Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačního inženýrství

Bakalářská práce

Raspberry PI jako firewall pro domácnosti a malé firmy

Vítězslav Košina

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Vítězslav Košina

Informatika

Název práce

Raspberry PI jako firewall pro domácnosti a malé firmy

Název anglicky

Raspberry PI as a firewall for home and small businesses

Cíle práce

Cílem práce je vytvořit firewall postavený na Raspberry PI a enterprise distribuci Linuxu, který bude neustále aktualizovaný z pohledu bezpečnostních rizik. Součástí řešení bude i instalace a konfigurace dodatečných síťových služeb. Řešení je určeno pro domácnosti a malé firmy, které nedisponují vlastními IT specialisty.

Metodika

Z minimální enterprise distribuce se připraví instalace, která se doplní vytvořenými skipty v shellu, které provedou automatickou instalaci a konfiguraci řešení. Současně připraví dodatečné nastavení, které se stará o periodickou, plně automatizovanou aktualizaci celého systému. Součástí celého řešení bude jak firewall, tak i servery DNS, DHCP, WebProxy, web server, SMTP, IMAP4 a XMPP server. Řešení bude připraveno tak, aby poskytovalo v rámci vnitřní sítě jak wifi acccess point, tak i bylo připravené pro připojení do domácí nebo firemní infrastruktury pomocí kabelu. Výsledné řešení bude možné naklonovat na více SD karet.

Doporučený rozsah práce

30-40 stran

Klíčová slova

Raspberry PI, Linux, Firewall, aktualizace, SOHO server, enterprise distribuce

---

Doporučené zdroje informací

Bhaskarjyoti Roy,Mohamed Alibi; CentOS 7 Linux Server Cookbook; 2016 Packt Publishing; ISBN 1785887289

Christine Bresnahan; Richard Blum; Linux Command Line and Shell Scripting Bible; 2015 Wiley; ISBN 978-1118983843

Lucian Gheorghe; Designing and Implementing Linux Firewalls and QoS using netfilter, iproute2, NAT and L7-filter; 2006 Packt Publishing; ISBN 978-1-90481-165-7

Soham Kamani; Full Stack Web Development with Raspberry Pi 3; 2017 Packt Publishing; ISBN 1788295897

Steve Suehring; Linux Firewalls: Enhancing Security with nftables and Beyond (4th Edition); 2015 Addison-Wesley Professional; ISBN 0134000021

Timothy Boronczyk; CentOS 7 Server Deployment Cookbook; 2016 Packt Publishing; ISBN 1783288884

Předběžný termín obhajoby

2018/19 LS – PEF

Vedoucí práce

Ing. Marek Pícka, Ph.D.

Garantující pracoviště

Katedra informačního inženýrství

Elektronicky schváleno dne 24. 1. 2019

Ing. Martin Pelikán, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 24. 1. 2019

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 10. 03. 2019

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci " Raspberry PI jako firewall pro domácnosti a malé firmy " jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 11.3.2019 _____

Poděkování

Rád bych touto cestou poděkoval Ing. Markovi Píckovi, Ph.D. za odbornou a pedagogickou pomoc při zpracování této bakalářské práce a rodině za maximální podporu během tvorby této práce.

Raspberry PI jako firewall pro domácnosti a malé firmy

Abstrakt

Bakalářská práce se zaměřuje na problematiku konfigurace firewallu a dalších služeb na jednodeskovém počítači Raspberry PI s využitím enterprise distribuce Linuxu v prostředí domácí nebo malé firemní počítačové sítě. V teoretické části jsou probrány jednotlivé typy firewallů, síťové protokoly a vzájemná vazba. Na tomto základě je pak popsána implementace firewallu v operačním systému Linux a následně rozepsány další aplikační protokoly, které tvoří celý rámec řešení firewallu prostřednictvím enterprise distribuce Linuxu. Praktická část se pak zaměřuje na popis a nastavení řešení dle požadovaných kritérií. Tedy síťové vrstvy, firewallu pomocí prostředků operačního systému a dalších síťových služeb. Řešení poskytuje ochranu jak proti nežádoucí aktivitě z Internetu, tak i proti nežádoucí aktivitě vycházející z vnitřní sítě. Dále je řešeno filtrování nevyžádaného, případně škodlivého obsahu a nežádoucí komunikace.

Výstupem je tedy konfigurace počítače Raspberry PI, který chrání menší domácí nebo podnikovou síť, poskytuje vybrané síťové služby a díky zvolené enterprise distribuci Linuxu i budoucí bezpečností update.

Klíčová slova: Raspberry PI, SOHO, Linux, CentOS, firewall, router, netfilter, počítačové síťě, TCP/IP, bezpečnost

Raspberry PI as the firewall for home and small enterprises

Abstract

Bachelor thesis focuses on the solution of the firewall configuration and other services running on the Raspberry PI computer. Solution is based utilizing enterprise Linux distribution in a small enterprise or home computer network. In the theoretical part of the thesis are described individual types of firewalls, network protocols and mutual relationship. On this basis, the implementation of the firewall in the Linux operating system is described, followed by other application protocols that provide the complex firewall framework solution in the enterprise Linux distribution. In the practical part, focuses on setting the solution itself according to the required criteria of the operating system, the network layer, the firewall using the operating system resources and other network services, including the recognition and prevention of known network attacks. The solution provides protection against both unwanted Internet activity and unwanted activity originating from the internal network. Additionally, filtering unwanted, potentially dangerous content and unwanted communications is addressed.

The result is a Raspberry PI configuration that protects a smaller home or enterprise network, provides selected network services. Usage of Enterprise Linux distribution provides future security update.

Keywords: Raspberry PI, SOHO, Linux, CentOS, firewall, router, netfilter, computer network, TCP/IP, security

Obsah

Seznam obrázků

Seznam tabulek

1 Úvod

V současné době je drtivá většina domácností, OSVČ i malých firem připojena k internetu pomocí různých technologií. Z pohledu koncového uživatele vše začíná prvkem, který ho fakticky spojuje s okolním světem, což je nejčastěji router, případně přístupový bod. Je téměř nepsaným pravidlem, že toto zařízení uživatel začíná vnímat pouze v okamžiku, kdy nefunguje a naopak je mimo jeho zorné pole po celý zbytek provozu.

Přestože se zvyšuje úroveň počítačové gramotnosti jakožto celku, nelze tvrdit, že se stejnou mírou stoupá i povědomí o IT bezpečnosti. To lze velmi snadno demonstrovat na jednoduchém příkladu, kdy v rámci domácnosti nebo malé firmy dramaticky roste počet tzv. chytrých zařízení, které jsou připojené do Internetu, nicméně neexistuje žádný přehled o tom, co daná zařízení odesílají a za jakým účelem.

Pro velkou většinu výrobců otázka zabezpečení skončila okamžikem zahájení výroby těchto zařízení a jakýkoliv update firmware nemají v plánu. Tento přístup se netýká pouze chytrých zařízení, ale dokonce i síťových prvků jako jsou routery nebo přístupové body. Je to mimo jiné dáno i tím, že cenová hladina, ve které jsou tato zařízení distribuována, je v řádu stovek resp. jednotek tisíců korun, jedná se o jednoúčelová zařízení, a proto politika výrobců se přiklání k distribuci těchto zařízení bez další podpory. Velmi často se lze setkat se zařízeními, která obsahují původní firemní nastavení i s původními přihlašovacími údaji a jejichž firmware nemá aktuální update i několik let.

V důsledku toho se taková zařízení stávají cílem útoků a v mnoha případech nejsou následky viditelné jen proto, že nejsou mediálně zajímavé. Správně nakonfigurovaný firewall a další síťové služby včetně posledních bezpečnostních aktualizací pak představují nutnou podmínku pro ochranu dat uvnitř firmy nebo domácnosti.

Cesta, kterou je možné opustit spirálu specifického hardware i software je použití univerzálního hardware a opensource operačního systému.

Problematikou vytvoření firewallu na bázi univerzálního jednodeskového počítače Raspberry PI 3B+ se zaobírá autor této práce.

2 Cíl práce a metodika

2.1 Cíl práce

Cílem této práce je vytvořit firewall pomocí hardware Raspberry PI a enterprise distribuce Linuxu. V rámci práce budou probrána a zhodnocena možná nastavení nejenom firewallu, aktualizací software, ale i dodatečných síťových služeb, které mohou být současně s firewalem provozovány na stejném hardware, nebo jsou z čistě praktických a bezpečnostních důvodů provozovány na jiném zařízení.

Implementaci získaných poznatků autor předvede v ukázkové konfiguraci nainstalovaného řešení dle definovaných požadavků.

Na závěr bude zhodnocena vhodnost Raspberry PI 3B+ pro případ užití jako firewallu pro malou firmu nebo domácnost.

2.2 Metodika

V teoretické části jsou vysvětleny především principy fungování firewallu v současných enterprise distribucích Linuxu, ale i v kontextu budoucích rozšíření a možností. Jsou rámcově probrány síťové služby, které mohou být dodatečně provozovány a které prakticky každá menší firma nebo domácnost využívá. V neposlední řadě je zde nastíněna možnost filtrování přístupu i v případech, kdy je spojení zabezpečeno pomocí TLS.

V praktické části je postupováno dle následujícího scénáře. Je zvolena konkrétní enterprise distribuce Linuxu a provedena instalace image na SD kartu. Následně je tato distribuce spuštěna na Raspberry PI 3B+ a je provedena dodatečná manuální konfigurace zařízení a služeb. Výsledná konfigurace je porovnána s původní konfigurací z čisté instalace a na základě rozdílového obrazu je vytvořena parametrická instalace, která se řídí konfiguračním souborem. Pomocí této parametrické instalace je vytvořena ukázková konfigurace firewallu a služeb v rámci reálného použití v malé firmě.

Dále je provedeno testování a zhodnocení reálného výkonu firewallu a síťových služeb v kontextu domácnosti, nebo malé firmy a provedeno doporučení optimalizace poskytovaných síťových služeb tak, aby řešení mohlo být následně provozováno bezobslužně v řádu jednotek let.

Platforma Raspberry Pi je primárně podporována na komunitních distribucích Linuxu a i referenční implementace OS je založená na komunitní distribuci Debian. Pro užití na firewallu je vhodné použít distribuce Linuxu určená pro podniková prostředí.

Distribuce operačního systému Linux, použitá v této práci je CentOS 7, která je binárně kompatibilní s distribucí RedHat Enterprise Linux 7. Volba této distribuce spočívala v definovaných klíčových požadavcích a to je stabilita a délka podpory. Tyto kritéria beze zbytku splnil právě CentOS 7, jelikož podpora této distribuce končí 30. června 2024[1], zatímco Respberry Pi 3B+ plánuje výrobce uvádět na trh do roku 2023[2].

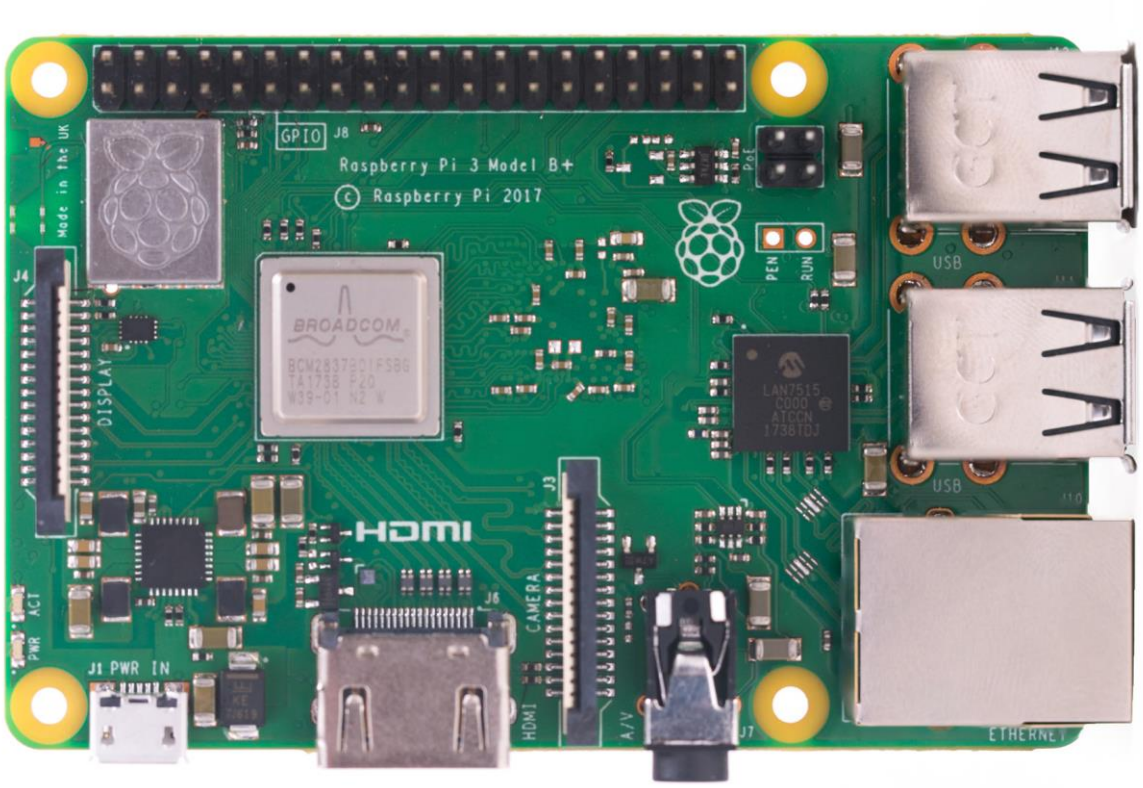
Během tvorby bakalářské práce byly využity zdroje uvedené v seznamu literatury.

3 Teoretická východiska

3.1 Raspberry PI

Raspberry PI [1] je malý jednodeskový počítač vyvinutý nadací Raspberry Pi Foundation původně za účelem podpory výuky robotiky, informatiky a počítačové vědy na školách. Vzhledem k jeho konstrukci a vybavení se jedná o cenově dostupný produkt. Verze 3B+ na obrázku (Obrázek 1) použitá v této bakalářské práci je dostupná v cenové hladině pod 1 000,- Kč včetně DPH. Hardwarové vybavení však činí z tohoto počítače univerzální prostředek pro provoz široké škály enterprise aplikací.

Obrázek 1: Rapberry PI 3B+



Zdroj: raspberrypi.org [1]

Model Raspberry PI 3B+ použitý v této bakalářské práci disponuje následující hardwarovou konfigurací [1]:

- Čtyř jádrový 64/32-bit procesor ARM Cortex-A53 s taktovací frekvencí 1.4 GHz

- 1 GB RAM, která je sdílená s GPU
- 10/100/1000 Mbit/s Ethernet rozhraní (jehož reálná rychlost je limitovaná cca. 300 Mbit/s díky sdílenému USB 2.0 rozhraní),
- bezdrátová dvoupásmová síť (2.4/5 GHz) na bázi standardů 802.11b/g/n/ac, Bluetooth 4.2 LS BLE
- čtyři konektory USB 2.0
- HDMI verze 1.4.

Vnitřní datové úložiště je realizované pomocí MicroSD karty nebo může být využité úložiště v rámci USB portů, např. USB flash disku, popř. reálných diskových periférií jako HDD nebo SSD.

Pro Raspberry PI byl připraven jako referenční operační systém Raspbian, což je upravená distribuce Debian pro toto zařízení. Nicméně výčet operačních systémů, které je možné provozovat na tomto zařízení je impozantní a zahrnuje Linux, FreeBSD, NetBSD, OpenBSD, Plan 9, RISC OS a Windows 10 IoT Core [3]. Pro účely této bakalářské práce byla zvolena enterprise distribuce Linuxu CentOS 7.

3.2 Enterprise distribuce Linuxu

Enterprise distribuce Linuxu je označení pro takovou distribuci, která je určena pro podnikové použití. Specifikum podnikového užití spočívá zejména ve stabilitě distribuce a délce podpory. Typická délka podpory jedné verze takové distribuce je pak 10 let, což je standardní doba provozování běžné podnikové aplikace.

3.2.1 CentOS

CentOS je komunitní distribuce OS Linux, které je 100% binárně kompatibilní s RedHat Enterprise Linux, což je nejrozšířenější distribuce OS Linux v podnikovém prostředí. Jelikož je RedHat Enterprise Linux založený na svobodném a otevřeném software, jsou v souladu s licencí publikovány zdrojové kódy této distribuce. Ty jsou následně převzaty tvůrci komunitních distribucí, přeloženy a distribuovány jakožto binárně kompatibilní distribuce. Nicméně mezi RedHat Enterprise Linux a CentOS existuje výrazně užší spolupráce, než pouhé převzetí kódů [4].

Hlavní výhodou CentOS, díky tomu, že vychází z distribuce RedHat Enterprise Linux je aktualizace programových balíčků. Pro verzi 7 tyto aktualizace skončí 30. června 2024[1]. Již v tuto chvíli existuje beta program RedHat Enterprise Linux

verze 8, což bude další verze této enterprise distribuce a s podporou procesorů ARM a tedy i Raspberry PI se zde počítá. Výše uvedená data pak dávají záruku aktualizací v časových horizontech, které jsou očekávané ve firemním prostředí.

3.2.2 Ubuntu LTS

Ubuntu LTS [5] je další rozšířená distribuce OS Linux v podnikovém prostředí. Na rozdíl od CentOS, který vychází z RedHat Enterprise Linuxu a používá tedy balíčkovací systém RPM, je Ubuntu LTS založena na distribuci Debian a její balíčkovací systém je tedy odlišný. Vztah k Raspberry PI je zde bližší, jelikož referenční implementace OS pro Raspberry PI Raspbian[3] vychází z distribuce Debian. Hlavní výhodou Ubuntu LTS na Rasperry PI je jeho velké podobnost s referenční implementací, což může být podstatné především pro začínající uživatele.

Ubuntu LTS má také garantovaný cyklus aktualizací. Pro verzi 18.04 tyto aktualizace skončí 30. dubna 2023[5]. Oproti CentOS resp. RedHat Enterprise Linux je tedy poloviční, což je pro enterprise aplikace a podnikové užití poměrně podstatný signál.

3.3 Firewally

Jako firewall se obecně označuje řešení, jehož hlavním úkolem je řízení a provádění zabezpečení komunikace v počítačové síti [6]. V kontextu komunikace v síti může jít jak o zabezpečení externí komunikace tj. z vnitřní sítě do Internetu a opačně, tak i interní tedy v rámci vnitřní počítačové sítě. Pro popis fungování firewallu je nejprve nutné popsat principy fungování síťových protokolů a jejich mapování na činnosti firewallu.

3.4 Síťové protokoly

Síťové protokoly [7], [8] jsou definovány jako formálními standardy zahrnující pravidla, postupy a formáty, které definují komunikaci mezi dvěma nebo více zařízeními v síti. Síťové protokoly tedy zahrnují všechny procesy, požadavky a omezení při iniciování, průběhu i ukončování komunikace mezi počítači, servery, směrovači a dalšími podpůrnými zařízeními v počítačové síti tzn., definují jak formátovat, přenášet a přijímat data.

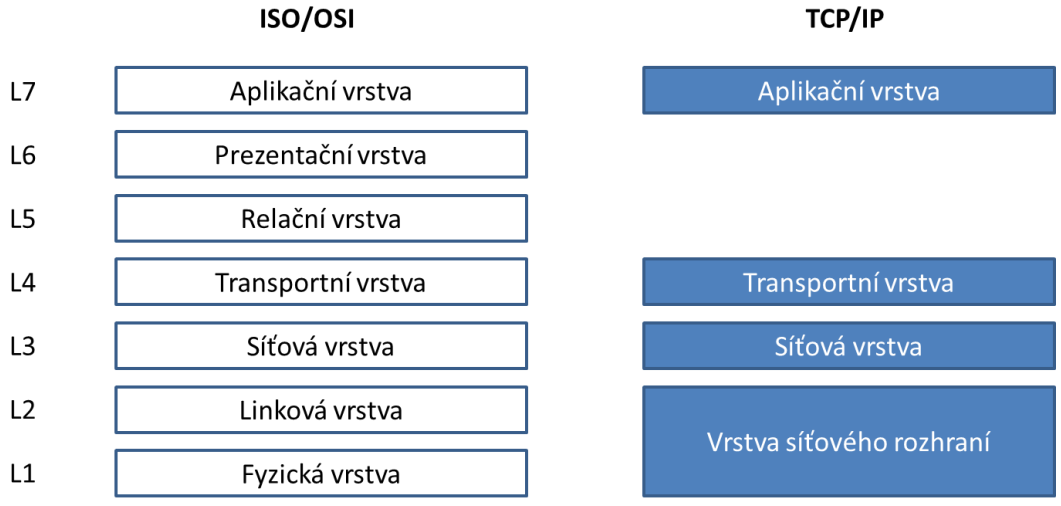
3.4.1 Referenční model ISO/OSI vs. TCP/IP

Model OSI [7] vytvořený Mezinárodní organizací pro standardizaci (ISO) popisuje sedm hierarchických vrstev, které zajišťují fungování sítě. Každá vrstva má na starosti obsluhu určité definované množiny funkcí tj. poskytuje určité služby. Klíčovým pravidlem tohoto modelu je, že každá vyšší vrstva modelu využívá služeb bezprostředně nižší vrstvy. Vzhledem k tomu, že skupina, která tento model tvořila, byla primárně ze sféry telekomunikací a její práce je především teoretická i model OSI tuto skutečnost reflektuje.

Oproti tomu TCP/IP [7], [8] je protokol, který vznikl opačným způsobem jako praktická realizace komunikačního protokolu pro síť Internet, která byla následně standardizována a je zde patrný akcent na praktické užití. Na rozdíl od OSI mám pouze 4 vrstvy, přičemž všechny je možné použít při realizaci firewallu s vědomím, že se k nim přistupuje odlišně.

OSI myšlenkově vychází z modelu, že existují tři spodní vrstvy, které se starají o spolehlivý přenos, tři horní vrstvy poskytují podporu aplikacím a spojuje je „přizpůsobovací“ transportní vrstva. TCP/IP naopak implementuje myšlenku, že spolehlivost přenosu je věcí až koncových účastníků tedy transportní vrstvy. I to je důvodem proč TCP/IP nemá vrstev sedm, ale pouze čtyři jak je vidět ba obrázku (Obrázek 2).

Obrázek 2: Vztah mezi OSI a TCP/IP



Zdroj: Vlastní zpracování s využitím [7]

3.5 Typy firewallů

Stejně tak jako existuje mapování mezi OSI a TCP/IP, existuje i vztah mezi TCP/IP a různými typy firewallů

3.5.1 Nestavové paketové filtry

Jedná se o nejjednodušší typ firewallu, který pracuje na síťové vrstvě TCP/IP. Paketový filtr [8] získal svůj název podle paketů, což je označení bloků dat na síťové vrstvě. Každý paket obsahuje hlavičku a vlastní data. Zatímco o obsah dat se paketový filtr vůbec nezajímá, z hlavičky přečte informaci o zdrojové adrese, cílové adrese a cílovém portu. Pokud se tyto informace shodují s definovaným pravidlem, provede odpovídající akci, což může být např. zahození nebo propuštění dál daného paketu. Jelikož se jedná o nestavový filtr a pracuje se na síťové vrstvě, mohou být využity pouze informace na úrovni IP hlaviček a nemůže být prověřován stav, ve kterém se paket nachází [9].

3.5.2 Stavové paketové filtry

Stavový paketový filtr pracuje na transportní vrstvě, tzn. kromě informací z IP hlavičky má i informace o stavech paketu a další informace z hlaviček TCP paketů. Je možné rozlišovat jednotlivé pakety ze stejného zdroje a/nebo cíle i na základě stavů, které byly nastaveny předchozím paketem. Lze tedy rozpoznat spojení, které spolu vzájemně souvisí, což má dopady jak na rychlost zpracování, tak i na typická pravidla stavového paketového filtru[8].

3.5.3 Stavové paketové filtry s inspekcí protokolů a IDS

Tento typ stavových filtrů [9] pracuje na transportní vrstvě, nicméně zajímá se nejenom o část hlaviček, ale i o datový obsah paketu. Jsou tedy schopné ověřit validitu datového obsahu, aniž by rozuměli vlastnímu datovému obsahu[8]. Vzhledem k tomu, že stále větší množství komunikace na Internetu probíhá jako zabezpečená komunikace pomocí TLS (jedná se tedy o šifrovanou komunikaci mezi dvěma koncovými body tedy na aplikační úrovni), je někdy výhodné tuto inspekci provádět. Firewall nebude rozumět vlastnímu obsahu komunikace, ale je schopen detekovat, zda nedochází k nekorektnímu použití konkrétního aplikačního protokolu. Jako příklad lze uvést P2P komunikaci, která se vydává za HTTPS komunikaci.

Z druhé strany pak firewall dokáže identifikovat, zda v rámci datového toku neprobíhá nějaký známý vektor útoku a takový útok zastavit. To je označováno jako inline IDS (Intrusion Detection Systems – systémy pro detekci útoků). Tento typ paketového filtru je výrazně náročnější na hardware, uvádí se 2-3 krát oproti běžnému stavovému paketovému filtru.

3.5.4 Aplikační brány

Zásadním rozdílem mezi paketovým filtrem a aplikační bránou [6] (někdy se lze setkat s označením proxy firewall) je vrstva, ve které pracují. Tak je v případně aplikační bran jak napovídá již název vždy aplikační. Současně mají za cíl zcela oddělit sítě, mezi které jsou postaveny. Komunikace probíhá tak, že aplikační brána na jedné straně přijímá požadavek konkrétního aplikačního protokolu, kterému rozumí (např. smtp nebo http), požadavek zpracuje např. tak, že provede volání konkrétního serveru za volajícího a výsledek operace předá původnímu volajícímu. Tento typ firewallů je v současné době v porovnání s paketovými filtry využíván méně.

3.6 Paketový filtr v operačním systému Linux

Vlastní paketový filtr v operačním systému Linux (označovaný jako netfilter) se skládá z několika navazujících komponent.

Na nejnižší úrovni, jako součást jádra, je netfilter hook API, které se stará o propojení vlastního síťového stacku jádra a obslužných modulů v user space. V principu, každý paket, který projde síťovou vrstvou jádra, je dále zpracován mechanismem hooků netfilteru[8].

Na hooky jsou navázány základní tabulky filter, mangle, nat, raw a security. Ty jsou propojeny s vybranými základními sadami pravidel definovaných v řetězcích pravidel. Jejich aplikace je tedy obsluha jednotlivých hooků. V systému je definováno pět základních řetězců pravidel [8]:

- **PREROUTING**
- **INPUT**
- **FORWARD**
- **OUTPUT**
- **POSTROUTING**

Tyto řetězce současně definují standardní chování, které je aplikováno pro případ, že řetězec pravidel žádné pravidlo neobsahuje.

Paket postupně prochází řetězci pravidel, která jsou definovaná v rámci jednotlivých částí řetězu popsaných dále. Platí, že každá část řetězu je spojena s jiným typem zpracování paketů. Pravidlo může způsobit provedená akce, chybu nebo skok do jiného pravidla a toto může být libovolně opakováno (je při tom zapamatováno, odkud skok proběhl). Každý síťový paket, který přichází nebo odchází z počítače, prochází nejméně jedním řetězcem pravidel, tedy i alespoň jednou tabulkou a je na něj aplikována alespoň politika řetězce pravidel tedy nějaká akce.

Existují tyto akce [8]:

- **ACCEPT** propustí paket po aplikaci pravidla dále.
- **DROP** paket zahodí.
- **REJECT** je speciální případ chování DROP, který paket také zahodí, ale dá vědět volajícímu pomocí ICMP zprávy.
- **RETURN** má za následek ukončení procházení řetězcem pravidel a pokračování v dalším pravidle v předchozím (volajícím) řetězci. Pokud žádné další pravidlo neexistuje, osud paketu definuje standardní politika řetězce pravidel.

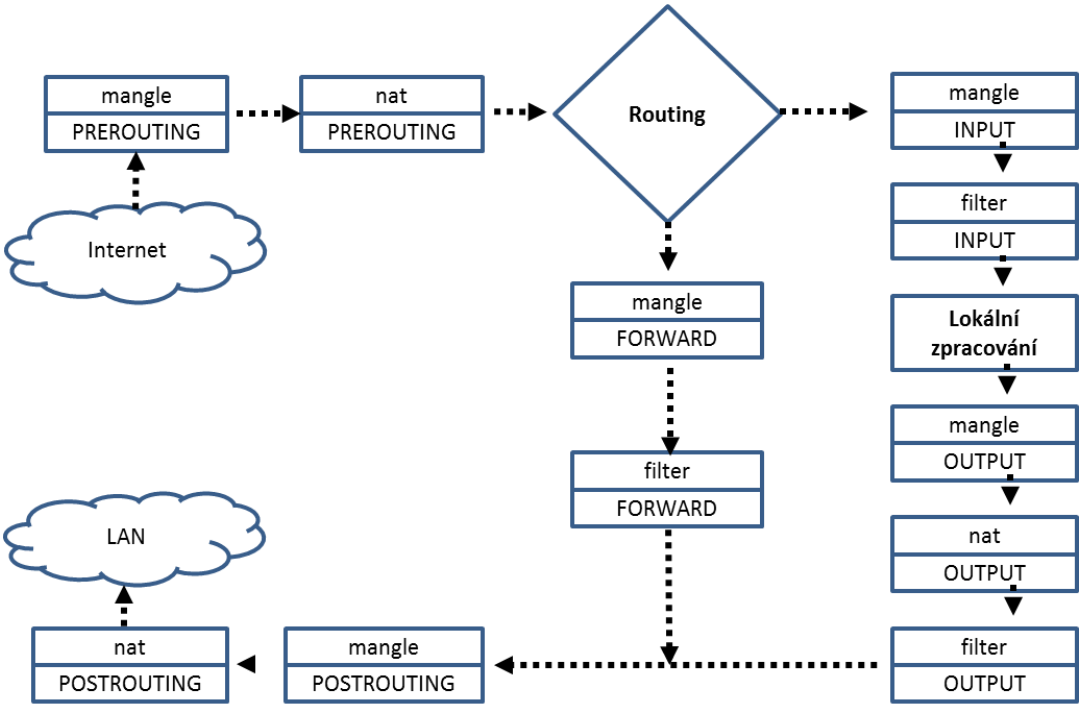
3.6.1 Popis základních tabulek filter, nat a mangle

V systému existuje celkem pět tabulek [8], které jsou využitelné pro filtrování paketů, nejčastěji jsou využívány následující:

- **filter** je standardní tabulka, která je využívá pro filtrování paketů. Pokud není v definici pravidla uvedena explicitně jiná tabulka, předpokládá se použití tabulky filter. Typicky se tato tabulka využívá v kombinaci s řetězci pravidel INPUT, FORWARD a OUTPUT.
- **nat** je tabulka určená pro překlad adres a její využití je především v případě, že směrujeme provoz z a do Internetu z vnitřní sítě přes jednu IP adresu brány. Typicky se tato tabulka využívá v kombinaci s řetězci pravidel PREROUTING, INPUT, OUTPUT a POSTROUTING.
- **mangle** je tabulka určená pro práci s hlavičkami paketů např. pro označení paketů, nastavení priorit apod. Tuto tabulku lze využít v kombinaci se všemi řetězci pravidel.

3.6.2 Schéma toku paketů netfilterem

Obrázek 3: Tok paketů netfiltrem



Zdroj: Vlastní zpracování s využitím [8], [9], [10]

3.6.3 Nftables vs. iptables

Od jádra linuxu verze 3.13 je k dispozici modul nftables, který nahrazuje část netfilteru (především pak moduly iptables, ip6tables, arptables a ebtables) a slučuje je celé do jednoho logického celku. To přináší zjednodušení, pokud jde o velikost a duplikování kódu v kernelu, ale i pro vlastní obsluhu paketového filtru. Nově je zde implementován jednoduchý virtuální stroj, který interpretuje pravidla jako bytecode, vyhodnocuje a provádí jednoduché operace s paketem[8].

Nastavení pravidel, která se aplikují je pak věcí uživatelských nástrojů jako iptables, ip6tables, arptables, ebtables nebo nft v závislosti na použitých modulech jádra.

Standardně používá distribuce CentOS 7 na procesorech x86 jádro linuxu postavené na verzi 3.10, a proto lze používat pouze netfilter. Na procesorech ARM, Raspberry PI distribuce v distribuci CentOS 7 využité jádro verze 4.14 s tím, že v komunitě je diskutován přechod na 4.19 LTS. Existují dva možné přístupy, jak nastavit paketový

filtr a to pomocí nástroje iptables nebo pomocí nástroje nft. Pro snadný přechod mezi iptables a nftables existuje nástroj iptables-translate [8], který převádí původní pravidla do nové formy zápisu a značně usnadňuje přechod. V této práci použijeme nastavení pomocí iptables a to především proto, že CentOS 7 jej používá jako výchozí nastavení. Naopak dle avizované nové verze CentOS 8, která bude vycházet z RedHat Exterprise Linux 8, bude nově výchozí platforma pro paketový filtr nftables [10].

3.7 Filtr na aplikační vrstvě

V současné době probíhá většina komunikace zabezpečeně pomocí SSL [11] a navíc webové prohlížeče začaly nově uživatele varovat před použitím nezabezpečených stránek [12]. Proto je filtrování komunikace na aplikační vrstvě prakticky nemožné, pokud nemá docházet k MITM (útok Man-in-the-middle).

3.7.1 Realizace pomocí aplikační brány

Varianta s MITM je často realizována u velkých korporací a probíhá tak, že uživatelé musí přistupovat přes proxy, která terminuje SSL spojení a následně sama vyřizuje požadavek s cílovým serverem, čímž funguje jako aplikační brána. Nicméně toto vyžaduje, aby měl uživatel nainstalovaný certifikát korporátní certifikační autority v prohlížeči. Tento postup ovšem není vhodný pro užití v kombinaci s Raspberry PI, jelikož vyžaduje poměrně velké prostředky.

3.7.2 SNI

Pro filtrování je možné využít i vlastností protokolů TLS 1.2 [13], konkrétně filtrování pomocí Server Name Indication (SNI). Jedná se o vlastnost protokolu TLS, který umožňuje na jedné IP provozovat více virtuálních serverů. Nejčastěji se s tímto lze setkat u velkých cloudových poskytovatelů jako je Amazon. Google nebo Microsoft, kdy na jedné IP je provozováno větší množství služeb, tedy virtuálních serverů.

Při zahájení komunikace, předtím, než je sestaven SSL tunel, klient odesílá jméno serveru, se kterým chce komunikovat, v otevřené formě. To je prakticky jediná část komunikace, která v protokolu TLS není šifrovaná, vše ostatní je už nečitelná šifrovaná komunikace.

Samotný paketový filtr Linuxu umožňuje omezit pouze komunikaci na třetí resp. čtvrté vrstvě tedy na úrovni, zatímco pomocí SNI [13] se lze dostat na aplikační úroveň. To vyžaduje implementaci modulu paketového filtru, který bude se SNI pracovat. Standardně takový modul není součástí enterprise distribucí, ale je nutné ho do distribuce přidat.

3.8 Nastavení pomocí iptables

Informaci o používané verzi programu získáme pomocí příkazu [8]

```
$ iptables -V
```

Odpověď může vypadat např. takto:

```
iptables v1.8.0 (legacy)
```

Pokud odpověď obsahuje řezezec „(legacy)“ pak to znamená, že se používá původní netfilter nikoliv nftables [8].

Detailní popis chování uživatelského aktuální verze programu iptables, který je dostupný pouze pro superuživatele (roota) je možné najít v manuálových stránkách operačního systému nebo na [14].

3.8.1 Práce s řetězy

Standardní nastavení paketového filtru v Linuxu je v režimu propouštění všech paketů dále do systému [8]. Označuje se jako politika řetězu a všechny jsou standardně v režimu ACCEPT. Znamená to, že pro efektivní nastavení firewallu je nutné politiku nastavit na režim DROP nebo se politika ponechá a jako poslední pravidlo je aplikováno odmítnutí všech paketů, které došly až k tomuto pravidlu. Výhodou druhého přístupu je možnost takové pakety logovat pro další využití. To se provádí přidáním pravidla pro logování těsně před pravidlo pro definitivní odmítnutí paketu na konci řetězce.

Kromě standardní řetězů definovaných systémem, je možné uživatelsky definovat nové řetězy [8] a do nich vkládat nová pravidla a následně je i rušit za předpokladu, že již neobsahují žádná uživatelem definovaná pravidla.

3.8.2 Práce s pravidly

Práce s jednotlivými pravidly se provádí pomocí tohoto obecného zápisu:

```
iptables -t < tabulka > < příkaz > < řetěz > < specifikace
pravidla >
```

Tabulky a řetězy jsou popsány v kapitole 3.6. Příkazy slouží pro manipulaci s pravidly, tedy pro vložení na přesnou pozici nebo na konec řetězu a zrušení pravidla. Pravidla je také možné hromadně odebrat. Specifikace pravidla obsahuje typicky zdroj a cíl, což může být síťový interface, IP adresa nebo rozsah adres, specifikace protokolu, případně příznaků v rámci protokolu, port a akci, která se má vykonat.

3.9 Síťové služby

Na firewallu v případě domácnosti nebo malé firmy mohou být provozovány i další služby [15] a to jak přímo na vlastním hardware firewallu nebo na dalším počítači, virtuálním stroji apod. Základní hledisko pro členění služeb, je jejich dostupnost z vnitřní sítě nebo z internetu. V případě, že služba je dostupná pouze z vnitřní sítě, pak naslouchá pouze na vnitřním síťovém rozhraní a firewall k ní jakýkoliv přístup odjinud blokuje. Pokud je služba dostupná z Internetu, pak typicky běží v tzv. demilitarizované zóně, což je logická podsíť, oddělená od vnitřní sítě. Firewall řídí přístup k této službě jak z Internetu, tak i z vnitřní sítě.

3.9.1 DHCP

Typická malá počítačová síť v domácnosti nebo malém podniku pracuje na protokolu IPv4 a používá dynamické přidělování IP adres. V případě podnikové sítě by mělo platit, že se důsledně rozlišuje mezi firemním zařízením, což jsou typicky veškeré počítače a síťová zařízení patřící firmě nebo domácnosti a pak hostujícími zařízeními, tedy takovými, která se v síti objevila např. v důsledku návštěvy apod.
O přidělování odpovídajících IP adres ve vnitřní síti se pak stará server DHCP [16] který rozlišuje zařízení známá a neznámá a podle toho přidělí odpovídající IP adresu. Přidělená IP adresa pak má vliv na aplikovaná pravidla na firewallu. Zároveň DHCP přiděluje odpovídající adresy DNS serverů.
Služba DHCP je vždy firewallem řízena tak, aby byla dostupná pouze do vnitřní sítě.

3.9.2 DNS

DNS slouží pro překlad IP na doménové jméno počítače nebo naopak pro překlad doménové jména počítače na IP adresu [15]. DNS je jednak označení pro hierarchický systém doménových jmen, jednak i označení serveru [17], který tuto službu realizuje a v neposlední řadě i označení protokolu pro realizaci této služby [18].

V rámci řešení je DNS server použit ve dvou rolích. Jedna se jedná o server pro vnitřní síť, kde poskytuje služby DNS a dále pak o tzv. kešovací DNS, tedy server, který zprostředkovává a ukládá dotazy na jiné DNS servery. Uživatel z vnitřní sítě je tedy v každém okamžiku obsluhován tímto DNS server bez ohledu na to, na jakou IP adresu nebo doménové jméno se ptá.

V současné době probíhá ohledně protokolu DNS rozsáhlá debata ohledně zabezpečení DNS, jelikož protokol sám o sobě je nezabezpečený a je tedy možné jeho obsah odposlouchávat. Na jedné straně to znamená jisté riziko pro soukromí, na druhou stranu je pak možná na základě obsahu tohoto protokolu filtrovat nežádoucí obsah, jako je například nevyžádaná reklama, útočné webové stránky apod. Ve hře jsou zejména dva standardy a to DNS over TLS[19] a DNS over HTTPS [20].

Služba DNS je standardně firewallem řízena tak, aby byla dostupná pouze do vnitřní sítě, a směrem k Internetu se chová pouze jako klient.

3.9.3 NTP

V každé počítačové síti se musí synchronizovat čas, jelikož úspěšnost mnoha operací je na aktuálním čase závislá. Současně platí, že čas by měla určovat autorita, nikoliv jednotlivé počítače v roli klienta v síti. Právě k tomuto účelu slouží NTP server [21], který čas pravidelně synchronizuje s hlavními časovými servery prostřednictvím počítačové sítě. Podobně jako DNS používá i NTP hierarchický systém, kdy nejvýše jsou hodiny synchronizované pomocí např. GPS a ostatní servery níže v hierarchii od nich čas následně odvozují.

Služba NTP je vždy firewallem řízena tak, aby byla dostupná pouze do vnitřní sítě, a směrem k Internetu se chová pouze jako klient.

3.9.4 OpenVPN

V případě, že je potřeba se připojit do vnitřní sítě z prostředí Internetu, je nutné nejprve sestavit šifrovaný tunel a tímto tunelem vést komunikaci. Na sestavení tohoto tunelu slouží software OpenVPN [22], který pro sestavení používá klientský certifikát vydaný důvěryhodnou certifikační autoritou Pro účely malé firmy nebo domácnosti je možné využít i vlastní certifikační autoritu. Na základě informací v certifikátu, zejména pak na základě informací o vydavateli certifikátu a CN je pak sestaveno odpovídající zabezpečené připojení.

Pro připojení prostřednictvím OpenVPN do vnitřní sítě musí být dostupná veřejná IP adresa, jinak není možné spojení přes VPN navázat přímo. Variantou je pak navázání spojení na VPN veřejně dostupnou v Internetu, která následně zprostředkuje připojení do tunelu VPN, která je vytvořena z prostředí, které nemá vlastní veřejnou adresu nebo aktivní port forwarding.

Firewall tedy směrem do Internetu poskytuje právě jeden UDP port a každý uživatel musí mít vlastní certifikát[23]. V případě požadavku na dodatečné zabezpečení je možné OpenVPN dále doplnit např. o jednorázová hesla [23], která jsou generovaná pomocí mobilního telefonu uživatele, čímž lze prakticky eliminovat jakýkoliv, pokud o průnik do vnitřní sítě.

4 Vlastní práce

4.1 Způsob realizace vlastní práce

Realizace vlastní práce vychází z metodiky. Autor nejprve provede instalaci minimální image enterprise distribuce CentoOS 7 na SD kartu a to bude výchozí stav pro následné kroky. Minimální instalace obsahuje základní programy a počáteční konfiguraci, která ovšem není vhodná pro provoz firewallu. Autor během své práce provede sérii dodatečných instalací a konfigurací síťových prvků a služeb tak, aby vznikl provozuschopný firewall se základními síťovými službami. Tento výstup bude následně porovnán s původní instalací a vznikne tak diferenční soubor, který bude základem pro vytvoření instalačního skriptu, který se bude řídit konfigurací.

Následně bude tento skript vytvořen a provedena ukázková konfigurace zařízení na základě předem definované konfigurace. Samotná konfigurace je relativně jednoduchý soubor, nicméně autor v práce naznačí i možnost jeho vytvoření pomocí jednoduché webové aplikace.

V další části jsou zhodnoceny praktické možnosti tohoto firewallu, zejména v kontextu jeho výkonu a současných možností. Bude provedeno měření propustnosti síťových prvků a srovnání se statistickými daty o rychlosti internetového připojení v ČR. Na základě srovnání těchto údajů a dat z reálného provozu bude vyhodnocena vhodnost Raspberry PI 3B+ pro využití jako firewall pro domácí a menší podnikové sítě.

4.2 Požadavky, předpoklady a výchozí nastavení

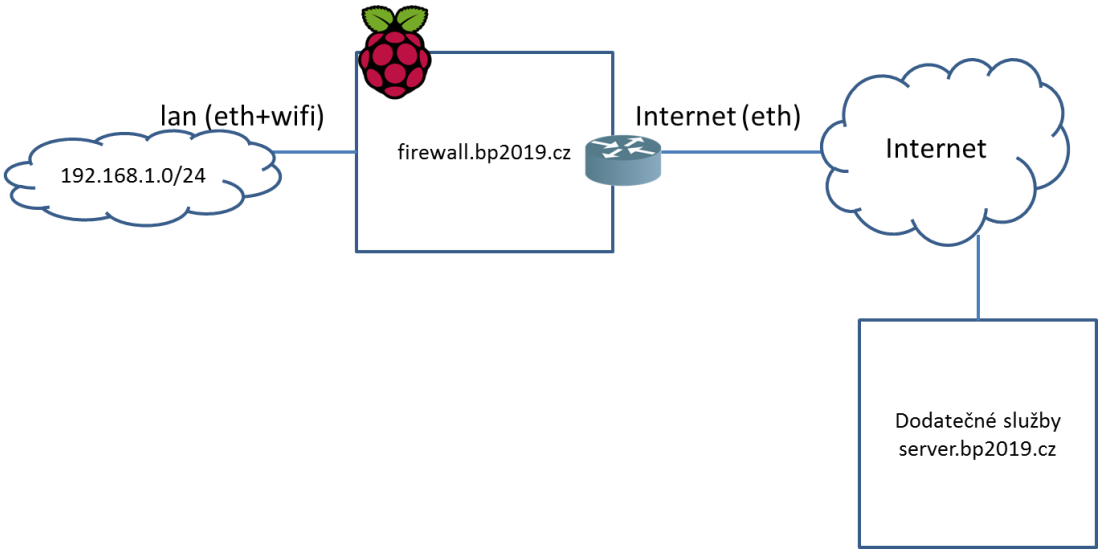
Předpokládáme, že počítačová síť, kterou bude náš firewall zabezpečovat je malého rozsahu, tzn., bude obsahovat řádově desítky zařízení, jako jsou počítače, mobilní telefony, herní konzole, televize a jinou „chytrou“ spotřební elektroniku. Většina zařízení bude připojena bezdrátově (některá ani jinou možnost připojení nenabízí), nicméně některá zařízení mohou být připojena ii pomocí síťového kabelu.

4.2.1 Použitá počítačová síť

Rasperry PI tedy plní jednak roli směrovače, dále pak firewallu a wifi přístupového bodu. Má 3 síťová rozhraní, jedno je realizované zabudovanou síťovou kartou, do které je přiveden Internet od poskytovatele, dále pak vestavěné wifi rozhraní

a pomocí USB připojenou další síťovou kartu, která je připojena do vnitrofiremního switche s RJ-45 porty. Celé schéma je znázorněno na obrázku (Obrázek 4)..

Obrázek 4: Schéma sítě



Zdroj: Vlastní zpracování

4.2.2 Požadované služby

Raspberry PI disponuje dostatečnou kapacitou na to, aby zvládlo roli směrovače a firewallu. Navíc bude plnit i roli následujících serverů:

- Webového serveru, který bude poskytovat statickou firemní prezentaci prostřednictvím protokolu HTTPs.
- SSH serveru pro připojení administrace s jedinou možnou autentizací pomocí klíče
- VPN serveru, který pomocí OpenVPN umožní uživatelům přístup do vnitřní sítě
- DNS a DHCP serveru pro užití ve vnitřní síti. V případě DNS se očekává, že bude probíhat filtrování nežádoucího obsahu.
- Emailového serveru v roli SMTP relay

Dále bude směrovat požadavky na vybraný server, poskytující služby těchto serverů:

- Emailového serveru, který bude obsluhovat po řádově stoky uživatelů, přičemž každý uživatel bude mít diskovou kvótu a podporované protokoly budou pouze zabezpečený IMAP a SMTP včetně jeho zabezpečené verze.

- XMPP serveru, který bude zabezpečovat služby chatu pro řádově stovky uživatelů

Současně budou instalované aplikace pravidelně aktualizované na poslední verzi

4.3 Instalace a konfigurace Raspberry PI

4.3.1 Instalace dodatečného hardware na Raspberry PI

Vlastní Raspberry PI je doplněno o další síťové rozhraní pomocí USB karty. Po přidání síťové karty se objeví, následují hlášení v logu, čímž OS informuje, že došlo k přidání nového zařízení a je připraveno k použití. Může být použita libovolná USB síťová karta kompatibilní s operačním systémem. Je doporučeno použít gigabitové síťové karty, jinak dojde k degradaci výkonu tohoto rozhraní. V našem konkrétním je použito zařízení ASIX AX88179, které je označeno v systému jako eth1 a má MAC adresu 00:13:3b:9c:93:bb. K tomuto zařízení bude připojen kabel vedoucí od poskytovatele Internetu.

```
[16.075774] ax88179_178a 1-1.1.3:1.0 eth1: register 'ax88179_178a'
at usb-3f980000.usb-1.1.3, ASIX AX88179 USB 3.0 Gigabit Ethernet,
00:13:3b:9c:93:bb
[25.032204] ax88179_178a 1-1.1.3:1.0 eth1: ax88179 - Link status
is: 1
[25.038428] IPv6: ADDRCONF(NETDEV_CHANGE): eth1: link becomes
ready
```

Po hardwarové stránce tedy platí, že do síťové karty připojené prostřednictvím USB se vždy připojí kabel od poskytovatele Internetu (ISP), do síťového rozhraní přímo v Raspberry PI se připojí patch kabel od lokálního firemního switche nebo hubu pokud takový ve firmě existuje. Pokud firma nebo domácnost nepoužívá strukturovanou kabeláž, ale pouze wifi připojení, pak zdířka bude neobsazená.

4.3.2 Základní instalace operačního systému na Raspberry PI

Základní instalace operačního systému se provádí přenesením image obsahující distribuci CentOS 7 na SD kartu. Pro firewall a základní servery dostačuje SD karta

o kapacitě 16 GB. Po instalaci a konfiguraci lze SD katu snadno naklonovat a vytvořit tak zálohu pro případ, že by došlo k poškození karty.

Nejprve si na libovolném stoji stáhneme image s CentOS 7 pro ARM z nejbližšího zrcadla ze seznamu dostupném na adrese [24] a provedeme jeho instalaci na SD kartu. Na instalaci image existuje vícero postupů pro různé operační systémy. Následující příkazy provedou instalaci image CentOS 7 revize 7.6 1810, která je uložena v souboru s názvem centos.raw.xz na SD kartu v prostředí operačního systému Linux, větší detail je možné načerpat v literatuře [24].

```
$ xzcat centos.raw.xz | sudo dd of=/dev/mccblk0 bs=4M
$ sync
```

Poté je možné s touto SD kartou nabootovat Raspberry PI a na zařízení přihlásit pod uživatelem **root** a heslem **centos.**

Následuje krok prvotní konfigurace systému, který probíhá manuálně a autor ho provádí proto, aby následně získal údaje potřebné pro vytvoření konfiguračního souboru pro skript automatizované instalace.

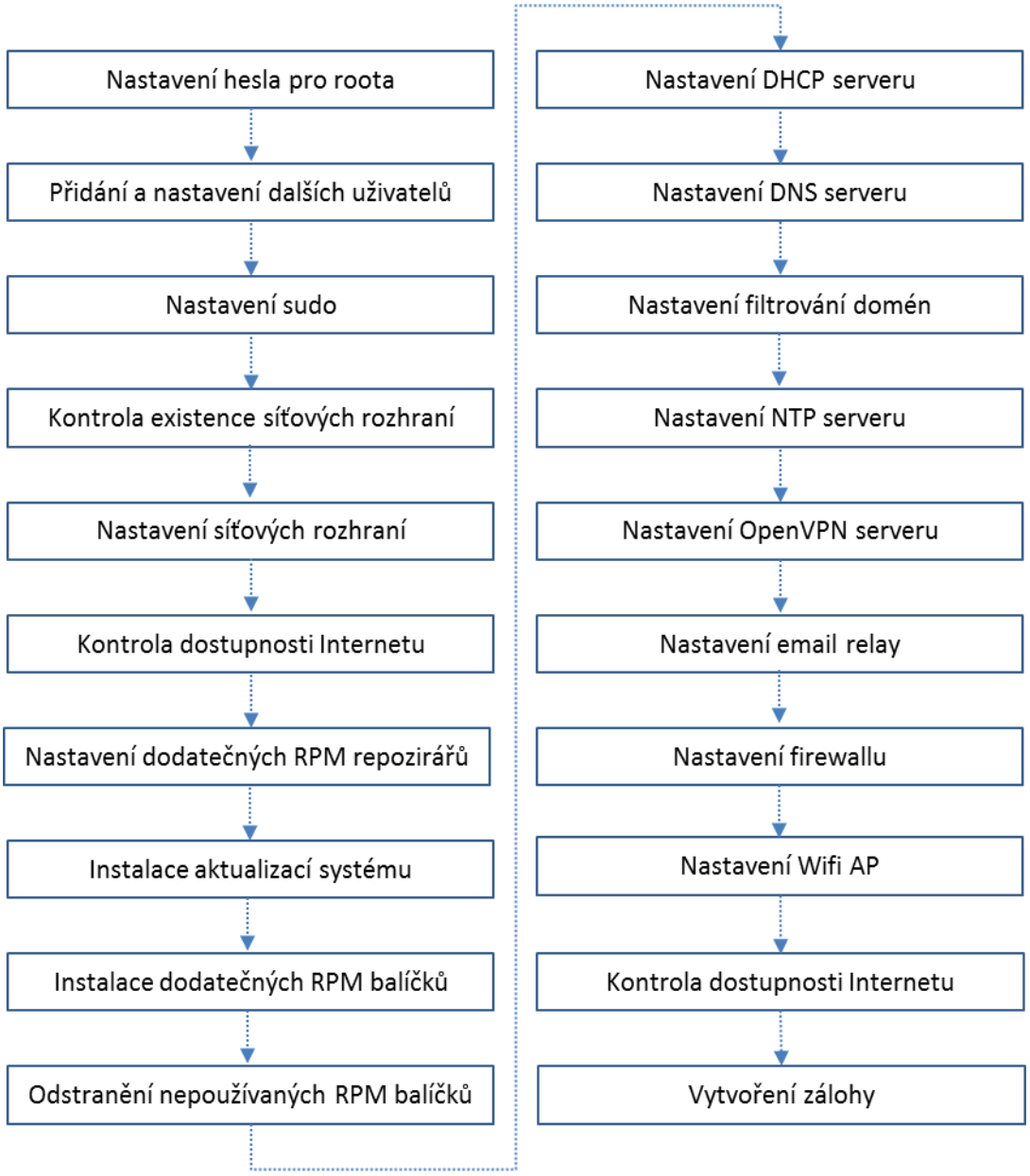
4.4 Prvotní konfigurace systému

Prvotní konfigurace systémů spočívá v postupném manuálním provedení sledu kroků znázorněných na obrázku (Obrázek 5), přičemž úspěšnost některých podmiňuje kroky následující. Pro konfiguraci byla vypracována metodika, která zachycuje sled kroků, které jsou provedeny pro prvotní nakonfigurování systému a jejich výstup slouží jako podklad pro vypracování konfigurace pro a automatickou instalaci.

Přestože by se mohlo na první pohled dát, že některé kroky je možné označit jako volitelné a jejich neprovedení, případně neúspěšné provedení tedy není fatální, není tomu tak. Některé chyby by se projevili hned (např. chybějí rozhraní nebo chybějící RPM balíček) a jiné až po chvíli (např. neprovedená nebo chybná konfigurace DHCP nebo DNS).

V prvotní konfiguraci byl použit modelový příklad nastavení sítě, aby bylo možné následně jednoduchým způsobem dohledat části, které budou parametrizovatelné prostřednictvím konfigurace pro automatickou instalaci.

Obrázek 5: Schématické znázornění instalačního procesu



Zdroj: Vlastní zpracování

Předpokládá se, že většina uživatelů především z domácího prostředí nemá statickou IP adresu, zatímco firemní uživatelé ano, nicméně velmi často z privátního rozsahu a následně je tato adresa překládána 1:1 na veřejnou IP adresu dle standardu IPv4. To je dáno tím, že velké množství ISP si za pevnou a veřejnou IP adresu účtuje buď jednorázovou odměnu, nebo měsíční poplatek.

V textu dále se popisují kroky, které provádí instalační skript, nikoliv detailně jednotlivé příkazy prováděné instalačním skriptem a jeho výstupy, pokud to není

výslovně nutné. Pro účely této bakalářské práce byla pro prvotní nastavení použita statická IP adresa přidělená ISP 10.56.71.230 s maskou 255.255.255.224 a IP adresa brány 10.56.71.225, přičemž adresa z vnitřního rozsahu je překládána ISP 1:1 na veřejnou adresu. Pro vnitřní adresy byla použita IP adresa 192.168.1.1 pro rozhraní určené pro pevné připojení do switche pomocí patch kabelu a adresa 192.168.2.1 pro rozhraní bezdrátové sítě.

4.4.1 Nastavení uživatelů a sudo

Změna hesla superuživatele root je provedena v interaktivním režimu příkazem ***passwd***. Jelikož by takový režim nebyl vhodný pro neinteraktivní použití ve skriptu [25], pak je použita následující kombinace příkazů. Heslo je záměrně generované, jelikož nikdy v budoucnu se nepředpokládá jeho běžné používání.

```
# Vygeneruje, zobrazí a nastaví heslo, dlouhé 12 znaků
$ openssl rand –base64 12 | echo - | passwd --stdin root
```

Přidání dalších uživatelů a skupin se provede pomocí příkazu useradd a groupadd. Je přidána skupina admin a sshusers, uživatel admin, je zahrnut do skupin admin, sshusers a wheel.

```
# Přidá skupiny admin a sshusers
$ groupadd admin
$ groupadd sshusers

# Založí uživatele admin, přidá do skupin a nastaví heslo
$ useradd –c Administrator –g admin –G sshusres, wheel admin
$ openssl rand –base64 8 | echo - | passwd --stdin admin
```

Díky přidání do skupiny wheel [26], může uživatel využívat mechanismu sudo, tedy provádět vybrané příkazy jako superuživatel bez nutnosti znát heslo superuživatele. Proto mohlo být heslo superuživatele nastaveno jako velmi složité, jelikož se nepředpokládá jeho budoucí užívání. Mechanismus sudo může být používán jednak se znalostí hesla uživatele nebo bez této znalosti a jednak umožňuje nastavit omezení příkazů, které je možné pustit. Pro účely této práce se používá nastavení, kdy

administarátor, tedy uživatel admin může spustit libovolný příkaz pomocí mechanismu sudo.

4.4.2 Nastavení síťových rozhraní

Nejprve je provedena kontrola existence síťových rozhraní. pomocí příkazu ***mncli,*** který zobrazí, která rozhraní jsou dostupná. Pro pokračování je nutné, aby první sloupec obsahoval rozhraní eth0, eth1 a wlan0 a byla správně zapojena kabeláž. Správné zapojení a nastavení je indikováno následujícím příkazem.

```
$ nmcli d
DEVICE  TYPE      STATE             CONNECTION
eth1    ethernet  připojeno         System eth1
wlan0   wifi      připojeno         local-ap
eth0    ethernet  připojeno         System eth0
lo      loopback  není pod správou  --
```

Nastavení se provede pomocí série příkazů ***nmcli*** a restartování služby sítě. V příloze číslo 1 práce je přiložena ukázka konfigurace síťových rozhraní pomocí tohoto příkazu. Příkaz ***mncli*** pouze modifikuje konfigurační soubory. Stejné nastavení může být provedeno tak, že jsou pouze uloženy výsledné konfigurační soubory do adresáře /etc/sysconfig/network-script a následně restartována služby network. Konfigurační soubor pro rozhraní eth0 bude vypadat např. takto.

```
DEVICE=eth0
BOOTPROTO=static
IPADDR=192.168.1.1
NETMASK=255.255.255.0
ONBOOT=yes
IPV6INIT=no
IPV6_AUTOCONF=no
IPV4_FAILURE_FATAL=no
DEFROUTE="no"
```

Test připojení do Internetu je proveden následujícím jednoduchým skriptem.

```
#!/bin/bash

nm-online > /dev/null 2>&1
if [ $? -eq 0 ]; then
    echo "Internet připojen"
else
    echo "Internet nepřipojen"
fi
```

Alternativně může být realizováno i pomocí příkazu wget. Pro správnou funkčnost sítě musí být přidány další RPM repozitáře a nakonfigurovány tři základní servery a to DHCP, DNS a NTP.

4.4.3 Nastavení dodatečných repozitářů

Distribuce CentOS 7 obsahuje poměrně velké množství RPM balíčků, ale ne všechny, které jsou použity v rámci našeho řešení. Proto jsou přidány další repozitáře např. Extra Packages for Enterprise Linux (EPEL) pomocí následujícího příkazu.

```
$ cat > /etc/yum.repos.d/epel.repo << EOF
[epel]
name=Epel rebuild for armhfp
baseurl=https://armv7.dev.centos.org/repodir/epel-pass-1/
enabled=1
gpgcheck=0

EOF
```

4.4.4 Nastavení DHCP serveru

DHCP server [26] slouží pro přidělení IP adres síťovým rozhraní. V případě této bakalářské práce jsou IP adresy přidělovány do rozhraní eth0 a wlan0, přičemž je využito balíčku dnsmasq. Adresy na rozhraní eth0 jsou z rozsahu 192.168.1.128 - 192.168.1.253. Rozsah 192.168.1.2 – 192.168.1.127 je rezervován pro užití v rámci

pevně přidělovaných adres. To mohou být například síťové tiskárny. Adresy na rozhraní wlan0 jsou z rozsahu 192.168.2.128 - 192.168.2.253. Rozsah 192.168.2.2 – 192.168.2.127 je rezervován pro užití v rámci pevně přidělovaných adres.

```
subnet 192.168.1.0 netmask 255.255.255.0 {
        option routers                  192.168.1.1;
        option subnet-mask              255.255.255.0;
        option broadcast-address        192.168.1.255;
        option domain-name              "bp2019.cz";
        ddns-domainname                 "bp2019.cz";
        ddns-rev-domainname             "in-addr.arpa";
        option domain-name-servers      192.168.1.1, 1.1.1.1;
        option ntp-servers              192.168.1.1;
        authoritative;
        range dynamic-bootp 192.168.1.128 192.168.1.253;
        default-lease-time 21600;
        max-lease-time 43200;

}
```

```
subnet 192.168.2.0 netmask 255.255.255.0 {
        option routers                  192.168.2.1;
        option subnet-mask              255.255.255.0;
        option broadcast-address        192.168.2.255;
        option domain-name              "bp2019.cz";
        ddns-domainname                 "bp2019.cz";
        ddns-rev-domainname             "in-addr.arpa";
        option domain-name-servers      192.168.2.1, 1.1.1.1;
        option ntp-servers              192.168.2.1;
        authoritative;

        range dynamic-bootp 192.168.2.128 192.168.2.253;
        default-lease-time 21600;
        max-lease-time 43200;
}
```

Pokud by to bylo užitečné pro konkrétní síť, např. speciálně se jedná o použití síťových tiskáren, je možné vynutiti přidělení konkrétní IP adresy na základě MAC adresy zařízení.

```
host tisk_cb {
               hardware ethernet 00:18:F3:A7:45:15;
                fixed-address 192.168.1.10;
                }

host tisk_b {
               hardware ethernet 00:18:F3:A7:45:16;
                fixed-address 192.168.1.11;
                }
```

4.4.5 Nastavení DNS serveru

DNS server je možné realizovat pomocí balíčku bind [26] nebo pomocí balíčku dnsmasq. V rámci této práce se použije balíček bind. Na firewallu se realizuje DNS server autoritativní pouze pro vnitřní síť a pro ostatní adresy a IP funguje pouze jako tzv. caching nameserver. Tzn., že nové požadavky se vyřizují prostřednictvím jiných nameserverů a hodnoty již použité se uchovávají pro další použití po určitou dobu lokálně (další stejný požadavek je tedy vyřízen přímo tímto serverem bez dotazu na další hierarchii). Veřejné služby DNS pro doménu bp2019.cz jsou realizovány pomocí DNS serveru ISP a jsou dostupné na veřejných IP adresách.
Nejprve je třeba upravit základní konfigurační soubor ***named.conf***. V následující ukázce jsou naznačeny pouze ty části, které jsou v rámci konfigurace měněny.
Specifickou oblastí DNS je filtrování obsahu, zejména pak nežádoucích reklam, útočných stránek, malware apod.

Za tímto účelem je vytvořena následující konfigurace, která tyto nežádoucí domény blokuje.

```
options {
        // Naslouchámе jen na vybraných rozhraních
        listen-on port 53 { 192.168.1.1; 192.168.2.1; };
...
        // Povolený rozsah, odkud je možné se serveru dotazovat
        allow-query { localhost; interni_sit; };
...
         // Blokování nežádoucího obsahu
        response-policy {
           zone "rpz.example.com";
        };
};

# Zahrnutí omezení pro přístup
include "pristup.conf"
# Zahrnutí nastavení pro vlastní síť
include "sit.conf"
```

Konfigurační soubor řídící přístup k DNS ***pristup.conf*** serveru obsahuje následující konfiguraci, která zahrnuje vnitřní sítě, které mohou využívat služeb poskytovaných DNS serverem:

```
acl interni_sit { 192.168.1.0/24; 192.168.20/24; };
acl sekundarni_dns { none; };
```

Konfigurační soubor nastavení interních zón DNS serveru ***sit.conf*** má tento obsah:

```
zone "1.168.192.in-addr.arpa" {
        type master;
        file " 192.168.1.db";
        allow-transfer { sekundarni_dns; };
        allow-query { internal_net; };
        notify yes;
        };
zone "2.168.192.in-addr.arpa" {
        type master;
        file " 192.168.2.db";
        allow-transfer { sekundarni_dns; };
        allow-query { internal_net; };
        notify yes;
        };
zone "bp2019.cz" {
        type master;
        file "bp2019.cz.db";
        allow-transfer { sekundarni_dns; };
        allow-query { internal_net; };
        notify yes;
        };
zone "rpz.bp2019.cz" {
        type master;
        file "rpz.bp2019.cz.db";
        allow-query { none; };
};
```

Konfigurační soubor interních zón DNS serveru se definují v souborech 192.168.1.db, 192.168.2.db, bp2019.cz.db a rpz.bp2019.cz.db.

Příklad souboru 192.168.1.db, obsahující definici zóny 192.168.1, tedy vnitřní sítě připojené do firewallu pomocí kabelu:

```
$ORIGIN .
$TTL 86400      ; 1 day
1.168.192.in-addr.arpa IN SOA  firewall.2019.cz. dns-
admin.bp2019.cz. (
                                2019020201 ; serial
                                3600       ; refresh (1 hodina)
                                1800       ; retry (30 minut)
                                2419200    ; expire (4 týdny)
                                604800     ; minimum (1 týden)
                                )
                        NS      firewall.bp2019.cz.
                        MX      0 server.bp2019.cz.

$ORIGIN 1.168.192.in-addr.arpa.
1                       PTR     firewall.bp2019.cz.
2                       PTR     pevne-2.bp2019.cz.
3                       PTR     pevne-3.bp2019.cz.
...
128                     PTR     pevne-dyn128.bp2019.cz.
...
253                     PTR     pevne-dyn253.bp2019.cz.
```

Pro soubor 192.168.2.db, obsahující definici zóny 192.168.2, tedy vnitřní sítě připojené do firewallu pomocí wifi acces pointu je analogické, odlišnosti panují zejména u určení sítě a názvech reverzních záznamů jednotlivých IP adres:

```
$ORIGIN 2.168.192.in-addr.arpa.
1                       PTR     ap.bp2019.cz.
2                       PTR     wifi-2.bp2019.cz.
...
```

Zóna bp2019.cz je definována následujícím způsobem. Opět se jedná o zkrácenou verzi, ze které je zřejmé ostatní nastavení:

```
$ORIGIN .
$TTL 86400      ; 1 day
bp2019.cz               IN SOA svsovh.dalvi.cz. dns-
admin.bp2019.cz. (
                                2019020201 ; serial
                                3600       ; refresh (1 hodina)
                                1800       ; retry (30 minut)
                                2419200    ; expire (4 týdny)
                                604800     ; minimum (1 týden)
                                )
                        NS      firewall.bp2019.cz.
                        MX      0 server.bp2019.cz.

$ORIGIN bp2019.cz.
localhost               A       127.0.0.1

_autodiscover._tcp      SRV     0 0 443 server.bp2019.cz.
_imaps._tcp             SRV     0 0 993 server.bp2019.cz.
_submission._tcp        SRV     0 0 465 server.bp2019.cz.
_xmpp-client._tcp       SRV     0 5 5222 server.bp2019.cz.
_xmpp-server._tcp       SRV     0 5 5269 server.bp2019.cz.
_xmpp-server._tcp.chat  SRV     0 5 5269 server.bp2019.cz.
_xmpp-server._tcp.chat  SRV     0 5 5222 server.bp2019.cz.
_xmppconnect            TXT     "_xmpp-client-
xbosh=https://server.bp2019.cz:5280/bosh"
_xmppconnect            TXT     "_xmpp-client-
xbosh=https://server.bp2019.cz:5281/bosh"

server                  A       54.36.XXX.XXX
firewall                A       192.168.1.1
ap                      A       192.168.1.2
wifi                    CNAME   ap
...
```

Poslední zónový soubor se týká blokování nevyžádaného obsahu. Jelikož je tento soubor proměnný v čase, bude zde uveden pouze jeho výchozí stav při prvotním nastavení:

```
@ 3600 IN SOA @ admin.bp2019.cz. 0 86400 7200 2592000 86400
@ 3600 IN NS firewall.bp2019.cz.
```

4.4.6 Nastavení NTP serveru

NTP server slouží pro synchronizaci času v lokální síti a pro správnou činnost mnoha služeb je synchronizace času nezbytnou podmínkou. V prostředí CentOS 7 [26] je možné použít balík ntp nebo chrony. Synchronizace času bude dostupná pouze do prostředí vnitřní sítě. Pro správnou funkci služby je potřeba v konfiguračním souboru ***ntp.conf*** přidat na konec souboru následující nastavení:

```
restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap
restrict 192.168.2.0 mask 255.255.255.0 nomodify notrap
```

4.4.7 Nastavení SSH serveru

Připojení na firewall prostřednictvím SSH musí být velmi restriktivní a omezené na vybraný okruh osobu. To zahrnuje především následující omezení v rámci konfigurace SSH serveru:

- Přihlášení je možné pouze pomocí RSA klíče, který musí být nejméně 2048 bitů dlouhý.
- Uživateli admin je uložen veřejný pro vzdálenou administraci klíč do domovského adresáře.
- Není možné se přihlásit, jako uživatel root.
- Přihlásit se mohou pouze ti uživatelé, kteří patří do skupiny sshusers.
- Při nečinnosti proběhne automatické odhlášení za 120 sekund.

Pro správnou funkci služby je potřeba v konfiguračním souboru ***sshd_config*** přidat na konec souboru následující nastavení:

```
PermitRootLogin no
AllowGroups sshusers
LoginGraceTime 2m
```

4.4.8 Nastavení OpenVPN serveru

Instalace OpenVPN serveru [22] se skládá z několika částí, jelikož OpenVPN server je konfigurován jak v módu klient tak i v módu server. Pro oba módy je nezbytně nutné, aby byly správně nastaveny klíče a certifikáty. Instalace předpokládá, že klíče a certifikáty budou vygenerovány na lokální certifikační autoritě. Pro tento účely je použitý balík easyRSA[27]

Mód server slouží pro připojení uživatelů z prostředí Internetu do vnitřní sítě, typicky se tedy jedná o zaměstnance firmy nebo členy domácnosti. Připojení může být díky použité technologii realizováno prakticky z jakéhokoliv zařízení, tedy jak PC, tak i například mobilního telefonu. Scénář pro mobilní telefon má i tu výhodu, že díky filtrování nežádoucího obsahu dochází při prohlížení internetu k významné úspoře dat. Tento fakt je při současné ceně mobilních dat v ČR v rámci nekorporátních nebo retailových uživatelů velmi významný.

V rámci konfigurace jsou provedeny následující úkony:

- Otevření UDP portu 1194 pro komunikaci.
- Načtení certifikátu autority.
- Načtení privátního klíče OpenVPN serveru a jeho certifikátu.
- Spuštění OpenVPN serveru pod neprivilegovaným uživatelem.
- Nastavení odpovídajících vlastností a parametrů síťového přenosu.
- Nastavení bezpečnostní politiky pro spouštěné skripty.
- Vytvoření odpovídající směrovací tabulky po navázání spojení na klientovi.
- Nastavení odpovídajících DNS a NTP záznamů na klientovi.
- Nastavení viditelnosti klientů mezi sebou při více připojeních v rámci VPN.
- Nastavení komprese přenášených dat.
- Nastavení udržování spojení v rámci tunelu.

Nastavení zařízení pro připojení pomocí openVPN v módu serveru se realizuje následujícím konfigurací:

```
port 1194
proto udp
dev tun0
ca ca.crt
cert server.crt
key server.key
dh dh2048.pem
server 192.168.3.0 255.255.255.0
up ./dalvi.up
push "route 192.168.1.0 255.255.255.0"
push "route 192.168.2.0 255.255.255.0"
push "dhcp-option DNS 192.168.1.1"
push "dhcp-option DOMAIN bp2019.cz"
push "dhcp-option SEARCH bp2019.cz "
push "dhcp-option NTP 192.168.1.1"
client-to-client
keepalive 10 120
compress lz4
user nobody
group nobody
persist-key
persist-tun
script-security 2
txqueuelen 1000
```

V případě, že se firewall připojuje jako klient např. na server, kde běží pošta a další je konfigurace jiná a obsahuje následující úkony:

- Sestavení tunelu jako klient.
- Načtení privátního klíče OpenVPN serveru a jeho certifikátu.
- Spuštění OpenVPN serveru pod neprivilegovaným uživatelem.
- Nastavení odpovídajících vlastností a parametrů síťového přenosu.
- Nastavení komprese přenášených dat.
- Připojení na server přes UDP port 1194.

Nastavení zařízení pro připojení pomocí OpenVPN v módu serveru je následující:

```
client
dev tun1
proto udp
remote server.bp2019.cz 1194
resolv-retry infinite
nobind
persist-key
persist-tun
compress lz4
auth-nocache
verb 3
ns-cert-type server
ca /etc/openvpn/ca.crt
cert /etc/openvpn/test-s-dr-web10.crt
key /etc/openvpn/test-s-dr-web10.key
tun-mtu 1500
mssfix 1450
tun-mtu-extra 32
cipher AES-256-CBC
```

4.4.9 Nastavení firewalu

Nastavení firewallu se skládá ze dvou částí [8]. První část obsahuje obecně platná nastavení oproti běžný útokům a blokování nevalidních spojení. Druhá část pak obsahuje vlastní nastavení firewallu v závislosti na požadovaných službách.

V případě běžných útoků se nejčastěji jedná o pakety s nevalidními TCP flagy nebo fragmenty[10].

```
# Zahození fragmentů
$ iptables –i eth1 -A INPUT -f -j DROP
```

```
# Blokování paketů, které nemají validní TCP hlavičky
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags
FIN,SYN,RST,PSH,ACK,URG NONE -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags
FIN,SYN FIN,SYN -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags
SYN,RST SYN,RST -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags
FIN,RST FIN,RST -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags
FIN,ACK FIN -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags
ACK,URG URG -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags
ACK,FIN FIN -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags
ACK,PSH PSH -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags ALL
ALL -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags ALL
NONE -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags ALL
FIN,PSH,URG -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags ALL
SYN,FIN,PSH,URG -j DROP
$ iptables -t mangle -A PREROUTING -i eth1 -p tcp --tcp-flags ALL
SYN,RST,ACK,FIN,URG -j DROP
```

Další výčet je možné najít v příloze 2 práce.

Vlastní výkonná část firewallu funguje dle následující šablony:

- Pokud paket přichází z rozraní, které je považován za bezpečný, pak je paket puštěn dále. Na začátku předpokládáme, že bezpečná rozhraní jsou eth0 a wlan0, tedy rozhraní z vnitřní sítě.
- Dovolíme na venkovním rozraní pouze přístup na UDP port 1194 a o ten omezíme na 4 pokusy za minutu
- Takto nakonfigurovaný firewall spustíme jako službu iptablesa můžeme dál přidávat pravidla

Výchozí firewall lze vytvořit následujícími příkazy:

```
# Pakety co patří již existujícímu spojení se pustí dále
$ iptables -A INPUT -i eth1 -m state --state RELATED,ESTABLISHED -
j ACCEPT

# ICMP např. ping se neblokuje
$ iptables -A INPUT -p icmp -j ACCEPT
$ iptables -A INPUT -i lo -j ACCEPT
$ iptables -A INPUT -i eth0 -j ACCEPT
$ iptables -A INPUT -i wlan0 -j ACCEPT
$ iptables -A INPUT -i tun+ -j ACCEPT

# Povolí se OpenVPN zvenčí
$ iptables -A INPUT -i eth1 -p udp -m state --state
NEW,ESTABLISHED --dport 1194 -j ACCEPT
$ iptables -A OUTPUT -o eth1 -p udp -m state --state ESTABLISHED -
-sport 1194 -j ACCEPT

# A vše ostatní zvenčí se odmítne
$ iptables --A INPUT $ iptables -A INPUT -i eth1 -j REJECT --
reject-with icmp-host-prohibited

# Povolí se forward paketů z vnitřní sítě
$ iptables -A FORWARD -i eth0 -j ACCEPT
$ iptables -A FORWARD -o eth0 -j ACCEPT
$ iptables -A FORWARD -i wlan0 -j ACCEPT
$ iptables -A FORWARD -o wlan0 -j ACCEPT
$ iptables -A FORWARD -i tun+ -j ACCEPT
$ iptables -A FORWARD -o tun+ -j ACCEPT

# A vše ostatní na forward se odmítne
$ iptables -A FORWARD -j REJECT --reject-with icmp-host-prohibited
# Provede se NAT na rozhraní k ISP
$ iptables -A POSTROUTING -o eth1 -j SNAT --to-source 10.56.71.230
```

4.4.10 Blokování vybraných HTTPS spojení

Na úrovni TCP spojení je možné pracovat i s částmi dat, které nejsou šifrovaná. Protokol TLS, který se v současnosti používá pro HTTPS spojení během úvodní iniciace posílá nešifrovaně název serveru, se který se chce spojit, tzv. SNI. Díky modulu [28] pro netfilter je možné filtrovat vybrané pakety. V modelovém příkladu se zakáže komunikace s Facebookem.

```
# Zakáže komunikaci se serverem www.facebook.com
$ iptables -A OUTPUT -p tcp --dport 443 -m tls --tls-host
"www.facebook.com" -j DROP


# Zakáže komunikaci se servery obsahující doménu facebook.com
$ iptables -A OUTPUT -p tcp --dport 443 -m tls --tls-host
"\*.facebook.com" -j DROP
```

4.4.11 Konfigurace emailové relay

Jelikož Raspebarry PI slouží pouze jako emailové relay, tedy pouze zprostředkovává odesílání emailů, je konfigurace realizovaná prostřednictví nastavení v souboru ***/etc/postfix/main.cf***

```
# Nastavení relay
relayhost = [server.bp2019.cz]:587
```

4.4.12 Nastavení automatické aktualizace

Je předpokládáno, že firewall poběží bez zásahu obsluhy dlouhou dobu (řádově roky) a po celou tuto dobu musí být automaticky aktualizován z pohledu bezpečnosti. O to se stará služba yum-cron. Správný běh této služby je podmíněn nastavením konfiguračního souboru ***/etc/ /yum/yum-cron.conf*** a spustěním služby automatických aktualizací.

```
# Nastavení automatických aktualizací
update_cmd = security
download_updates = yes
apply_updates = yes
```

4.5 Automatizace konfigurace

V předcházejícím textu byla popsána prvotní konfigurace celého řešení tak, aby výsledkem byl funkční firewall na bázi Raspberry PI 3B+. Nyní bude proveden návrh zobecnění instalace tak, aby mohla být aplikována na základě konfigurace a jednotlivé kroky nebyly prováděny ručně, ale v dávce, která se řídí touto konfigurací

Konfigurační soubor má formát prostého textu a svou strukturou odpovídá standardnímu skriptu pro bash. Konfigurace je prováděna pomocí proměnných, tedy prostá editace tohoto konfiguračního souboru stačí pro plnou parametrizaci skriptu. Konfigurační skript je při spuštění instalačního skriptu zahrnut do provádění a tím je dosaženo efektu parametrizace.

4.5.1 Struktura automatické instalace

Automatická instalace se skládá ze tří částí:

- Instalačního skriptu, který obsahuje jednotlivé operace konfigurace, které jsou postupně prováděny a kontrolu výstupu těchto operací, pokud je jejich úspěšné provedení nutné pro další pokračování.
- Šablon konfigurací, které obsahují konfigurační soubory uzpůsobené pro parametrizaci. Šablony vychází z prvotní konfigurace.
- Konfiguračního souboru, který obsahuje nastavení, které se použije pro parametrizaci. Konfigurační soubor je zahrnut do instalačního skriptu po spuštění.

4.5.2 Konfigurační soubor instalace

Konfigurační soubor, který parametrizuje automatickou instalaci, má následující strukturu popsanou v tabulce (Tabulka 1). Hodnoty uvedené jako defaultní byly použité v modelovém případě, pro vytvoření konfigurace shodné s prvotní

konfigurací. Výstup automatické instalace je tedy shodný s výstupem původní manuální konfigurace.

Tabulka 1: Parametry instalace

Název parametru	Defaultní hodnota	Popis
LAN1	eth0	Označení (název) sítového rozhraní pro připojení do vnitřní sítě prostřednictvím kabelu
WLAN1	wlan0	Označení (název) sítového rozhraní pro bezdrátovou komunikaci, kde bude vytvořen přístupový bod (AP)
WAN	eth1	Označení (název) sítového rozhraní pro připojení do Internetu prostřednictvím kabelu
LAN1IP	192.168.1.1	IP adresa pevného připojení do LAN přes kabel
LAN1MASK	255.255.255.0	Síťová maska adresa pevného připojení do LAN přes kabel
WLAN1IP	192.168.2.1	IP adresa bezdrátového připojení do LAN přes kabel
WLAN1MASK	255.255.255.0	Síťová maska adresa bezdrátového připojení do LAN přes kabel
WANIP	10.45.28.235	IP adresa pevného připojení do Internetu
WANMASK	255.255.255.248	Síťová maska adresa

		pevného připojení do Internetu
WANGW	10.45.28.234	Defaultní brána pro připjení do Internetu
DNS1	1.1.1.1	IP adresa serveru, na který se bude lokální DNS server obracet pro vyřízení požadavků
DNS2	8.8.8.8	IP adresa serveru, na který se bude lokální DNS server obracet pro vyřízení požadavků
AP	BP2019	Název Access Pointu
VPNPORT	1194	UDP port pro připojení přes VPN
SMTPSERVER	server.bp2019.cz	Název SMTP serveru, přes který jsou odesílány emaily
IMAP4SERVER	server.bp2019.cz	Název IMAP4 serveru, přes který jsou přijímány emaily
XMPPSERVER	server.bp2019.cz	Název XMPP serveru pro odesílání a příjem zpráv v rámci chatu
SNI	1	Pokud je nastaveno na 1, pak se provádí blokace TLS spojení, která jsou uvedena v souboru sni.txt
HTTPSONLY	1	Pokud je nastaveno na 1, pak se blokuje veškerý nový odchozí provoz vyjme explicitně

		povoleného mimo port 443.

Zdroj: Vlastní zpracování

4.6 Ověření použitelnosti

V rámci metodiky je realizováno i závěrečné ověření použitelnosti firewallu postaveném na Raspberry PI 3B+. Vlastní řešení je nakonfigurováno v prostředí reálné sítě v malé firmě, za následujících podmínek:

- Připojení do Internetu bylo realizováno pomocí kabelu a konektivita udávaná ISP byla 60/20 Mbit s agregací 1:4
- Vnitřní bezdrátová síť byla realizována pomocí Wifi v pásmu 5GHz
- Vnitřní kabelová síť byla připojena do malého switche NETGEAR GS605

4.6.1 Provedená měření

V rámci testů byla provedena měření rychlosti přenosu jednotlivých rozhraní, přičemž bylo využito programu iperf. Na každém rozhraní je provedeno 10 měření rychlosti. Jako protistrana sloužil notebook Dell s GBit síťovou kartou a duální 2,4/5 GHz Wifi. Během měřená byly naměřeny níže uvedené výsledky.

Naměřené hodnoty rychlosti přenosu na rozhraní eth1 (připojeno do switche) jsou v tabulce (Tabulka 2).

Tabulka 2: Naměřené hodnoty na rozhraní eth1.

Číslo měření	Objem přenesených dat [MB]	Propustnost [MBit/s]
1	390	326
2	389	325
3	390	326
4	392	327
5	390	326
6	390	326
7	390	326
8	389	325
9	390	326

10	389	325
Průměr	389,9	**325,8 ± 0,6**

Zdroj: Vlastní zpracování

Naměřené hodnoty rychlosti přenosu na rozhraní eth0 (připojeno do switche) jsou v tabulce (Tabulka 3).

Tabulka 3: Naměřené hodnoty na rozhraní eth0.

Číslo měření	**Objem přenesených dat [MB]**	**Propustnost [MBit/s]**
1	194	162
2	192	160
3	198	165
4	197	165
5	207	173
6	199	166
7	203	170
8	194	162
9	198	165
10	202	169
Průměr	198,4	**165,7 ± 3,8**

Zdroj: Vlastní zpracování

Naměřené hodnoty rychlosti přenosu na rozhraní wlan0 2,4 GHz (připojeno jako přístupový bod) jsou v tabulce (Tabulka 4).

Tabulka 4: Naměřené hodnoty na rozhraní wlan0 / 2,4 GHz.

Číslo měření	**Objem přenesených dat [MB]**	**Propustnost [MBit/s]**
1	89	74
2	99	83
3	97	81
4	96	80
5	97	81
6	99	83

7	101	84
8	101	84
9	98	82
10	99	83
Průměr	97,6	**81,5 ± 2,8**

Zdroj: Vlastní zpracování

Naměřené hodnoty rychlosti přenosu na rozhraní wlan0 5 GHz (připojeno jako přístupový bod) jsou v tabulce (Tabulka 5).

Tabulka 5: Naměřené hodnoty na rozhraní wlan0 / 5 GHz.

Číslo měření	**Objem přenesených dat [MB]**	**Propustnost [MBit/s]**
1	98	82
2	99	83
3	98	82
4	98	82
5	98	82
6	99	83
7	101	84
8	101	84
9	98	82
10	99	83
Průměr	98,9	**82,7 ± 0,8**

Zdroj: Vlastní zpracování

Naměřené hodnoty rychlosti přenosu dat z Internetu pomocí služby https://www.speedtest.net na rozhraní wlan0 v pásmu 5 GHz (notebook připojen k Internetu přes AP na Raspberry Pi 3B+ jsou v tabulce (Tabulka 6).

Tabulka 6: Naměřené hodnoty rychlosti Internetového připojení.

Číslo měření	**Download**	**Upload**
1	49,2	19,4

2	51,3	19,2
3	50,9	19,3
4	52,8	19,4
5	53,1	19,2
6	49,9	19,4
7	52,8	19,4
8	52,4	19,2
9	50,8	19,1
10	51,3	19,2
Průměr	**51,45 ± 1,25**	**19,28 ± 0,11**

Zdroj: Vlastní zpracování

5 Výsledky a diskuse

5.1 Výsledky měření použitelnosti

Z naměřených výsledků v rámci testů použitelnosti bylo zjištěno, že přenosová kapacita je vyšší, než aktuální nejvyšší rychlost Internetu, kterou je možné ideálně u ISP dosáhnout. Z toho vyplývá, že dané řešení lze bez problémů provozovat i na rychlejších sítích. Pokud budeme uvažovat nejpomalejší variantu, tedy čistě bezdrátové připojení, daný firewall by byl vhodný pro rychlosti připojení Internetu do 80 MBit/s. Pokud budeme uvažovat i agregaci, lze takový firewall připojit i do rychlejších sítí.

Dále se nabízí varianta použití Raspberry PI jako firewallu čistě na kabelové technologii a wifi realizovat dodatečných zařízením. Zde jsou limity v rychlostech cca. 160 MBit/s

5.2 Vyhodnocení použitelnosti

Na základě údajů převzatých z [29] bylo provedeno srovnání s průměrnou rychlostí Internetu na pevné síti, která je dosahovaná na různých typech připojení. Pro srovnání uvažujeme, že se ve vnitřní síti dominantně používá Wifi připojení k firewallu v pásmu 5 GHz. Vychází z naměřené průměrné propustnosti Wifi 82,7 MBit/s. Výsledky srovnání jsou v tabulce (Tabulka 7)

Tabulka 7: Rezerva propustnosti Wifi AP pro různé typy připojení

Typ připojení	Průměrná rychlost Mbit/s	Rezerva Mbit/s
xDSL	19,16	63,54
Kabel	40,32	42,38
Optika	31,57	51,13
Wifi	17,22	65,48

Zdroj: Vlastní zpracování

6 Závěr

V rámci bakalářské práce byl realizován firewall na bázi počítače Raspberry PI 3B+, instalace byla převedena do opakovatelné formy a bylo provedeno měření propustnosti tohoto zařízení v rámci malé firmy a domácnosti.

Na základě výsledků práce lze konstatovat, že Raspberry PI může sloužit jako velmi dobrý firewall pro malou firmu nebo domácnost, který zároveň dokáže poskytnout velmi dobré připojení dovnitř a současně významným způsobem reguluje nežádoucí obsah, jako je například reklama, spam a další typy nevyžádaného obsahu.

V cenové relaci do 1 000,- včetně DPH je tak možné realizovat řešení, které bude provozuschopné několik let bez obsluhy.

Během zpracování práce narůstala komplexita celého řešení. Lze konstatovat, že pokud bude řešení dále vyvíjeno a rozšířeno o další moduly jako je monitoring, alerting, web GUI, REST API pro správu apod. mohl by vzniknout velmi zajímavý projekt. Ten by mohl být následně prezentován v rámci diplomové práce.

7 Seznam použitých zdrojů

[1] CentOS Wiki. *CentOS Product Specifications* [online]. [cit. 2019-03-10]. Dostupné z: https://wiki.centos.org/About/Product

[2] Rasperry PI Foundation. *Raspberry Pi 3 Model B+* [online]. (PDF). [cit. 2019-03-10]. Dostupné z: https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf

[3] Raspberry Pi Downloads - Software for the Raspberry Pi. *Raspberry Pi — Teach, Learn, and Make with Raspberry Pi* [online]. [cit. 2019-03-10]. Dostupné z: https://www.raspberrypi.org/downloads/

[4] Red Hat, Inc . *Red Hat and the CentOS Project Join Forces to Speed Open Source Innovation.* [online]. Copyright ©2019 Red Hat, Inc. [cit. 2019-03-10]. Dostupné z: https://www.redhat.com/en/about/press-releases/red-hat-and-centos-join-forces

[5] Ubuntu Wiki. *LTS - Ubuntu Wiki* [online]. [cit. 2019-03-10]. Dostupné z: https://wiki.ubuntu.com/LTS

[6] ZWICKY ,Elizabeth D., COOPER, Simon & CHAPMAN, D. Brent. *Building Internet Firewalls, Building Internet Firewalls Second Edition*. O'Reilly,. June 2000 890 s. ISBN: 1-56592-871-7

[7] DOSTÁLEK, Libor, KABELOVÁ, Alena. *Velký průvodce protokoly TCP/IP a systém DNS.* Computer Press 2008. ISBN 978-80-251-2236-5

[8] SUEHRING, Steve. *Linux Firewalls: Enhancing Security with nftables and Beyond (4th Edition)*. 2015 Addison-Wesley Professional. ISBN 0134000021

[9] GHEORGHE, Lucian. *Designing and Implementing Linux Firewalls and QoS using netfilter, iproute2, NAT and L7-filter.* 2006 Packt Publishing; ISBN 978-1-90481-165-7

[10] Red Hat Customer Portal. *8.0 Beta release notes.* [online]. Copyright © 2019 Red Hat, Inc. [cit. 2019-03-10]. Dostupné z: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8-beta/html-single/8.0_beta_release_notes/index

[11] W3Techs - extensive and reliable web technology surveys. *Usage Statistics of Default protocol https for Websites, March 2019* [online]. [cit. 2019-03-10]. Dostupné z: https://w3techs.com/technologies/details/ce-httpsdefault/all/all

[12] GoDaddy. *Google Chrome Not Secure warning.* [online]. Copyright © 1999 [cit. 2019-03-10]. Dostupné z: https://www.godaddy.com/garage/google-chrome-not-secure-warning-google-ups-ante-on-website-security/

[13] IETF, The Transport Layer Security (TLS) Protocol Version 1.2 [online]. [cit. 2019-03-10]. Dostupné z: https://datatracker.ietf.org/doc/rfc5246/

[14] Netfilter.org. *IPTABLES* [online]. [cit. 2019-03-10]. Dostupné z http://ipset.netfilter.org/iptables.man.html

[15] BHASKARJYOTI Roy,Mohamed Alibi; CentOS 7 Linux Server Cookbook; 2016 Packt Publishing; ISBN 1785887289

[16] Internet Systems Consortium . *ISC's open source DHCP software system* [online]. Copyright ©2019 Internet Systems Consortium, Inc. [. 2019-03-10]. Dostupné z: https://www.isc.org/downloads/dhcp/

[17] Internet Systems Consortium. *BIND 9 Open Source DNS Server* [online]. Copyright ©2019 Internet Systems Consortium, Inc. [. 2019-03-10]. Dostupné z: https://www.isc.org/downloads/bind/

[18] IETF, *DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION*. [online]. [cit. 2019-03-10]. Dostupné z: https://datatracker.ietf.org/doc/rfc1035/

[19] IETF. *Specification for DNS over Transport Layer Security (TLS).* [online]. [cit. 2019-03-10]. Dostupné z: https://datatracker.ietf.org/doc/rfc7858/

[20] IETF. *DNS Queries over HTTPS (DoH).* [online]. [cit. 2019-03-10]. Dostupné z: https://datatracker.ietf.org/doc/rfc7858/

[21] ntp.org: *Home of the Network Time Protocol.* [online]. [cit. 2019-03-10]. Dostupné z: http://www.ntp.org/

[22] OpenVPN. *VPN Software Solutions & Services For Business* [online]. Copyright © 2019 OpenVPN Inc. [cit. 2019-03-10]. Dostupné z: https://openvpn.net/

[23] KEIJSER, Jan Just. *OpenVPN Cookbook, 2nd Edition*. 2017 Packt Publishing. ISBN 1786463121

[24] CentOS Wiki. *SpecialInterestGroup/AltArch/armhfp* [online]. [cit. 2019-03-10]. Dostupné z: https://wiki.centos.org/SpecialInterestGroup/AltArch/armhfp

[25] BRESNAHAN, Christine, BLUM, Richard. *Linux Command Line and Shell Scripting Bible*. 2015 Wiley; ISBN 978-1118983843

[26] BORONCZYK, Timothy. *CentOS 7 Server Deployment Cookbook.* 2016 Packt Publishing. ISBN 1783288884

[27] GitHub. OpenVPN/easy-rsa: easy-rsa - Simple shell based CA utility. [online]. [cit. 2019-03-10]. Dostupné z: https://github.com/OpenVPN/easy-rsa

[28] GitHub.com. *Filter TLS traffic with IPtables* [online]. [cit. 2019-03-10]. Dostupné z: https://github.com/Lochnair/xt_tls

[29] DSL.cz. *Naměřené rychlosti internetu na DSL.cz v lednu 2019* [online]. Copyright © 2018 dsl.cz [cit. 2019-03-10]. Dostupné z: http://www.dsl.cz/clanky/namerene-rychlosti-internetu-na-dsl-cz-v-lednu-2019

8 Přílohy

8.1 Příloha č. 1 – skript pro nastavení síťových rozraní

```
#!/bin/sh

ZELENA=$(tput setaf 2)
STANDARDNI=$(tput sgr0)

# Nastavení parametrů IPv4 rozraní
function nastav_if()
{
   [ ! -f /usr/bin/nmcli ] && echo $ZELENA \
                           && echo "Network Manager musí být nainstalován,
provádím instalaci ..." \
                           && echo $STANDARDNI \
                           && sleep 2 \
                           && yum -y install NetworkManager
   ip=$1
   gw=$2
   dns=$3
   aktivif=`nmcli connection show --active | grep ethernet | cut -d' ' -f1`
   ifconf=/etc/sysconfig/network-scripts/ifcfg-$aktivif
   if [ -f $ifconf ]
   then
      txt=`cat $ifconf | grep BOOTPROTO`
      txt1=`cat $ifconf | grep IPADDR`
      txt2=`cat $ifconf | grep GATEWAY`
      txt3=`cat $ifconf | grep DNS`
      echo "Původní nastavení: nameserver IF $txt $txt1 $txt2 $txt3"
      nmcli con mod $aktivif ipv6.addresses "" ipv6.gateway "" ipv6.dns ""
ipv6.method auto
      nmcli con mod $aktivif ipv4.addresses $ip/27 ipv4.gateway $gw
ipv4.dns $dns ipv4.method manual
      txt=`cat $ifconf | grep BOOTPROTO`
      txt1=`cat $ifconf | grep IPADDR`
      txt2=`cat $ifconf | grep GATEWAY`
      txt3=`cat $ifconf | grep DNS`
      echo "After: nameserver IF $txt $txt1 $txt2 $txt3"
      systemctl restart network
   fi
}

# Kontola správnosti IPV4 adresy
function ipv4_test()
{
    local  ip=$1
    local  stat=1
    if [[ $ip =~ ^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$ ]]; then
       OIFS=$IFS
       IFS='.'
       ip=($ip)
       IFS=$OIFS
       [[ ${ip[0]} -le 255 && ${ip[1]} -le 255 \
           && ${ip[2]} -le 255 && ${ip[3]} -le 255 ]]
       stat=$?
    fi
    return $stat
```

```
}

# Nastaví název stroje
function nastav_hostname()
{
   hn=$1
   hostnamectl set-hostname $hn
   systemctl restart systemd-hostnamed
   hostnamectl status
}

if [ -z $1 ] || [ -z $2 ] || [ -z $3 ]
then
   echo ""
   echo "Použití skriptu je následující:"
   echo "$0 hostname ipv4.address ipv4.gateway"
   echo "$0 firewall.bp2019.cz 10.56.71.230 10.56.71.225"
   echo ""
   exit
fi


hn=$1
ip=$2
gw=$3
dns="1.1.1.1"
if [ ! -z "$4" ]
then
   dns=$4
fi
# check ip
ipv4_test $ip
if [[ $? -ne 0 ]]
then
   echo "$ip není platná IPv4 IP adresa"
   exit
fi
# check gw
ipv4_test $gw
if [[ $? -ne 0 ]]
then
   echo "$gw není platná IPv4 IP adresa"
   exit
fi
# check dns
ipv4_test $dns
if [[ $? -ne 0 ]]
then
   echo "$dns není platná IPv4 IP adresa"
   exit
fi


# Set hostname
nastav_hostname $hn

# Set IPv4 stuff
nastav_if $ip $gw $dns

exit 0
```

8.2 Příloha č. 2 – všeobecná pravidla firewallu

Ochrana proti zahlcení SYN pakety

```
$ iptables -N syn_flood
$ iptables -A INPUT –i eth1 -p tcp --syn -j syn_flood
$ iptables -A syn_flood –i eth1 -m limit --limit 1/s --limit-burst 4 -j
RETURN
$ iptables -A syn_flood –i eth1 -j DROP
$ iptables -A INPUT –i eth1 -p icmp -m limit --limit  1/s --limit-burst 1 -
j ACCEPT
$ iptables -A INPUT –i eth1 -p icmp -m limit --limit 1/s --limit-burst 1 -j
LOG --log-prefix PING-DROP:
$ iptables -A INPUT –i eth1 -p icmp -j DROP
$ iptables -A OUTPUT –i eth1 -p icmp -j ACCEPT
```

Omezení zahlcení SYN pakety prostřednitcím SYNPROXY

```
$ iptables -t raw -A PREROUTING –i eth1 -p tcp -m tcp --syn -j CT --notrack
$ iptables -A INPUT –i eth1 -p tcp -m tcp -m conntrack --ctstate
INVALID,UNTRACKED -j SYNPROXY --sack-perm --timestamp --wscale 7 --mss 1460
$ iptables -A INPUT –i eth1 -m conntrack --ctstate INVALID -j DROP
```

Blokování nových paketů, které nejsou SYN

```
$ iptables -t mangle -A PREROUTING –i eth1 -p tcp ! --syn -m conntrack --
ctstate NEW -j DROP
```

Zahození všech nulových paketů

```
$ iptables -A INPUT –i eth1 -p tcp --tcp-flags ALL NONE -j DROP
```

Blokování paketů s neplatnou MSS hodnotou

```
$ iptables -t mangle -A PREROUTING –i eth1 -p tcp -m conntrack --ctstate
NEW -m tcpmss ! --mss 536:65535 -j DROP
```

Zahození XMAS paketů

```
$ iptables -A INPUT –i eth1 -p tcp --tcp-flags ALL ALL -j DROP
```