Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií

Diplomová práce

Debian jako hraniční router

Bc. Marek Fučík

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Katedra informačních technologií

Provozně ekonomická fakulta

ZADÁNÍ DIPLOMOVÉ PRÁCE

Fučík Marek

Informatika

Název práce

Debian jako hraniční router

Anglický název

Debian as border router

Cíle práce

Cílem diplomové práce je instalace a praktické užití serveru s operačním systémem Debian. Vhodným užitím tohoto serveru je obsluha hraniční části počítačové sítě

Metodika

Metodika práce je založena na studiu odborných informačních zdrojů. Praktická část se věnuje oživení reálného serveru s operačním systémem Debian.

Harmonogram zpracování

1. Příprava a studium informačních zdrojů, upřesnění dílčích cílů - 6/2012-10/2012
2. Přehled řešené problematiky - 10/2012-12/2012
3. Zpracování praktické části - 12/2012-2/2013
4. Odevzdání práce a teze - 2/2013-3/2013

Rozsah textové části

60-80 stran

Klíčová slova

Debian, server, network, router, gateway

Doporučené zdroje informací

1. KOFLER, Michael. Linux12. Addison Wesley Verlag , 2012. German. ISBN 3827331471.
2. HUNT, Craig. Linux Síťové servery. SoftPress, 2006. 672 s. ISBN 80-86497-59-3.
3. WILLIAMS, Graham. GNU/Linux Desktop Survival Guide. eBook. Ang. Togaware.com, 2006. ISBN/ASIN 0975710915.
4. BASTA, Alfred - FINAMORE, Dustin, A. - BASTA, Nadine. Linux Operations and Administration. Course Technology, 2012. Angl. 848 s. ISBN 111103530X
5. BARRET, Daniel - SILVERMAN, Richard - BYRNES, Robert. SSH, The Secure Shell. O'Reilly Media, 2011. Angl. 668 s. ISBN 0596008953.
6. SOSINSKY, Barrie. Mistrovství – počítačové sítě. COMPUTER PRESS, 2010. 840 s. ISBN 9788025133637.
7. CASAD, Joe. Sams Teach Yourself TCP/IP in 24 Hours. U.S. Corporate and Goverment Sales, 2011. Angl. 515 s. ISBN 0672335719.

Vedoucí práce

Vasilenko Alexandr, Ing.

Termín odevzdání

březen 2013

doc. Ing. Zdeněk Havlíček, CSc.
Vedoucí katedry

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

prof. Ing. Jan Hron, DrSc., dr.h.c.
Děkan fakulty

V Praze dne 15.1.2013

Čestné prohlášení

Prohlašuji, že svou diplomovou práci "Debian jako hraniční router" jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 29.3.2013 _____

Poděkování

Rád bych touto cestou poděkoval Ing. Alexandru Vasilenkovi za konzultace a vstřícný přístup při návrhu a specifikaci podrobností mé práce. Dále bych chtěl poděkovat Ing. Martinovi Ottmárovi, administrátorovi sítě UNHfree.net o.s., za konzultace technických detailů řízení provozu paketů v počítačových sítích.

Debian jako hraniční router

Debian as a border router

Souhrn

Diplomová práce se v úvodu teoretické části zabývá charakteristikou linuxových operačních systémů a licenční politikou. V dalších kapitolách práce pozvolně přechází k charakteristice operačního systému Debian GNU/Linux. Tyto kapitoly pojednávají zejména o modularitě, struktuře a způsobech instalace tohoto operačního systému.

Teoretická část pokračuje popisem procesů správy, směrování a řízení počítačových sítí. Řešená problematika je uváděna do kontextu s operačním systémem Debian. Některé postupy jsou na příkladech konfigurace tohoto sytému demonstrovány.

Praktická část práce je zaměřena na stavbu, instalaci, konfiguraci a testování směrovače na modelové počítačové síti. Cílem je vytvořit cenově dostupný směrovač s pokročilými funkcemi řízení síťového provozu. Tento směrovač by měl být po drobných změnách konfigurace použitelný pro široké spektrum aplikací ve středně velkých podnikových sítích. Mezi přednosti tohoto směrovače by měly patřit nízká cena a spotřeba elektrické energie proti vysoké spolehlivosti.

Klíčová slova:

Debian
Směrovač
Směrování
Síťový provoz
Firewall
Shorewall
IP adresa
VPN
VLAN
QoS

Summary

At the beginning of the theoretical part, this diploma thesis deals with characteristics of Linux operating systems and licensing policy. In subsequent chapters, the work goes slowly to the characteristics of the operating system Debian GNU/Linux. These chapters deal mainly about modularity, a structure and ways of installation of this operating system.

The theoretical part continues with describing of the administration processes, routing and management of computer networks. The solved problematic is put into the context with the operating system Debian. Some procedures are demonstrated on examples of the configuration of this system.

The practical part is focused on the construction, installation, configuration and testing of routers on a network model. The aim is to create an affordable router with advanced network traffic management capabilities. The router should be after minor configuration changes applied to a wide range of applications in medium-sized enterprise networks. Among the advantages of this router should be low cost and power consumption to high reliability.

Keywords:

Debian
Router
Routing
Network traffic
Firewall
Shorewall
IP address
VPN
VLAN
QoS

1 ÚVOD

Celosvětová síť Internet je systém vzájemně propojených počítačových sítí. Tento systém se skládá ze všech účastníků jednotlivých sítí, serverů a prvků, které slouží pro datový přenos. Internet se stal hlavním komunikačním kanálem a zdrojem informací soudobé společnosti. Data a jednotlivé služby na něm provozované jsou rozmístěny po celém světě. Je tedy nasnadě, že musí existovat mechanizmy, které se starají o to, aby bylo možné navázat datovou komunikaci mezi stanicemi napříč Internetem. Obdobně jako zásilkové poštovní služby, které mají různá agregační místa zásilek a třídírny, tak i Internet a počítačové sítě disponují uzly, které obstarávají obdobnou funkci. Tyto uzly se nazývají směrovače. Jak už název napovídá, jejich úkolem je směrovat datový tok napříč Internetem. Starají se jak o komunikaci s koncovými stanicemi jednotlivých sítí, tak o komunikaci mezi sebou navzájem.

Směrovače se nestarají pouze o propojování jednotlivých sítí. Někdy jsou nazývány termínem gateway, tedy brána. Tato brána má za úkol chránit stanice za sebou před potenciálními hrozbami ze sítě Internet. Tak jak jsou schopny nasměrovat data ke svému cíli, jsou schopny i komunikaci zabránit.

Na trhu je k dispozici mnoho komerčních řešení směrovačů od malých domácích směrovačů v kombinaci Wi-Fi přístupovým bodem, až po výkonné profesionální řešení pro velké datové přenosy v řádech desítek Gb/s. Jako zástupce profesionálních produktů lze jmenovat směrovače společnosti CISCO a Mikrotik. Další možností je stavba instalace vlastního směrovače na míru. Touto volbou lze získat možnost definovat pravidla pro řízení síťového provozu, které u hotových univerzálních řešení nejsou k dispozici. Další výhodou je výběr ze široké škály volně dostupných operačních systémů a k nim dostupného programového vybavení.

Stavba a instalace směrovače s linuxovou distribucí Debian GNU/Linux a popis jednotlivých procesů směrování je náplní této práce.

2 CÍL PRÁCE A METODIKA

Cílem této práce je demonstrace alternativního využití distribuce Debianu pro specifické odvětví počítačových sítí. Jedná se o možnost využití Debianu pro směrování a zabezpečení počítačových sítí. Práce nemá za úkol detailně analyzovat jednotlivé mechanizmy směrovače, ani veškeré funkce Debianu, jelikož každý z nich by vydal na samostatné zpracování. Jejím úkolem je ukázat základní funkce směrovače a jejich vzájemnou kombinaci. V závěru práce je zpracován návrh a implementace funkčního modelu směrovače se zaměřením na hraniční směrovače podnikových sítí. V tomto modelu jsou v praxi otestovány nejběžnější nástroje řízení provozu a směrování počítačových sítí s využitím operačního systému Debian. Tímto je prokázána rozmanitost využití tohoto systému, kromě klasického serverového či desktopového řešení.

3 PŘEHLED ŘEŠENÉ PROBLEMATIKY

3.1 Linuxový server

Operační systémy na bázi Linuxu jsou vhodné k široké škále aplikací od malého serveru po komplexní řešení firemního informačního systému s vysokou dostupností. Přes 70 % serverů provozovaných v ČR je provozováno právě na Linuxu. Při výběru distribuce pro server je nutné se odrazit od požadavků - jaký server je potřeba postavit (jaký software má mít v repositářích), je-li preferována aktuálnost balíčků před stabilitou nebo naopak. Dalším faktorem výběru je zkušenost administrátora serveru s konkrétní distribucí. Nelze s určitostí říci, která distribuce je pro dané řešení nejvhodnější, každý administrátor preferuje jinou. Pro serverové užití je důležitý také faktor aktualizací a podpory ze strany vývojářů. Je zapotřebí, aby nebylo nutné celý server jednou za dva roky znovu instalovat, tedy aby podpora ve formě aktualizace a záplat byla k dispozici více let. Tato problematika se nazývá životní cyklus operačního systému.

3.2 Debian GNU/Linux

V prostředí Linuxu existuje mnoho distribucí, ve kterých se užívají stejné moduly a balíčky, popřípadě jejich kompilace. Mnoho distribucí přímo vychází z Debianu, takže se dá říci, že níže popsané informace platí pro jakýkoli server postavený na Linuxu. Debian je druhou nejstarší žijící distribucí vůbec (1. Slackware). Byl vytvořen v roce 1993 Ianem Murdockem a dnes je jednou z nejrozsáhlejších a nejuznávanějších distribucí.

Debian je ryze nekomerční a se svými uživateli uzavírá "společenskou smlouvu", ve které se zavazují mj. k tomu, že zůstane 100% svobodný, zaměřený na uživatele a otevřený (v tom smyslu, že před nimi nebude skrývat známé problémy). Stabilní verze Debianu k 1.3.2013 je 6.0.7. Debian je však více než jen samotný operační systém. Obsahuje přes 29 000 balíčků s (předkompilovanými) programy a dokumentací, připravených pro snadnou instalaci. Z Debianu vycházejí některé další distribuce, ať již nekomerční či komerční. Mezi tyto distribuce patří mj. Knoppix, Linspire, Ubuntu či Xandros.

Z hlediska nasazení na server vyniká Debian svou stabilitou a relativně dlouhým vývojovým cyklem, který představuje jistý kompromis mezi distribucemi s dlouhou

podporou jako CentOS nebo Ubuntu LTS, a distribucemi, které vycházejí každých 6-12 měsíců (Fedora, apod.).

3.3 Licenční politika

3.3.1 GNU

Projekt GNU je projekt zaměřený na svobodný software, inspirovaný operačními systémy UNIXového typu. Původní cíl byl vyvinout operační systém se svobodnou licencí, který však neobsahuje žádný kód původního UNIXu. Jeho jméno je rekurzivní zkratka pro GNU's Not Unix (česky „GNU Není Unix“).

3.3.2 GPL

GNU General Public License, GNU GPL (česky „všeobecná veřejná licence GNU“) je licence pro svobodný software, původně napsaná Richardem Stallmanem pro projekt GNU. Licence GPL vyžaduje, aby veškerá díla vycházejícího z díla původního pod touto licencí byla rovněž šířena s licencí GPL. Tato licence se snaží zajistit svobodné šíření software a zdrojových kódů pro možnost jejich využití dalšími vývojáři.

3.4 Struktura OS Linux

Operační systém se skládá z několika částí. Základem linuxových operačních systémů je jádro obstarávající spolupráci s hardwarem (paměť, procesor, síťové a zvukové karty, pevné disky, apod.). Poskytuje rozličné služby procesům. Rozděluje procesorový výkon. Většina programů volá funkce knihoven, které se poté předávají jádru k dalšímu zpracování. Některé programy potřebují přistupovat k jádru přímo pomocí jeho systémových volání.

3.4.1 Jádro

Jádro je středem operačního systému. Zajišťuje komunikaci s hardwarem, spravuje procesy, zajišťuje přístup k paměti. Obsahuje ovladače, které slouží ke komunikaci s hardwarem. Jádro rovněž obsahuje firewall netfilter, který obstarává síťovou komunikaci. V současné době je jádro modulární. To znamená, že dodatečné ovladače a funkce lze přidávat ve formě modulů bez nutnosti kompilace jádra za účelem konkrétního využití.

Tato skutečnost usnadnila využívání linuxových systémů široké veřejnosti, bez nutnosti znalosti vnitřních procesů jádra operačního systému

3.4.2 Knihovny

Velké množství kódu, který je často využíván různými aplikacemi, je uloženo v knihovnách. Knihovny obsahují nejčastěji zdrojové kódy pro matematické výpočty, operace grafického vykreslování a zobrazování. Knihovny se tak staly nedílnou součástí operačního systému.

3.4.3 Program vs. proces

Program je datový soubor uložený na disku počítače. Proces vzniká po spuštění toho programu. Po spuštění se program nahraje do paměti, kde dále funguje jako proces. Spuštěním programu může vzniknout více procesů. Rovněž vícenásobným spuštěním jednotlivých uživatelů systému vznikají další procesy. Pro vypsání všech procesů v paměti slouží příkaz **#ps –e**.

3.5 Instalace

Debian je distribuován svobodně přes Internet. Je možné jej nainstalovat několika způsoby: z CD (Comapct Disk), DVD (Digital Versatile Disc) a Bluray disků, přes síť, bootstrapped z jiné linuxové distribuce nebo ze systému MS Windows. Po dokončení instalace bude další aktualizace, upgrade a údržba prováděna pomocí integrovaných nástrojů pro správu balíčků. Je možné upgradovat velké softwarové komponenty nebo dokonce i přechod mezi verzemi Debianu bez přeinstalování systému.

3.5.1 Instalace z media

Celé instalační sady je možné stáhnout ve formátu ISO z http://www.debian.org/distrib a následně zapsat na CD či DVD či vhodně nakopírovat na flash disk. Další možností je koupě instalační sady u obchodníka. Při instalaci z media není vyžadováno připojení k Internetu, jelikož se celá instalace včetně nejdůležitějších a nejpoužívanějších balíčků nachází na mediu.

3.5.2 Síťová instalace

Pro mnoho administrátorů, kteří v době instalace mají přístup ke spolehlivému vysokorychlostnímu připojení k Internetu, je vhodné použít síťovou instalaci. K provedení síťové instalace je zapotřebí stáhnout a zapsat CD se základními bootovacími a inicializačními soubory o velikosti cca 40 MB. Následně dojde k připojení k repozitářům a administrátor má možnost si vybrat jednotlivé balíčky dle potřeby. Podmínkou je však správná inicializace ethernetové karty počítače.

3.5.3 Instalace z OS

Instalace je rovněž možná přímo z operačního systému, nemusí se jednat o shodnou distribuci. Instalaci lze spustit i z operačního systému Windows, stačí pouze stáhnout patřičný obraz disku a spustit instalaci.

3.5.4 Debootstrap

Tento nástroj umožňuje instalaci z jiného operačního systému do volných oddílů disku, aniž by došlo k pozastavení činnosti již nainstalovaného systému. Prostřednictvím sítě lze instalovat i z jiného počítače.

3.6 Přehled instalačního procesu

Na začátku instalace proběhne zkopírování zaváděče Grub či Lilo, následně dojde ke spuštění debian-installeru. Jedná se o instalační program, který rozpoznává hardware a nahrává správné ovladače, rozděluje disky, instaluje jádro systému a dohlíží na některé moduly. Mezi tyto moduly patří dhcp-client, debootstrap a taskshel. Debian-installer končí s prvním zavedením nového systému. Díky modulu tasksel je možné jednoduše doinstalovat celé skupiny programů jako „webový server“ nebo „desktopové prostředí“ a přizpůsobit tak systém konkrétním potřebám. Mezi nejčastěji užívaná desktopová prostředí patří KDE (K Desktop Environment), GNOME (GNU Object Model Envoronment) a Xfce (Xforum, common environment).

3.7 Adresářová struktura

Adresářová struktura v Linuxu začíná tzv. kořenovým adresářem, který se označuje symbolem „/“. Často se také používá anglický výraz root (stejně se označuje i superuživatel-administrátor). Nacházejí se zde hlavní systémové adresáře.

3.7.1 Připojování disků a oddílů

Diskové oddíly jsou připojovány jako určité adresáře, které jsou obvykle připojovány při startu systému a výměnná média za běhu nebo po spuštění. Informace o tom, kam se mají oddíly připojit, je uložena v souboru **/etc/vstav**. Například zápis **/dev/hda6 / ext3** znamená, že 6. oddíl primárního IDE disku (hda6) se připojí jako kořen „/“ na souborovém systému ext3. A v případě **/dev/hda8 /home xfs** bude 8. oddíl téhož disku připojen do adresáře **/home** se souborovým systémem XFS.

Diskové oddíly, na nichž jsou nainstalovány nelinuxové operační systémy, jsou obvykle připojovány do adresáře **/mnt** jako adresáře označené příslušným názvem. Pro disky s operačním systémem MS Windows se většinou zobrazují názvy windows, win_c, win_d. Do adresáře **/mnt** jsou připojována i výměnná média CD a DVD-ROM mechaniky a disketové jednotky. Tyto adresáře jsou nejčastěji pojmenovány například cdrom, dvd, floppy removable. V souboru fstab je rovněž uloženo, do jakého konkrétního adresáře se dané výměnné médium či oddíl jiného OS (Operační Systém) připojí.

Adresář **/home** je vyhrazen pro domovské adresáře uživatelů a bývá vhodné mu vyhradit samostatný oddíl na disku již při instalaci. Důvodem je možnost přeinstalovat operační systém bez zásahu do uživatelských dat. V adresáři **/home** jsou podadresáře shodné s přihlašovacími jmény uživatelů. Jeden uživatel obvykle nemá právo zápisu do adresářů jiných uživatelů.

3.7.2 Absolutní a relativní cesta

Absolutní cesta vždy začíná kořenem a končí názvem konkrétního souboru či adresáře. Kupříkladu **/home/Pavel/Dokumenty/text.txt** jednoznačně určuje soubor text.txt v adresáři Dokumenty uživatele Pavel.

Oproti absolutní cestě je používána ještě cesta relativní. To je poloha adresáře či souboru, oproti adresáři, ve kterém se uživatel v danou chvíli nachází. Důsledkem je, že určení cílového souboru není jednoznačné, ale záleží na aktuální poloze v adresářové struktuře. Relativní cestu lze od absolutní rozeznat tak, že relativní nezačíná symbolem „/“.

Pokud se uživatel nachází v adresáři **/home/Dokumenty**, pak se k souboru text.txt dostane pomocí absolutní cesty **/home/Dokumenty/texty/text.txt** nebo pomocí relativní cesty **../Dokumenty/clanek.txt** .

Některé speciální symboly usnadňují práci s cestami. Symbol „~“ nahrazuje cestu do domovského adresáře přihlášeného uživatele, takže místo absolutní cesty je možné napsat **~/Dokumenty/text.txt**.

V relativní cestě lze použít symbol „..“ pro označení nadřazeného adresáře a symbol „.“ pro označení aktuálního adresáře. Takže **./text.txt** označuje soubor text.txt v aktuálním adresáři a **../../text.txt** soubor text.txt o dva adresáře výše.

Skryté soubory a adresáře se v Linuxu vytvářejí tak, že je před název souboru nebo adresáře vepsána tečka. Pokud je tedy soubor text.txt přejmenován na .text.txt, stane se skrytým, a pokud je správce souborů nastaven tak, aby nezobrazoval skryté soubory, pak se tento soubor nebude zobrazovat. Stejné platí pro adresáře.

Na rozdíl od OS Windows se adresáře v Linuxu oddělují běžným lomítkem „/“, kdežto ve Windows je k tomuto účelu využíván znak zpětné lomítko „\“.

3.7.3 Další adresáře

„V linuxovém systému existuje množství adresářů, které vznikají bez přímého přičinění běžných uživatelů. Jde tedy o adresáře nacházející se jinde než v adresáři /home. Je mimo Seznam nejdůležitějších a nejzajímavějších adresářů:

***/bin** – Adresář obsahuje několik málo základních programů potřebných pro start systému, jež mohou být využity i běžnými uživateli (po startu).*

***/sbin** – Obdobně jako adresář /bin, ale zde umístěné programy nejsou určeny pro běžné uživatele, nýbrž administrátora (superuživatele root).*

***/boot** – Zde jsou uloženy soubory zavaděče a jádra potřebné při spouštění systému.*

***/dev** – Soubory zařízení; jde o speciální soubory, které umožňují uživatelskou komunikaci se zařízeními systému (např. připojené diskové oddíly, sériové porty, zvuk, obraz aj.).*

***/etc** – Konfigurační soubory globální, systémové; další konfigurační soubory jednotlivých aplikací naleznete v domovských adresářích uživatelů.*

***/home** – Domovské adresáře jednotlivých uživatelů (jsou pojmenovány obvykle podle přihlašovacích jmen uživatelů) obsahující uživatelská data a uživatelskou konfiguraci (ta je uložena podle aplikací ve skrytých souborech nebo skrytých adresářích).*

***/lib** – Sdílené knihovny vyžadované programy v kořenovém adresáři.*

/mnt, /media *– Adresář určený pro dočasně připojované systémy jako disketová jednotka, CD-ROM aj.*

/opt *– Instalují se sem některé nestandardní součásti systému, dodatečné aplikace, např. OpenOffice.org aj.*

/proc *– Poskytuje informace o systému (původně jen o procesech, odtud název); obsahuje pseudosouborový systém (dokonce je přiřazen ve fstab), který nereprezentuje strukturu dat na disku (nezabírá na disku žádné místo), ale jde o strukturu vytvořenou v paměti umožňující přístup k informacím o procesech a nastavení systému a jádra.*

/root *– Domovský adresář administrátora root; obvykle není přístupný ostatním uživatelům.*

/tmp *– Prostor vyhrazený běžícím programům pro ukládání dočasných souborů.*

/usr *– Tento adresář bývá velmi objemný, protože jsou do něj instalovány všechny aplikace; vnitřní struktura je z části podobná té z kořenového adresáře – najdeme zde bin,* ***sbin, etc, lib****; zvláštní význam mají adresáře /usr/share, kam jsou umisťovány aplikacemi sdílené soubory, a /usr/local, kde jsou instalovány aplikace „mimo“ distribuci (např. při kompilaci; opět adresáře bin, sbin, etc, lib aj.).*

/var *– Obsahuje data měněná za normálního běhu systému; jsou zde kupříkladu adresáře pro uchovávání logů, tiskové a poštovní fronty, dlouhodobější tmp adresář aj.“* [11]

3.8 Autentizace

Uživatel či administrátor se do systému přihlašuje pomocí uživatelského jména, kde uživatel root je superuživatel, a hesla. Po startu systému je k dispozici konzole vyzývající k zadání přihlašovacích údajů. Přihlášení probíhá zadáním uživatelského jména, potvrzením klávesou Enter, zadáním hesla a opětovným potvrzením. Pro přihlášení může být použit i certifikát. V případě přihlašování v desktopu má uživatel k dispozici grafické uživatelské rozhraní. Informace o uživatelích a hesla mohou být uchovávány dvěma způsoby. Stínová hesla jsou prostředkem k lepšímu zabezpečení systému. Systémy bez stínových hesel uchovávají uživatelská hesla v zašifrované podobě v souboru **/etc/passwd** přístupném všem uživatelům. Tento soubor musí zůstat čitelný, poněvadž obsahuje důležité informace o uživatelích, například jak se mají převádět uživatelská jména na odpovídající číselné hodnoty. Kdokoliv, kdo získá soubor **/etc/passwd**, se může pokusit útokem hrubou silou

(automatizované zkoušení všech možných kombinací) odhalit, jaká hesla mají uživatelé systému.

Pokud je povoleno použití stínových hesel, hesla se budou uchovávat v souboru **/etc/shadow**, který je přístupný pouze správci systému. Je doporučeno používat stínová hesla.

3.8.1 LDAP

Zatím byly popsány pouze způsoby přihlášení k systému jako takovému. Mnohem sofistikovanějším způsobem uchovávání informací o jednotlivých uživatelích systému a aplikací, je využití adresářových služeb. Jde o adresářovou službu (dále jen adresář), sloužící jako centrální úložiště pro společná data, která umí velmi rychle poskytovat. Data, o která se adresářová služba stará, jsou spíše neměnná, velmi rychle dohledatelná. Nepotřebují transakční zpracování. Od adresářové služby je požadována velmi rychlá odpověď a stabilita (jedná se o centrální systém).

V adresáři LDAP (Lightweight Directory Access Protocol) jsou nejčastěji uchovávána identifikační data uživatelů, konfigurační data aplikací a jmenné služby. Klienty adresáře jsou operační systémy, systémy pro správu přístupu k aplikacím, databáze, web server, mailserver a další.

Adresář je na rozdíl od relační databáze optimalizován pro velký počet požadavků na čtení (tisíce čtení na jednu zápisovou operaci). Relační databáze jsou většinou optimalizovány pro časté zápisy. Nejčastěji používaným protokolem pro komunikaci s adresářovým serverem je právě LDAP.

Popis LDAPu

LDAP je standardizovaný komunikační protokol adresářových služeb provozovaný na protokolu TCP/IP (Transmission Kontrol Protocol/Internet Protocol). LDAP data jsou uchovávána v záznamech, které jsou organizovány ve stromové struktuře obdobné právě adresářové struktuře souborového systému. Z toho vychází pojem adresářová služba. Záznam v LDAPu se skládá ze tří základních prvků:

Distinguished name - unikátní název záznamu

Atributy - popisují záznam

Object classes - určují typ záznamu

Distinguished name

Každý záznam ve stromové struktuře LDAPu má přiřazený unikátní název záznamu. Tento název je složen z atributů jejich hodnot. Běžně se využívají atributy dc (domainComponent), o (organization), ou (organizationalUnit), uid (userid) a cn (common name). Jejich využití není nutné nebo je lze naopak doplnit.

Atributy

Atributy popisují záznam, který je obsahuje. Tímto obsahem mohou být například jména, hesla, adresy, email. Součástí každého atributu je jeho definice. Definicí atributu se rozumí, zda je citlivý na velikost písmen, jakého je datového typu a kolikrát se může opakovat.

cn - plné jméno – Karel Novák
sn - příjmení - Novák
uid - unikátní identifikátor (použitelný například jako login to aplikací) - novakk
userpassword - heslo - nějaký hash hesla
email - emailová adresa - nejaky.email@example.net

Zjednodušeně řečeno, pokud je potřeba uchovávat základní informace o uživatelích a skupinách do kterých jsou zařazeni, je vhodné užít tento systém. Díky tomuto řešení není potřeba konfigurovat přístupové údaje jednotlivých uživatelů pro každou aplikaci zvlášť, jak již bylo zmíněno výše, jde o centrální úložiště těchto dat s rychlou odezvou.

3.9 Vzdálený přístup

3.9.1 SSH

Název Secure Shell (SSH) byl odvozen o obdobného programu rsh. SSH ve skutečnosti nepředstavuje příkazový terminál. SSH oproti rsh a Telnetu zajišťuje bezpečnou šifrovanou komunikaci

Prostřednictvím SSH je možné zajistit bezpečnou komunikaci mezi počítači. SSH umožňuje vytvořit síťový tunel. Tímto tunelem lze realizovat všeobecný přenos dat nebo

přístup příkazovému terminálu. Server komunikuje na portu TCP/22. Zajišťuje šifrovanou komunikaci.

Autentizaci uživatelů je možné realizovat prostým uživatelským jménem a heslem. Už tato komunikace je šifrovaná. SSH však disponuje robustnějším řešením zabezpečení prostřednictvím klíčů. Na serveru dojde k vygenerování privátního a veřejného klíče, přičemž pouze veřejný klíč je dále uchováván na serveru. Privátní klíč je na vhodném dostatečně bezpečném místě uložen u klienta. Při pokusu o přihlášení server veřejným klíčem zašifruje řetězec dat, která jsou následně zaslána klientovi. Klient tento řetězec dešifruje a odešle zpět na server. Autentizace je úspěšná, pokud dešifrovaný řetězec odpovídá původnímu nešifrovanému řetězci. Privátní klíč není možné zachytit, protože se neúčastní datového přenosu. Jediný způsob jeho získání je jeho fyzická krádež. Tímto způsobem vznikne šifrovaný tunel pro možnost přístupu k příkazovému terminálu nebo komunikaci dalších aplikací, které mohou být chráněny uživatelským heslem, čímž se zabezpečení dále zesiluje.

3.9.2 VNC

VNC (Virtual Network Computing) slouží k přístupu na plochu vzdáleného počítače. Prostřednictvím VNC lze tento vzdálený počítač ovládat, tak jako by připojený uživatel seděl přímo před jeho obrazovkou. Uživatel má možnost definovat rozlišení a barevnou hloubku, ve které bude obraz přenášen. Mimo obrazu lze touto technikou přenášet i zvuk. Tohoto principu lze využívat i k vytváření tenkých klientů, kdy aplikace a data jsou uloženy na hostitelském počítači, který je uživateli vzdáleně ovládán. Tohoto lze využít u aplikací, které nepracují v módu klient-server. Další možností uplatnění je společné užívání jedné licence drahých aplikací více uživateli. Této výhody je možné užívat pokud je aplikace svou licencí vázána k pracovní stanici.

Pro spojení je vhodné využít některou z metod šifrovaného tunelování, jelikož VNC, tak jak je navrženo, nepodporuje šifrování, čímž jsou uživatelská jména a hesla vystavena nebezpečí odposlechu.

3.9.3 VPN

Na rozdíl od firewallů, které mají za úkol udržet neoprávněné uživatele vně zabezpečené sítě, u VPN (Virtual Private Network) je naopak potřeba umožnit

oprávněným uživatelům přístup k interním prostředkům. Ve VPN je třeba zajistit, aby spolu komunikovaly skutečně oprávněné strany, jejichž totožnost je ověřená, a aby měly přístup k prostředkům, které jim mají být k dispozici. Při vlastní komunikaci po veřejné síťové infrastruktuře by se mohlo stát, že by privátní data mohl někdo neoprávněný odposlechnout nebo dokonce změnit. Proto je třeba zajistit jejich utajení a integritu.

VPN autentizace

Autentizací je možné ověřit totožnost klienta navazujícího komunikaci s bránou VPN, respektive serverem poskytujícím tuto službu, nebo totožnost dvou vzájemně komunikujících bran VPN. Klientem bývá většinou uživatel jednotlivec s jedinečným autentizačním přístupem, který se ze svého počítače snaží přihlásit do privátní sítě. V případě komunikace dvou VPN bran se nejčastěji jedná o vzájemné propojení podnikových sítí nebo poboček s centrálou. V těchto sítích uživatelé sdílejí společné heslo, které nemusejí znát, jelikož autorizaci zprostředkovává brána VPN.

Šifrování VPN

Zabezpečení pomocí šifrovaných tunelů slouží k zabezpečení autentizace, autorizace i síťového provozu uživatelů připojených k VPN od potenciální hrozby odposlechu komunikace. Pro šifrování je možné definovat různé hashovací algoritmy a metody. Například jednostranný hashovací algoritmu SHA (Secure Hash Algorithm), nebo algoritmy privátních a veřejných klíčů DES (Data Encryption Standard) a RSA (Rivest, Shamir and Adleman), které mohou být rovněž doplněny o zabezpečení prostřednictvím hesla.

Většina VPN používá metodu veřejných klíčů a privátních klíčů. Tato metoda spočívá ve vygenerování náhodného řetězce na straně brány VPN, který je zašifrován veřejným klíčem sdíleným pro všechny klienty. Zašifrovaný řetězec je odeslán klientovi, který jej svým privátním klíčem dešifruje a odešle zpět na stranu brány VPN. Pokud se dešifrovaný řetězec shoduje s původním řetězcem znaků je autorizace povolena. V rámci jednoho spojení lze provádět i změny v šifrování. Tato skutečnost znemožňuje potenciálnímu útočníkovi odposlech komunikace, jelikož nemá dostatek času pro dešifrování.

VPN na bázi IPSec

Řešení VPN komunikujících na síťové vrstvě internetového protokolu je nejčastěji založeno na bázi zabezpečení IPSec (Internet Protocol SECurity). IPSec přináší možnost tunelování nejen pro VPN, ale jakékoli spojení, které je tímto tunelem zapouzdřeno. V případě zahájení komunikace se spolu komunikující stanice nejprve musí dohodnout na bezpečnostní politice ve formě šifrování a autentizace. IPSec je velice bezpečnou metodou pro vytváření tunelového VPN spojení. K jeho aplikaci je potřeba specifický software, který nemusí být vždy kompatibilní napříč operačními systémy.

VPN na bázi SSL

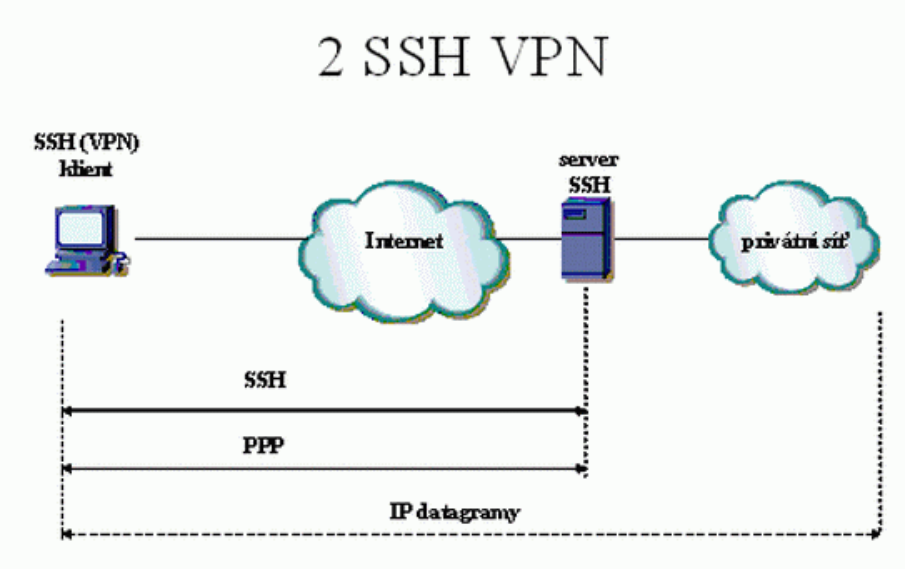
SSL (Secure Sockets Layer) zajišťuje bezpečnost komunikace aplikací typu klient-server prostřednictvím šifrovaného přenosu na aplikační vrstvě prostřednictvím protokolu TPC. Nevýhodou této metody v případě aplikace pro VPN je nižší bezpečnost v počáteční fázi přenosu. Přenos je nejdříve navázán pomocí asymetrického šifrování, kdy dojde k autentizace a vygenerování klíčů. Následně dojde k přechodu na symetrické šifrování na základě těchto klíčů.

Výhodou aplikace SSL je snadná implementace, jelikož většina aplikací, které předpokládají tento přenos disponuje šifrováním SSL. Jedná se zejména o webové prohlížeče a klienty elektronické pošty.

VPN na bázi SSH

SSH (Secure Shell) je metoda vzdáleného přístupu rozšířená zejména na linuxových operačních systémech. Pomocí SSH je možné vytvářet šifrované tunely, které jsou schopny zapouzdřit jiné protokoly. VPN na linuxových operačních systémem využívá rozhraní PPP (Point-to-Point Protocol). Odchozí provoz na tomto rozhraní je šifrován .

Obr. č. 1: SSH VPN



Zdroj: http://www.dsl.cz/clanek/515-bezpecnost-ve-vpn-ipsec-versus-ssl

3.10 Router

„Router (směrovač) je v počítačových sítích aktivní síťové zařízení, které procesem zvaným routování přeposílá pakety směrem k jejich cíli. Směrování probíhá na třetí vrstvě referenčního modelu ISO/OSI (síťová vrstva).

Charakteristika

Netechnicky řečeno, směrovač spojuje dvě sítě a přenáší mezi nimi data. Směrovač se podstatně liší od přepínače, který spojuje počítače v místní síti. Rozdílné funkce přepínačů a směrovačů si lze představit jako přepínače coby silnice spojující všechna města ve státě a směrovače coby hraniční přechody spojující různé země.

Obecně jako směrovač může sloužit jakýkoliv počítač s podporou síťování a pro směrování v menších sítích se často dodnes používají běžné osobní počítače, do vysokorychlostních sítí jsou však jako směrovače používány vysoce účelové počítače obvykle se speciálním hardwarem, optimalizovaným jak pro běžné přeposílání (forwarding) datagramů, tak pro specializované funkce jako šifrování u IPsec tunelů.

Jiné změny také zlepšují spolehlivost. Například používání stejnosměrného napájení (které se může v datových centrech odebírat z baterií) místo napájení přímo ze sítě, používání flash pamětí místo pevných disků. Velké moderní směrovače se tak podobají

spíše telefonním ústřednám, jejichž technologie k směrovačům (vzhledem ke stále častějšímu nasazování protokolu IP i ke spojování hovorů) konverguje a které směrovače případně nahradí, zatímco malé směrovače, kombinované například s kabelovými nebo DSL modemy, eventuálně WiFi přístupovými body, se stávají běžným vybavením domácností.

Směrovač se používá ke spojení alespoň dvou sítí. Speciálním případem je „jednoruký" směrovač, který používá jeden port a směruje pakety mezi virtuálními sítěmi VLAN provozovanými na tomto portu. V mobilních ad-hoc sítích si každý počítač směruje a forwarduje sám, zatímco v metalických a optických sítích je obvykle jen jeden směrovač pro celou broadcastovou doménu.

Směrovači, který připojuje klienty k vnější sítí (typicky Internetu), se říká „hraniční router" (edge router, někdy též „brána" – gateway, což je zastaralé označení pro směrovače obecně). Směrovač přenášející data mezi jinými směrovači se nazývá „vnitřní směrovač" (core router).

Směrovač používá routovací tabulku, která obsahuje nejlepší cesty k jistým cílům a routovací metriky spojené s těmito cestami

Směrovače se nyní implementují také jako „internetové brány", primárně pro malé sítě jako ty používané doma a v malých kancelářích. Používají se hlavě tam, kde je internetové připojení rychlé a „vždy připojené", jako kabelový modem nebo DSL. Tato zařízení ale nejsou v principu směrovače, protože počítače ve vnitřní síti efektivně skrývají pod svoji vlastní IP adresu ve vnější síti. Tato technika se nazývá NAT (network address translation, překlad adres)." [15]

3.11 IP adresy

IP (Internet protokol) adresa je číselné označení, pomocí něhož je možné určit síťové rozhraní v počítačové síti, která používá internetový protokol. V současné době je nejrozšířenější verze IPv4. Jednotliví výrobci zařízení již nabízejí nastupující verzi IPv6, na kterou se připravují i poskytovatelé. Rozlišujeme dva základní druhy IP adres dle použití. Jedná se o takzvané veřejné a privátní (vnitřní) IP adresy. Veřejné adresy si poskytovatel zprostředkovává buď od jiného ISP (Internet Service Provider) nebo se stane členem patřičné organizace a adresní prostor si nechá po zaplacení poplatků a

administrativní časti přidělit. Pro Evropu je organizací spravující adresní prostor RIPE NCC (Réseaux IP Européens Network Coordination Centre). Ta je podřízena celosvětovému koordinátorovi IANA (Internet Assigned Numbers Autority), u kterého si adresy alokuje.

3.11.1 IPv4

IPv4 je 32 bitovým zápisem čtyř 8 bitových čísel oddělených tečkami ve formátu xxx.xxx.xxx.xxx, zapisovaných v desítkové soustavě například 192.168.1.1. Pro orientaci v této adrese bylo zavedeno několik tříd dělení. Ve třídě A první číslo IP adresy označuje síť a zbylá tři čísla označují adresu hostitele. Ve třídě B jsou to první dvě pro síť a zbývající dvě pro hostitele. Síť třídy C používá první tři čísla pro označení sítě a poslední pro označení hostitele. Rozlišení na třídy A, B a C je stále značně hrubé. Je tedy zapotřebí rozdělit i tyto celky na menší, které byly nazvány podsítě. Toto rozdělení probíhá tak, že je možno umisťovat libovolně předěl mezi adresu sítě a lokální část adresy. Daná adresa se pak značí kombinací prefixu a délky ve formě 192.168.24.0/21, což znamená, že takto vytvořená síť je určena prvními 21 bity adresy (maska by byla 255.255.248.0), zbytek je adresa stanice (případně podsítě), takže tato síť používá rozsah adres 192.168.24.0–192.168.31.255.

Jelikož je již delší dobu patrné že rozsah IPv4 nebyl dimenzován na současný rozvoj Internetu, bylo nutné se zamyslet nad úsporou těchto adres. Jednu úsporu přináší takzvaný NAT, pro který byly organizací IANA vyčleněny tři rozsahy privátních adres. Jedná se o tyto rozsahy:

10.0.0.0 – 10.255.255.255 (10/8 prefix)

172.16.0.0 – 172.31.255.255 (172.16/12 prefix)

192.168.0.0 – 192.168.255.255 (192.168/16 prefix).

Tyto rozsahy adres jsou vyčleněny pro užití uvnitř soukromých sítí a nemělo by dojít k jejich propagování do Internetu. Naopak pokud správce sítě nastaví v soukromé vnitřní síti IP adresy mimo tyto rozsahy, muže se stát, že jím zvolený rozsah bude již přidělen jinému poskytovateli jako veřejná IP adresa. Pokud by se chtěl někdo z účastníků vnitřní sítě takto nesvědomitého správce připojit ke službě poskytované na této adrese v Internetu, nepodaří se mu to, jelikož ho směrovač vnitřní sítě nepřeklopí na bránu, ale

jeho komunikace se uskuteční pouze v rámci vnitřní sítě. Další cestou jak zpomalit vyčerpání IPv4, aby bylo více času na pro přípravu přechodu na IPv6, byla změna sazebníku cen udělených adres. U většiny komodit platí, že se vzrůstajícím odběrem kupujete další jednotku za nižší "zvýhodněnou" cenu. U IP adres je tomu naopak, čím více jich ISP poptává, tím více roste jejich cena.

3.11.2 IPv6

Úspěšným řešením problému nedostatku adres IPv4 by měla být nová verze protokolu, označovaná IPv6. Délka těchto adres je 128 bitů, jde o osm 8 bitových čísel zapisovaných v hexadecimální soustavě. Oddělení jednotlivých skupin čísel je realizováno ":". Příklad adresy: 2001:0718:1c01:0016:0214:22ff:fec9:0ca5, celý zápis je možné zkrátit vynecháním úvodních nul v každé skupině. Pokud jde o skupinu nul, je možné tyto vynechat všechny a zapsat pouze "::". Obhajobou toho, zda IPv6 poskytne dostatečný počet variant, je tvrzení o dostatku kombinací pro připojení každého zařízení k Internetu s jedinečnou IP adresou, které toto umožňuje. I tyto adresy podléhají dalšímu dělení, konkrétně do tří skupin:

Individuální (unicast) - identifikují jedno konkrétní síťové rozhraní.

Skupinové (multicast) – označují skupinu síťových rozhraní na způsob podsítí u Ipv4, jejímž účastníkům se mají data poslat. Tato data jsou rozdělena pro všechny účastníky této skupiny.

Výběrové (anycast) – označení skupiny síťových rozhraní. Data jsou zaslána pouze jednomu účastníkovi skupiny.

IPv6 neobsahuje všesměrové adresy jako IPv4 0.0.0.0. Používají se zde znaky jednotlivých skupin. Pokud je třeba zasílat požadavky všem uzlům, užije se speciální znak, který zahrne tyto uzly daného rozsahu. IPv6 zavádí také dosah, kdy je možné u adresy definovat dosah z celého Internetu nebo jen v rámci nějaké sítě. Adresy jsou udělovány náhodně. Není určeno, která část bude individuální či výběrová. Tato úloha rozhodnutí připadá uzlům sítě.

3.12 Přidělování IP adres

3.12.1 Statická alokace

V tomto případě konfigurace nejsou jednotlivé adresy IP přidělovány automaticky a správce dané sítě je ručně zapisuje do konfiguračních rozhraní jednotlivých zařízení. V případě koncových zařízení si je povětšinou mohou účastníci zapisovat i sami. Konfigurace probíhá prostřednictvím příkazového terminálu, editací konfiguračních souborů, nebo prostřednictvím GUI.

Konfigurace

V případě konfigurace síťového rozhraní u serveru a směrovačů se zpravidla užívá statické alokace adres. Konfigurace adres se ukládá do souboru **/etc/network/interfaces.** V tomto souboru jsou k dispozici konfigurace pro veškerá síťová fyzická i virtuální rozhraní. Při konfigurace je potřeba rovněž nastavit adresy pro DNS prostřednictvím terminálu nebo editací souboru **/etc/resolv.conf**. Příklad konfigurace přes terminál může vypadat následovně:

 # ip addr add 192.168.0.1/24 brd + dev eth0

 # ip route add default via 192.168.0.254

 # ip link set eth0 up

Příklad konfigurace **/etc/resolv.conf** :

```
search jmeno-domeny.cz;          # název domény
nameserver    10.98.231.66       # primární DNS
nameserver    10.98.0.243        # sekundární DNS
```

Tímto zápisem bylo docíleno aktivního rozhraní eth0 s IP adresou 192.168.0.1, maskou 255.255.255.0, branou 192.168.0.254 a DNS 10.98.231.66 a 10.98.0.243. Přidání dalších adresních rozsahů na jedno rozhraní, například pro komunikaci s dalšími servery, management okolních zařízení, či záložní konektivitu, se provádí stejným způsobem. Stačí zadat další adresy.

3.12.2 DHCP Server

Je systém pro dynamické přidělování vymezeného rozsahu adres jednotlivým účastníkům. Dynamické přidělování adres využívá většina poskytovatelů zejména k úspoře rozsahu adres, kterými disponuje. V reálu se totiž nestává, aby byli v jeden okamžik k síti připojeni všichni účastníci. Proto poskytovateli postačí méně adres, které jsou dynamicky přidělovány tomu, kdo o adresu požádá. Výhoda dynamického přidělování adres spočívá i v tom, že není zapotřebí odborná znalost účastníků sítě v protokolech IP, neboť většina zařízení má tuto volbu ve výchozím nastavení a není tedy třeba žádné, nebo jen minimální konfigurace.

DHCP (Dynamic Host Configuration Protocol) server zjednodušuje práci na síti tak, že sám automaticky přiděluje účastníkům parametry potřebné ke komunikaci. Mezi tyto parametry patří IP adresa, maska sítě, výchozí brána a mohou to být i adresy DNS. Princip spočívá v tom, že účastníci žádají DHCP server o adresu IP prostřednictvím UDP paketu na portu 67. Pokud jsou k dispozici volné adresy nebo není účastník, který vznáší dotaz na adresu IP, blokován, odpoví mu DHCP server na portu 68 paketem s nabídkou volných IP adres. Z těch si jednu vybere a odpoví serveru. Server tento výběr zpětně potvrdí a účastník může začít s provozem na síti. Přidělování IP adres je časově omezeno, proto musí účastník před vypršením tohoto limitu opět požádat o používání adresy nebo ji přestat užívat. Tento protokol je kompatibilní napříč všemi operačními systémy.

Instalace, konfigurace

Pokud nebyl balíček pro DHCP přidán v průběhu instalace systému, instalace probíhá příkazem **#apt-get install isc-dhcp-server**. Konfigurace DHCP serveru je v Debianu uložena v souboru #**/etc/dhcp/dhcpd.conf**.

```
option domain-name „DHCP server“;
option routers 192.168.0.254 192.168.0.253;
option domain-name-servers 192.168.0.1, 192.168.0.2;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.0.255;
default-lease-time 3600;
max-lease-time 7200;
```

```
subnet 192.168.0.0 netmask 255.255.255.0 {
        range 192.168.0.100 192.168.0.200;
}
host voip {
        hardware ethernet 00:ab:2f:dc:62:d3;
        fixed-address 192.168.0.100;
}
```

Při takovéto konfiguraci bude dhcpd přidělovat adresy v rozmezí 192.168.1.100 až 192.168.1.200 na jednu hodinu, ne však déle než na dvě hodiny. Klient voip je telefon, který potřebuje mít přidělovánu stálou adresu 192.168.1.100. Tato adresa je tedy rezervována a DHCP server zajistí, aby nebyla poskytnuta jiné stanici v síti. DHCP démon se nazývá dhcpd. Jeho spouštění se v Debianu provádí prostřednictvím příkazu **#/etc/init.d/dhcpd start**. Informace o přidělených adresách jsou po spuštění DHCP serveru ukládány do souboru **/var/lib/dhcp/dhcpd.leases**.

Deklarace

Rejstřík deklarace nejčastějších parametrů pro konfiguraci **/etc/dhcpd.conf** :

„group id; Tato deklarace umožňuje seskupit určité parametry a aplikovat je na vybranou skupinu hostitelů, může to ušetřit práci s jednotlivým vypisováním; id je v tomoto případě identifikace pro skupinu.

host id; Oproti group se volby v této deklaraci budou týkat jen počítače definovaného identifikátorem id.

subnet IP adresa netmask IP adresa; Definuje podsíť, které se následující volby týkají. Musí být následována maskou sítě. Celá syntaxe je popsána výše.

shared-network id; Umožňuje sdílet na jedné fyzické síti (rozhraní) různé podsítě.

range [dynamic-bootp] počáteční IP [koncová IP]; Tuto deklaraci jsem už snad dostatečně probral, jen si všimněte, že koncová IP je nepovinná.

pool; Slouží k rozdělení parametrů pro různé segmenty stejné sítě, v manuálu je uveden příklad jednoho poolu pro známé klienty a jednoho pro neznámé klienty.

authoritative; Je to povinná volba; a je dobré vědět, že server DHCP by měl být vždy autoritativní. Opakem této volby je not autoritative;.

max-lease-time sekundy; Maximální možná délka doby pronájmu adresy, bez ohledu na požadavky klienta v sekundách (ve výchozím nastavení server respektuje všechny klientské požadavky).

default-lease-time sekundy; Výchozí doba trvání pronájmu adresy.

filename "/cesta/k/souboru/"; Cesta k bootovacímu souboru pro bezdiskové stanice. Tato volba se často kombinuje s parametrem next-server, který klientovi řekne, ze kterého serveru má tyto zaváděcí údaje získat.

server-name "jméno/IP"; Definuje jméno serveru, na kterém se vzdáleně zavádí systém.

allow [denny] booting; Povoluje/zakazuje konfigurační požadavky klienta.

server-identifier "ip adresa;" Definuje IP adresu serveru posílaného klentům.

hardware [typ] hw adresa; Určuje typ linkové vrstvy (nejčastěji ethernet) a hardwarovou adresu (MAC adresu).

fixed-address adresa; Přiděluje adresu nebo skupinu adres klientovi.

option host-name "jméno"; Jméno pro klienta.

option domain-name "jméno"; Doménové jméno pro klienta.

option broadcast-address adresa; Určuje broadcast adresu.

option domain-name-servers jméno-serveru; Označuje DNS server (servery).

option routers adresa; Adresa routeru.

option lpr-servers adresa; Tato volba udává tiskový server.

option smtp-servers adresa; Tato zase poštovní server.

option subnet-mask maska; Maska hostitele.

ddns-update-style ad-hoc|inerim|none; Tento parametr musí být v konfiguračním souboru vždy přítomný. Říká, jestli je řešena synchronizace s DNS serverem.

always-reply-rfc1048; Komunikuje s klienty podle RFC1048, týká se klientů BootP.

use-host-decl-names příznak; Volba vhodná do direktivy host. Říká DHCP, aby posílal s TCP/IP údaji i hostitelský název počítače.“ [17]

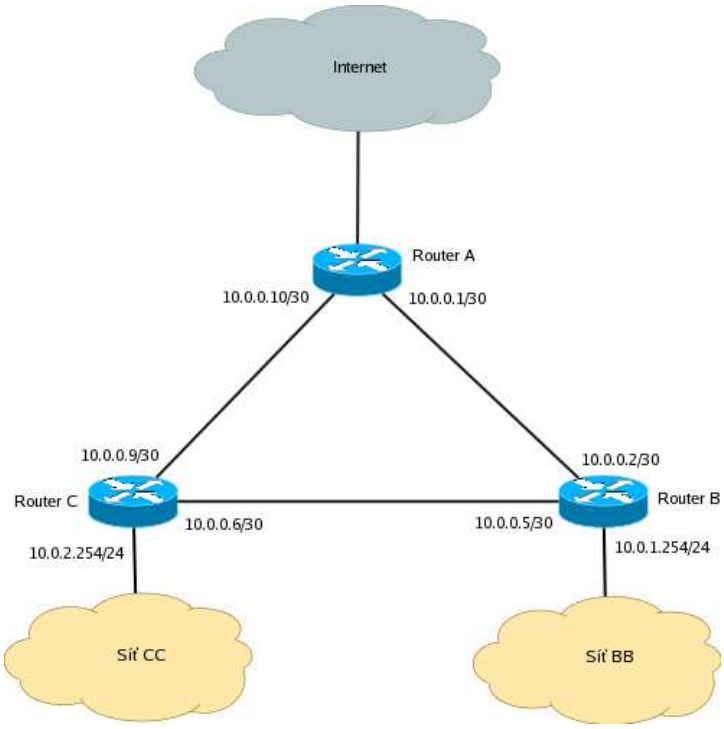
3.13 Dynamické routování

Při budování počítačových sítí osazených více servery či routery se administrátor většinou nevyhne nutnosti budování záložních tras k propojení jednotlivých serverů. Nemusí se jednat o servery rozmístěné po celém světě, postačí pouze jejich různé rozmístění v rámci kolokace. Další modelovou situací může být propojení dvou serverů jedné společnosti, který je každý v jiném sídle, přičemž tato společnost využívá záložní připojení k Internetu. Problémem není ani tak fyzické propojení těchto serverů, jako jejich konfigurace. Pro řešení tohoto problému užívá Linux routovací démon quagga a protokol OSPF.

3.13.1 Protokol OSPF

OSPF (Open Shortest Path First) je protokol používaný pro směřování uvnitř autonomního systému. Dynamické směrování prostřednictvím OSPF se používá v případech, kdy směrovač může komunikovat s ostatními směrovači v síti pomocí více tras. Nejkratší trasa, kterou směrovač volí pro komunikaci s okolními směrovači, se nazývá default route. Každý směrovač uvnitř autonomní sítě zná topologie celé této sítě. Existence a dostupnost okolních směrovačů je ověřována prostřednictvím hello paketů odesílaných v krátkých časových intervalech. Pokud směrovač neobdrží od sousedního směrovače odpověď, dojde k přepočtu nové nejkratší trasy. Každá trasa v síti je ohodnocena cenou cost. Právě z této ceny je prostřednictvím Dijkstrova algoritmu vypočítána nová trasa, která je předána démonu zebra. Zebra předá patřičné informace jádru operačního systému, které nastaví novou default route.

Obr. č. 2: Modelová situace autonomní sítě



Zdroj: http://www.abclinuxu.cz

Na obrázku č. 2 je zobrazena modelová situace autonomní sítě. V této síti je A hraničním směrovačem. Pokud se chtějí sítě BB a CC připojit prostřednictvím svých směrovačů B a C k Internetu, musejí být prostřednictvím OSPF propagovány default routes. Ty jsou voleny na základě nejnižší ceny jednotlivých tras. V případě, že budou mít všechny trasy stejnou hodnotu cost 100, bude oběma směrovačům propagován jako default route směrovač A. Jiná situace nastane, pokud by se hodnota cost mezi směrovači A a B změnila na 500 nebo by došlo k výpadku této trasy. V tom případě by se směrovači B propagoval jako default route směrovač C. Stejná pravidla platí i v opačném směru pro sítě BB a CC. Příklad konfigurace démona **/etc/zebra/zebra.conf**.

```
hostname router
password ******
enable password ******
log file /var/log/zebra.log
service advanced-vty
interface lo
```

```
interface eth0
interface eth1
interface eth2
```

V průběhu konfigurace démona **/etc/zebra/ospfd.conf**. Je potřeba nastavit zejména rozhraní na kterých má OSPF pracovat. Dále je potřeba nastavit hesla a další pokročilé parametry, mezi které patří například cost a časové intervaly hello paketů.

```
hostname router
password ******
enable password ******
log file /dev/null
service advanced-vty
interface lo
description system loopback
#interface eth0
interface eth1
description smer router A
ip ospf cost 100
ip ospf dead-interval 40
ip ospf hello-interval 10
interface eth2
description smer router C
ip ospf cost 100
ip ospf dead-interval 40
ip ospf hello-interval 10
router ospf
ospf router-id 10.10.10.1
redistribute connected metric-type 1
redistribute static metric-type 1
network 10.0.0.0/24 area 0
network 10.0.1.0/24 area 0
network 10.0.2.0/24 area 0
```

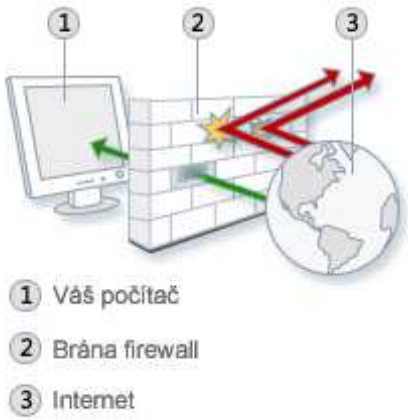
```
default-information originate always metric-type 1
```

Jak již bylo zmíněno, celý proces propagace funguje i v opačném směru. To znamená, že i směrovač A obdrží díky těmto nastavením default routes na sítě BB a CC. Navíc se default routes vypropagují i pro směrovače B a C do sítí CC a BB.

3.14 Firewall

Firewall je síťové zařízení, které slouží k řízení a zabezpečování síťového provozu mezi sítěmi s různou úrovní důvěryhodnosti a zabezpečení. Zjednodušeně se dá říci, že slouží jako kontrolní bod, který definuje pravidla pro komunikaci mezi sítěmi, které od sebe odděluje. V Linuxu je pod názvem Netfilter integrován v jádře. Jeho konfigurace je nejčastěji prováděna pomocí dalších programů. Zjednodušený pohled na firewall je vyobrazen na obrázku č. 3.

Obr. č. 3: Princip funkce firewallu



Zdroj: http://windows.microsoft.com/cs-cz/windows7/what-is-a-firewall

3.14.1 IPtables

IPtables nejrozšířenější Linuxový nástroj ke správě firewallu, který slouží k editaci tabulek Netfilteru. Umožňuje nastavení pravidel ovlivňujících průchod příchozích a odchozích paketů jádrem operačního systému a jejich směrování mezi rozhraními. Lze rozhodnout o tom, zda bude paket přijat, zamítnut, přesměrován nebo pozměněn. Jednotlivá pravidla se skládají do řetězců. V těchto řetězcích jsou pravidla aplikována postupně na základě průchodu paketu tímto řetězcem.

Tabulky

FILTER – filtrování paketů

NAT – překlad adres

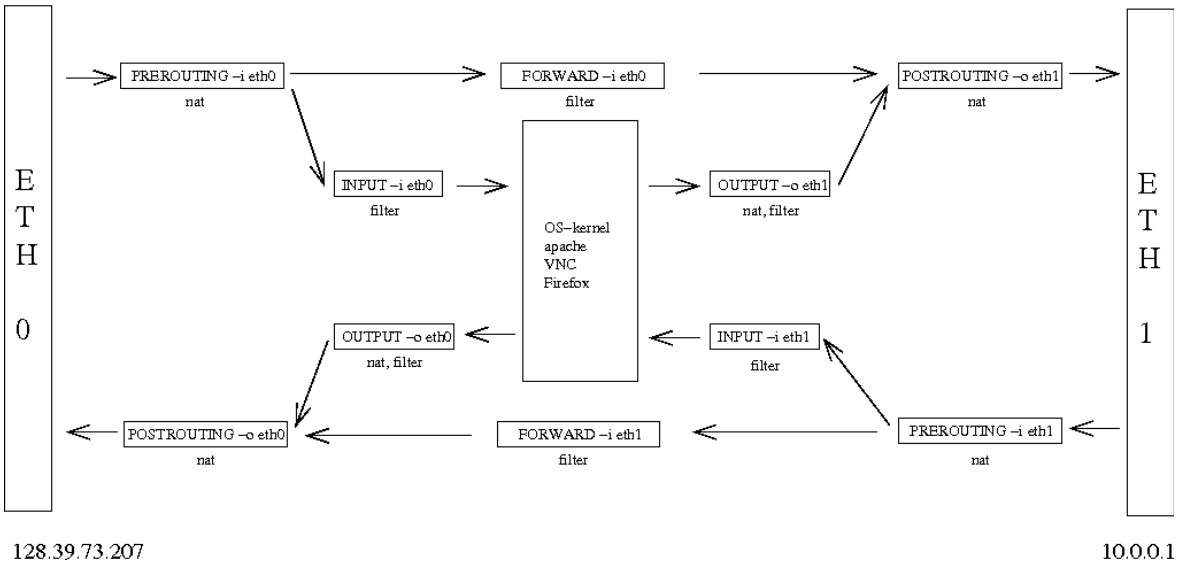
MANGLE – značkování paketů

RAW – odklonění od connection tracking

FILTER

Tato tabulka je nastavena jako výchozí. Změna výchozí tabulky je prováděna parametrem **–t**. FILTER obsahuje sadu pravidel INPUT, OUTPUT a FORWARD. V případě, že paket přichází dovnitř, jsou aplikována pravidla INPUT. Stejně tak jsou aplikována pravidla OUTPUT pro pakety směřující ven. Pokud stanice zastává funkci směrovače a paket je pouze tranzitní, aplikují se pravidla FORWARD. V tomto případě nejsou aplikována pravidla INPUT a OUTPUT. Schematické znázornění popisuje obrázek č. 3.

Obr. č. 4: Schematické znázornění IPtables



Zdroj:http://www.iu.hio.no/teaching/materials/MS004A/html/pictures/iptablesETH.gif

NAT

Tato tabulka obsahuje tři výchozí sady pravidel PREROUTING, POSTROUTING a OUTPUT, která slouží k úpravě cílové adresy příchozího paketu. Tuto tabulku je možné

využít v případě maskování vnitřní sítě za jednou adresou. Další možností je překlad veřejné adresy pro server umístěný v lokální síti nebo DMZ – takzvaný DNAT (Destination Network Address Translation). Všechna pravidla se dají aplikovat i v opačném směru jak je patrné na obrázku č. 2.

MANGLE

Tato tabulka obsahuje všechna pravidla uvedená v předchozích tabulkách, která doplňuje o možnost manipulace s hlavičkou paketu. Jedná se zejména a konfiguraci QoS a ToS.

Přidávání nebo modifikace pravidel je možné prostřednictvím příkazu **#iptables [tabulka] [akce] [chain] [ip_část] [match] [cíl] [cíl_info]**. Veškeré parametry, které lze použít v syntaxi tohoto příkazu, lze vypsat prostřednictvím příkazu **#iptables –help**. Zde je výtah nejpoužívanějších parametrů.

-A, --append – Přidání nového pravidla na konec řetězu.

-D, --delete – Smazání vybraného pravidla.

-R, --replace – Nahrazení pravidla.

-I, --insert – Vložení nového pravidla na začátek řetězu.

-L, --list – Výpis definovaných pravidel.

-F, --flush – Výmaz všech pravidel.

-N, --new-chain – Vytvoření nového řetěz.

-X, --delete-chain – Smazání uživatelem vytvořeného řetězu.

-P, --policy – Politika řetězu (DROP/zahoď, ACCEPT/přijmy).

-E, --rename-chain – Přejmenování uživatelem vytvořeného řetězu.

3.14.2 Shorewall

V Linuxu je rozšířeným a velice účinným nástrojem pro tuto funkci Shorewall. Jedná se o sadu skriptů usnadňující nastavení linuxového firewallu. Pokud budeme předpokládat užití Linuxu coby síťového serveru, budeme klást velký důraz i na jeho zabezpečení proti případným útokům. Právě Shorewall nám přináší možnost kontroly,

blokování a směrování síťového provozu zamýšleného serveru. Neslouží však pouze zabezpečení. Důležitým aspektem jeho nasazení je i takzvaný NAT, neboli síťový překlad adres. NAT umožňuje za jednu IP adresu připojit zařízení v jiném rozsahu sítě, čímž dojde k úspoře veřejných adres. Tato možnost nachází uplatnění zejména tam, kde potřebujeme více serverů připojit k jedné přidělené IP nebo je potřeba k danému serveru připojit další uživatele k Internetu. Oproti IPtables obsahuje sadu skriptů, které usnadňují spouštění jednotlivých pravidel.

Konfigurace Shorewall

Instalace Shorewallu se provádí příkazem **#apt-get install shorewall**. Po instalaci je připraven k použití, avšak ve velmi benevolentní podobě. Je proto nutné nastavit další parametry. Veškeré administrátorem přidané restrikce či povolení jsou nadřazeny výchozím.

Všechny soubory potřebné pro konfiguraci se nacházejí v adresáři **/etc/shorewall**. Konfigurace se provádí editací těchto souborů. Zde je výpis nejčastěji používaných:

/etc/shorewall/shorewall.conf – nejdůležitější parametr v tomto souboru je IP_FORWARDING, který zajišťuje předávání paketů.

/etc/shorewall/zones – slouží k přiřazení jednotlivých zón, které budou v Shorewallu použity.

/etc/shorewall/interfaces – slouží pro přidělení síťových zařízení k jednotlivým zónám.

/etc/shorewall/masq – nastavuje do jaké sítě se budou předávat příchozí pakety.

/etc/shorewall/policy – základní bezpečnostní politika.

/etc/shorewall/rules – nastavení jemnějších pravidel bezpečnostní politiky.

/etc/shorewall/tunnels – nastavení tunelů procházejících skrz shorewall.

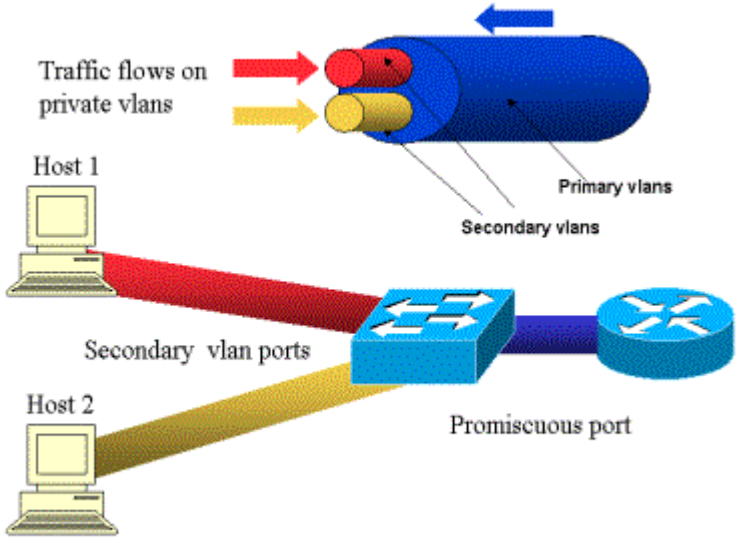
/etc/shorewall/tos – nastavení značkovacích priorit pro komunikační protokoly.

Platí obecné pravidlo, že to, co není výslovně zakázáno, je povoleno. Proto je důležité v nastavení politiky Shorewallu nejprve blokovat veškerý provoz z Internetu, a teprve poté povolit výjimky.

3.15 VLAN

Virtuální lokální sítě VLAN (Virtual Local Area Network) normy IEEE802.1Q umožňují vytváření vzájemně oddělených logických sítí na jedné fyzické síti, jak je zobrazeno na obrázku č. 4. Tato skutečnost šetří nároky na počet portů směrovačů a umožňuje sdílení síťové infrastruktury. Stanice připojené do různých VLAN se vzájemně nevidí, čímž je zajištěna jejich vzájemná bezpečnost. V případě směrovačů se nemusí jednat pouze o bezpečnost. Tímto způsobem lze sdílet komunikační trasy bez vzájemné kolize jednotlivých směrovačů. Těmto trasám mezi směrovači nebo přepínači se říká trunk.

Obr. č. 5: Schematické znázornění VLAN



Zdroj:http://www.cisco.com/en/US/products/hw/switches/ps700/products_tech_note09186a008013565f.shtml

Princip fungování spočívá ve značkování respektive tagování jednotlivých paketů. Paket opouštějící směrovač je označkován 32 bitovým doplňkem hlavičky. Při průchodu sítí je dle této značky směrován na patřičné porty přepínačů, kde dojde k jeho průchodu prostřednictvím trunku, nebo k jeho odtagování na patřičném portu. Po odtagování je s paketem možné běžně pracovat na vyšších vrstvách síťového modelu.

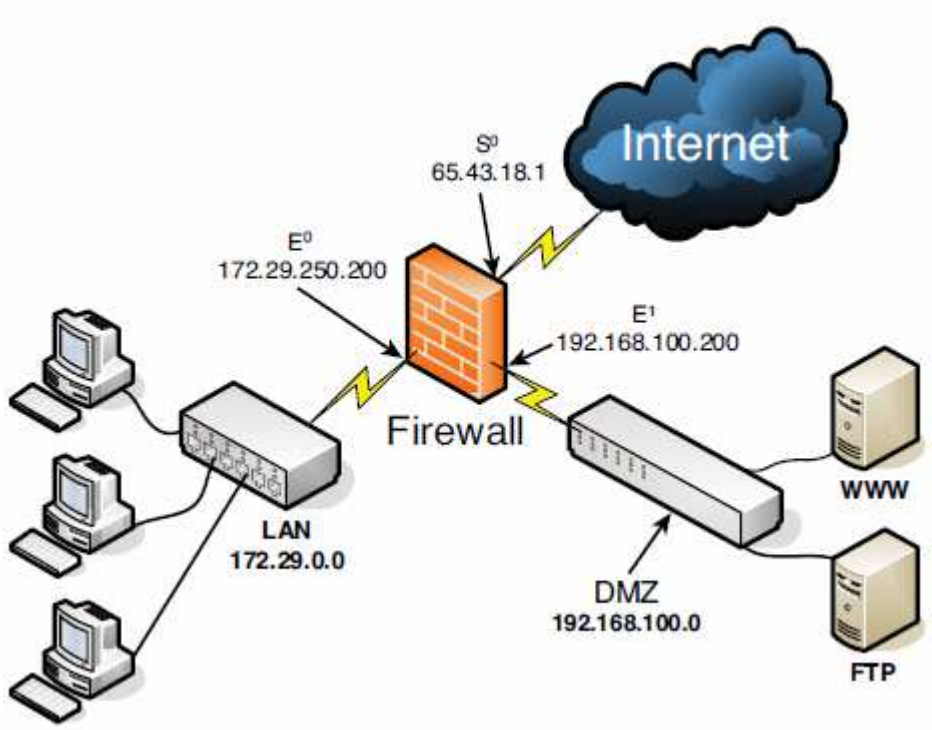
3.15.1 Konfigurace VLAN

V první řadě je nutné na fyzickém rozhraní daného směrovače vytvořit virtuální rozhraní, která budou komunikovat s jednotlivými virtuálními sítěmi. K tomu účelu slouží příkaz **#vconfig add [interface-name] [vlan-id]**. Pro vytvoření je nutné deklarovat parametry na jakém fyzickém rozhraní bude virtuální rozhraní vytvořeno, jeho název a ID. Vzhledem k tomu, že má doplněk hlavičky šířku 32 bitů, lze nastavovat i prioritu zpracování příchozích paketů na jednotlivých VLAN. Zbytek konfigurace je téměř totožný s konfigurací klasického síťového rozhraní prostřednictvím editace souboru **/etc/network/interfaces**.

3.16 Demilitarizovaná zóna

DMZ (Demilitarized Zone) je speciální druh počítačových sítí. Používají se v případech, kdy je potřeba zvýšit bezpečnost komunikace s Internetem. Klasickým případem aplikace DMZ je situace, kdy je potřeba oddělit komunikaci lokální sítě a Internetu. Stanice nebo servery umístěné v DMZ často mají veřejnou IP adresu. Z bezpečnostních důvodů je nežádoucí tyto počítače umisťovat do lokální sítě. Řešením je vytvoření DMZ, kam jsou tyto počítače umístěny viz. obrázek č. 6. Přístup ke službám provozovaných na těchto počítačích je následně možný jak z lokální sítě, tak ze sítě Internet, aniž by se zvyšovalo bezpečnostní riziko. V linuxových operačních systémech je aplikace DMZ řešena vytvořením zóny ve firewallu, která má nastavena specifická pravidla pro komunikaci s lokální sítí a Internetem.

Obr. č. 6: Schematické znázornění DMZ



Zdroj: http://thuansoldier.net/wp-content/uploads/2011/03/dmz.png

3.17 QoS

Kvalita služeb označována QoS (Duality of Service) se aplikuje za účelem zmírnění následků přetížené sítě. V případě, že jsou požadavky na datový přenos větší než je disponibilní kapacita sítě, bude nutné vyřešit, která data budou mít v provozu na síti přednost. Přenos standardně probíhá jako po sériové lince. Tedy data, která do linky vstoupí, ji opouštějí ve stejném pořadí a právě QoS je schopen podle druhu nebo označení paketů posoudit, které pakety budou mít přednost. V podstatě rozlišujeme několik základních protokolů:

UDP – neprovádí žádnou kontrolu chyb a neposkytuje informace o ztrátě paketů, z těchto vlastností vyplývá, že aplikace založené na tomto způsobu mohou být citlivé na ztrátu paketů.

TCP – tento protokol využívá algoritmů upozorňujících odesilatele na ztrátu paketů, ty pak mohou být znovu odeslány. Aplikace založené na tomto přenosu nebývají časově závislé a nejsou citlivé na ztrátu paketů

http – je založen na protokolu TCP, není proto časově závislý, případná ztráta paketů způsobí pomalejší načtení webové stránky.

FTP – je rovněž založen na protokolu TPC, není proto časově závislý, případná ztráta paketu způsobí zpomalení přenosu dat, jelikož je nutné čekat na zaslaní nového paketu.

SSH, telnet – založeny na protokolu TPC, pokud dojde ke ztrátě paketů, uživatel tuto skutečnost pozná zpomalením terminálového okna při psaní či výpisech.

VoIP – je založen zejména na protokolu UDP. Kvůli hlasovému proudu RTP (Real Time Protocol) při přenosu hlasového hovoru je velice důležité, aby pakety dorazily ve stejném pořadí. Pokud by se tak nestalo nebo by došlo ke ztrátě, účastník tyto výkyvy pozná tak, že dojde k výpadkům či zkreslení hlasu. Protokol UDP obecně není moc spolehlivý, jelikož nevykonává žádnou kontrolu, nicméně v případě hlasových hovorů či videohovorů je důležitá jeho rychlost.

3.17.1 Priority

V mnoha případech rozlišování podle druhu paketu nestačí, je proto zapotřebí přistoupit k dalšímu rozlišení paketů. Toto rozlišení lze provést pomocí označení paketů nebo jejich rámců. Jsou rozlišovány tři základní druhy označení – IP precedence, DiffServ (Differential Service) a CoS (Class of Service). IP precendece a CoS jsou schopny prostřednictvím 3 bitů rozlišovat osm stavů priorit. Hlavní rozdíl mezi těmito technikami označování paketů je ten, že zatímco CoS označuje rámce paketů, IP precedence označuje hlavičky paktů samotných. Ale ani těchto osm stavů v mnoha případech nepostačuje. Proto přišla na řadu technika DiffServ, která využívá 6 bitů. Hodnoty jejich kombinací jsou nazývány hodnotami DSCP (Differential Service Code Point). Stavy jednotlivých bitů těchto metod nalezneme v 8 bitovém poli ToS (Type of Service). V současné době se využívá všech metodik, případně jejich kombinací. Kombinací je možné docílit toho, aby na různých linkách stejné sítě probíhala priorizace jiných paketů.

3.18 Traffic shaping

Traffic shaping sleduje určitou mezní hodnotu přenosové kapacity. Pokud je tato kapacita překročena, nedojde k zahazování paketů, ale tyto pakety budou ukládány do paměti (bufferu) a při poklesu provozu budou odeslány. Ani traffic shaping není všemocný. Pokud dojde k přeplnění tohoto bufferu, pakety budou zahazovány. V tomto

případě hovoříme o přetížení sítě. Ve chvíli, kdy dojde k přetížení, začne značným způsobem stoupat latence a ztrátovost, což může vyústit až k naprostému zahlcení. Traffic shaping není vhodný pro pakety nesoucí hlasový hovor. Pokud by byly tyto pakety uloženy a následně odeslány či zahozeny, došlo by k přeskakování či výpadkům hovoru.

3.19 CompactFlash

CF (CompactFlash) paměť je záznamové zařízení typu NAND (NOT AND) ve standardizovaném pouzdře. Počet zápisů na toto médium je omezen přibližně na milion. Podporovaná kapacita činí 128 GB. Rychlost zápisu se pohybuje okolo 20 MB/s. Pro připojení využívá 50ti pinový konektor. Díky své ceně a vysoké spolehlivosti se dnes uplatňuje v široké škále aplikací. Vysoká spolehlivost tohoto média je dána absencí mechanický dílů. Výhodou oproti obdobným záznamovým médiím pro některé aplikace je podpora rozhraní IDE a ATA.

4 VLASTNÍ ŘEŠENÍ

Analytická část této práce se zabývá stavbou, instalací a konfigurací skutečného směrovače použitelného pro zabezpečení a směrování menší podnikové sítě obsahující několik desítek uživatelských stanic. Konfigurace jednotlivých částí systému jsou demonstrovány na konkrétní modelové situaci popsané v metodice práce. Výslednou konfiguraci lze dále modifikovat dle požadavků různých počítačových sítí.

4.1 Metodika

4.1.1 Vytyčené cíle

Cílem práce bylo vytvoření modulárního směrovače s pokročilými funkcemi směrování a zabezpečení počítačové sítě. Mezi hlavní parametry patří nízká pořizovací cena, vysoká spolehlivost a nízká spotřeba elektrické energie.

Modelová situace sestává ze směrovače se dvěma ethernetovými porty. Jeden z těchto portů je připojen k poskytovateli připojení k Internetu. Druhý port slouží k připojení vnitřní sítě. Vnitřní síť je rozdělena do tří virtuálních sítí, aby byla zajištěna bezpečnost vzájemným oddělením jednotlivých sítí. VLAN1 sloužit pro lokální síť, VLAN2 pro veřejnost a VLAN3 je určena pro demilitarizovanou zónu. Jednotlivé virtuální sítě jsou tagovány na patřičné porty přepínače propojeného s vytvořeným směrovačem.

Adresy pro stanice připojené ke směrovači jsou přidělovány z rozděleného rozsahu adres třídy C prostřednictvím DHCP serveru.

Správa směrovače je možná pouze prostřednictvím terminálového rozhraní. Přístup je možné navázat prostřednictvím SSH spojení. Vzdálený přístup do sítě je realizován prostřednictvím virtuální privátní sítě.

Na síti je předpokládán provoz VoIP telefonie. Směrovač proto musí disponovat nástroji pro zajištění spolehlivého přenosu hlasu v reálném čase.

4.1.2 Postup dosažení cílů

Vytyčeného cíle bylo dosaženo výběrem vhodného HW, operačního systému. Nezanedbatelnou součástí je výběr a konfigurace vhodných modulů umožňujících provoz síťových služeb.

4.2 Příprava HW a OS

4.2.1 Hardware

Při výběru HW, respektive počítače vhodného pro instalaci směrovače, byl kladen důraz zejména na spolehlivost, nízkou energetickou náročnost a hlučnost. Předpokladem aplikace tohoto směrovače je menší podniková síť. Pro dlouhodobě pozitivní zkušenosti autora práce byl vybrán mikropočítač ALIX.2D2 se spotřebou nepřesahující 4W. Pro instalaci operačního systému byla použita CF (CompactFlash) paměťová karta o kapacitě 4 GB od společnosti SanDisk. Na tuto kartu nejsou kladena žádná zvláštní specifika. Slouží pouze pro uložení operačního systému a jeho následné načítání z této karty. Následující běh systému je realizován pouze v rámci operační paměti počítače.

Specifikace ALIX.2D2:

• Procesor: 500 MHz AMD Geode LX800
• Paměť: 256 MB DDR DRAM
• Připojení paměti: slot CompactFlash
• Napájení: DC jack nebo pasivní POE, 7 V – 20 V
• Signalizace: 3x LED
• Rozšíření: 2 miniPCI slots, LPC bus
• Konektivita: 2 x Ethernet port (Via VT6105M 10/100)
• I/O: DB9 serial port, dual USB port
• Rozměry: 6 x 6 (152.4 x 152.4 mm)
• Firmware: tinyBIOS

Dalšími prvky použitými pro instalaci a testování modelové situace byly osobní počítač, dva notebooky, USB (Universal Seriál Bus) čtečka paměťových karet, USB flash disk, USB to RS232 konvertor, propojovací ethernetové kabely a inteligentní UBNT Tought Switch.

4.2.2 Operační systém

Instalovaným operačním systémem se stal Debian GNU/Linux. Tento systém byl vybrán zejména díky jeho značné rozšířenosti, což přináší možnost snadného hledání informací potřebných pro jeho konfiguraci. Dalším důležitým aspektem je dvouletý vývojový cyklus, který zajišťuje stabilitu systému tím, že nejsou vydávány nové

aktualizace, ale pouze bezpečnostní záplaty. V neposlední řádě je jeho nespornou výhodou dostupnost široké škály modulů pro různé aplikace tohoto systému. Platforma ALIX.2D2, respektive procesor AMD Geode LX800, pracuje na architektuře x86. Pro instalaci byl tedy zvolen balíček debian-6.0.7.i386-netinst.iso o velikosti pouhých 180 MB získaný z webu http://www.debian.org. Jak už název balíčku napovídá, jedná se o síťovou instalaci, která umožní pouze instalaci jádra a nejnutnějších modulů systému.

4.3 Instalace

4.3.1 Způsoby instalace

Instalace operačního systému na platformu ALIX je odlišná oproti klasické instalaci na počítač nebo server. Důvodem je absence některých komponent a funkcí. Jedná se o absenci VGA (Video Grapihics Array) výstupu pro zobrazovací zařízení a nepodporovanou funkci zavádění instalace z médií připojených prostřednictvím USB rozhraní.

Postup instalace byl testován dvěma způsoby. První variantu lze nazývat programovou. Tento postup spočívá v instalaci programu DAEMON Tools, který umožňuje emulaci ISO obrazů do podoby virtuální mechaniky CD-ROM. Dalším instalovaným programem byl VMware, který umožňuje emulaci virtuálního počítače. Prostřednictvím VMware byl vytvořen virtuální stroj, do kterého byla zavedena instalace z virtuální mechaniky obsahující instalační obraz. Jakožto cíl instalace byla nastavena CompactFlash paměťová karta připojená prostřednictvím čtečky paměťových karet.

Druhý způsob instalace spočíval ve využití osobního počítače, čtečky paměťových karet a USB flash disku. Prostřednictvím programu LinuxLive USB creator byl z instalačního obrazu vytvořen spustitelný USB flash disk. Z tohoto disku byla po zapnutí počítače spuštěna instalace do CF paměťové karty připojené prostřednictvím čtečky paměťových karet. Před spuštěním počítače bylo nutné v BIOSu základní desky nastavit spouštění z výměnných médií.

Jako mnohem rychlejší a spolehlivější instalace operačního systému se jeví druhý způsob z výše zmíněných. Hlavním důvodem je zejména několikanásobně vyšší rychlost oproti emulaci a vyšší stabilita. Během instalace na emulovaném stroji došlo k selhání

emulovaného stroje a pádu celé instalace, což se negativně projevuje degradací životnosti CF paměťové karty.

4.3.2 Instalační proces

Po spuštění instalace se nejprve zobrazí dialogové okno s výběrem způsobu instalace. Zde se nabízí možnost textového nebo grafického postupu. Na výběru nezáleží, jelikož oba způsoby nabízejí totožné kroky a volby instalace. Několik následujících kroků se zabývá volbou jazykové mutace, lokalizace a rozvržením klávesnice. Zde lze doporučit volbu US QWERTY rozvržení, kvůli snadnějšímu psaní některých znaků během konfigurace. V dalším kroku dochází k detekci síťového rozhraní pro komunikaci se serverem obsahujícím instalační balíčky.

Během instalace dochází automaticky k vytvoření účtu superuživatele root. Uživatel je pouze vyzván k nastavení hesla a vytvoření běžného uživatelského účtu.

Důležitým krokem je rozvržení disku a volba souborového systému. Na disku je potřeba vytvořit pouze jedinou primární jednotku, která bude obsahovat kořenový adresář „**/“**. Při výběru souborového systému byl paradoxně vybrán zastaralý ext2. Důvodem tohoto výběru je absence odkládacího prostoru právě probíhajících transakcí pro případ pádu systému. Tento odkládací prostor by způsoboval degradaci CF paměťové karty.

V dalších krocích probíhá instalace jádra, modulů, výběr repozitářů až po krok s hrubým výběrem několika modulů, které předurčují budoucí zaměření systému. Z uvedené nabídky byl pro tento případ vybrán pouze SSH server. Grafické prostředí se na severy a směrovače zpravidla neinstaluje z důvodu bezpečnostního rizika. V tomto případě je možné grafické prostředí opomenout i z důvodu absence grafického adaptéru na platformě ALIX.2D2.

Po úspěšné instalaci bylo nutné systém spustit v osobním počítači a provést konfiguraci výstupního zobrazovače pro sériové rozhraní RS232. Jednalo se o nastavení výstupu zobrazovače na sériové rozhraní. Po přihlášení do systému pod účtem superuživatele root se nastavení provádí editací konfiguračního souboru zaváděče systému **/etc/default/grub**. Níže uvedená část konfigurace mimo jiné zkracuje dobu čekání volby zaváděče GRUB na 3 s, nastavuje přenosovou rychlost na 57 600 bitů a výstup terminálu je směrován na rozhraní tty0 a ttyS0, což je právě sériová linka.

```
GRUB_DEFAULT=0
GRUB_HIDDEN_TIMEOUT=0
GRUB_HIDDEN_TIMEOUT_QUIET=true
GRUB_TIMEOUT="3"
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="console=tty0 console=ttyS0,57600n8
reboot=bios"
GRUB_CMDLINE_LINUX=""
GRUB_TERMINAL=serial
GRUB_SERIAL_COMMAND="serial --speed=57600 --unit=0 --word=8 --parity=no --
stop=1"
```

Dále je potřeba u patřičného řádku v souboru **/etc/inittab** zrušit komentování a nastavit přenosovou rychlost. Touto editací je umožněna inicializace sériového rozhraní při startu systému.

```
T0:23:respawn:/sbin/getty -L ttyS0 57600 vt100GRUB_HIDDEN_TIMEOUT=0
```

Po těchto úpravách již byla CF paměťová karta připravena pro první spuštění v platformě ALIX. V tuto chvíli ještě nebylo nakonfigurováno žádné síťové rozhraní. Bylo tedy nutné systém spustit a ovládat terminálem prostřednictvím sériové linky přes program puTTy, nebo Hyperterminál.

4.4 Konfigurace síťových rozhraní

Po přihlášení do systému prostřednictvím sériové linky RS232 je možné vypsat aktuální konfiguraci síťových rozhraní prostřednictvím příkazu **#ifconfig** . V tomto případě se na rozhraní eth0 zobrazila nefunkční konfigurace klientského nastavení dynamického přidělování adres. Rozhraní bylo nefunkční, jelikož byla konfigurace přiřazena k fyzické adrese rozhraní počítače, na kterém probíhala instalace. Bylo nutné odstranit soubor s přiřazením fyzické adresy pro konfiguraci daného rozhraní a editovat soubor **/etc/network/interfaces**. V tomto souboru se nastavují IP adresy jednotlivých síťových rozhraní.

```
# The loopback network interface
auto lo
iface lo inet loopback
# The primary network interface
auto eth0
iface eth0 inet static
        adress 10.98.243.2
        netmask 255.255.255.252
        broadcast 10.98.243.4
auto eth1
iface eth1 inet static
       address 192.168.0.254
       netmask 255.255.255.252
       broadcast 192.168.0.255
```

Došlo ke konfiguraci statických adres pro rozhraní eth0 a eth1. Pro rozhraní eth0 byla použita konfigurace adres přidělená od poskytovatele připojení k Internetu. Rozhraní eth1 reprezentuje vnitřní síť, jejíž další konfigurace je dále popsána. Přidělená adresa 192.168.0.254 slouží management směrovače a jeho možnou správu v případě pádu DHCP serveru, který bude dále konfigurován.

Aby bylo připojení kompletně funkční, bylo nutné provést ještě konfiguraci adres pro jmenné servery DNS (Domain Name Systém) editací souboru **/etc/resolv.conf**. Zde byly použity adresy přidělené poskytovatelem připojení k Internetu.

```
domain unhfree.czf
search unhfree.czf
nameserver 10.98.231.66
nameserver 10.98.0.243
```

Po dokončení konfigurace síťových rozhraní je potřeba změny aplikovat příkazem **#/etc/init.d/networking restart** a otestovat funkčnost připojení.

```
root@debian:/# ping google.cz
PING google.cz (173.194.44.255) 56(84) bytes of data.
```

```
64 bytes from 173.194.44.255: icmp_req=1 ttl=56 time=3.50 ms
64 bytes from 173.194.44.255: icmp_req=2 ttl=56 time=3.03 ms
64 bytes from 173.194.44.255: icmp_req=3 ttl=56 time=2.78 ms
```

Po konfiguraci a otestování síťových rozhraní již bylo možné použít pro připojení ke směrovači lokální počítačovou síť namísto sériové linky. Pro připojení bylo užito SSH spojení prostřednictvím programu puTTy, v němž byla nastavena cílová adresa 192.168.0.254. Spojení prostřednictvím SSH tunelu zajišťuje SSH server instalovaný v průběhu instalačního procesu systému. Jeho spuštění lze před odpojením sériové linky ověřit v seznamu běžících procesů po zadání příkazu **#ps –A**, kde je tento proces reprezentován záznamem sshd.

4.5 Instalace modulů

Pro další konfiguraci směrovače bylo nutné nainstalovat některé moduly, jelikož nebyly nainstalovány během instalačního procesu systému. Během instalace modulů není potřeba zadávat cestu jejich zdroje, jelikož je tato cesta nastavena v souboru **/etc/apt/sources.list**, který lze měnit nebo doplňovat o další zdroje.

Moduly je vhodné nainstalovat najednou i přes to, že je jejich konfigurace je prováděna později. Je totiž nutné co nejdříve přesměrovat vytváření logů těchto modulů do operační paměti ALIXu, aby nedocházelo k degradaci CF paměťové karty. Instalace probíhá příkazem **#apt-get install název_modulu.** Byly instalovány tyto moduly:

Shorewall – sada skriptů pro editaci firewallu,

OpenVPN – brána virtuální lokální sítě,

MC – vizuální souborový manažer,

DHCP3-server – server pro dynamické přidělování adres,

VLAN – modul s ovladači pro virtuální sítě.

4.6 Přesměrování logů

Jak již bylo zmíněno, opakovaný zápis na CF paměťovou kartu vede k jejímu postupnému zničení. Při běhu směrovače dochází pouze k zápisu logů jednotlivých modulů. Výstup logů je zapisován do adresáře **/var/log** . Pro vyřešení tohoto problému se

nabízejí dvě varianty. První spočívá v přesměrování obsahu adresáře **/var/log** do **/dev/null**, což v podstatě znamená vyhození těchto logů do prázdného souboru, kde nedochází k zápisu. Tato varianta není doporučována, jelikož by v budoucnu nebylo možné sledovat logy o připojených stanicích, pokusech o útok, kolizích v systému apod. Lepší, ovšem mnohem náročnější na konfiguraci, je druhá varianta, která spočívá ve vytvoření odkládacího prostoru v operační paměti počítače, respektive ALIXu. V souboru **/etc/fstab** je potřeba vytvořit záznam pro přípojné místo adresáře **/var/log** v souborovém systému s možností zápisu do operační paměti.

# <file system>	<mount point>	<type>	<options>	<dump>	<pass>
proc	/proc	proc	defaults	0	0
shm	/dev/shm	tmpfs	nodev,nosuid	0	0
tmpfs	/tmp	tmpfs	rw	0	0
tmpfs	/var/log	tmpfs	rw	0	0

Adresář **/var/log** neobsahuje pouze soubory, ale i další adresáře. Init skript, který bude spouštěn během načítání systému pro vytvoření přípojného místa v operační paměti, musí být proto naprogramován tak, aby v tomto přípojném místě vytvořil adresáře pro odkládání logů některých modulů. Pokud by tak nebylo učiněno, mohlo by vlivem neexistující cesty dojít k pádu procesů těchto modulů. Situace je ošetřena následujícím skriptem.

```
#! /bin/sh
PATH=/sbin:/usr/sbin:/bin:/usr/bin
. /lib/init/vars.sh
. /lib/lsb/init-functions
CONFIG_FILE=/etc/default/tmpfs_mount
LIST=""
if [ -f $CONFIG_FILE ]; then
        . $CONFIG_FILE
fi
do_start() {
        for i in $LIST ; do
                DIR=`echo $i | cut -d : -f 1`
```

```
            USER=`echo $i | cut -d : -f 2`
            GROUP=`echo $i | cut -d : -f 3`
            if [ -z "$USER" ]; then
                USER="root"
            fi
            if [ -z "$GROUP" ]; then
                GROUP="root"
            fi
            echo "mkdir /var/log/$DIR; chown $USER:$GROUP /var/log/$DIR"
            mkdir /var/log/$DIR && chown $USER:$GROUP /var/log/$DIR
        done
}
case "$1" in
    start)
        do_start
        ;;
    restart|reload|force-reload)
        echo "Error: argument '$1' not supported" >&2
        exit 3
        ;;
    stop)
        ;;
    *)
        echo "Usage: $0 start|stop" >&2
        exit 3
        ;;
esac
```

Tento skript je pod názvem **tmps_mount** uložen v adresáři **/etc/init.d**. Pro správnou funkci je nutné jeho zavedení před spuštěním procesů, které vytvářejí logy. Toto bylo ošetřeno vytvořením symbolického linku do adresáře **/etc/rcS.d/**, kde číselná hodnota za znakem S nastavuje pořadí spouštění během načítání systému.

4.7 Nastavení NTP

Platforma ALIX není vybavena baterií, která by zajišťovala takt pro běh vnitřních hodin po vypnutí nebo restartu systému. Aktuální čas, který je potřeba například pro analýzu údajů z logů, je nutné po startu systému synchronizovat prostřednictvím NTP serveru. Konfigurace NTP klienta probíhá editací souboru **/etc/ntp.conf**. V tomto souboru je potřeba nastavit adresu NTP serveru.

```
# pool: <http://www.pool.ntp.org/join.html>
server 0.debian.pool.ntp.org iburst
server 1.debian.pool.ntp.org iburst
server 2.debian.pool.ntp.org iburst
server 3.debian.pool.ntp.org iburst
```

4.8 Konfigurace VLAN

Jedním z cílů práce bylo rozdělit vnitřní síť do tří segmentů prostřednictvím virtuálních sítí. Tato konfigurace umožní například oddělit podnikovou síť od sítě návštěvníků na konferencích a demilitarizovanou zónu určenou pro servery. Pro vnitřní síť byl vybrán rozsah adres 192.168.0.0/24, tedy celý rozsah 256 adres třídy C. Tento rozsah byl rozdělen do pěti segmentů. Jedná se o podsítě 192.168.0.0/26, 192.168.0.64/26, 192.168.0.128/26, 192.168.0.252/30 a zbytek, který je nepřidělen pro případ budoucího užití.

Nejprve bylo potřeba ověřit zda se modul 802.1q pro virtuální sítě načítá během startu systému. Toto je možné ověřit v souboru **/etc/modules**.

```
# This file contains the names of kernel modules that should be loaded at boot time
loop
8021q
```

Konfigurace virtuálních sítí VLAN1, VLAN2, VLAN3 byla provedena editací souboru **/etc/network/interfaces**. Jednotlivé VLAN se nastavují stejně jako ostatní síťová zařízení, jelikož na ně systém stejně nahlíží. Odlišení sítí VLAN probíhá zápisem číselné hodnoty VID z rozsahu 0-4094 za tečku u konkrétního rozhraní. V konfiguračním souboru byly pro jednotlivé VLAN nastaveny adresní rozsahy, které budou užívány DHCP

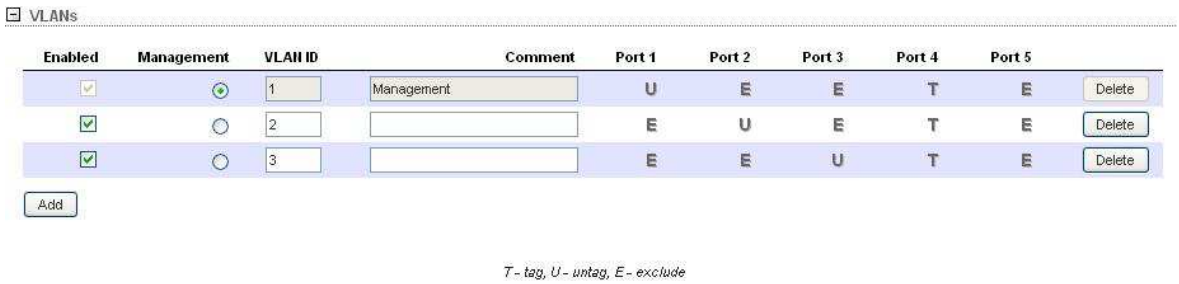
serverem. Zápis iface eth1.1 znamená VLAN 1 na rozhraní eth1, které je určeno pro vnitřní síť. Konfigurace pro fyzické rozhraní eth1 byla ponechána pro případ konfigurace bez použití přepínače, který by zrušil tagování.

```
auto eth1
iface eth1 inet static
        address 192.168.0.254
        netmask 255.255.255.252
        broadcast 192.168.0.255
auto eth1.1
iface eth1.1 inet static
        address 192.168.0.1
        netmask 255.255.255.192
        broadcast 192.168.0.63
auto eth1.2
iface eth1.2 inet static
        address 192.168.0.65
        netmask 255.255.255.192
        broadcast 192.168.0.127
auto eth1.3
iface eth1.3 inet static
        address 192.168.0.129
        netmask 255.255.255.192
        broadcast 192.168.0.191
```

Po dokončení konfigurace síťových rozhraní a VLAN je potřeba změny aplikovat příkazem **#/etc/init.d/networking restart** a otestovat funkčnost připojení.

Přepínač připojený ke směrovači byl nastaven tak, aby byly jednotlivé VLAN směrovány na příslušné porty, kde je zrušeno tagování. Porty 1-3 korespondují s VLAN1-3. Porty 4 je použit jako uplink přepínače. Konfigurace je znázorněna na obrázku č. 7.

Obr. č. 7: Konfigurace VLAN UBNT Touht Switch



Zdroj: Autor práce

4.9 Pravidla pro Shorewall

Pro správnou funkčnost směrovače bylo nutné provést konfiguraci firewallu. Pro tuto konfiguraci byla zvolena sada skriptů a konfiguračních souboru Shorewall. Po instalaci Shorewallu se v adresáři **/etc/shorewall/** nachází pouze soubor shorewall.conf, jehož editace v tomto případě nebyla potřeba. Předloha všech konfiguračních souborů v prázdném stavu je uložena v adresáři **/usr/share/doc/shorewall/shorewall-config/**. Z tohoto adresáře byly do adresáře **/etc/shorewall/** zkopírovány a editovány soubory **zones, interfaces, policy, rules, masq, tos** a **tunnels**.

4.9.1 zones

V první řadě byly nadefinovány v souboru **/etc/shorewall/zones** zóny, mezi kterými bude řízen datový provoz. Jako typ byl zvolen ipv4, neboť veškeré adresy budou přidělovány právě z tohoto rozsahu. Zóna loc slouží pro lokální síť, mgmt pro management, dmz pro demilitarizovanou zónu, quest pro veřejnou část sítě, net je zóna připojená přímo do Internetu, vpn je zóna pro OpenVPN a fw reprezentuje vnitřní zónu směrovače.

#ZONE	TYPE	OPTIONS	IN	OUT
loc	ipv4			
mgmt	ipv4			
dmz	ipv4			
guest	ipv4			
net	ipv4			
vpn	ipv4			

fw	firewall

4.9.2 interfaces

Pro identifikaci jednotlivých zón byly tyto zóny v souboru **/etc/shorewall/interfaces** přiřazeny ke konkrétním síťovým rozhraním. Ke každému rozhraní byla přiřazena právě jedna zóna, aby byla zajištěna jejich vzájemná izolace. Do parametru broadcast je možné nastavit broadcast jednotlivých adresových rozsahů nebo si usnadnit práci a tuto hodnotu detekovat prostřednictvím parametru detect. Parametr options byl doplněn o poznámku, že je na některých rozhraních provozován DHCP server.

#ZONE	INTERFACE	BROADCAST	OPTIONS
net	eth0	detect	
mgmt	eth1	detect	
loc	eth1.1	detect	dhcp
guest	eth1.2	detect	dhcp
dmz	eth1.3	detect	
vpn	tap0	detect	dhcp

4.9.3 policy

V konfiguračním souboru **/etc/shorewall/policy** byla nastavena hrubá pravidla o tom, mezi kterými zónami je možná komunikace a v jakém směru. Parametr ACCEPT komunikaci povoluje, DROP pakety zahazuje. REJECT komunikaci blokuje, ale odesilateli paketů podává informaci o tom, že byl provoz blokován. Parametru REJECT je využito ve vnitřní síti, aby bylo možné identifikovat, že jsou jednotlivé části sítě v provozu i ze sítí, které do nich nemají bezprostřední přístup. Parametr DROP byl využit pro blokaci ze sítě Internet. Potenciálním útočníkům se zařízení v síti budou jevit jako nedostupná.

Zóna loc je privilegována nad ostatními, neboť má přístup do všech sítí. Jedná se o privátní síť využívanou pouze oprávněnými osobami. Ze zóny pro návštěvníky je umožněn přístup pouze do Internetu. Zóna mgmt slouží pouze pro správu směrovače. Přístup do všech sítí včetně Internetu je u této zóny blokován. Zóně firewallu směrovače je umožněn přístup do všech sítí za účelem kontroly nad veškerým provozem. Ze zóny vpn je umožněn přístup do zóny loc.

Ve směru komunikace, který by mohl být terčem útoku, bylo nastaveno vytváření logů o provozu paketů prostřednictvím parametru ULOG.

#SOURCE	DEST	POLICY	LOG LEVEL	LIMIT:BURST
# Accept rules here				
loc	all	ACCEPT		
guest	net	ACCEPT		
guest	dmz	ACCEPT		
dmz	net	ACCEPT		
mgmt	fw	ACCEPT		
fw	all	ACCEPT		
net	dmz	ACCEPT	ULOG	
vpn	loc	ACCEPT	ULOG	
# Drop and Reject rules here				
guest	loc	REJECT	ULOG	
guest	mgmt	REJECT	ULOG	
guest	fw	REJECT	ULOG	
dmz	loc	DROP	ULOG	
dmz	guest	DROP	ULOG	
dmz	fw	DROP	ULOG	
dmz	mgmt	DROP	ULOG	
mgmt	loc	REJECT		
mgmt	dmz	REJECT		
mgmt	net	REJECT		
mgmt	guest	REJECT		
net	loc	DROP		
net	guest	DROP		
net	mgmt	DROP	ULOG	
net	fw	DROP	ULOG	
net	vpn	DROP	ULOG	

4.9.4 rules

V konfiguračním souboru **/etc/shorewall/rules** byla nastavena jemnější pravidla, která specifikují protokoly a porty, kterými lze mezi jednotlivými zónami komunikovat. Zde bylo využito pravidla, že co není povoleno, je striktně zakázáno. Povolena je komunikace prostřednictvím SSH protokolu ke správě směrovače a přístup prostřednictvím VPN do vnitřní sítě na portu 1194. Na portu 5060 je nastaveno přesměrování portu na adresu VoIP telefonu v lokální síti. a port 5060. Povolen je rovněž přenos doménových záznamů a údajů pro časovou synchronizaci. Protokol igmp byl povolen pro správnou funkci DHCP serveru a icmp pro možnost ověření dostupnosti stanic v jednotlivých sítích.

#ACTION	SOURCE	DEST	PROTOCOL	PORT
ACCEPT	all	fw	tcp	ssh
DNAT	net	loc:192.168.0.20	udp	5060
ACCEPT	all	vpn	udp	1194
ACCEPT	fw	all	udp	domain,ntp
ACCEPT	all	fw	udp	domain
ACCEPT	quest	net	tpc	http
ACCEPT	fw	all	igmp	
ACCEPT	all	fw	igmp	
ACCEPT	all	mgmt	icmp	
ACCEPT	mgmt	all	icmp	
ACCEPT	loc	all	icmp	
ACCEPT	all	loc	icmp	
ACCEPT	fw	all	icmp	
ACCEPT	all	fw	icmp	

4.9.5 masq

Maškaráda, neboli maskování vnitřní sítě pod jednu adresu propagovanou do Internetu je konfigurováno v souboru **/etc/shorewall/masq**. Parametr INTERFACE označuje vstupní rozhraní a SUBNET definuje síť na kterou bude provoz směrován. V tomto případě byl použit celý rozsah 192.168.0.0/24. Není tedy nutné definovat zvlášť jednotlivé podsítě nebo rozhraní. Konfigurace tohoto souboru je velice důležitá pro správnou funkčnost připojení k Internetu pro vnitřní síť.

#INTERFACE	SUBNET	ADDRESS	PROTO	PORT(S)	IPSEC
eth0	192.168.0.0/24				

4.9.6 tos

Pro případ přetížení sítě byla zavedena priorizace některých služeb konfigurací souboru **/etc/shorewall/tos**. Maximální prioritu získalo ssh spojení, aby byla možná správa směrovače i případě přetížení sítě způsobeném například pirátským útokem. Velice citlivou službou na latenci a přednost zpracování paketů je telefonie VoIP na portu 5060. Tato služba není nikterak datově náročná, ale v případě zpoždění paketů dochází k výpadkům hlasu a přeslechům hovoru.

#SOURCE	DEST	PROTOCOL	SOURCE PORT	DEST PORT	TOS
all	all	tcp	ssh	ssh	16
loc	all	udp	5060	5060	8
all	loc	udp	5060	5060	8

4.9.7 tunnels

Tunel vytvořený v konfiguračním souboru **/etc/shorewall/tunnels** umožňuje prostřednictvím OpenVPN přistupovat do zóny vpn ze všech IP adres Internetu. Tento přístup je možný pouze za pravidel nastavených v předchozích souborech.

#TYPE	ZONE	GATEWAY	GATEWAY
openvpn	net	0.0.0.0/0	vpn

Validace všech konfiguračních souborů byla provedena příkazem **#shorewall** check. V případě chyby je administrátor upozorněn na konkrétní soubor a záznam obsahující chybnou konfiguraci. Spuštění validní konfigurace bylo provedeno příkazem **#shorewall restart**. Alternativou jsou příkazy zadané v tomto pořadí **#shorewall stop #shorewall start**.

4.10 Konfigurace DHCP serveru

Implementace DHCP serveru byla provedena, aby byl zajištěn maximální komfort připojování stanic jednotlivých účastníků privátní a veřejné sítě. Pro účely testování byl DHCP server provozován i na rozhraní pro demilitarizovanou zónu. Jeho konfigurace byla

odstraněna, jelikož servery mívají zpravidla statickou konfiguraci adres kvůli překladu veřejné IP adresy.

Prvním krokem byla definice rozhraní, na kterých bude DHCP server přidělovat adresy. Tato definice se nastavuje v konfiguračním souboru **/etc/default/isc-dhcp-server**. Pořadí, ve kterém jsou jednotlivá rozhraní zapsána v tomto souboru, koresponduje s pořadím konfigurace rozsahů přidělovaných IP adres v následujícím kroku.

```
#       Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACES="eth1.1 eth1.2 eth1.3"
```

Konfigurace rozsahů přidělovaných IP adres a dalších parametrů byla provedena editací souboru **/etc/dhcp/dhcpd.conf**. IP adresy pro konfiguraci jednotlivých rozhraní byly nastaveny dle definice v souboru **/etc/network/interfaces**. Rozsahy přidělovaných IP adres byly sníženy o adresu sítě, broadcast a výchozí bránu. IP adresa 192.168.0.20 byla rezervována k fyzické adrese VoIP telefonu umístěného v lokální síti.

```
# nastaveni pro eth1.1
subnet 192.168.0.0 netmask 255.255.255.192 {
        range 192.168.0.2 192.168.0.62;
        option routers 192.168.0.1;
        option domain-name-servers 10.98.231.66, 10.98.0.243;
        option subnet-mask 255.255.255.192;
        option broadcast-address 192.168.0.63;
        default-lease-time 6000;
        max-lease-time 12000;
        server-identifier 192.168.0.65;
}
}
host voip {
   hardware ethernet 00:0a:3b:2c:11:af;
   fixed-address 192.168.0.20;
}
# nastaveni pro eth1.2
subnet 192.168.0.64 netmask 255.255.255.192 {
```

```
range 192.168.0.66 192.168.0.126;
option routers 192.168.0.65;
option domain-name-servers 10.98.231.66, 10.98.0.243;
option subnet-mask 255.255.255.192;
option broadcast-address 192.168.0.127;
default-lease-time 600;
max-lease-time 7200;
server-identifier 192.168.0.65;
```

Spuštění DHCP serveru se provádí prostřednictvím příkazu **#/etc/init.d/isc-dhcp-server start.** DHCP server je v činnosti již po instalaci modulu, vhodnější je proto použít příkaz **#/etc/init.d/isc-dhcp-server restart**.

4.11 OpenVPN

Na směrovači bylo nakonfigurováno tunelové spojení do privátní sítě prostřednictvím OpenVPN serveru. Toto spojení umožňuje využívat služby provozované v privátní lokální síti z celého Internetu. Spojení bylo zabezpečeno prostřednictvím hesel a certifikační autority.

4.11.1 Certifikační autorita

Pro server a klienty byly před konfigurací OpenVPN serveru vygenerovány klíče a certifikáty. Vzorové skripty pro generování těchto souborů se nacházejí v adresáři **/usr/share/doc/openvpn/examples/easy-rsa/**. Adresář se skripty byl zkopírován do adresáře **/etc/openvpn**. Před generováním klíčů byl editován soubor **/etc/openvpn/easy-rsa/2.0/vars** za účelem vytvoření podkladových řetězců pro generování.

```
export KEY_SIZE=1024
export KEY_COUNTRY=CZ
export KEY_PROVINCE="Czech Republic"
export KEY_CITY="Mesto"
export KEY_ORG="Spolecnost"
export KEY_EMAIL="email@email.cz"
```

V dalším kroku byly v adresáři **/etc/openvpn/easy-rsa/2.0/vars** příkazem **#clean-all** vymazány vzorové klíče. Příkazem **#buil-ca** byla vytvořena certifikační autorita. Příkaz **#buil-key-server** vytvořil veřejný klíč pro server a postupným zadáváním příkazu **#buil-key-pass** byly generovány privátní klíče klientů. Během generování privátních klíčů byly zobrazeny výzvy pro zadání přihlašovacích hesel budoucích klientů. Celý proces generování trval přibližně 15 minut. Po tomto procesu je nezbytné certifikační autoritu a privátní klíče předat klientům a tyto klíče ze serveru z bezpečnostních důvodů odstranit.

4.11.2 OpenVPN server

Pro konfiguraci OpenVPN serveru byl do adresáře **/etc/openvpn** zkopírován vzorový konfigurační soubor **/etc/openvpn/vpn_server.conf** z adresáře **/usr/share/doc/openvpn/examples,** u kterého byla provedena editace některých parametrů. Port pro komunikaci byl ponechán na výchozí hodnotě, tedy 1194. Bylo zvoleno virtuální rozhraní dev tap, které je součástí jádra, a protokol udp, který zajišťuje rychlejší spojení. Ze zbylého rozsahu IP adres, který byl ponechán pro budoucí užití, bylo odebráno 8 adres pro přidělení VPN klientům. V tomto souboru je rovněž potřeba definovat cesty k certifikační autoritě a veřejnému klíči. Parametr keepalive udržuje prostřednictvím icmp spojení naživu i při nečinnosti klienta. push "route 192.168.0.0 255.255.255.192" směruje provoz VPN klientů na rozsah lokální sítě. Parametr push "dhcp-option DNS 10.98.231.66" umožňuje klientům využívat stejné DNS jako užívá lokální síť směrovače. Spuštění OpenVPN serveru se provádí příkazem **#/etc/init.d/openvpn start**.

```
mode server
tls-server
port 1194
proto udp-server
dev tap
server 192.168.0.192 255.255.255.248
ca /etc/openvpn/ca.crt
cert /etc/openvpn/server.crt
key /etc/openvpn/server.key
log-append /var/log/openvpn.log
```

```
status /var/run/vpn.status 10
user nobody
group nogroup
keepalive 10 120
comp-lzo
verb 3
push "route 192.168.0.0 255.255.255.192"
push "dhcp-option DNS 10.98.231.66"
```

4.12 Testování

V průběhu testování byla nejprve ověřena správná funkčnost DHCP serveru. K přepínači připojeného do eth1 byly připojeny tři počítače. U těchto počítačů bylo ověřeno zda mají přidělenu IP adresu ze správného rozsahu. Následně byl z jednotlivých počítačů prostřednictvím příkazového řádku odeslán icmp **#ping 192.168.0.X** na výchozí bránu DHCP serveru směrovače.

Přístup do sítě internet byl testován nejprve z terminálu směrovače prostřednictvím icmp **#ping google.cz**. Následně byl proveden test na jednotlivých počítačích. Prostřednictvím webového prohlížeče byla zadána stránka http://www.google.cz čímž byla ověřena i správná funkce překladu pomocí DNS resolveru.

Správné směrování paketů na portu 5060 pro adresu 192.168.0.20 bylo ověřeno připojením VoIP telefonu, z něhož byl uskutečněn hovor na mobilní telefon.

Vzájemné oddělení jednotlivých sítí prostřednictvím VLAN bylo ověřováno pokusem o přihlášení na FTP server spuštěný na počítači umístěném v privátní lokální síti. Tento server byl pro stanice z ostatních podsítí nedostupný, čímž se potvrdilo fungování VLAN. Mezi jednotlivými sítěmi byla možná pouze odpověď na icmp dotazy.

Přístup do privátní sítě prostřednictvím VPN byl ověřován pomocí OpenVPN klienta nainstalovaného na počítači, který se připojoval do sítě prostřednictvím mobilního připojení od společnosti Vodafone. Test byl proveden připojením na FTP server umístěný v privátní lokální síti.

5 ZHODNOCENÍ VÝSLEDKŮ

Výsledkem vlastního řešení autora práce je provozuschopný plně funkční směrovač, který splňuje všechny stanovené podmínky. Jedná se o cenově dostupný směrovač se spotřebou elektrické energie do 4W. Tento směrovač je možné rozšiřovat o další moduly, čímž se dá značně rozšířit jeho funkcionalita. Materiálové náklady na stavbu směrovače byly vyčísleny na 3 000 Kč včetně kovového pouzdra a CF paměťové karty. Nejlevnější konkurenční řešení v obdobné kvalitativní a výkonové hladině přináší produkt Mikrotik RouterBoard RB800 256 MB RAM L6. Cena této alternativy je přibližně 7 000 Kč. Navíc je uživatel omezen pouze funkcionalitou umožněnou operačním systémem tohoto produktu.

Instalace a konfigurace vlastního směrovače s operačním systémem Debian je oproti komerčním již hotovým řešením časově velmi náročná, což může značně zvýšit náklady na lidské zdroje při jednorázové implementaci do počítačové sítě. Přínosem je možnost klonování výsledného řešení. U dalších kopií CF paměťových karet tak lze pouze modifikovat konfigurace jednotlivých modulů, což není zdaleka tak náročná činnost. V případě širšího nasazení se časová náročnost ztrácí a finanční náklady hovoří ve prospěch vlastního řešení.

Konfigurace výše zmíněné modelové situace přináší podrobný ucelený návod a postup konfigurace modulů směrovače podnikové sítě. Záměr jednotlivých kroků je podrobně rozebrán což přináší snadnou orientaci pro případné rozšíření funkcionality směrovače, nebo jeho modifikace pro uplatnění v rozličných počítačových sítích.

Námětem pro další zlepšení a zvýšení konkurenceschopnosti předkládaného řešení by mohlo být doplnění administrátorského rozhraní. Tímto rozhraním není myšleno prostředí pracovní plochy, ale webové rozhraní propojené s funkcionalitou jednotlivých modulů. I pro tuto oblast existuje mnoho rozšiřujících modulů, které lze snadno nainstalovat.

6 ZÁVĚR

V této práci autor popisuje obecné charakteristiky operačního systému Debian s návazností na řízení síťového provozu. Funkcionalita směrovačů je uváděna v kontextu s operačním systémem Debian. Autor předkládá jednu z možností, kterou se administrátoři mohou ubírat při výběru vhodného řešení pro řízení provozu počítačových sítí. Svým řešením směrovače doplňuje oblast počítačových sítí, pro kterou na trhu není dostatek alternativních produktů. Jedná se o středně velké podnikové sítě, které nevyžadují výkonný směrovač v řádu desítek tisíc korun, avšak vyžadují jeho funkcionalitu. Tuto funkcionalitu nedokáží poskytnout stolní směrovače určené pro domácí použití.

Alternativním řešením mohou být směrovače na jiných platformách včetně linuxové distribuce, vybrané dle preferencí administrátora sítě. I pro tato řešení může být tato práce cenným přínosem. Na Internetu lze nalézt širokou škálu postupů pro konfiguraci linuxového směrovače, ale nejedná se o ucelené návody, jak celkové řešení z jednotlivých modulů sestavit. Mnoho prostudovaných postupů bylo značně zastaralých. Z tohoto důvodu se autor domnívá, že tato práce může být po nějaký čas přínosem pro mnoho začínajících administrátorů v linuxovém prostředí, kteří zvolí obdobnou variantu směrovače.

Každý administrátor počítačové sítě by při výběru směrovače měl předvídat v delším časovém horizontu a zvažovat, které služby budou na síti provozovány. Na základě těchto předpokladů by měl volit směrovač s potřebnou funkcionalitou a upřednostňovat řešení, která je možné do budoucna rozšiřovat.

7 SEZNAM POUŽITÝCH ZDROJŮ

1. HUNT, Craig. *Linux Síťové servery*. 2006. Praha: SoftPress, 672 s. ISBN 80-86497-59-3

2. KOFLER, Michael. *Linux 2012*. 2012. Německo: Adison Wesley Verlag, 1208. ISBN 3827331471.

3. WILLIAMS, Graham. GNU/Linux Desktop Surfoval Guide. 2006. eBook. Ang. Togaware.com. ISBN/ASIN: 0975710915.

4. BASTA, Alfred., FINAMORE, Dustin, A., BASTA, Nadine.Linux Operations and Administrations. 2012. Angl.Course Technology. 858 s. ISBN 111103530X.

5. BARET, Daniel., Silverman, Richard., BYRNES, Robert. SSH, The Secure Shell. 2011. Angl. O´Reilly Media. 668 s. ISBN 0596008953.

6. SOSINSKY, Barrie. Mistrovství – počítačové sítě. 2010. COMPUTER PRESS, 2010. 840 s. ISBN 9788025133637.

7. CASAD, Joe.Sams. Tlach Yourself TCP/IP in 24 Hours.2011. Angl.U.S. Corporate and Goverment Sales.515 s. ISBN 0672335719

8. DEBIAN, GNU/Linux. Dokumentace. [online] 10.3.2013 [citace 20.3.2013]. Dostupný z WWW: <http://www.debian.org/doc/>

9. Wikipedie. GNU General Public Licence. [online] 9.8.2012 [citace 11.2.2013]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/GNU_General_Public_License>

10. ZAPLETAL, Lukáš. Lehký úvod do LDAP. [online] 24.7.2000 [citace 12.03.2013]. Dostupný z WWW: <http://www.root.cz/clanky/lehky-uvod-do-ldap/>

11. HANOUSEK, Tomáš. Linuxákův průvodce po adresářích. [online] 22.10.2007 [citace 15.3.2013]. Dostupný z WWW: <http://www.linuxexpres.cz/praxe/linuxakuv-pruvodce-po-adresarich>

12. Wikipedie. Secure Shell. [online] 19.1.2013 [citace 2.2.2013]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Secure_Shell>

13. ŠTRAUCH, Adam. VNC a Linux: vzdálená plocha. [online] 2.1.2009 [citace 5.2.2013]. Dostupný z WWW: <http://www.root.cz/clanky/vnc-a-linux-vzdalena-plocha/>

14. OpenVPN. Dokumentation. [online] 2013 [citace 10.3.2013]. Dostupný z WWW: <http://openvpn.net/index.php/open-source/documentation.html>

15. Wikipedie. Router. [online] 13.8.2004 [citace 10.1.2013]. Dostupný z WWW: <http://www.linuxsoft.cz/article.php?id_article=302>

16. ČEČÁK, Ondřej. Linux v příkazech – konfigurace sítě. [online] 2.1.2009 [citace 5.2.2013]. Dostupný z WWW: <http://www.linuxsoft.cz/article.php?id_article=302>

17. ŽALUD, Vladimír. DHCP-1 (instalace a konfigurace serveru). [online] 24.8.2006 [citace 5.2.2013]. Dostupný z WWW: <http://www.abclinuxu.cz/clanky/site/dhcp-1-instalace-a-konfigurace-serveru>

18. Shorewall. Dokumentation. [online] 2012 [citace 20.12.2012]. Dostupný z WWW: <http://shorewall.net/>

19. KLOS, Tomáš. Shorewall, 1. díl. [online] 4.10.2005 [citace 19.12.2012]. Dostupný z WWW: <http://www.linuxsoft.cz/article.php?id_article=302>

20. HOKŮV, OSPF – Dinamické routování. [online] 3.8.2005 [citace 2.2.2013]. Dostupný z WWW: <http://www.abclinuxu.cz/clanky/site/ospf-dynamicke-routovani>

21. FUČÍK, Marek. Bakalářská práce – Internet Service Provider. ČZÚ, 2011. 53 s.

8 REJSTŘÍK POUŽITÝCH OBRÁZKŮ

9 PŘÍLOHY

Příloha č. 1: Seznam použitých zkratek

Příloha č. 1

CD - Comapct Disk

CF – Compact Flash

CoS - Class of Service

ČR – Česká republika

DES - Data Encryption Standard

DHCP - Dynamic Host Configuration Protocol

DiffServ - Differential Service

DMZ - Demilitarized Zone

DNS - Domain Name System

DSCP - differential service code point

DSL – Digital Subscribe Line

DVD - Digital Versatile Disc

FTP - File Transfer Protocol

GNOME - GNU Object Model Envoronment

GPL - General Public License

GUI - Graphical User Interface

HTTP - Hypertext Transfer Protocol

HW – Hardware

IANA - Internet Assigned Numbers Autority

IP - Internet Protokol

IPSec - Internet Protocol SECurity

ISP - Internet Service Provider

KDE - K Desktop Environment

LDAP - Lightweight Directory Access Protocol

MB - Mega Byte

MS – Microsoft

NTP - Network Time Protocol

NAND - NOT AND

NAT - Network Address Translation

OFDM - Orthogonal Frequency Division Multiplexing

OLT - Optical Line Terminal

ONU/ONT - Optical Network Unit/ Optical Network Termination

OS - Operační Systém

OSPF - Open Shortest Path First

PPP - Point-to-Point Protocol

QoS - Quality of Service

RIPE NCC - Réseaux IP Européens Network Coordination Centre

ROM – Read Only Memory

RSA - Rivest, Shamir and Adleman

SHA - Secure Hash Algorithm

SSH - Secure Shell

SSL - Secure Sockets Layer

TCP- Transmission Control Protocol

ToS - Type of Service

UDP - User Datagram Protocol

US – Uninted States

USB - Universal Serial Bus

VGA - Video Grapihics array

VLAN - Virtual Local Area Network

VoIP - Voice over Internet Protocol

VNC - Virtual Network Computing

VPN - Virtual Private Network

W – Watt

Wi-Fi - Wireless Fidelity

Xfce - Xforum common environment