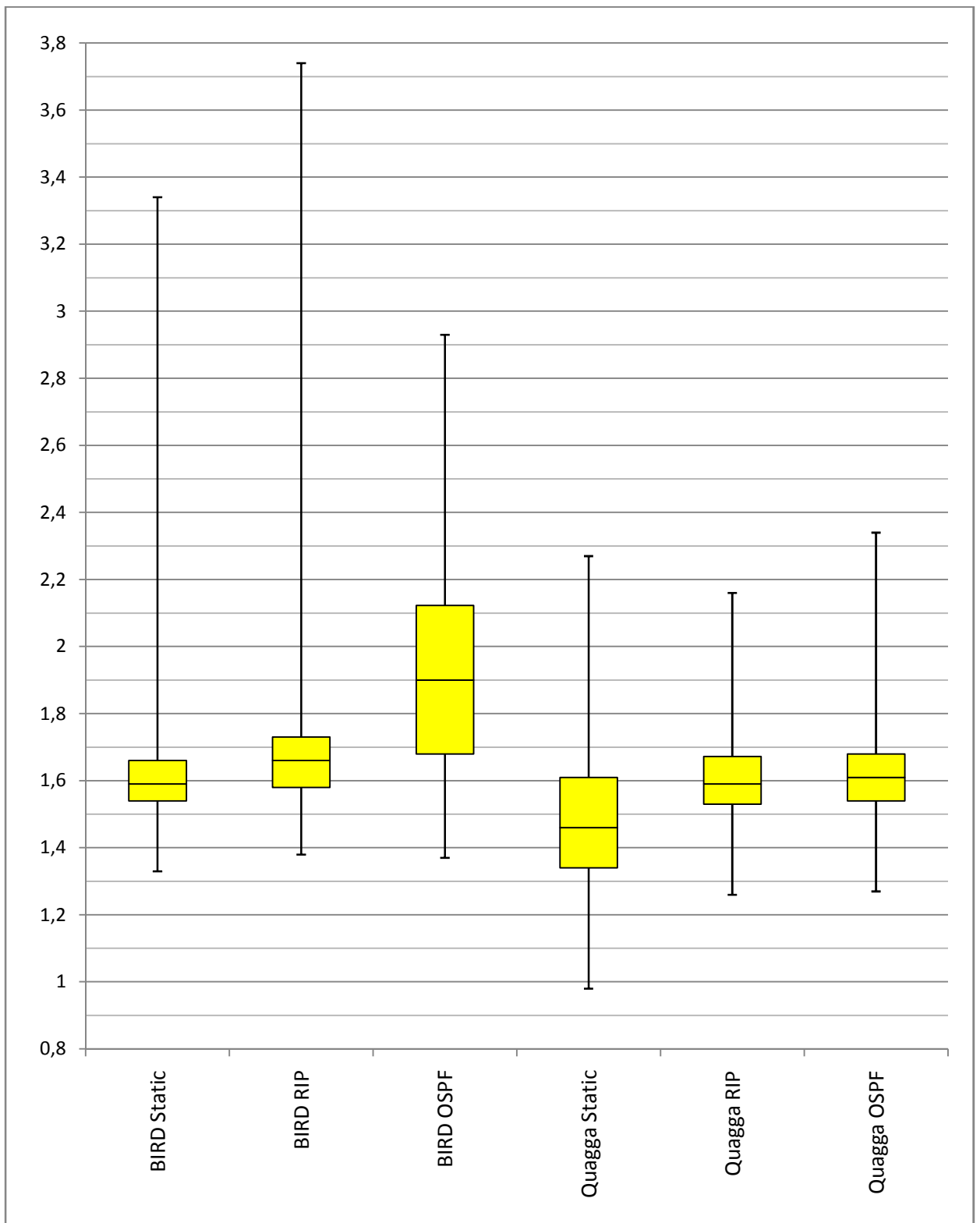
Tímto poněkud kontroverzním označením jsme se pokusili o vzájemnou komparativní analýzu rychlosti směrování. Zmíněné hodnoty vyjadřují čas v milisekundách, za který koncová stanice v rámci naší topologie (obrázek 20) dostane odpověď ve formě ICMP paketu od svého adresáta, umístěného přesně na druhé straně naší síťové topologie.



OBRÁZEK 13: NÁMI POUŽITÁ SÍŤOVÁ TOPOLOGIE

K naměření těchto hodnot jsme využili SW směrovačů, implementovaných na platformách Raspberry PI a HummingBoard, přesně tak, jak bylo popsáno v předchozích kapitolách. Postupně jsme využili směrovací platformu BIRD i Quagga, na nichž jsme realizovali statické a dynamické směrování pomocí směrovacího protokolu RIP a OSPF. Pro každé z námi realizovaných měření jsme se snažili zajistit vzájemné homogení prostředí. Vždy jsme zasílali právě 500 ICMP paketů.

Veškeré naměřené statisticky zpracované hodnoty, jsou názorně zobrazeny na následujícím grafu 1.



GRAF 1: VEŠKERÉ NAMĚŘENÉ HODNOTY

Na základě grafu 1 znázorněného na předchozí straně jsme schopni jednoduše a intuitivně rozpoznat jednotlivé minimální, průměrné a maximální hodnoty odezvy pro každou z námi zkoumaných variant směrování. Rovněž jsme schopni jednoduše identifikovat hodnoty jednotlivých kvartilů (Q_1 , Medián a Q_3).

Kromě výše graficky znázorněných dat je dále vhodné uvést konkrétní statistické výpočty a hodnoty přímo vycházející z našeho měření (průměrná doba odezvy, variační rozpětí (R), modus, rozptyl ($\text{Var}(x)$), šikmost (G_1) a špičatost (G_2).

TABULKA 5: STATISTICKÉ (ZAKROUHLLENÉ) VÝPOČTY

	BIRD			Quagga		
	Static	RIP	OSPF	Static	RIP	OSPF
MIN	1,335	1,388	1,373	0,980	1,263	1,274
AVG	1,607	1,681	1,906	1,489	1,608	1,618
MAX	3,347	3,745	2,938	2,279	2,165	2,341
Q1	1,540	1,580	1,680	1,340	1,530	1,540
Median	1,590	1,660	1,900	1,460	1,590	1,610
Q3	1,660	1,730	2,123	1,610	1,673	1,680
MOD	1,590	1,670	2,160	1,510	1,550	1,540
R	2,012	2,357	1,565	1,299	0,902	1,067
Var(x)	0,019	0,034	0,066	0,042	0,016	0,016
G1	0,013	0,031	0,004	0,005	0,001	0,001
G2	0,022	0,052	0,011	0,007	0,001	0,001

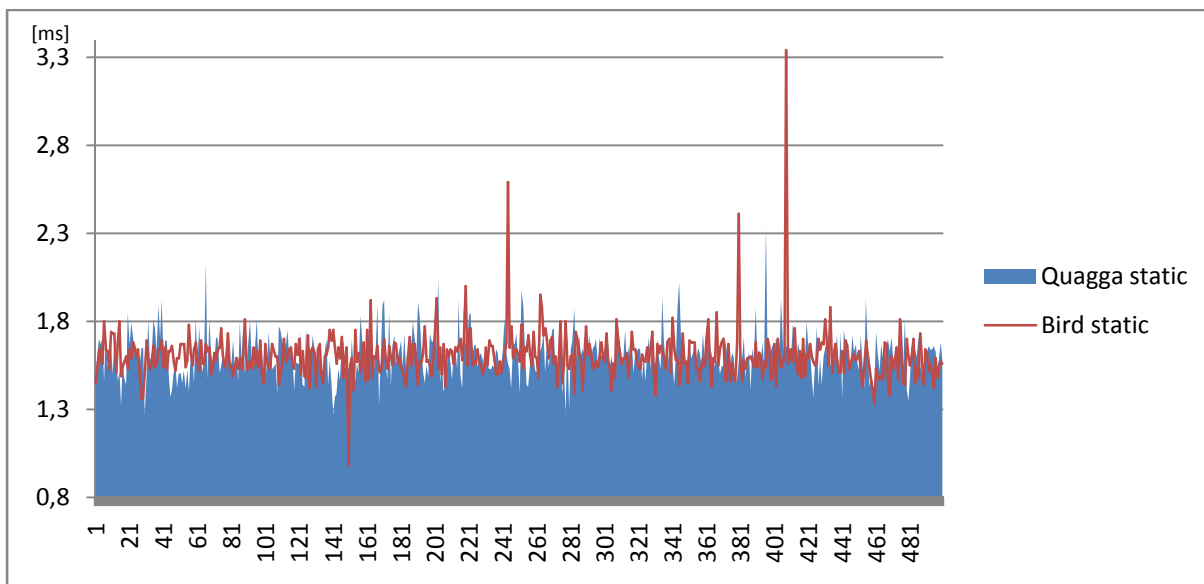
Na základě informací uvedených v grafu 1 a tabulky 6 docházíme k závěru, že v rámci našeho testu odezvy se vítězem stala volba směrovacího démona Quagga. Z hlediska směrovací platformy je pro nás tento výsledek poněkud překvapením, jelikož jsme očekávali, že relativně novější směrovací platforma BIRD se umístí lépe.

Úspěchy při použití statického směrování, ať už při využití SW démona BIRD, či platformy Quagga jsou pro nás potvrzením naší původní hypotézy, jelikož je všeobecně

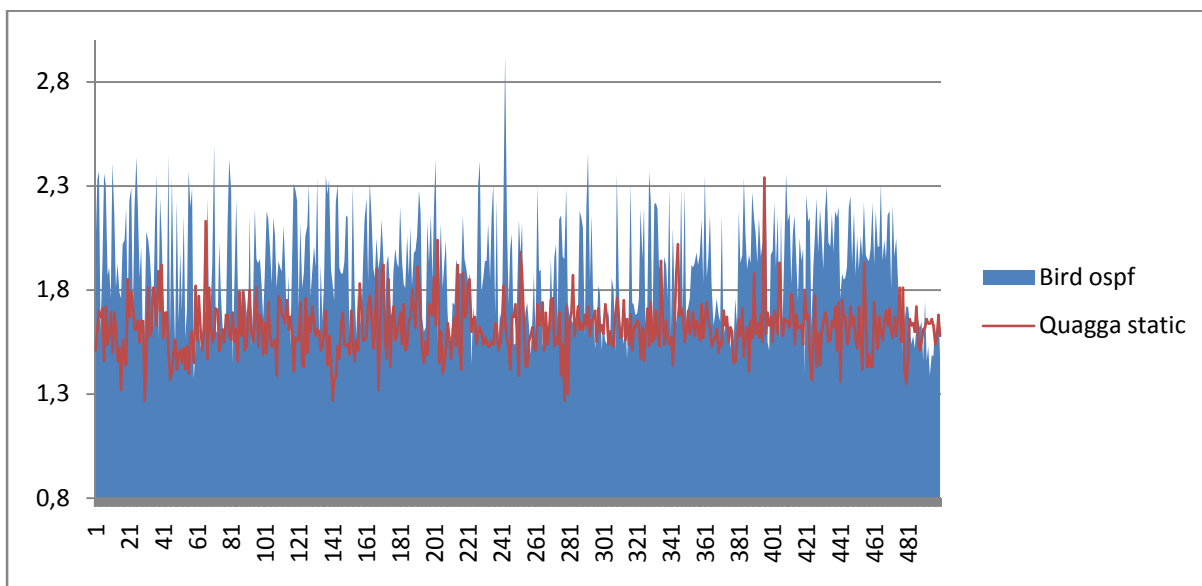
známo, že statické směrování je při použití na malých topologiích optimální volbou a to především kvůli minimálním nárokům na správu topologie sítě.

V rámci vzájemného srovnání dynamických směrovacích protokolů docházíme nezávisle na použité směrovací platformě rovněž k obdobným výsledkům, jež říkají, že použití směrovacího protokolu RIP je z hlediska námi použité síťové topologie vhodnější, než použití směrovacího protokolu OSPF. Tento závěr se může na první pohled jevit poněkud překvapivý, jelikož starší směrovací protokol RIP dosáhl lepších výsledků než jeho v dnešní době více používaný sok OSPF. Je však nutné poznamenat, že k těmto závěrům jsme došli při testování na námi zvoleném síťově nenáročném prostředí. Na základě tohoto výsledku lze říci, že směrovací protokol RIP nalezne stále své uplatnění.

Celkovým vítězem našeho jednoduchého testu se tak s nejnižšími hodnotami minimální a průměrné odezvy všech kvartilů a modusu stalo použití statického směrování, realizovaného díky směrovacímu démonu Quagga. Nejhorších výsledků jsme naopak dosáhli při použití směrovací platformy BIRD při použití protokolu OSPF.



GRAF 2: SROVNÁNÍ STATICKEHO SMĚROVÁNÍ



GRAF 3: SROVNÁNÍ NEJRYCHLEJŠÍHO A NEJPOMALEJŠÍHO VÝSLEDKU

Na výše uvedených grafech je názorně ukázáno srovnání statického směrování při použití na námi zkoumaných směrovacích demonech (graf 2) a porovnání nejlepšího s nejhorším výsledkem (graf 3).

Komparativní analýza softwarových démonů

Díky informacím uvedených v předchozí kapitole, jsme dále schopni na základě průměrné doby odezvy a modusu dob odezev jednotlivých měření vzájemně komparativně analyzovat jednotlivé směrovací platformy.

Na základě vzájemného srovnání průměrných časů odezvy jednotlivých měření docházíme k závěru, že v rámci našeho síťového prostředí je optimálním řešením implementace směrovacího démona Quagga. K tomuto závěru docházíme především z toho důvodu, že ve všech námi provedených měřeních byla průměrná doba odezvy nižší právě při použití tohoto směrovacího démona.

Komparativní analýza směrovacích protokolů

Zaměříme-li se na vzájemné srovnání jednotlivých směrovacích protokolů, dojdeme k závěru, že absolutním vítězem je statické směrování, které jak při použití SW démona BIRD tak i při použití směrovací platformy Quagga docílilo nejkratší průměrné doby odezvy a nejnižších hodnot jednotlivých modusů. Pomyslné druhé místo v rámci našeho testování není jednoduché specifikovat. Budeme-li se soustředit pouze na průměrnou dobu odezvy, potom docházíme k závěru, že rychlejších výsledků dosáhl směrovací protokol RIP. Zaměříme-li se na nejčtetnější hodnotu v jednotlivých výběrech, potom docházíme k závěru, že při použití SW démona Quagga je vhodnější využití směrovacího protokolu OSPF.

7.1.3 Softwarové nároky

V rámci komparativní analýzy jsme se rovněž zaměřili na srovnání zátěže, kterou si jednotliví směrovací démoni nárokují během své činnosti na CPU jednotlivých směrovačů. Pro zjištění takového parametru jsme využili následujícího příkazu.

```
#ps aux
```

Tento příkaz vypíše tabulku všech běžících procesů daného OS i s jejich aktuálními nároky na CPU konkrétního stroje.

Za účelem vytvoření zátěže jsme byli dále nuceni využít ICMP paketů. Ty jsme generovali na všech síťových prvcích znázorněných na obrázku 20 a to tak, že vždy v jeden konkrétní moment, každý z prvků v počítačové síti vysílal ICMP pakety na všechny ostatní stanice v topologii.

Tímto přístupem jsme se při nejnáročnějším z námi provedených měření dostali na hodnotu odpovídající 1,2% CPU při použití směrovacího protokolu OSPF, realizovaného SW démonem Quagga nasazeným na méně výkonném Mini PC Raspberry PI. Při použití směrovací platformy BIRD jsme docílili maximálního zatížení 0,3% a to při realizaci směrování pomocí směrovacího protokolu OSPF realizovaného právě na Raspberry PI.

V ostatních případech realizace statického směrování či směrování za pomoci směrovacího protokolu RIP byla zátěž vyvíjená SW démonem BIRD nulová.

Na základě těchto výsledků docházíme k potvrzení již předem očekávaného závěru jímž je, že směrovací platforma BIRD je méně náročná na provoz.

```
root 18046 0.0 0.3 2456 1692 ? Ss 13:18 0:00 bird -c /home/pi/Documents/ospf.conf
root 18052 0.0 0.4 4728 2116 pts/0 R+ 13:22 0:00 ps aux
root@pi:/home/pi/Documents#
```

OBRÁZEK 14: ZÁTĚŽ VYVINUTÁ SOFTWAREM DĚMONEM BIRD

Jelikož jsme byli schopni otestovat zátěž pouze v malém síťovém prostředí, vyvstává otázka, zda-li by se zátěž s nárůstem složitosti topologie zvětšovala či nikoliv. Na tuto otázku odpovídá jeden z tvůrců směrovací platformy BIRD, který uvádí ve zdroji [31] porovnání námi zkoumaných démonů z hlediska SW nároků právě při větší zátěži. Autor zde uvádí princip testování spotřeby CPU a paměti směrovače při a bez synchronizace se směrovací tabulkou OS obsahující 300 000 IPv4 adres při implementaci směrovacího protokolu BGP. Na základě výsledků těchto testů potom uvádí, že Quagga spotřebovává výkon směrovače až několikanásobně vícekrát.

Zdroj [63] konkrétně uvádí, že se jedná až o 4 násobně menší spotřebu paměti. Mimo to autor dále vyzvihuje stabilitu, se kterou na rozdíl od Quaggy zvládá BIRD pracovat při použití v rozsáhlých sítích.

Na základě námi provedených měření a výše uvedených informací docházíme tedy k závěru, že směrovací platforma BIRD vyvíjí menší SW nároky na CPU směrovače.

7.1.4 Rozšířenost

Poslední z námi zmíněných vlastností je míra používání dané směrovací platformy. Jedná se o pomyslnou hodnotu vyjadřující četnost používání daného SW. Základním předpokladem je přímá úměra rozšířenosti s velikostí komunity uživatelů pracujících s daným SW. Na základě rozsáhlé komunity uživatelů lze potom předpokládat, že v případě potřeby řešení potencionálního problému nalezneme pomoc při řešení

jednodušeji. Čím více uživatelů používá konkrétní SW, tím více se o něm dá dohledat informací, které mohou být užitečné během jeho používání.

Díky našemu výzkumu a sběru informací o zkoumaných SW demonech docházíme k závěru, že mezi běžnými uživateli je směrovací platforma Quagga známější. Ostatně samotní autoři démona BIRD uvádějí ve zdrojích [30], [31] a [64] totéž.

Na základě tohoto výsledku však nelze říci, že český SW démon BIRD nemá svoje uplatnění z hlediska směrování. Vzhledem k minimálním SW nárokům, popsaným v předchozí kapitole, a perfektní stabilitě při vysokém zatížení v kombinaci s jednoduše konfigurovatelnými filtry je BIRD oblíben především v peeringových centrech, jež tvoří páteř celého Internetu. Zde se využívají především vnější směrovací protokoly (např. BGP).

Zdroj [65] uvádí, že v roce 2014 byl BIRD nasazen na více jak 50% propojovacích uzlů tvořících páteř celého Internetu (IXP). Jako příklady nasazení autor zmiňuje například největší peeringové centrum na světě – frankfurtský DE-ICX, amsterodamský AMS-IX, londýnský LINX či běžnými uživateli vysoce známou společnost NETFLIX. Zdroj [63] mimo výše zmíněných uvádí dále americký PAIX a moskevský MSK-IX.

BIRD samozřejmě funguje i v českém peeringovém centru NIX, které mělo s ohledem k původu této směrovací platformy tu čest stát se prvním uzlem, na kterém BIRD běžel.

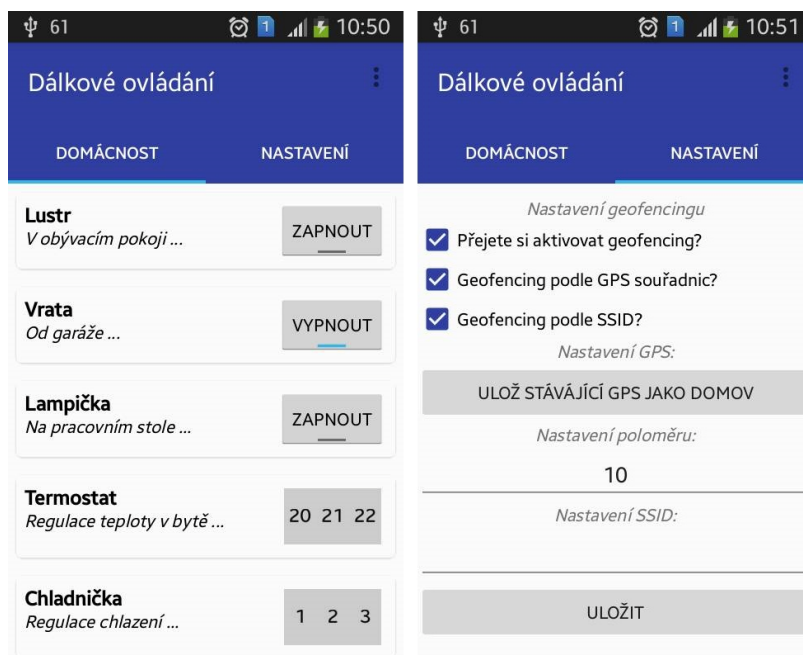
Výše uvedené zdroje dále zmiňují ocenění LINX Awards uděleném britským IXP LINX, které v roce 2010 BIRD získal. Jedná se o cennou trofej, která předtím několik let nebyla nikomu udělena a která oceňuje osobnosti, organizace nebo produkty, které přispěly k rozvoji internetových uzlů.

7.2 Využití směrovače v Internetu věcí

Na úplný závěr této práce bychom rádi inspirovali čtenáře a poukázali na jedno z potenciálních využití námi navržených SW směrovačů v IoT. V rámci výuky

programování mobilních aplikací vyučovaných na UHK/FIM jsme se již dříve zabírali využitím směrovače v IoT a to v rámci implementace chytrého domova. Vytvořili jsme jednoduchou mobilní aplikaci – Dálkové ovládání chytrého domu.

Jelikož se jedná pouze a doplnění námi vedeného výzkumu, pokusíme se nyní stručně popsat princip fungování této aplikace.



OBRÁZEK 15: DÁLKOVÉ OVLÁDÁNÍ CHYTRÉHO DOMU

Uživatel využívající chytrý telefon s OS Android může používat na svém chytrém telefonu námi navrhnoutou mobilní aplikaci, která umožňuje dálkové zapisování řídicích instrukcí do textových souborů. Tyto textové soubory mohou být umístěny v libovolném Mini PC a zároveň sloužit k ovládání vstupních a výstupních rozhraní. Skrze tato rozhraní potom může Mini PC posílat řídicí signály specifickým prvkům v domě a díky tomu je lze ovládat či získávat data.

Mimo samotné dálkové ovládání jsme rovněž implementovali tzv. geofencing. Jedná se o techniku určení polohy uživatele (jeho chytrého telefonu) na základě GPS souřadnic. Jelikož uživatel zadá GPS souřadnice svého domova, dokážeme na základě rozdílu jeho aktuální polohy a polohy domova využitím Pythagorovy věty jednoduše

vypočítat danou vzdálenost. Na základě tohoto rozdílu tedy poznáme, jak daleko se nachází uživatel od svého domova. Obdobně jsme rovněž implementovali geofencing vycházející z dostupnosti SSID domácí sítě. Je-li k dispozici domácí Wi-Fi síť, pak se uživatel nachází doma. Na základě těchto výpočtů lze potom aktivovat vybrané prvky v chytré domácnosti (jmenujme například centrální topení, rádio a jiné).

8 Závěr

Cílem práce bylo navrhnout a implementovat vybrané SW směrovače na platformě Mini PC.

V úvodu práce byl nejprve stručně vysvětlen pojem IoT a byly zmíněny prognózy předních světových organizací z oblasti informatiky (Intel, MIT, Cisco, IBM a Gartner) popisující enormní nástup počtu zařízení participujících v IoT. Následně byla provedena rešerše stávajících řešení, na základě které byl popsán úspěšný český projekt sdružení CZ.NIC – Turrus Omnia. S ohledem na vysokou pořizovací cenu tohoto zařízení, bylo rozhodnuto navrhnout obdobné, avšak značně levnější řešení. V následující kapitole 3 byly dále velmi stručně představeny vybrané principy směrování, OS a především SW směrovačů. Čtvrtá kapitola byla věnována podrobné analýze HW platform. V rámci našeho výzkumu jsme se rozhodli o přiblížení dvou potencionálních řešení, všeobecně známým standardem Raspberry Pi a relativně neznámou, avšak parametry vysoce atraktivní, platformou HummingBoard. Dále byl proveden výběr relevantních OS. Byla popsána jejich základní charakteristika a požadavky na jejich nastavení. Na základě těchto poznatků bylo navrženo komplexní modelové řešení, využívající jednotlivé komponenty popsané v předchozích kapitolách. V kapitole 7 byla podrobně popsána realizace navrhnutého řešení a to především s důrazem na popis implementace, konfigurace a spouštění směrovacích démonů Quagga a BIRD. Příkladů konfigurací bylo uvedeno hned několik druhů. Následně bylo provedeno kritické zhodnocení základní funkčnosti formou komparativní analýzy obou námi realizovaných řešení. Na úplný závěr práce bylo poukázáno na možné využití námi navrhnutých řešení v rámci IoT.

Výsledkem našeho výzkumu je, že po zkušenostech a problémech, se kterými jsme se setkali během práce s izraelským Mini PC HummingBoard, bychom raději doporučili více než čtyřikrát levnější, zaběhnutý standard, Raspberry Pi a to i přes značně nižší výkon tohoto zařízení.

Jelikož OS Raspbian, použitý při implementaci SW směrovače na platformě Raspberry Pi je pouze derivátem OS Debian GNU/Linux, který byl použit při výzkumu na platformě HummingBoard, docházíme z hlediska OS k neutrálnímu závěru, že není žádný charakteristický rozdíl, na základě kterého bychom upřednostnili jeden OS před druhým.

Z hlediska testování základní funkčnosti jsme se rozhodli o vzájemnou komparativní analýzu námi realizovaných řešení. Výsledky této analýzy jsou podrobně popsány v kapitole 8. Směrovací démon Quagga, který umožňuje implementovat více směrovacích protokolů, dosáhl lepších výsledků v rámci našeho testování rychlosti směrování. Tento výsledek byl pro nás poněkud překvapením, jelikož jsme očekávali výhru oblíbeného českého SW démona BIRD. Na základě našich měření jsme si dále potvrdili námi předem stanovené hypotézy – statické směrování je v rámci malých topologií nejvhodnější volbou. Démon BIRD naopak dominoval v otázce SW nároků. Posledním vzájemně analyzovaným prvkem bylo nedeterministické určení toho, který ze SW směrovačů je více rozšířen mezi uživateli. Odpověď na tuto otázku je závislá na prostředí, do kterého směrovač integrujeme. SW směrovač Quagga sice zvítězil v rámci našich testování, avšak v případě velké zátěže je vhodnější nasazení SW démona BIRD. Ostatně jak uvádí zdroj [7] 2/3 propojovacích uzlů, tvořících páteř celého Internetu, jej již aktivně používá. Úplný závěr kapitoly je věnován stručnému popisu principu využití SW směrovače jako dálkového ovládání chytrého domova.

9 Použitá literatura

- [1] Intel IoT: What Does The Internet of Things Mean? In: *Youtube* [online]. Santa Clara: Intel, 2014 [cit. 2017-10-29]. Dostupné z: <https://www.youtube.com/watch?v=Q3ur8wzzhBU>
- [2] Internet of Things: Roadmap to a Connected World. *MIT: xPRO* [online]. Cambridge (United States of America): MIT, 2016 [cit. 2017-10-29]. Dostupné z: https://mitxpro.mit.edu/courses/course-v1:MITProfessionalX+IOTx+2016_T1/about
- [3] Internet of Things (IoT): The IoT links objects to the Internet, enabling data and insights never available before. *Cisco: Internet of Things (IoT)* [online]. San Jose: Cisco Systems, 2017 [cit. 2017-10-29]. Dostupné z: <https://www.cisco.com/c/en/us/solutions/internet-of-things/overview.html#~stickynav=1>
- [4] IBM: Watson Internet of Things (IoT). *IBM* [online]. New York: IBM, 2017 [cit. 2017-10-29]. Dostupné z: <https://www.ibm.com/internet-of-things/>
- [5] Gartner Says 8.4 Billion Connected "Things" Will Be in Use in 2017: Up 31 Percent From 2016. *Gartner* [online]. Egham: Gartner, 2017 [cit. 2017-10-29]. Dostupné z: <https://www.gartner.com/newsroom/id/3598917>
- [6] DRDLA, Matěj. *Komparativní analýza softwarových směrovačů na platformě GNU/Linux*. Hradec Králové, 2014. Bakalářská práce. Univerzita Hradec Králové, Fakulta informatiky a managementu, Katedra informačních technologií. Vedoucí práce Josef Horálek.
- [7] CZNIC. CZ.NIC ve 256 vteřinách. In: *Youtube* [online]. [cit. 2017-10-26]. Dostupné z: <https://www.youtube.com/watch?v=glJd5bXvnw>
- [8] O sdružení. *CZ.NIC* [online]. Praha: CZ.NIC, 2017 [cit. 2017-10-26]. Dostupné z: <https://www.nic.cz/page/351/>
- [9] *Turris Omnia* [online]. Praha: CZ.NIC, ©2017 [cit. 2017-10-27]. Dostupné z: <https://omnia.turris.cz/cs/>

- [10] Národní CSIRT České republiky. *CSIRT.CZ* [online]. Praha: CZ.NIC, 2017 [cit. 2017-10-26]. Dostupné z: <https://csirt.cz/>
- [11] Introducing FRED. *FRED* [online]. Prague: CZ.NIC, 2016 [cit. 2017-10-26]. Dostupné z: <https://fred.nic.cz/>
- [12] Turris Omnia: hi-performance & open-source router. *INDIEGOGO* [online]. Indiegogo, ©2017 [cit. 2017-10-26]. Dostupné z: <https://www.indiegogo.com/projects/turris-omnia-hi-performance-open-source-router-security-network#/>
- [13] FILIP, Ondřej. Nejžádanější přístroj na Indiegogo je... router? *ROOT.CZ* [online]. Praha: ROOT.CZ, 2016 [cit. 2017-10-26]. Dostupné z: https://www.root.cz/clanky/nejzadanejsi-pristroj-na-indiegogo-je-router/?utm_source=FB&utm_medium=post&utm_campaign=FB
- [14] *OpenWrt: Wireless Freedom* [online]. OpenWrt Project, 2003 [cit. 2017-11-04]. Dostupné z: <https://openwrt.org/>
- [15] GANNON, Mary. How and where are SFP connectors and cabling used? *CONNCTORTIPS: An EE World Online Resouce* [online]. CONNCTORTIPS, 2016 [cit. 2017-11-04]. Dostupné z: <http://www.connectortips.com/how-and-where-are-sfp-connectors-and-cabling-used/>
- [16] TUNNISSEN, Jacco. *Intrusion Detection, Honeypots and Incident Handling Resources* [online]. ©2002-2017 [cit. 2017-10-26]. Dostupné z: <http://honeypots.net/>
- [17] KHAN ACADEMY LABS. What is information theory?: Journey into information theory. In: *Youtube* [online]. [cit. 2017-10-26]. Dostupné z: <https://www.youtube.com/embed/d9aIWZRzBWk?enablejsapi=1&fs=0&hl=cs&rel=0&controls=0&showinfo=0&html5=1&playsinline=1&origin=https%3A%2F%2Fkhanovaskola.cz&start=0>

- [18] GRAY, Robert M. *Entropy and Information Theory First Edition, Corrected* [online]. New York: Springer-Verlag, ©2013 [cit. 2017-10-27]. Dostupné z: <https://ee.stanford.edu/~gray/it.pdf>. Stanford University, Electrical Engineering Department, Information Systems Laboratory.
- [19] *Velký průvodce protokoly TCP/IP a systémem DNS*. 2. aktualizované vydání. Praha: Computer Press, 2000. ISBN 80-7226-323-4.
- [20] SOBĚSLAV, Vladimír, Josef HORÁLEK, Petr ŠUBA a Agáta BODNÁROVÁ. *Architektury směrovačů*. Hradec Králové, 2012. Univerzita Hradec Králové, Fakulta informatiky a managementu, Katedra informačních technologií.
- [21] PÁV, Miroslav, Jan SYŘÍNEK a Jana HOŠKOVÁ. *CCNA Exploration - Směrování, koncepce a protokoly* [online]. Plzeň, 2010 [cit. 2017-10-27]. Dostupné z: [https://is.muni.cz/el/1433/podzim2013/PV233/um/CCNA Exploration 2.pdf](https://is.muni.cz/el/1433/podzim2013/PV233/um/CCNA%20Exploration%202.pdf). Studijní materiál. VOŠ a SPSE Plzeň.
- [22] HORÁLEK, Josef. *Úvod do teorie operačních systémů*. Hradec Králové, 2012. Univerzita Hradec Králové, Fakulta informatiky a managementu, Katedra informačních technologií.
- [23] *Linux: dokumentační projekt*. 2. aktualizované vydání. Praha: Computer Press, 2001. ISBN 80-7226-503-2.
- [24] KLIMEŠ, Cyril. *Úloha a struktura operačních systémů*. Ostrava, 2012. Ostravská univerzita, Lékařská fakulta.
- [25] Kde získat Linux a jaký si vybrat. *Linux.cz* [online]. ©2005-2013 [cit. 2017-10-26]. Dostupné z: <http://proc.linux.cz/kde-ziskat.html#debian>
- [26] FILIP, Ondřej. *Směrovací démon BIRD* [online]. CZ.NIC, 2010 [cit. 2017-10-26]. Dostupné z: [https://www.nic.cz/public media/IT10/prezentace/den 2 1 Filip.pdf](https://www.nic.cz/public%20media/IT10/prezentace/den%202%201%20Filip.pdf)

- [27] *GNU Operating System: GNU General Public License* [online]. Free Software Foundation, 2007 [cit. 2017-11-04]. Dostupné z: <https://www.gnu.org/licenses/gpl-3.0.en.html>
- [28] HÁTAŠ, Martin. *Virtuální laboratoř*. Univerzita Hradec Králové, 2012. Diplomová práce. Univerzita Hradec Králové, Fakulta informatiky a managementu, Katedra informačních technologií. Vedoucí práce Josef Horálek.
- [29] *Quagga Software Routing Suite* [online]. NonGNU, 2013 [cit. 2017-10-27]. Dostupné z: <http://www.nongnu.org/quagga/index.html>
- [30] ZAJÍČEK, Ondřej. Dynamický routing a BIRD. In: *Stream* [online]. 2010 [cit. 2017-03-26]. Dostupné z: <http://www.stream.cz/uservideo/611670-dynamicky-routing-a-bird-ondrej-zajicek>
- [31] FILIP, Ondřej. *Směrovací démon BIRD* [online]. 2010 [cit. 2017-10-26].
- [32] FILIP, Ondřej. *The BIRD Internet Routing Daemon Project* [online]. Praha: CZ.NIC, 2004 [cit. 2017-10-26]. Dostupné z: <http://bird.network.cz/>
- [33] *SolidRun: Simple Robust Embedded Computing Solutions* [online]. Yokne'am Illit: SolidRun, ©2017 [cit. 2017-10-26]. Dostupné z: <https://www.solid-run.com/>
- [34] *Product Guide* [online]. Yokne'am Illit: SolidRun, 2017 [cit. 2017-10-26]. Dostupné z: https://www.solid-run.com/wp-content/uploads/2017/04/SolidRun-Catalog_32-for-Web.pdf
- [35] HummingBoard-Gate. In: *Youtube* [online]. Yokne'am Illit: SolidRun, 2015 [cit. 2017-10-26]. Dostupné z: <https://www.youtube.com/watch?v=KigPkFQ9cxl>
- [36] *Click Boards* [online]. Belgrade: MikroElektronika, ©1998-2017 [cit. 2017-10-26]. Dostupné z: <https://shop.mikroe.com/click>
- [37] *MikroBUS: Standard specifications* [online]. Rev. 2. Belgrade: MikroElektronika, 2015 [cit. 2017-10-26]. Dostupné z:

<http://download.mikroe.com/documents/standards/mikrobus/mikrobus-standard-specification-v200.pdf>

[38] *NXP-Freescale i.MX6 SOM i2: Dual Core SOM(System-On-Module)* [online]. Rev. 1.5. Yokne'am Illit: SolidRun, 2017 [cit. 2017-10-26]. Dostupné z: <https://wiki.solid-run.com/lib/exe/fetch.php?media=imx6:microsom:docs:sr-imx6-microsom-i2-rev15-datasheet.pdf>

[39] IMX6 SOM – System On Module: Documentation. *SolidRun: Simple Robust Embedded Computing Solutions*[online]. Yokne'am Illit: SolidRun, ©2017 [cit. 2017-10-26]. Dostupné z: <https://www.solid-run.com/freescale-imx6-family/imx6-som/imx6-som-documentation-block-diagram/>

[40] What is a Raspberry Pi? In: *Vimeo* [online]. Cambridge (United Kingdom): Raspberry Pi Foundation, 2014 [cit. 2017-10-26]. Dostupné z: <https://vimeo.com/90103691>

[41] *Raspberry Pi* [online]. Cambridge (United Kingdom): RASPBERRY PI FOUNDATION, 2017 [cit. 2017-10-26]. Dostupné z: <https://www.raspberrypi.org/>

[42] *Annual Review 2016: Raspberry Pi Foundation* [online]. Cambridge (United Kingdom). Raspberry Pi Foundation, 2016 [cit. 2017-10-26]. Dostupné z: <https://www.raspberrypi.org/files/about/RaspberryPiFoundationReview2016.pdf>

[43] FRIED, Limor. A Tour of the Pi Zero. *Adafruit: A Tour of the Pi Zero* [online]. New York City: adafruit, 2017 [cit. 2017-10-26]. Dostupné z: <https://learn.adafruit.com/introducing-the-raspberry-pi-zero/a-tour-of-the-pi-zero>

[44] BIEN-KAHN, JOSEPH. The New Raspberry Pi Zero W Is Your Key to the Hackable Future. *Wired.com* [online]. Boone (Iowa): Wired Magazine, 2017 [cit. 2017-10-26]. Dostupné z: <https://www.wired.com/2017/03/new-raspberry-pi-zero-w-key-hackable-future>

- [45] Raspberry Pi Compute Module 3 launched. TYSON, Mark. *Hexus.net* [online]. United Kingdom: HEXUS, 2017 [cit. 2017-10-26]. Dostupné z: <http://hexus.net/ce/news/gadgets/101509-raspberry-pi-compute-module-3-launched/>
- [46] Raspberry Pi - Model B. *SparkFun.com* [online]. Niwot: SparkFun Electronics, 2016 [cit. 2017-10-26]. Dostupné z: <https://www.sparkfun.com/products/retired/11546>
- [47] ARM1176JZF-S: Technical Reference Manual. *Infocenter.arm.com* [online]. Cambridge (United Kingdom): ARM, ©2004-2009 [cit. 2017-10-26]. Dostupné z: <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ddi0301h/index.html>
- [48] Classic Processors: Arm 11. *Developer.arm.com* [online]. Cambridge (United Kingdom): ARM, ©1995-2017 [cit. 2017-10-26]. Dostupné z: <https://developer.arm.com/products/processors/classic-processors>
- [49] IMX6 SOM – System On Module: Specification. *Solid-run.com* [online]. Yokne'am Illit: SolidRun, ©2017 [cit. 2017-10-26]. Dostupné z: <https://www.solid-run.com/freescale-imx6-family/imx6-som/imx6-som-specifications/>
- [50] USB 2.0 Fast Ethernet Adapter. *Edimax.com* [online]. New Taipei City: Edimax, ©2014 [cit. 2017-10-26]. Dostupné z: http://www.edimax.com/edimax/mw/cufiles/files/download/datasheet/transfer/USB/EU-4208/EU-4208_v2_Datasheet.pdf
- [51] Official IMX6 Distributions. *Wiki.solid-run.com* [online]. Yokne'am Illit: SolidRun, 2016 [cit. 2017-10-26]. Dostupné z: <https://wiki.solid-run.com/doku.php?id=products:imx6:software:os:official>

- [52] Introducing the Yocto Project. *Yoctoproject.org* [online]. San Francisco: Linux Foundation, ©2016 [cit. 2017-10-26]. Dostupné z: <https://www.yoctoproject.org/tools-resources/videos/introducing-yocto-project>
- [53] Vítejte na stránkách Debianu CZ. *Debian.cz* [online]. 2017 [cit. 2017-10-26]. Dostupné z: <http://www.debian.cz/>
- [54] *Debian* [online]. Software in the Public Interest, ©1997-2017 [cit. 2017-10-26]. Dostupné z: <https://www.debian.org/index.en.html>
- [55] KRČMÁŘ, Petr. Linuxová distribuce Debian: Konference LinuxAlt. In: *Stream* [online]. Brno, 2009 [cit. 2017-03-27]. Dostupné z: <http://www.stream.cz/uservideo/531922-linuxova-distribuce-debian-petr-krcmar>
- [56] Debian. *Distrowatch.com* [online]. Copenhagen, ©2001-2017 [cit. 2017-10-26]. Dostupné z: <https://distrowatch.com/table.php?distribution=debian>
- [57] Debian Releases. *Debian.org* [online]. Indianapolis: Software in the Public Interest, ©1997-2017 [cit. 2017-10-26]. Dostupné z: <https://www.debian.org/releases/index.en.html>
- [58] KEMP, Steve. Debian Stretch Released. *Debian-administration.org* [online]. Helsinki, 2017 [cit. 2017-10-26]. Dostupné z: [https://debian-administration.org/article/728/Debian Stretch Released](https://debian-administration.org/article/728/Debian%20Stretch%20Released)
- [59] Raspbian: Welcome to Raspbian. *Raspbian.org* [online]. [cit. 2017-10-26]. Dostupné z: <https://www.raspbian.org/>
- [60] Raspbian. *Distrowatch.com* [online]. Copenhagen, ©2001-2017 [cit. 2017-10-26]. Dostupné z: <https://distrowatch.com/table.php?distribution=raspbian>

- [61] RaspberryPi: Can I put Debian on my Raspberry Pi? *Wiki.debian.org* [online]. Indianapolis: Software in the Public Interest, ©1997-2017 [cit. 2017-10-26]. Dostupné z: <https://wiki.debian.org/RaspberryPi>
- [62] Linux Kernel and Its Architecture. *Knowstuffs* [online]. 2012 [cit. 2017-10-26]. Dostupné z: <https://knowstuffs.wordpress.com/2012/06/11/linux-kernel-and-architecture/>
- [63] SEDLÁK, Jan. Český systém BIRD porazil Cisco a je největší na světě – Connect.cz. *Connect.cz* [online]. Brno: CN Invest, 2012 [cit. 2017-10-26]. Dostupné z: <http://connect.zive.cz/clanky/cesky-system-bird-porazil-cisco-a-je-nejvetsi-na-svete/sc-320-a-165474/default.aspx>
- [64] FILIP, Ondřej. Směrovací démon BIRD: Konference LinuxAlt. In: *Stream.cz* [online]. CZ.NIC, 2009 [cit. 2017-03-26]. Dostupné z: <http://www.stream.cz/uservideo/532199-smerovaci-demon-bird-ondrej-filip>
- [65] FILIP, Ondřej. *BIRD Internet Routing Daemon: Route servery* [online]. Praha: CZ.NIC, 2014 [cit. 2017-10-26]. Dostupné z: https://www.linuxdays.cz/2014/video/Ondrej_Filip-Routovaci_demon_BIRD.pdf
- [66] Libelium Smart World. In: *Libelium* [online]. Zaragoza: libelium, 2013 [cit. 2017-11-04]. Dostupné z: http://www.libelium.com/wp-content/themes/libelium/images/content/applications/libelium_smart_world_infographic_big.png

10 Seznam obrázků

Obrázek 1: Základní deska směrovače Turris Omnia [9]	7
Obrázek 2: Architektura směrovací platformy Quagga (zpracováno dle [28]).....	16
Obrázek 3: Architektura směrovací platformy BIRD [autor]	18
Obrázek 4: HummingBoard Gate [33].....	27
Obrázek 5: I2 SOM [38]	29
Obrázek 6: Blokové schéma i.MX6 MicroSOM [39].....	30
Obrázek 7: Raspberry Pi 1 Model B [46].....	35
Obrázek 8: Blokové schéma mikroprocesoru ARM1176JZF-S [48]	36
Obrázek 9: Externí síťový adaptér Edimax [50]	37
Obrázek 10: Architektura modelového řešení [autor].....	43
Obrázek 11: Konkrétní verze operačního systému GNU/Debian.....	47
Obrázek 12: Konkrétní verze operačního systému Raspbian	47
Obrázek 13: Skript konfigurace operačního systému	47
Obrázek 14: Obsah konfiguračního souboru zebra.conf	49
Obrázek 15: Obsah konfiguračního souboru ripd.conf	50
Obrázek 16: Obsah konfiguračního souboru ospfd.conf	50
Obrázek 17: Obsah konfiguračního souboru realizujícího statické směrování	52
Obrázek 18: Obsah konfiguračního souboru specifikující směrovací protokol RIP	53
Obrázek 19: Obsah konfiguračního souboru specifikující směrovací protokol OSPF.....	54
Obrázek 20: Námi použitá síťová topologie	59
Obrázek 21: Zátěž vyvinutá softwarovým démonem BIRD	65
Obrázek 22: Dálkové ovládání chytrého domu	67

11 Seznam tabulek

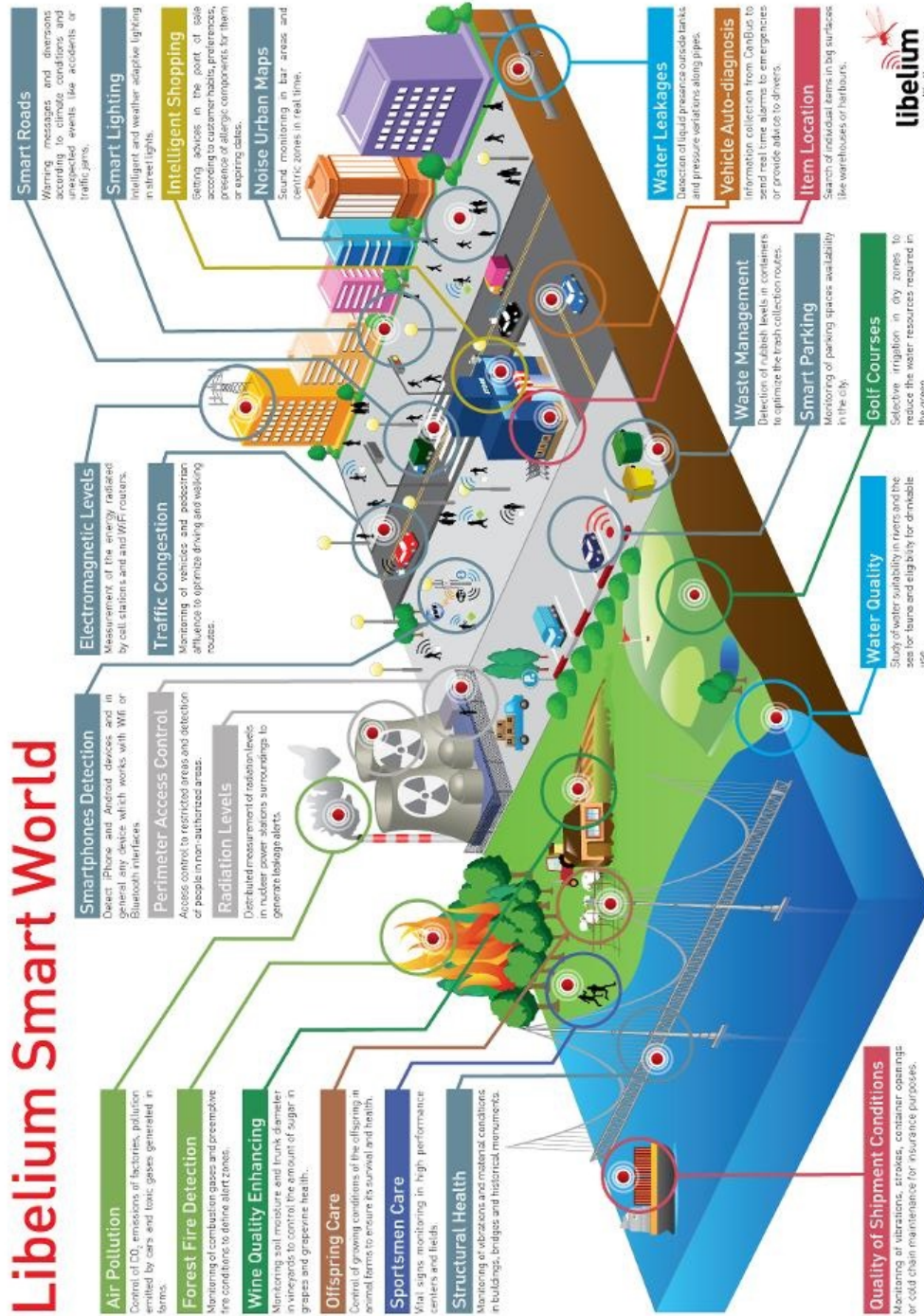
Tabulka 1: Obecné vlastnosti Turris Omnia [9]	6
Tabulka 2: Verze HummingBoardu [33]	25
Tabulka 3: Stručná specifikace konkrétního mikroprocesoru (převzato ze zdroje [33])	28
Tabulka 4: Rozdíly mezi modely Raspberry Pi [41]	34
Tabulka 5: Obecné srovnání směrovacích platforem	58
Tabulka 6: Statistické (zaokrouhlené) výpočty.....	61

12 Seznam grafů

Graf 1: Veškeré naměřené hodnoty.....	60
Graf 2: Srovnání statického směrování.....	62
Graf 3: Srovnání nejrychlejšího a nejpomalejšího výsledku	63

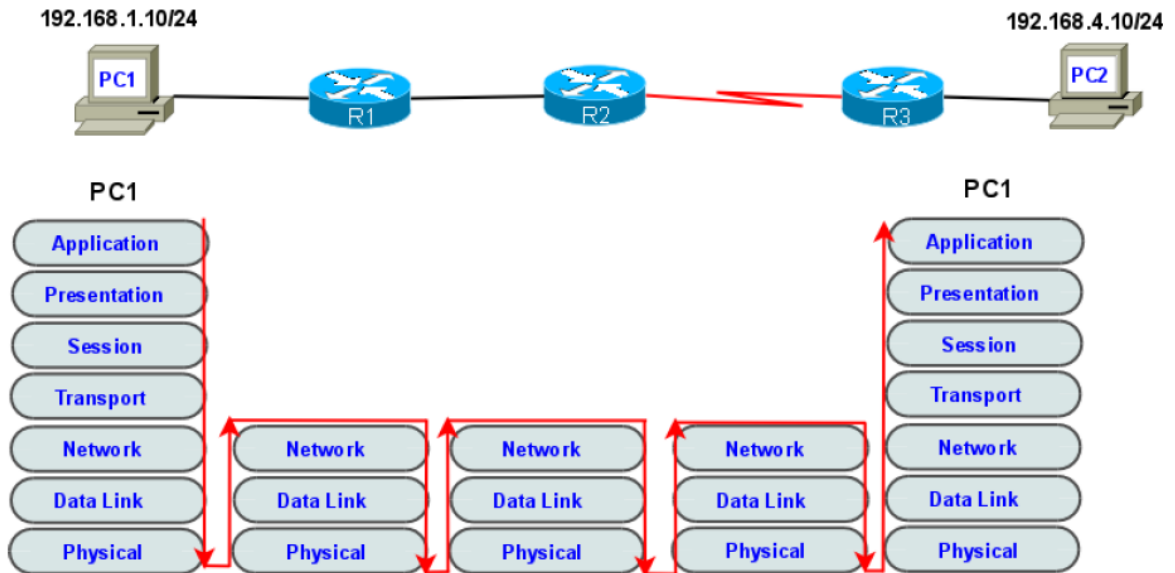
13 Přílohy

Příloha A: Smart World [66]

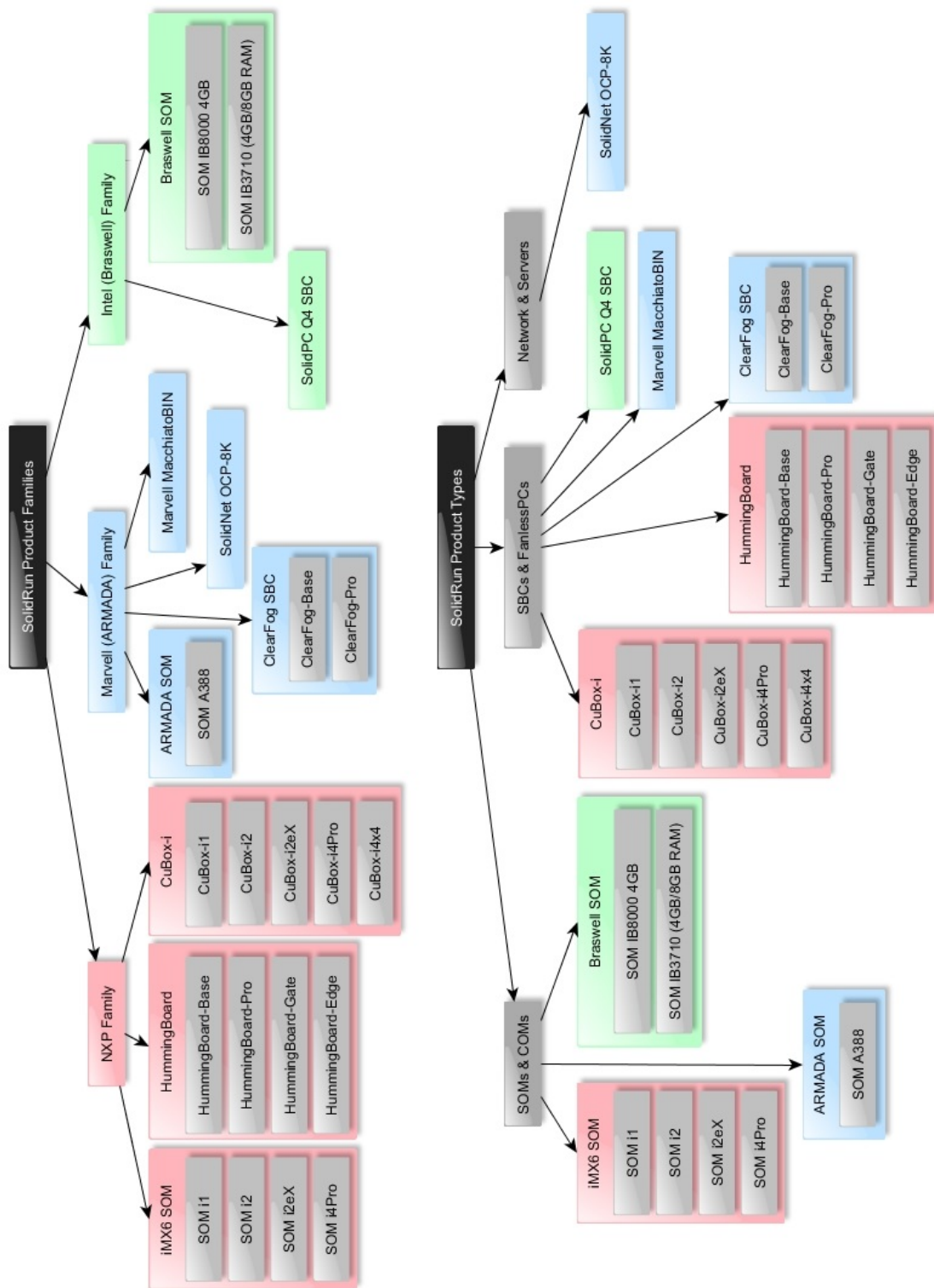


Příloha B: Princip práce směrovače [21]

Směrovač pracuje na vrstvách 1, 2 a 3 modelu OSI



Příloha C: SolidRun – kategorizace produktů [autor]



Příloha D: Komplexní srovnání jednotlivých SoM [33]

	SOM i1	SOM i2	SOM i2eX	SOM i4Pro
System On Chip	i.MX6 Solo	i.MX6 Dual Lite	i.MX6 Dual	i.MX6 Quad
Processor core	Single core ARM A9	Dual core ARM A9	Dual core ARM A9	Quad core ARM A9
Processor speed	1GHz (up to 1.2GHz)	1GHz (up to 1.2GHz)	1GHz (up to 1.2GHz)	1GHz (up to 1.2GHz)
Floating Point	VFPv3	VFPv3	VFPv3	VFPv3
SIMD	NEON	NEON	NEON	NEON
Graphics Processing Unit	Vivante GC880	Vivante GC880	Vivante GC2000	Vivante GC2000
3D GPU Support	OpenGL ES1.1/2.0	OpenGL ES1.1/2.0	OpenGL ES1.1/2.0 OpenCL 1.1E	OpenGL ES1.1/2.0 OpenCL 1.1E
HW Video Dec/Enc	Multi Format	Multi Format	Multi Format	Multi Format
Memory	32 bit, 512MB DDR3 @ 800Mbps	64 bit, 1GB DDR3 @ 800Mbps	64 bit, 1GB DDR3 @ 1066Mbps	64 bit, 2GB DDR3 @ 1066Mbps
Wired Network	10/100/1000 Mbps* Mbps	10/100/1000 Mbps* Mbps	10/100/1000 Mbps* Mbps	10/100/1000 Mbps* Mbps
Wireless Network	Optional (WL1831)	Optional (WL1831)	Optional (WL1831)	Optional (WL1831)
Bluetooth	Optional (WL1831)	Optional (WL1831)	Optional (WL1831)	Optional (WL1831)
Display Max resolution	HDMI: 1080p, LCD: WUXGA (1920 x 1200)	HDMI: 1080p, LCD: WUXGA (1920 x 1200)	HDMI: 1080p, LCD: WUXGA (1920 x 1200)	HDMI: 1080p, LCD: WUXGA (1920 x 1200)
Dual display support	Yes	Yes	Yes	Yes
Supported External Storage	NOR-Flash, eMMC, SD/microSD, PCIe SSD	NOR-Flash, eMMC, SD/microSD, PCIe SSD	NOR-Flash, eMMC, mSATA, SD/microSD, PCIe SSD	NOR-Flash, eMMC, mSATA, SD/microSD, PCIe SSD
Supported Internal Storage	eMMC, SPI ROM (Optional)	eMMC, SPI ROM (Optional)	eMMC, SPI ROM (Optional)	eMMC, SPI ROM (Optional)
SD/MMC	3	3	3	3
USB 2.0 host	1	1	1	1
USB OTG	1	1	1	1

Serial ports	3	3	3	3
Digital audio serial interface	1	1	1	1
Camera interface port	2 Lane CSI	2 Lane CSI	4 Lane CSI	4 Lane CSI
CAN bus	Yes	Yes	Yes	Yes
S-ATA	-	-	Gen II, 3Gbps	Gen II, 3Gbps
PCI-Express 2.0	X1	X1	X1	X1
Second Ethernet	Via PCIe or USB NIC	Via PCIe or USB NIC	Via PCIe or USB NIC	Via PCIe or USB NIC
I2C	3	3	3	3
SPI	3	3	3	3
PWM	4	4	4	4
GPIO	75	75	75	75
JTAG	Test Point Header	Test Point Header	Test Point Header	Test Point Header
S/PDIF Input	Yes	Yes	Yes	Yes
S/PDIF Output	Yes	Yes	Yes	Yes
1-Wire	1-Wire Transceiver on carrier board	1-Wire Transceiver on carrier board	1-Wire Transceiver on carrier board	1-Wire Transceiver on carrier board
RTC	On Carrier	On Carrier	On Carrier	On Carrier
Linux Support	Yes	Yes	Yes	Yes
Temperature range	Commercial, Industrial, Automotive	Commercial, Industrial, Automotive	Commercial, Industrial, Automotive	Commercial, Industrial, Automotive
Main Voltage	5V	5V	5V	5V
IO Voltage	3.3V	3.3V	3.3V	3.3V
SoM interface	Hirose DF40 connectors 1.5 up to 4.0mm mating heigh	Hirose DF40 connectors 1.5 up to 4.0mm mating heigh	Hirose DF40 connectors 1.5 up to 4.0mm mating heigh	Hirose DF40 connectors 1.5 up to 4.0mm mating heigh
Dimensions (W x L)	47mm x 30mm	47mm x 30mm	47mm x 30mm	47mm x 30mm
Price	From 48 USD	From 66 USD	From 81 USD	From 115 USD

Podklad pro zadání DIPLOMOVÉ práce studenta

PŘEDKLÁDÁ:	ADRESA	OSOBNÍ ČÍSLO
Bc. Drdla Matěj	Riegrova 128, Hrochův Týnec	11459

TÉMA ČESKY:

Využití a implementace softwarového směrovače na platformě Mini PC

TÉMA ANGLICKY:

Implementation and application of software routers on Mini PC platform

VEDOUCÍ PRÁCE:

Mgr. Josef Horálek, Ph.D. - KIT

ZÁSADY PRO VYPRACOVÁNÍ:

Cílem práce je navrhnout a implementovat vybrané softwarové směrovače na platformě miniPC s důrazem na jejich využití v internet of things (IoT).

V teoretické části autor představí vybrané principy SW směrovačů, provede analýzu vhodné HW platformy, požadavků na plnou funkcionalitu a výběr relevantního operačního systému. Dále autor představí návrh komplexního řešení a postup k jeho implementaci.

V praktické části autor realizuje navržené řešení s důrazem na jeho využitelnost pro IoT a provede základní testování funkčnosti.

Návrh osnovy:

Úvod

Rešerše stávajících řešení

Představení principů SW směrovačů

Analýza HW

Analýza a výběr OS

Návrh modelového řešení

Implementace a testování

Kritické zhodnocení navrženého řešení

Závěr

SEZNAM DOPORUČENÉ LITERATURY:

GREENGARD, Samuel. The internet of things. Cambridge, Massachusetts: MIT Press, 2015, xviii, 210 pages. ISBN 9780262527736.

HUCABY, David. CCNP routing and switching switch 300-115 official cert guide. 1st edition. Indianapolis, IN: Cisco Press, 2014, pages cm. ISBN 9781587205606.

Podpis studenta:

Datum:

Podpis vedoucího práce:

Datum: