

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ  
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

UNIVERZÁLNÍ LINUXOVÝ SERVER PRO MALÉ A STŘEDNÍ  
FIRMY UMOŽŇUJÍCÍ JEDNODUCHÝ DOHLED NAD SÍTÍ

DIPLOMOVÁ PRÁCE  
MASTER'S THESIS

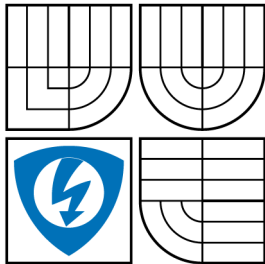
AUTOR PRÁCE  
AUTHOR

BC. STANISLAV JUŘENA

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY  
A KOMUNIKAČNÍCH TECHNOLOGIÍ  
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND  
COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

UNIVERZÁLNÍ LINUXOVÝ SERVER PRO MALÉ A STŘEDNÍ  
FIRMY UMOŽŇUJÍCÍ JEDNODUCHÝ DOHLED NAD SÍTÍ  
UNIVERSAL LINUX SERVER FOR SMALL AND MEDIUM COMPANIES ENABLING  
SIMPLE NETWORK CONTROL

DIPLOMOVÁ PRÁCE  
MASTER'S THESIS

AUTOR PRÁCE  
AUTHOR

BC. STANISLAV JUŘENA

VEDOUCÍ PRÁCE  
SUPERVISOR

DOC. ING. KAREL BURDA, CSC.

BRNO 2008

ZDE VLOŽIT LIST ZADÁNÍ

Z důvodu správného číslování stránek

ZDE VLOŽIT PRVNÍ LIST LICENČNÍ  
SMOUVY

Z důvodu správného číslování stránek

ZDE VLOŽIT DRUHÝ LIST LICENČNÍ  
SMOUVY

Z důvodu správného číslování stránek

## **ABSTRAKT**

Cílem této práce bylo navrhnout počítačovou síť pro malou a střední firmu, která bude mimo jiné tvořena síťovým serverem, umožňujícím snadný dohled nad touto sítí. Dalším úkolem serveru je poskytnutí připojení k síti internet účastníkům vnitřní sítě, jejich zabezpečení a přístup k základním službám.

V teoretické části je diskutována volba distribuce operačního systému Linux s ohledem na požadované služby, stabilitu a dlouhodobý provoz. Součástí práce je i teoretický úvod k jednotlivým službám, důvod jejich použití a jejich slabá místa.

Praktická část se věnuje instalaci a konfiguraci operačního systému Debian, zprovoznění základních služeb a nastavení vybraných monitorovacích programů.

## **KLÍČOVÁ SLOVA**

Linux, Debian, DNS, WWW server, DHCP, Samba, firewall, FTP server, monitorování, MRTG, Ipac-ng, kvalita služeb.

## **ABSTRACT**

The main object of this thesis was to design a computer network for small and medium companies which will be made among others from network server providing simple network control. The next task was to provide internet connection to subscribers of local area network, their security and access to common services.

There had been discussed the choice of distribution of Linux operation system with regarding to demanded services, stability and long lasting operation in theoretical part. One part of the work is a theoretic preliminary to separate services, to the purpose of their using and to their weaknesses.

The practical part deals with an installation and configuration of Debian operating system, launching the base services and the setting of selected monitoring programs.

## **KEYWORDS**

Linux, Debian, DNS, WWW server, DHCP, Samba, firewall, FTP server, monitoring, MRTG, Ipac-ng, Quality of Service.

## PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Univerzální linuxový server pro malé a střední firmy umožňující jednoduchý dohled nad sítí“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne .....

.....

(podpis autora)

# OBSAH

<b>1</b>	<b>Úvod</b>	<b>11</b>
<b>2</b>	<b>Linuxová řešení firemních sítí</b>	<b>12</b>
2.1	Úvod . . . . .	12
2.2	Linuxové možnosti . . . . .	12
2.2.1	WWW server (Web Server) . . . . .	13
2.2.2	DNS server . . . . .	13
2.2.3	Emailový server . . . . .	13
2.2.4	DHCP server . . . . .	13
2.2.5	Firewall . . . . .	13
2.2.6	Samba server . . . . .	14
2.2.7	FTP server . . . . .	14
2.2.8	Proxy server . . . . .	14
2.2.9	Monitorování sítě . . . . .	14
2.3	Služby popsané v této práci . . . . .	14
<b>3</b>	<b>Návrh počítačové sítě</b>	<b>16</b>
3.1	Úvod . . . . .	16
3.2	Návrh . . . . .	16
3.3	Není síť jako síť . . . . .	17
3.4	Realizovaná topologie sítě . . . . .	17
<b>4</b>	<b>Instalace Linuxu - Debian</b>	<b>19</b>
4.1	Úvod . . . . .	19
4.2	Instalace systému Debian . . . . .	19
4.2.1	Prerekvizity . . . . .	19
4.3	První spuštění . . . . .	21
<b>5</b>	<b>Firewall</b>	<b>22</b>
5.1	Úvod . . . . .	22
5.2	IPTables . . . . .	22
5.3	Prerekvizity . . . . .	22
5.4	Konfigurace . . . . .	22
5.4.1	Popis skriptu . . . . .	22
<b>6</b>	<b>Samba server</b>	<b>25</b>
6.1	Úvod . . . . .	25
6.2	Instalace . . . . .	25

6.3	Konfigurace serveru . . . . .	25
6.4	smb.conf . . . . .	25
6.5	Správa serveru . . . . .	26
6.5.1	SWAT . . . . .	26
6.5.2	SWAT a SSL . . . . .	27
6.5.3	Webmin . . . . .	27
<b>7</b>	<b>DHCP server</b>	<b>33</b>
7.1	Úvod . . . . .	33
7.2	Popis funkce . . . . .	33
7.3	Výhody použití DHCP . . . . .	33
7.4	Instalace Serveru . . . . .	34
7.5	Konfigurace serveru . . . . .	34
<b>8</b>	<b>FTP server</b>	<b>36</b>
8.1	Úvod . . . . .	36
8.2	ProFTPD . . . . .	36
8.3	Požadavky . . . . .	36
8.4	Instalace Serveru . . . . .	36
8.5	Šifrovaný přenos . . . . .	37
8.6	Konfigurace serveru . . . . .	37
8.7	Proftpd.conf . . . . .	37
8.7.1	Test spojení . . . . .	38
<b>9</b>	<b>Web server</b>	<b>39</b>
9.1	Úvod . . . . .	39
9.2	HTTP protokol . . . . .	39
9.3	Apache . . . . .	39
9.3.1	Instalace Apache . . . . .	39
9.3.2	Generování certifikátu . . . . .	39
9.3.3	Konfigurace Apache2 serveru . . . . .	40
9.4	MySQL . . . . .	40
9.4.1	Instalace MySQL . . . . .	40
9.4.2	phpMyAdmin . . . . .	41
<b>10</b>	<b>Webmin monitoring plugin</b>	<b>42</b>
10.1	Úvod . . . . .	42
10.2	Instalace . . . . .	42
10.3	Konfigurace . . . . .	42
10.4	Výsledek . . . . .	42

<b>11 Ipac-ng</b>	<b>46</b>
11.1 Úvod . . . . .	46
11.2 Instalace . . . . .	46
11.3 Konfigurace . . . . .	46
11.3.1 Vytváření grafů . . . . .	47
11.3.2 Propojení s MySQL . . . . .	47
11.4 Výsledek . . . . .	48
<b>12 MRTG</b>	<b>49</b>
12.1 Úvod . . . . .	49
12.2 Instalace . . . . .	49
12.3 Konfigurace . . . . .	49
12.3.1 Vytváření souborů . . . . .	49
12.3.2 Generování mrtg.cfg . . . . .	50
12.3.3 Aktualizace dat přes Cron . . . . .	51
12.4 Výsledek . . . . .	51
<b>13 Kvalita služeb</b>	<b>53</b>
13.1 Úvod . . . . .	53
13.2 Prerekvizity . . . . .	53
13.3 Konfigurace . . . . .	53
13.3.1 Popis skriptu . . . . .	55
<b>14 Praktická část</b>	<b>58</b>
14.1 Konfigurace routeru . . . . .	58
14.2 Zadání . . . . .	58
14.3 Postup řešení . . . . .	58
14.3.1 Výsledky scanování portů . . . . .	59
<b>15 Závěr</b>	<b>61</b>
<b>Literatura</b>	<b>63</b>
<b>A Příloha - skript IPTables</b>	<b>64</b>
<b>B Příloha - konfigurační soubor FTP serveru</b>	<b>68</b>
<b>C Příloha - skript zajišťující QoS</b>	<b>71</b>

## SEZNAM OBRÁZKŮ

3.1	Návrh síťové topologie malé firmy či dílčí části (vlan) . . . . .	16
3.2	Návrh síťové topologie střední firmy . . . . .	17
3.3	Zvolená síťové topologie k realizaci . . . . .	18
4.1	Konfigurace sítě při instalačním procesu Debianu . . . . .	19
4.2	Vytváření oddílů na pevném disku při instalačním procesu Debianu .	20
4.3	Dotaz na vytvoření zavaděče v MBR při instalačním procesu Debianu	20
6.1	Úvodní menu správy samba serveru přes SWAT . . . . .	28
6.2	Možností nastavení serveru SWAT . . . . .	29
6.3	Zobrazení stavu démonů a aktuálně přihlášených uživatelů . . . . .	30
6.4	Úvodní menu správy samba serveru Webmin . . . . .	31
6.5	Možností nastavení serveru Webmin . . . . .	32
6.6	Nastavení serveru pro upřesnění vlastností sítě . . . . .	32
7.1	Přiřazení dat z DHCP serveru síťové kartě ve Windows . . . . .	35
9.1	Rozhraní phpMyAdmin . . . . .	41
10.1	Základní okno Webmin-sysstats pluginu . . . . .	43
10.2	Sledování vytížení procesoru pomocí Webminu . . . . .	44
10.3	Sledování využití paměti pomocí Webminu . . . . .	45
11.1	Statistika přenosu a grafy v ipac-ng . . . . .	48
12.1	Základní správa MRTG . . . . .	52
13.1	Rozdělení šířky pásma pro příchozí spojení (download) . . . . .	54
13.2	Rozdělení šířky pásma pro odchozí spojení (upload) . . . . .	54

# 1 ÚVOD

Vzhledem k neustálému růstu počtu malých firem, vzniká i poptávka po levném a současně kvalitním řešení počítačových sítí. Co se týče firem středních, které jsou definovány rozsahem zaměstnanců v rozmezí 50 - 250, řeší se většinou rozdělením firemní infrastruktury na dílčí části. Málokdy bývá firma tohoto rozsahu situována do jedné budovy bez satelitních poboček. A právě do těchto, ať už virtuálních (vlan), či fyzických podsítí je možné implementovat řešení, popsané v této práci.

Operační systém Linux je pro účely vybudování sítě takřka bezkonkurenční volbou. Je zcela zdarma a současně nabízí velmi širokou škálu bezplatných programů, pomocí kterých je možné síť „vyladit“ do posledního detailu.

Linux je založen ve většině distribucí<sup>1</sup> na open-source přístupu, což znamená, že na vylepšování a získávání nových funkcí se podílejí převážně uživatelé tohoto operačního systému.

Tato práce si ukládá za cíl seznámit čtenáře s problematikou návrhu a realizace počítačové sítě pro malé a střední firmy. Dále se zabývá zabezpečením uživatelských stanic lokální sítě a snaží se jim poskytnout potřebné základní služby, usnadňující jim každodenní práci. V neposlední řadě je také kladen důkaz na snadnou správu celé sítě a jednoduchý monitoring jejího vytížení.

Po dohodě s vedoucím diplomové práce bude tato práce použita jako součást studijních materiálů k předmětu MNSB (Návrh, správa a bezpečnost počítačových sítí), proto je podrobněji popsán i instalační proces jednotlivých aplikací.

---

<sup>1</sup>distribuce = společně šířená ucelená sada jádra Linuxu, systémových knihoven, utilit, uživatelských nástrojů a aplikací. Popis některých distribucí a pomoc s výběrem můžete najít třeba na <http://www.root.cz>

## 2 LINUXOVÁ ŘEŠENÍ FIREMNÍCH SÍTÍ

### 2.1 Úvod

Tato kapitola se zabývá možnostmi, které nabízí realizace firemní sítě na platformě Linux. Čtenář se bude po jejím přečtení orientovat v jednotlivých aplikacích a bude schopen si sám vybrat, které služby využije a které jsou pro něj zcela zbytečné.

### 2.2 Linuxové možnosti

Linux nabízí prakticky neomezené možnosti nejen co se týče sítě, ale i uživatelské stanice, případně i mobilních zařízení. Instalace jednotlivých programů (rozšíření) je závislá prakticky jen na vývojářích daného programu, případně na podpoře jednotlivých distribucí.

Vzhledem k tomu, že se tato práce zabývá malou firemní sítí, nabízí se hned několik možností, jak si vybudovanou síť rozšířit a zjednodušit tak například práci administrátorům či uživatelům.

Mezi služby, používané ve firemních sítích patří:

1. WWW server (Web Server)
2. DNS server
3. Emailový server
4. DHCP server
5. Firewall
6. Samba server
7. FTP server
8. Proxy server
9. Monitorování sítě

Jelikož z výčtu pochopí jen pokročilí uživatelé, o co se vlastně jedná, seznámíme se podrobněji s jednotlivými službami.

### 2.2.1 WWW server (Web Server)

Tento server zajišťuje prezentaci vaší firmy na Internetu prostřednictvím webových stránek. Požadavkem je vlastnictví veřejné IP adresy, případně doménového jména. Samozřejmostí u tohoto serveru je i vytvoření tzv. intranetových stránek, které obsahují interní firemní informace a uživatelé k nim mohou přistupovat pouze z vnitřní sítě. Tento server bývá současně spojen i s monitorováním sítě - stará se o zobrazení statistik.

### 2.2.2 DNS server

Hlavním úkolem DNS serveru je překlad IP adres na doménové jména. Jeho funkce je však v poslední době rozšířena o elektronickou poštu a IP telefonii.

Pro srozumitelnost si funkci DNS serveru ukážeme na příkladu prohlížení webových stránek. Uživatelé po otevření prohlížeče zadají např. adresu `www.seznam.cz`. Pokud by však nebyl v síti, která má DNS server, musel by ke stejnému serveru přistupovat pomocí IP adresy (`77.75.72.3`), což je značně nepraktické a velmi náročné na paměť uživatele.

### 2.2.3 Emailový server

Emailový server, jak již název napovídá, nabízí možnost odeslání emailů. To by samo o sobě bylo velmi jednoduché, ovšem pokročilé emailové servery skenují přílohy emailů a hledají virovou nákazu. Stejně tak mohou zabraňovat hromadnému odesílání nevyžádané pošty (spam). Další službou, kterou nabízí emailový server, je odesílání emailů z určité domény. Bylo by v celku nepraktické používat emailovou adresu veřejných severů. Proto pokud komunikujeme se zástupcem některé firmy, nalezneme za jeho jménem název firmy (`@cpress.cz`).

### 2.2.4 DHCP server

Hlavním cílem DHCP serveru je zjednodušení práce s nastavováním síťových adres klientům v síti. Pokud tedy přijdete s notebookem do práce, připojíte síťový kabel a máte nastaveno načítání síťových adres s DHCP serveru, bude vám automaticky přiřazena IP adresa a budete moci bez složitých nastavování přistupovat do sítě.

### 2.2.5 Firewall

Vzhledem k tomu, že nejcennějším majetkem firem je tzv. Know How, bylo by v celku nebezpečné používat nezabezpečenou firemní síť, aby k těmto datům měl prakticky

kdokoliv přístup. K zabezpečení se používá Firewall, který zabraňuje přístupu z venkovní sítě do vnitřní nepovoleným uživatelům.

### 2.2.6 Samba server

Vzhledem k různorodosti uživatelů a současně i oblibě operačních systémů je nutné zajistit přístup do sítě všem uživatelům, nezávisle na platformě. Tohoto úkolu se zhostil Samba server, který umožňuje sdílení souborů a tiskáren mezi různými operačními systémy.

### 2.2.7 FTP server

Snadný přístup k datům nabízí FTP server. Přístup na něj je velmi rychlý a není závislý na operačním systému. Vzhledem ke stáří samotného FTP protokolu se v dnešní době stále více využívá služeb SFTP - tzv. zabezpečený FTP server. Útočník má ztíženo odchytení uživatelského jména a hesla, které je nutné zadat pro vstup na tento server.

### 2.2.8 Proxy server

Oddělení lokální počítačové sítě od sítě Internet zajišťuje Proxy server. Prakticky schovává uživatele a komunikace mezi klientem ve vnitřní síti a klientem v Internetu vypadá tak, jako by na straně LAN komunikoval přímo Proxy server. V případě pokročilejšího aplikačního proxy serveru je možné blokovat přístup na některé webové stránky, odstraňovat reklamy z http adres, apod. Příkladem tohoto typu serverů je např. Squid.

### 2.2.9 Monitorování sítě

Vzhledem k rozsáhlosti firemních je potřeba zobrazení aktuálně připojených uživatelů, vytížení linky případně vyhledávání poruch. Celkový dohled nad sítí zajišťují tzv. monitorovací programy. Nejuznávanějším a současně nejobtížnějším na konfiguraci je Nagios.

## 2.3 Služby popsané v této práci

S ohledem na důležitost jednotlivých služeb byly pro tuto práci vybrány pouze Web server, DHCP server, Firewall, Samba server, FTP server a monitorování sítě. Důvodem byla omezená délka této práce, možná počáteční nezkušenost čtenářů a vlastní pořadí, v jakém byly instalovány služby na síťový server.

O neuvedení DNS serveru rozhodla hlavně skutečnost, že DNS servery nabízejí poskytovatelé internetu (ISP). Emailový server je velmi složitá a na konfiguraci velmi náročná aplikace, která by svým rozsahem vydala za samotnou diplomovou práci. Obzvláště s rozšířením o funkci antivirového a antispamového filtru. Proxy server jsem v této práci neřešil hlavně kvůli tomu, že schovávání klientů za jeden server řeší funkce NAT implementována do Firewall. Zákaz přístupu na webové stránky a filtrování reklam je ponecháno na klientech - přece jen, jejich omezováním nic nezískáme.

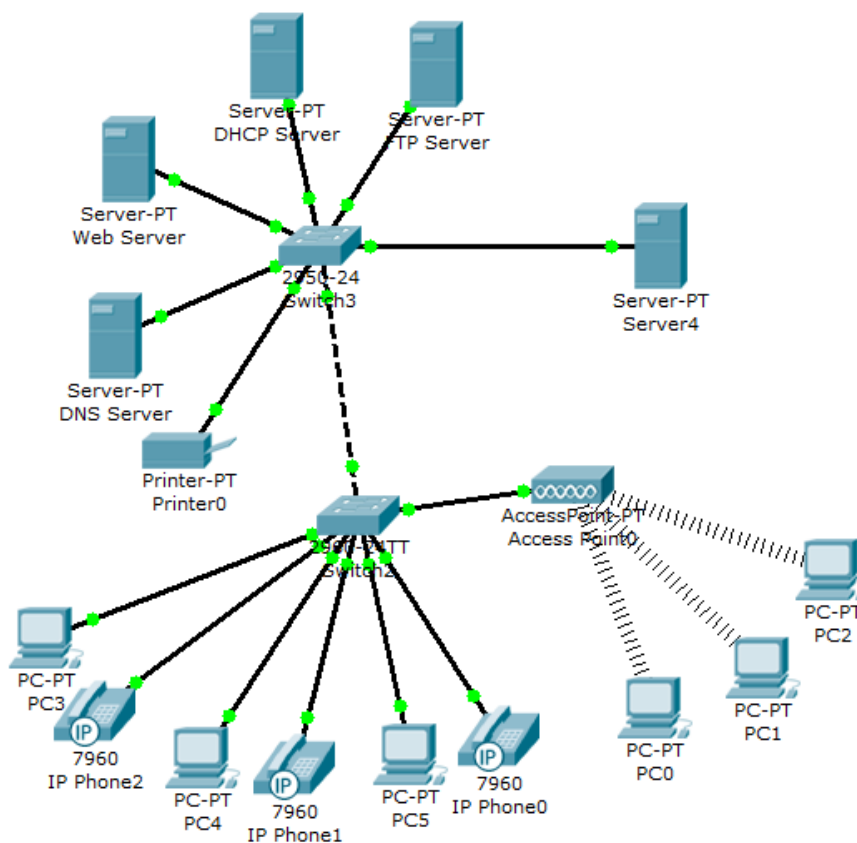
## 3 NÁVRH POČÍTAČOVÉ SÍTĚ

### 3.1 Úvod

Návrh počítačové sítě je výchozím a také nejdůležitějším bodem celé realizace sítě. Cílem je tedy propojení jednotlivých uživatelských stanic do ucelené struktury a jejich následné připojení do sítě Internet pomocí aktivních síťových prvků.

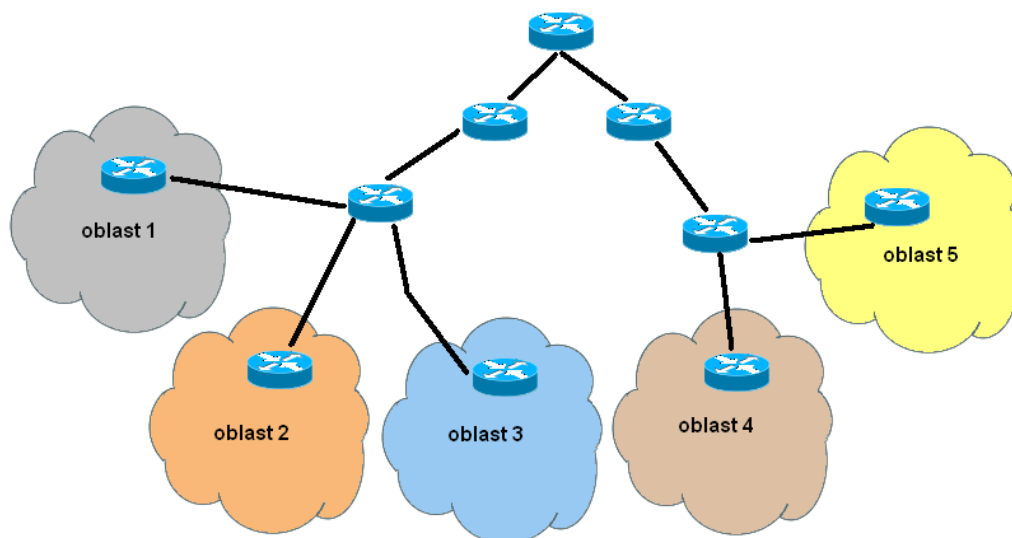
### 3.2 Návrh

Před jakoukoliv akcí je dobré předběžně určit celkový počet klientů, tiskáren a dalších prvků v síti a přibližně si rozvrhnout topologii. Příklad takového nákresu je na obr.3.1.



Obr. 3.1: Návrh síťové topologie malé firmy či dílčí části (vlan)

V případě střední firmy zavedeme pro přehlednost substituci sítě na obr.3.1 hraničním směrovačem v dané oblasti. Vznikne nám schéma podobné tomu na obr.3.2, kde v každé oblasti nalezneme podobnou strukturu sítě.



Obr. 3.2: Návrh síťové topologie střední firmy

Thusté čáry na obr.3.2 znázorňují některý z druhů spojení (ethernetový kabel, bezdrátový spoj, . . .). Stejně tak může být zvolená struktura založena na virtuálním rozdělení a přestože budou uživatelé ve stejné budově a třeba i místnosti, budou se nacházet v oddělených sítích. Dobrou volbou pro rozdělení sítě může být přihlašování pomocí programu založeném na VPN, či využití pokročilých směrovačů (Cisco), které umožňují snadné rozdělení sítě do podsítí.

### 3.3 Není síť jako síť

Základem každé lokální sítě je jeden nebo více aktivních prvků, které jsou propojeny strukturovanou kabeláží mezi sebou. Důležitým rozhodnutím je určení výchozího místa, kde se bude nacházet centrální bod celé sítě - ve většině případů se jedná o serverovnu. Poté je dobré si předem rozmístit jednotlivé uživatelské stanice do prostoru budovy, čímž zjistíme, kam umístit přístupové body, switche a koncové výstupy pro snadný přístup do sítě. Při výpočtech délky kabelů se vždy vyplatí přidat aspoň metr navíc - je mnohem jednodušší kabel svázat než dělat nové rozvody, počítejte také s chybným nacvaknutím konektorů apd.

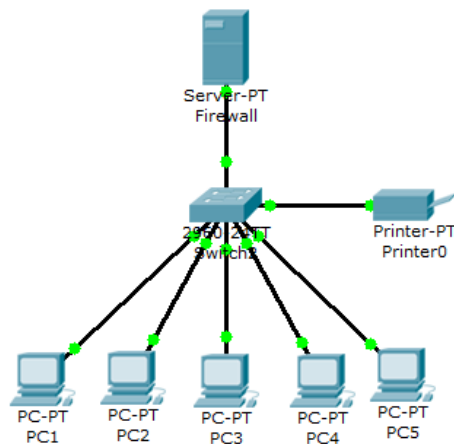
### 3.4 Realizovaná topologie sítě

V síti, která byla zvolena pro realizaci se nachází pět uživatelů, kteří využívají kromě připojení k internetu i intranetový Web server, sdílejí mezi sebou soubory a preferují tisk na síťové tiskárně. Z důvodu zabezpečení firemních dat bylo rozhodnuto, že v síti

nebude využíváno bezdrátové technologie. Celá síť tak bude realizovaná pomocí kabelových spojů.

Ze síťových serverů byly vybrány pouze DHCP, FTP a Web server. Překlad adres zajišťuje DNS server poskytovatele internetového připojení. Z důvodu malého síťového rozsahu bylo zvoleno, že se v síti bude nacházet jen jeden počítač, který bude plnit veškeré potřebné funkce. Toto rozhodnutí bylo učiněno vzhledem k nízkým finančním nákladům na elektrickou energii, ceně pořizovaného hardwaru a snadné správě síťových prostředků. IP telefony byly nahrazeny softwarovým řešením zejména kvůli zbytečným pořizovacím nákladům.

Výslednou topologii můžete vidět na obr.3.3.



Obr. 3.3: Zvolená síťové topologie k realizaci

## 4 INSTALACE LINUXU - DEBIAN

### 4.1 Úvod

Jako operační systém, na kterém poběží celý router byl zvolen Linux z distribuce Debian, a to zejména pro jeho cílené zaměření na správu sítě. Výhodou je také velmi jednoduchá práce spolu se snadnou instalací programů. Nevýhodou této distribuce je pak pomalá aktualizace jednotlivých balíčků s programy.

### 4.2 Instalace systému Debian

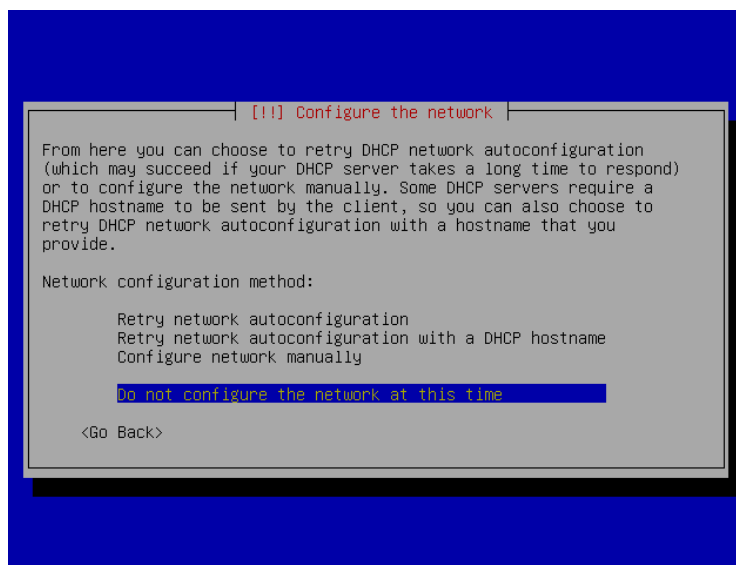
Zvolený Linux s označením Debian má spousty verzí. Pro funkci serveru ale bez problému postačí verze Sarge s označením NetInstall, která obsahuje jen základní potřebné balíčky. Vše ostatní se pak podle potřeby stahuje z internetu.

#### 4.2.1 Prerekvizity

Pro tento druh instalace je potřeba stáhnout instalační obraz ze stránky:

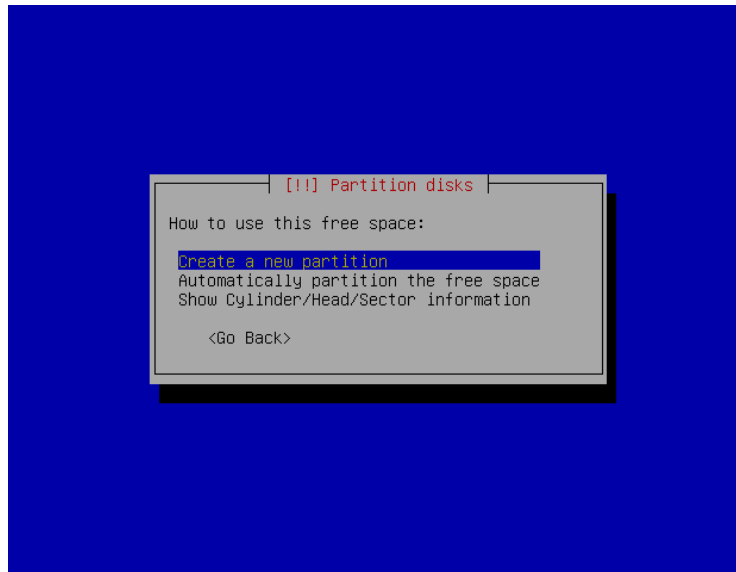
<http://www.us.debian.org/devel/debian-installer/>

Po jeho vypálení zvolíme bootování z CD mechaniky. V zápětí se objeví černá obrazovka s logem Debianu, která slouží k výběru jádra a typu instalace. Možnosti instalace se ukáží po stisku klávesy F1. Volba **linux26** znamená instalaci jádra 2.6. Následuje volba jazyka celého systému a rozložení klávesnice. Instalace pokračuje hledáním hardwaru počítače a konfigurace sítě, což je vidět na obr. 4.1.



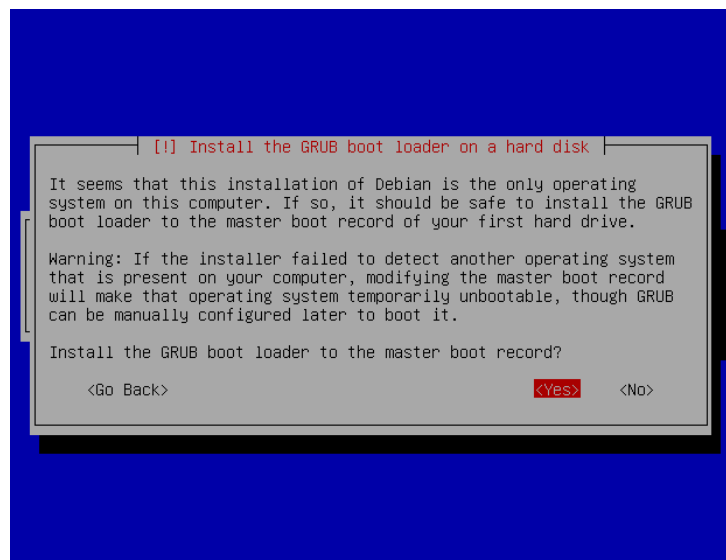
Obr. 4.1: Konfigurace sítě při instalačním procesu Debianu

Nyní je na řadě výběr disku a vytvoření tzv. oddílů (partitions). Vše se provádí stále ve stejném prostředí, které je na obr. 4.2.



Obr. 4.2: Vytváření oddílů na pevném disku při instalačním procesu Debianu

Při instalaci byla zvolena varianta automatického rozdělení místa do adresářů **/boot**, **/swap**, **/**, **/var**, **/usr** a **/home**. Tato možnost je ideální, pokud nemáte konkrétní představu o cílovém diskovém využití jednotlivých sekcí. Po proběhnutí instalace základního systému se ještě objeví dotaz na zavaděč, viz obr. 4.3.



Obr. 4.3: Dotaz na vytvoření zavaděče v MBR při instalačním procesu Debianu

Tímto instalace končí a při následujícím spuštění počítače se již objeví nabídkové menu, zprostředkované zavaděčem GRUB, ve kterém je volba spuštění nainstalovaného systému.

### 4.3 První spuštění

Po prvním spuštění a přihlášení se zobrazí pouze příkazový řádek. Pokud nebyla konfigurace sítě úspěšná při instalaci, je nutné ji přenastavit. Soubor, obsahující informace následně přiřazené síťovým zařízením naleznete v adresáři `/etc/network` a jmenuje se **interfaces**. K jeho úpravě poslouží editor **vim**:

```
vi /etc/network/interfaces
```

Je nutné v tomto souboru upravit nastavení ekvivalentně k:

```
auto eth0
iface eth0 inet static
    address 192.168.1.1
    netmask 255.255.255.0
```

Jednotlivé řádky znamenají, že se při spuštění zapne zařízení eth0, kterému bude nastavena statická adresa 192.168.1.1. s maskou sítě 255.255.255.0. Pokud je v síti přiřazování adres realizováno pomocí DHCP serveru, stačí napsat:

```
auto eth0
iface eth0 inet dhcp
```

Po zprovoznění internetu je již další počínání otázkou instalace potřebných programů, zajišťujících funkce serveru. K tomu je možné použít správce balíčků **apt**, který slouží zejména k usnadnění práce při administraci systému Debian.

## 5 FIREWALL

### 5.1 Úvod

Tato kapitola je věnována implementaci firewall, vytvořené pomocí programu IPTables. Firewall (česky ohnivá stěna) slouží k zabezpečení stanice (v tomto případě routeru a jeho podsítě) proti útokům zvenčí. Snahou bylo vytvořit skript, se kterým se bude snadno manipulovat, přesto si však zachová velmi komplexní a kvalitní úroveň zabezpečení.

### 5.2 IPTables

Program IPTables je paketovým filtrem, sloužícím k nastavení směrovacích tabulek v OS Linux. Funguje na principu zadávání příkazů, které pracují s pakety. Tento program je součástí samotného jádra již od verze 2.4. Odpadá tak nutnost instalačního procesu.

### 5.3 Prerekvizity

Konfigurace se provádí z prostředí příkazové řádky zadáváním jednotlivých příkazů. Jelikož je tato metoda časově náročná a při případném výpadku serveru by se ztratila takto vytvořená nastavení, je mnohem jednodušší vytvoření skriptu.

V `/etc/init.d/` vytvoříme soubor „firewall“, do kterého napíšeme námi potřebné příkazy. Přejdeme do konzole a napíšeme: **update-rc.d firewall defaults 25**, čímž se nám zavede obsah souboru „firewall“ do spouštěcí části Linuxu (rc.d) na 25 pozici.

### 5.4 Konfigurace

Jak již bylo řečeno, zapisujeme pravidla do řádků v souboru „firewall“. Celý soubor je v příloze A.

#### 5.4.1 Popis skriptu

Nejznámější zabezpečovací politikou je vycházet z uzavřeného stavu, což zajišťuje trojice řádků:

```
$IPTABLES -P INPUT DROP
$IPTABLES -P OUTPUT DROP
$IPTABLES -P FORWARD DROP
```

Pokud by skript obsahoval pouze tyto tři pravidla, zahazoval by veškeré příchozí i odchozí pakety. Takto nastavený router by však ztrácel smysl, a proto povolíme směrování paketů z a do vnitřní sítě.

```
$IPTABLES -A POSTROUTING -t nat -o $INET_IFACE -j SNAT --to $INET_IP
$IPTABLES -A FORWARD -i eth1 -s 192.168.1.2 -m mac
--mac-source 00:16:E6:75:53:7A -j ACCEPT
$IPTABLES -A FORWARD -i eth1 -o eth2 -j ACCEPT
$IPTABLES -A FORWARD -i eth2 -o eth1 -m state
--state ESTABLISHED,RELATED -j ACCEPT
```

První řádek zajišťuje překlad z IP adresy vnitřní sítě na adresu vnějšího rozhraní. Na pohled to tedy vypadá, že komunikuje pouze router, nikoliv za ním ukryté počítače.

Druhým řádkem povolíme komunikaci z vnější sítě počítači, který má současně IP adresu 192.168.1.2 a fyzickou MAC adresu 00:16:E6:80:50:7A. Pokud by splňoval pouze jedno z těchto pravidel, nebude mu povolena komunikace.

Následuje samotné povolení komunikace z vnitřní sítě do sítě vnější.

Poslední řádek zajišťuje příjem odpovědí z vnější sítě, jejichž stav spojení je již navázán.

## Řetězec INPUT

Pravidla v řetězci INPUT slouží k definování pravidel s určitými pakety na vstupu ať už z pohledu z internetu, tak z vnitřní sítě.

První příkaz zajišťuje povolení příchozích TCP spojení na portu 22 přes rozhraní eth2. Obdobně se vytvářejí pravidla pro další příchozí spojení, která chceme přijímat. Druhý řádek umožňuje přijímat severu žádosti příkazu ping a následně na ne odpovídat. Pokud byste tedy chtěli server "zneviditelnit", stačí nepovolovat odezvu na příkaz ping.

```
$IPTABLES -A INPUT -i eth2 -p TCP --dport 22 -j ACCEPT
$IPTABLES -A INPUT -i eth2 -p ICMP -j ACCEPT
```

Velmi užitečným zabezpečením je povolování vstupu do sítě na základě kombinace IP a MAC adresy. Tyto dva řádky nejprve vytvoří pravidlo, které udává vhodnou kombinaci adres a následně zakáže celý rozsah, tudíž ty počítače, které nevyhovují nebudou komunikovat.

```
$IPTABLES -A INPUT -m mac --mac-source 00:16:E6:80:50:7A
-i eth1 -s 192.168.1.10 -j ACCEPT
$IPTABLES -A INPUT -i eth1 -m iprange
--src-range 192.168.1.2-192.168.1.254 -j DROP
```

Následující dva řádky povolují odchozí pakety z počítače ve vnitřní síti do jeho síťového okolí a do sítě internet.

```
$IPTABLES -A INPUT -i eth1 -d 192.168.1.1/32 -j ACCEPT
$IPTABLES -A INPUT -i eth1 -d 84.242.66.121 -j ACCEPT
```

Pokud komunikuje někdo ze sítě internet s někým uvnitř naší sítě a toto spojení je již ve stavu navázaném nebo spojeném, povolíme tyto pakety. Ostatní komunikace bude zakázána.

```
$IPTABLES -A INPUT -d 84.242.66.121 -m state
--state ESTABLISHED,RELATED -j ACCEPT
```

### Řetězec OUTPUT

Pro odchozí pakety většinou nebývají téměř žádná pravidla. Pouze se povolí odchozí traffic z vnitřní a vnější IP adresy. Čili:

```
$IPTABLES -A OUTPUT -s $LAN_IP -j ACCEPT
$IPTABLES -A OUTPUT -s $INET_IP -j ACCEPT
```

## 6 SAMBA SERVER

### 6.1 Úvod

Samba je jeden z nejdůležitějších serverů na síti, protože umožňuje vzájemnou komunikaci mezi počítači s operačními systémy Linux, Solaris, BSD, MAC OS a samozřejmě Windows. Výhodou těchto serverů je bezpečnost, stabilita, nízká hardwarová náročnost a samozřejmě potěší i to, že je tento program zdarma. Tyto důvody vedou k tomu, že je tento server často používán i v sítích, které jsou složeny výhradně ze stanic s OS Windows. Komunikace probíhá na SMB protokolu, který se nad TCP používá ke sdílení souborů a tiskáren a nad UDP pro prohlížení adresářové struktury.

Z hlediska bezpečnosti je možné použít jeden ze dvou zabezpečovacích módů:

**[User level]** - je základním zabezpečením samby a využívá uživatelská přístupová práva k jednotlivým souborům na serveru. Každý uživatel pak po přihlášení dostane UID číslo, které používá k veškeré komunikaci se serverem.

**[Share level]** - na rozdíl od user levelu probíhá zabezpečení na úrovni sdílených prostředků. Každý tento sdílený prvek může mít nastaveno heslo, po jehož zadání má klient přístup ke všem souborům, které prostředek obsahuje.

### 6.2 Instalace

Sambu lze nainstalovat buď z balíčků, které jsou dostupné pro většinu UNIXových či Linuxových distribucí, nebo zkompileovat a nainstalovat ze zdrojových kódů dostupných na stránce [samba.org](http://samba.org).

### 6.3 Konfigurace serveru

Server se nastavuje v konfiguračním souboru **smb.conf**. Obsahuje základní nastavení vlastností Samby a specifikuje sdílené složky.

### 6.4 smb.conf

Nastavení Samba serveru s několika rozšiřujícími prvky:

```
[global]
  workgroup = SKUPINA
  server string = Thunder's server
  unix charset = ISO8859-2
```

```

security = user
dos charset = CP852
log file = /var/log/samba/log.%m
max log size = 50
dns proxy = no
hosts allow = 192.168.1.1/32
domain logons = yes
encrypt passwords = true
passdb backend = tdbsam #druh databáze s hesly
invalid users = root #zakázání přístupu uživateli root
unix password sync = yes #synchronizace hesel s databází passwd v linuxu
local master = yes #tato samba je dominantni v siti
interfaces = eth1 #definuje rozhraní, ze kterého je přístup do sítě
bind interfaces only = yes #přístup pouze z definovaných rozhraní

```

[homes]

```

comment = Home Directories
browseable = no
readable = yes
writable = yes
create mask = 0777
directory mask = 0777

```

kde:

[**global**] - obsahuje základní konfiguraci serveru, skupinu, jméno, znakovou sadu,...

[**homes**] - konfigurace domovských adresářů jednotlivých uživatelů

Obdobně se vytvářejí další adresáře přístupné určitým uživatelům.

## 6.5 Správa serveru

Díky velkému rozšíření, se Samba server dočkal i možnosti správy přes webové rozhraní. K tomu slouží hned několik programů, jako třeba SWAT nebo Webmin.

### 6.5.1 SWAT

Program SWAT, který je zobrazen na obr.6.1, je vyvíjen stejnou organizací jako Samba. Překvapivé je, že umožňuje v mnoha směrech lepší nastavení, než zdrojový soubor smb.conf. Ukázka možností nastavení je na obr.6.2. Po nainstalování je SWAT

nastaven na výchozí adresu "http://localhost:901". Z toho vyplývá, že SWAT nepodporuje šifrovaný přenos. To je při přenášení hesla pro uživatele root, se kterým SWAT pracuje velmi nebezpečné, proto je vhodné ho HTTPS naučit. Tomu bude věnována kapitola 6.5.2.

Při práci je nutné dát si pozor při ukládání jakýchkoliv úprav. Ukládá se kompletní aktuální konfigurace, takže pokud máme vytvořený konfigurační skript a chceme změnit jen jednu část, bude celý smazán a vytvořen znova podle zadaných pravidel.

Další úskalí skýtá vytvoření nového sdíleného adresáře. Program ho nedokáže vytvořit a je tedy nutné ho vytvořit manuálně a následně k němu teprve nastavit požadované akce.

Výhodou správy přes SWAT je velmi přehledná tabulka se spuštěnými démony a přihlášenými uživateli, což je patrné z obr.6.3. S démony lze provádět standardní akce jako start, stop a restart.

### 6.5.2 SWAT a SSL

Protože SWAT nemá podporu šifrovaného přenosu, je nutné použít program **stunnel**, který tvoří virtuální šifrovaný tunel nad spojením. Na jedné straně pracuje šifrovaně přes HTTPS, informace z něj dešifruje a posílá cílovému programu. K přenosu je samozřejmě potřeba vytvoření certifikátů pomocí OpenSSL. Ty se generují pomocí příkazu:

```
/usr/bin/openssl req -new -x509 -days 365 -nodes -config  
/usr/share/doc/stunnel/examples/stunnel.cnf  
-out /etc/stunnel/stunnel.pem -keyout /etc/stunnel/stunnel.pem
```

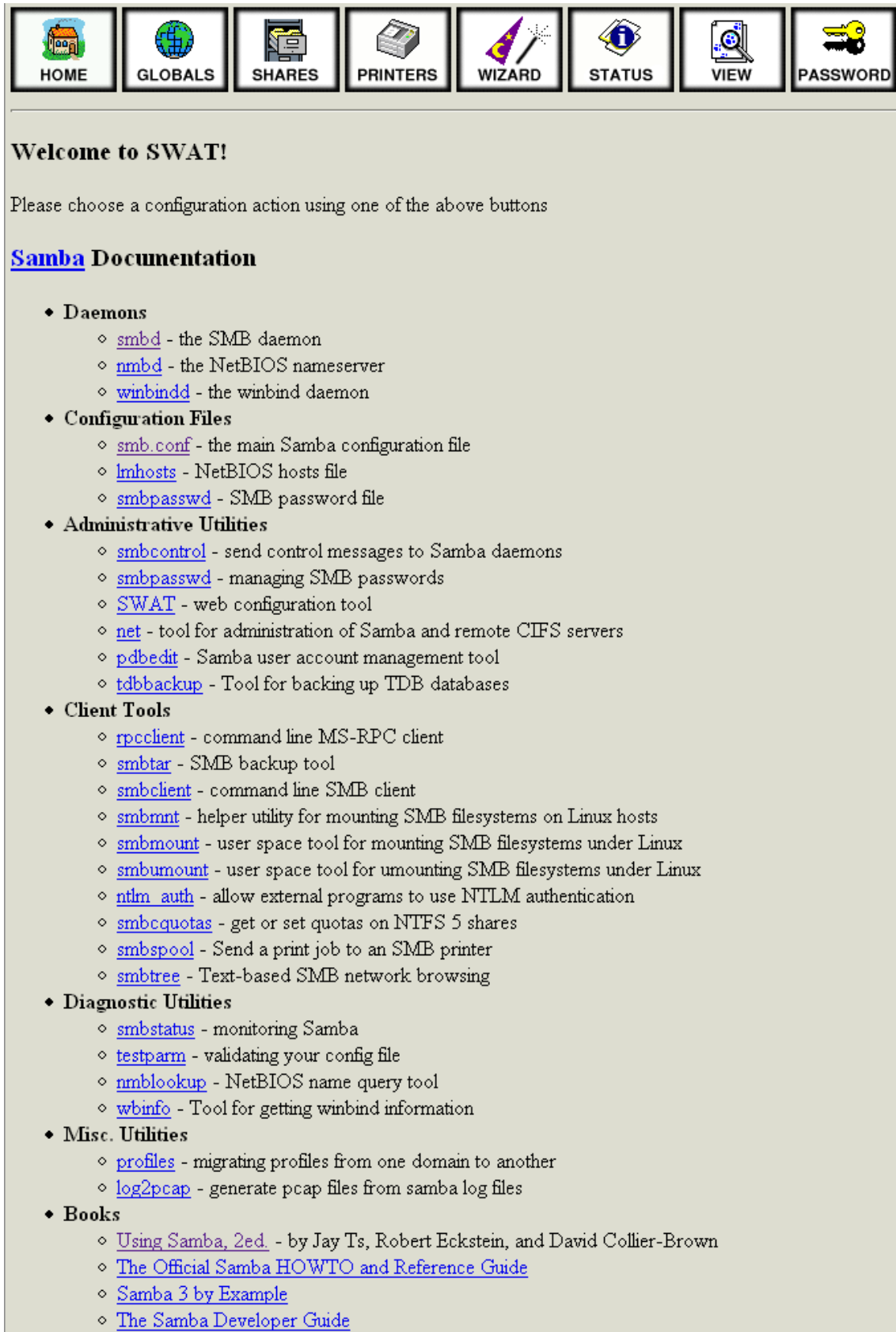
Dalším krokem je vymazání odkazu na SWAT z démona inetd nebo xinetd, podle toho, který je používán. Následuje už jen spuštění stunnelu s následujícími parametry:

```
stunnel -p /etc/stunnel/stunnel.pem -d 901 -l /usr/sbin/swat swat
```

### 6.5.3 Webmin

Program Webmin je určený ke kompletní správě systému přes webové rozhraní, což je ukázáno na obr.6.4. Používá rozšíření, která jsou nadefinována pro správu jednotlivých aplikací. Při správě samba serveru se velice osvědčil, protože umožňuje stejně jako SWAT velmi snadnou a rozsáhlou správu.

Bezproblémové vytváření adresářů ke sdílení, stejně tak i tiskáren, patří k hlavním výhodám oproti SWATu. Na obr.6.5 a obr.6.6 jsou pak zobrazeny možné nastavení severu pro Linux i klientské stanice na Windows.



The screenshot shows the SWAT web interface. At the top, there is a navigation bar with eight buttons: HOME (house icon), GLOBALS (globe icon), SHARES (folder icon), PRINTERS (printer icon), WIZARD (magic wand icon), STATUS (info icon), VIEW (magnifying glass icon), and PASSWORD (key icon). Below the navigation bar, the text reads "Welcome to SWAT!" followed by "Please choose a configuration action using one of the above buttons". A section titled "Samba Documentation" contains a list of links organized into categories:

- ◆ **Daemons**
  - ◇ [smbd](#) - the SMB daemon
  - ◇ [nmbd](#) - the NetBIOS nameserver
  - ◇ [winbindd](#) - the winbind daemon
- ◆ **Configuration Files**
  - ◇ [smb.conf](#) - the main Samba configuration file
  - ◇ [lmhosts](#) - NetBIOS hosts file
  - ◇ [smbpasswd](#) - SMB password file
- ◆ **Administrative Utilities**
  - ◇ [smbcontrol](#) - send control messages to Samba daemons
  - ◇ [smbpasswd](#) - managing SMB passwords
  - ◇ [SWAT](#) - web configuration tool
  - ◇ [net](#) - tool for administration of Samba and remote CIFS servers
  - ◇ [pdbedit](#) - Samba user account management tool
  - ◇ [tdbbackup](#) - Tool for backing up TDB databases
- ◆ **Client Tools**
  - ◇ [rpcclient](#) - command line MS-RPC client
  - ◇ [smbtar](#) - SMB backup tool
  - ◇ [smbclient](#) - command line SMB client
  - ◇ [smbmnt](#) - helper utility for mounting SMB filesystems on Linux hosts
  - ◇ [smbmount](#) - user space tool for mounting SMB filesystems under Linux
  - ◇ [smbumount](#) - user space tool for unmounting SMB filesystems under Linux
  - ◇ [ntlm\\_auth](#) - allow external programs to use NTLM authentication
  - ◇ [smbcquotas](#) - get or set quotas on NTFS 5 shares
  - ◇ [smbspool](#) - Send a print job to an SMB printer
  - ◇ [smbtree](#) - Text-based SMB network browsing
- ◆ **Diagnostic Utilities**
  - ◇ [smbstatus](#) - monitoring Samba
  - ◇ [testparm](#) - validating your config file
  - ◇ [nmblookup](#) - NetBIOS name query tool
  - ◇ [wbinfo](#) - Tool for getting winbind information
- ◆ **Misc. Utilities**
  - ◇ [profiles](#) - migrating profiles from one domain to another
  - ◇ [log2pcap](#) - generate pcap files from samba log files
- ◆ **Books**
  - ◇ [Using Samba, 2ed.](#) - by Jay Ts, Robert Eckstein, and David Collier-Brown
  - ◇ [The Official Samba HOWTO and Reference Guide](#)
  - ◇ [Samba 3 by Example](#)
  - ◇ [The Samba Developer Guide](#)

Obr. 6.1: Úvodní menu správy samba serveru přes SWAT

## Global Parameters

Current View Is:  Basic  Advanced  
 Change View To:

### Base Options

<a href="#">Help</a>	dos charset	CP852	<input type="button" value="Set Default"/>
<a href="#">Help</a>	unix charset	ISO8859-2	<input type="button" value="Set Default"/>
<a href="#">Help</a>	display charset	LOCALE	<input type="button" value="Set Default"/>
<a href="#">Help</a>	workgroup	SKUPINA	<input type="button" value="Set Default"/>
<a href="#">Help</a>	realm		<input type="button" value="Set Default"/>
<a href="#">Help</a>	netbios name	DEBIAN	<input type="button" value="Set Default"/>
<a href="#">Help</a>	netbios aliases		<input type="button" value="Set Default"/>
<a href="#">Help</a>	netbios scope		<input type="button" value="Set Default"/>
<a href="#">Help</a>	server string	Thunder's server	<input type="button" value="Set Default"/>
<a href="#">Help</a>	interfaces		<input type="button" value="Set Default"/>
<a href="#">Help</a>	bind interfaces only	No <input type="button" value="Set Default"/>	

### Security Options

<a href="#">Help</a>	security	USER <input type="button" value="Set Default"/>	
<a href="#">Help</a>	auth methods		<input type="button" value="Set Default"/>
<a href="#">Help</a>	encrypt passwords	Yes <input type="button" value="Set Default"/>	
<a href="#">Help</a>	update encrypted	No <input type="button" value="Set Default"/>	
<a href="#">Help</a>	client schannel	Auto <input type="button" value="Set Default"/>	
<a href="#">Help</a>	server schannel	Auto <input type="button" value="Set Default"/>	
<a href="#">Help</a>	allow trusted domains	Yes <input type="button" value="Set Default"/>	
<a href="#">Help</a>	map to guest	Never <input type="button" value="Set Default"/>	
<a href="#">Help</a>	null passwords	No <input type="button" value="Set Default"/>	
<a href="#">Help</a>	obey pam restrictions	Yes <input type="button" value="Set Default"/>	
<a href="#">Help</a>	password server	*	<input type="button" value="Set Default"/>
<a href="#">Help</a>	smb passwd file	/etc/samba/smbpasswd	<input type="button" value="Set Default"/>
<a href="#">Help</a>	private dir	/etc/samba	<input type="button" value="Set Default"/>
<a href="#">Help</a>	passwd backend	tdbsam	<input type="button" value="Set Default"/>
<a href="#">Help</a>	algorithmic rid base	1000 <input type="button" value="Set Default"/>	
<a href="#">Help</a>	root directory		<input type="button" value="Set Default"/>
<a href="#">Help</a>	guest account	nobody	<input type="button" value="Set Default"/>
<a href="#">Help</a>	enable privileges	Yes <input type="button" value="Set Default"/>	

Obr. 6.2: Možností nastavení serveru SWAT

**Server Status**

Auto Refresh

Refresh Interval:

version: 3.0.23d

smbd: running

nmbd: running

winbindd: running

**Active Connections**

PID	Client	IP address	Date	Kill
26600	thunder	192.168.1.5	Fri Jan 26 09:23:11 2007	<input type="button" value="X"/>
26600	thunder	192.168.1.5	Fri Jan 26 09:23:14 2007	<input type="button" value="X"/>

**Active Shares**

Share	User	Group	PID	Client	Date
IPC\$	sirthunder	sirthunder	26600	thunder	Fri Jan 26 09:23:11 2007
linux server	sirthunder	sirthunder	26600	thunder	Fri Jan 26 09:23:14 2007

**Open Files**

PID	Sharing	R/W	Oplock	File	Date
26600	1000	DENY_DOS	RDWR	NONE	Fri Jan 26 09:23:14 2007

Obr. 6.3: Zobrazení stavu démonů a aktuálně přihlášených uživatelů

Samba Windows File Sharing
Samba version 3.023


[Create a new file share.](#)  
 [Create a new printer share.](#)  
 [Create a new copy.](#)  
 [View all connections.](#)


Share Name	Path	Security
<a href="#">homes</a>	<i>All Home Directories</i>	Read/write to all known users
<a href="#">sdilenej plac</a>	/home/sdileno	Read/write to all known users
<a href="#">linux server</a>	/	Read/write to all known users


[Create a new file share.](#)  
 [Create a new printer share.](#)  
 [Create a new copy.](#)  
 [View all connections.](#)


---


### Global Configuration


  
[Unix Networking](#)


  
[Windows Networking](#)


  
[Authentication](#)

  
[Windows to Unix Printing](#)

  
[Miscellaneous Options](#)


  
[Winbind Options](#)


  
[File Share Defaults](#)


  
[Printer Share Defaults](#)


---


### Samba Users


  
[Edit Samba users and passwords](#)

  
[Convert Unix users to Samba users](#)

  
[Configure automatic Unix and Samba user synchronisation](#)

  
[Add and edit Samba groups](#)

  
[Configure automatic Unix and Samba group synchronisation](#)

  
[Bind to Domain](#)

---

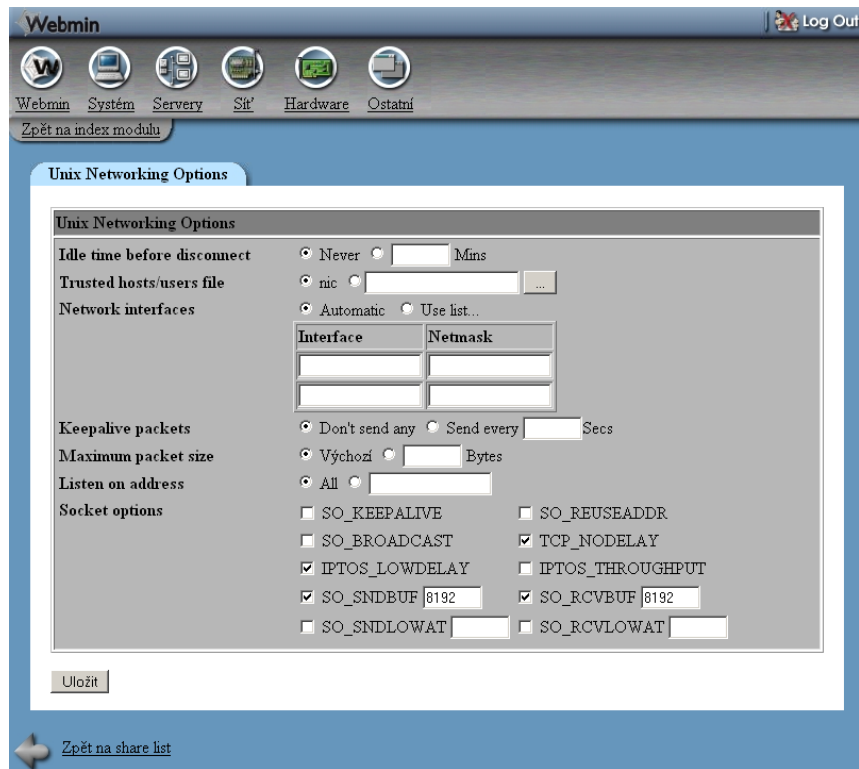
Restart Samba Servers

Stop Samba Servers

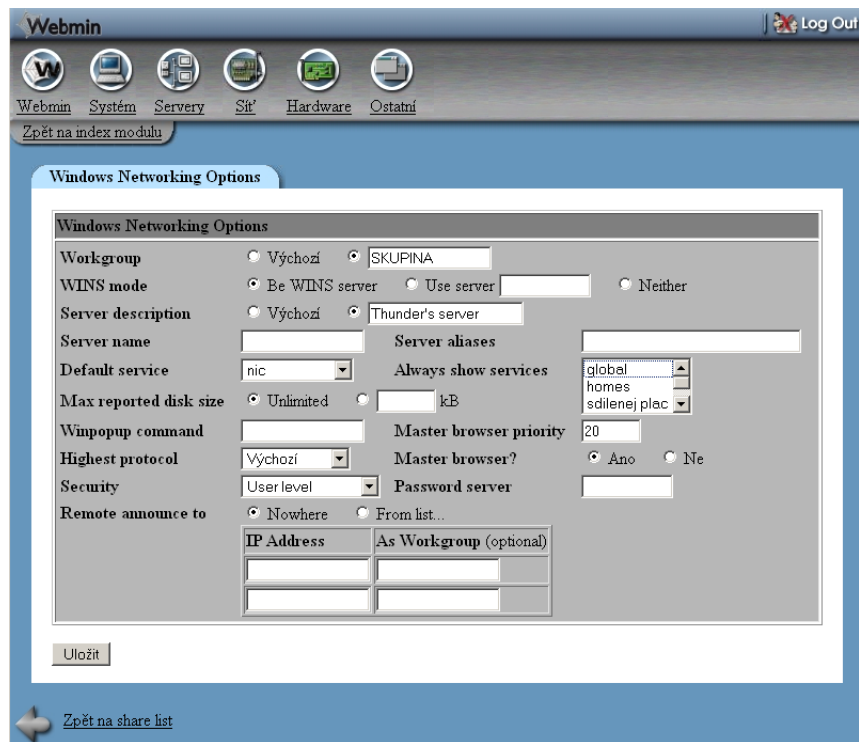
Click this button to restart the running Samba servers on your system. This will force the current configuration to be applied. This will also disconnect any connections to the server, so if you do not want the current configuration to be applied immediately you should just wait 1 minute until Samba reloads the configuration automatically.

Click this button to shut down the running Samba servers on your system. All currently logged in users will be forcibly disconnected.

Obr. 6.4: Úvodní menu správy samba serveru Webmin



Obr. 6.5: Možností nastavení serveru Webmin



Obr. 6.6: Nastavení serveru pro upřesnění vlastností sítě

## 7 DHCP SERVER

### 7.1 Úvod

DHCP (Dynamic Host Configuration Protokol) je aplikační protokol z rodiny TCP/IP, který se primárně používá pro automatické přidělování IP adres koncovým uživatelským stanicím v síti podle určitých nastavených pravidel. Současně s IP adresou však server může posílat klientské stanici například informace o DNS serverech, masce sítě, adrese nejbližšího směrovače, případně i adrese doporučených NTP, WINS, SMTP serverů. Přiřazování probíhá buď pomocí vazby IP adresa - MAC adresa, nebo přiřazuje jednotlivým připojeným uživatelům náhodné adresy z povoleného rozsahu. Jeho vývojem a správou se zabývá organizace ISC (Internet System Consortium), na jejíž stránkách ([www.isc.org](http://www.isc.org)) lze balík DHCP stáhnout.

### 7.2 Popis funkce

1. Do sítě se připojí nová stanice, která vyšle broadcastem paket DHCPDISCOVER.
2. Na ten odpoví DHCP server paketem DHCPOFFER, který nese nabídku IP adres. Pokud je v síti více DHCP serverů, odpovídají všichni.
3. Klient si z obdržných nabídek vybere jednu IP adresu a zažádá o ni požadavkem DHCPREQUEST.
4. Příslušný server mu danou IP adresu může buď přiřadit a odpovědět DHCPACK, nebo zamítnout její nepřirazení s odpovědí DHCPNACK.
5. Pokud ji přiřadil, klient může automaticky IP adresu využívat. Pokud jeho žádost byla zamítnuta, opět posílá do sítě paket DHCPDISCOVER.
6. Platnost takto přiřazené IP adresy je časově omezena (tzv. Lease time). Pokud klient v daném časovém úseku neprodlouží platnost dané IP adresy, je mu odebrána a nemůže ji nadále používat.

### 7.3 Výhody použití DHCP

- Uživatelé na PC nemusejí nic nastavovat.
- Zabraňuje konfliktu IP adres (dvě stejné v jedné síti).
- Jednoduchá obsluha z pozice správce, která na uživatele nemá vliv.

## 7.4 Instalace Serveru

Pro instalaci se používá balík, který je obsažen přímo v distribuci Debian. Instalace se provádí příkazy:

```
apt-get install dhcp a apt-get install dhcp-client.
```

## 7.5 Konfigurace serveru

Konfigurace se provádí v souboru **dhcpd.conf** a skýtá mnoho možných nastavení.

### dhcpd.conf

Pro příklad uvádím použitý konfigurační soubor s vysvětlivkami:

```
#Doménové jméno
option domain-name "sirthunder.cz";

#Maska sítě
option subnet-mask 255.255.255.0;

#Broadcast adresa
option broadcast-address 192.168.1.255;

#Přiřazování adres DNS serverů
option domain-name-servers 81.27.192.33, 81.27.192.97;

#Základní doba propůjčování adres
default-lease-time 3600;
#Maximální propůjčovací doba
max-lease-time 7200;

#Definování rozsahu podsítě a masky
subnet 192.168.1.0 netmask 255.255.255.0 {
    option routers 192.168.1.1;
    range 192.168.1.5 192.168.1.11;
    deny unknown-clients;
}

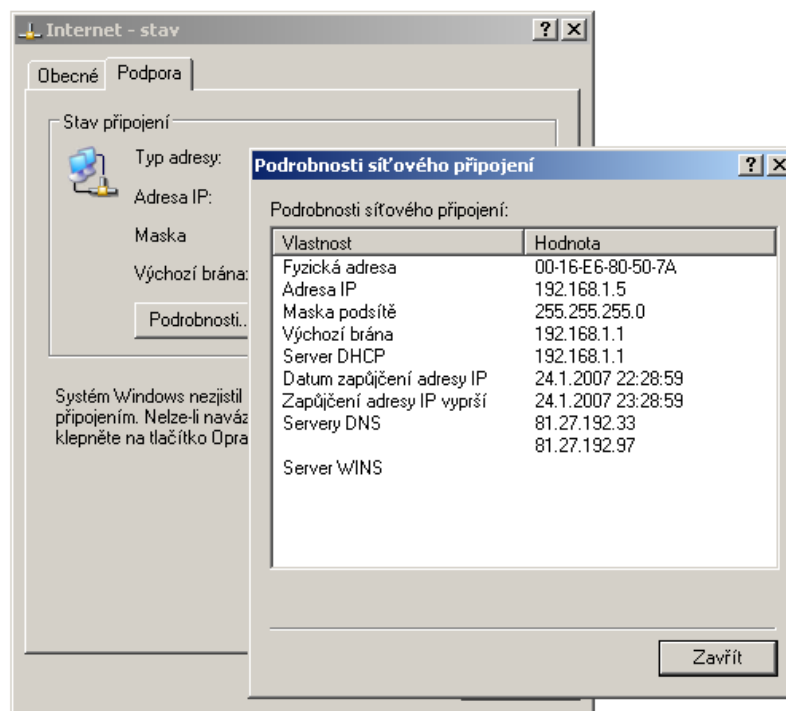
#Definování uživatelů
host user1 {
    hardware ethernet 00:00:0X:XA:05:XB;
```

```

    fixed-address 192.168.1.5;
  }
host user2 {
    hardware ethernet 00:00:0X:XA:05:XA;
    fixed-address 192.168.1.6;
  }

```

V systému Windows pak po načtení informací z DHCP serveru dostaneme konfiguraci sítě uvedenou na obr.7.1.



Obr. 7.1: Přiřazení dat z DHCP serveru síťové kartě ve Windows

## 8 FTP SERVER

### 8.1 Úvod

Hojně využívanou možností jak sdílet data, je použití protokolu FTP (File Transfer Protocol). Tento protokol patří mezi historicky nejstarší internetové protokoly a díky tomu má velmi širokou podporu. Využívá se k přenosu souborů, neboť pro něj existuje podpora mezi různými operačními systémy. Pracuje vždy ve spojení klient - server. Pro navázání spojení a zadávání příkazů se ve výchozím stavu používá síťový port 21, pro přenos je pak určen port 20. Porty však není problém změnit a zvýšit tak zabezpečení proti útokům.

Podporována je také autentizace jménem a heslem, ovšem velkou nevýhodou je přenášení těchto informací ve formě prostého textu, což znamená, že při odposlechu konkrétního kanálu se k heslům dříve či později útočník dostane.

Existuje však již náprava využívající šifrovaného přenosu. To ale musí samozřejmě podporovat i klientská stanice. Důležitým faktorem při výběru FTP serveru je bezesporu rozsáhlost konfigurace, čili nabídka požadovaných funkcí. Je tedy nutné si promyslet, kolik uživatelů budeme v síti mít, jak chceme spravovat jejich uživatelské účty, jestli chceme použít šifrovaný či nešifrovaný přenos, nebo je dokonce provozovat souběžně.

### 8.2 ProFTPD

Program ProFTP plní funkcí FTP serveru s mnoha rozšiřujícími vlastnostmi a také s rozsáhlou možností konfigurace. Je velmi intuitivní, podporuje šifrovaný přenos a správa jednotlivých uživatelů se dá kupříkladu provádět přes MySQL databázi v phpMyAdmin.

### 8.3 Požadavky

Pro instalaci je potřeba stáhnout instalační balík `proftpd-1.3.0.tar.gz` ze stránek výrobce (<http://www.proftpd.org/>), nebo použít balík, který je obsažen přímo v distribuci Debian.

### 8.4 Instalace Serveru

Instalace z balíku poskytovaného vývojáři probíhá v krocích `configure`, `make` a `make install`. Přičemž u příkazu `configure` je možné použít mnoho specifických

parametrů ovlivňujících průběh instalace a počet instalovaných rozšíření.

Mnou použité nastavení při kompilaci vypadalo následovně:

```
./configure --with-modules=mod_sql:mod_sql_mysql:mod_tls,
```

Takto specifikovaný příkaz umožňuje použití TLS<sup>1</sup> a propojení s MySQL databází.

## 8.5 Šifrovaný přenos

Pro zprovoznění šifrovaného přenosu je nutné vygenerovat patřičné certifikáty. Pro jejich generování se využívá program OpenSSL a jeho přidružené knihovny.

Pro vygenerování je možné použít příkaz:

```
openssl req -new -x509 -days 365 -nodes -out ftpd-rsa.pem -keyout ftpd-rsa-key.pem
```

kde:

openssl označuje název programu pro generování klíčů

req - práce s certifikační žádostí

new - vytvoří novou certifikační žádost

x509 - vytváří se tzv. certifikát, podepsaný sám sebou (self-signed)

days - certifikát bude platit daný počet dnů

nodes - soukromý klíč nebude zašifrován

out - certifikát se zapisuje do souboru ftpd-rsa.pem

keyout - klíč zapsat do souboru ftpd-rsa-key.pem

Vzniklé soubory je nutné někde nakopírovat a zavést na ně v konfiguračním souboru odkaz.

## 8.6 Konfigurace serveru

Všechna nastavení FTP serveru se provádějí v konfiguračním souboru **proftpd.conf**, odpadá tedy procházení mnoha adresářů a vyhledávání svázaných souborů.

## 8.7 Proftpd.conf

Konfigurační soubor naleznete v příloze B.

---

<sup>1</sup>Transport Layer Security = používá se při šifrování komunikace mezi počítači

### 8.7.1 Test spojení

Pro test bylo použito programu IglooFTP v 3.9.

```
Connecting to 84.242.66.121 on Port 21 ... (Try 1)
Connected to FTP server. Waiting for welcome message ...
220 Welcome to the FTP server. Please login...
AUTH TLS
234 AUTH TLS successful
Starting SSL/TLS negotiation ...
The server certificate issued by ProFTP to ProFTP was found in
the trusted certificates directory and accepted.
SECURE CONNECTION ESTABLISHED.
PBSZ 0
200 PBSZ 0 successful
Protected buffer size set to 0 bytes.
PROT P
200 Protection set to Private
Encryption of file transfers activated.
USER breta
331 Password required for breta.
PASS xxx
230 User breta logged in.
Login successful.
```

## 9 WEB SERVER

### 9.1 Úvod

V 90. letech začal vývoj, v dnešní době nejpoužívanějšího webového serveru poskytujícího internetovým klientům (Internet Explorer, Firefox, Opera, . . .) odpovědi na jejich dotazy. Pracuje na protokolu HTTP (HyperText Transfer Protocol), což je síťový protokol aplikační vrstvy používaný k předávání dat mezi serverem a klienty.

### 9.2 HTTP protokol

Jak již bylo zmíněno, bezstavový protokol HTTP je postaven na principu požadavku a odpovědi a definuje pravidla při komunikaci mezi serverem a klienty. Ve výchozím nastavení běží na portu 80, což se dá ovšem lehce změnit. Jedná se bezesporu o nejrozšířenější službu v síti internet, který je aktuálně ve verzi 1.1.

### 9.3 Apache

Z produkce Apache Software Foundation, která byla založena skupinkou webmasterů v roce 1994, pochází Apache Web Server. Svou popularitu si vydobyl zejména díky open source přístupu, z čehož pak vyplývá, díky práci mnoha uživatelů, velký výběr rozšiřujících modulů a rozsáhlé možnosti zabezpečení.

#### 9.3.1 Instalace Apache

Stejně tak jako mnoho dalších programů, je i Apache obsažen ve většině distribucí. Je tedy možné ho nainstalovat z distribučních balíčků, nebo si ze zdrojových kódů, které jsou na adrese: (<http://www.apache.org/>) zkompilevat verzi přímo na míru.

Pro standardní používání však postačí verze z distribuce. Je tedy potřeba příkazem **apt-get install apache2 openssl ssl-cert libapache2-mod-php5 php5-cli php5-common php5-cgi** nainstalovat veškeré potřebné součásti. Takto nainstalovaný server pracuje s PHP verze 5 a bude umožňovat šifrovaný přenos.

#### 9.3.2 Generování certifikátu

Pro využívání podpory šifrovaného přenosu je nutné pomocí openssl vygenerovat certifikát pro apache. K tomu slouží příkaz:

```
openssl req -new -x509 -days 365 -nodes -out /etc/apache2/ssl/apache.pem  
-keyout /etc/apache2/ssl/apache.pem
```

Při generování budete dotazováni na otázky ohledně umístění serveru a jména.

V základním nastavení však Apache funguje jen na portu 80, je tedy nutné v `/etc/apache2/ports.conf` povolit sledování portu 443, čehož docílíme vložením řádku **Listen 443**. Povolení SSL pak provést příkazem **a2enmod ssl** a následně restartovat Apache pomocí `/etc/init.d/apache2 restart`.

Teď už stačí jen upravit `/etc/apache2/sites-available/default`, případně v souboru `/etc/apache2/apache2.conf` povolit šifrování jen pro určitou část stránek.

### 9.3.3 Konfigurace Apache2 serveru

Většina nastavení se provádí v souboru `/etc/apache2/apache2.conf`. Je možné nastavit například počet spuštěných serverů, uživatele, pod kterým apache bude pracovat, logování či vytváření virtuálních hostů (stránek).

Důležitou součástí jsou také moduly. Výpis aktuálně nainstalovaných se nachází v adresáři `/etc/apache2/mods-available` a jejich aktivace se provádí vytvořením závislosti příslušného modulu do adresáře `/etc/apache2/mods-enabled`.

## 9.4 MySQL

MySQL je relační databázový systém. Komunikace s databází probíhá pomocí jazyka SQL. Velkou výhodou je, že je k dispozici pod bezplatnou licenci GPL, což z ní v kombinaci s Apache a PHP činí základ většiny webových serverů.

### 9.4.1 Instalace MySQL

Databázový server MySQL se instaluje příkazem **apt-get install mysql-server**. V základním nastavení však umožňuje připojení jen z adresy localhost, pokud tedy chcete server využívat jako databázový stroj i pro jiné počítače v síti internet, je nutné zakomentovat v souboru `/etc/mysql/my.cnf` řádku **bind-address = 127.0.0.1**. Takto nainstalovaný databázový server má ještě jednu bezpečnostní trhlinu a to absence hesla uživatele root, je tedy nutné pomocí příkazu **mysqladmin -u root password heslo** heslo vytvořit.

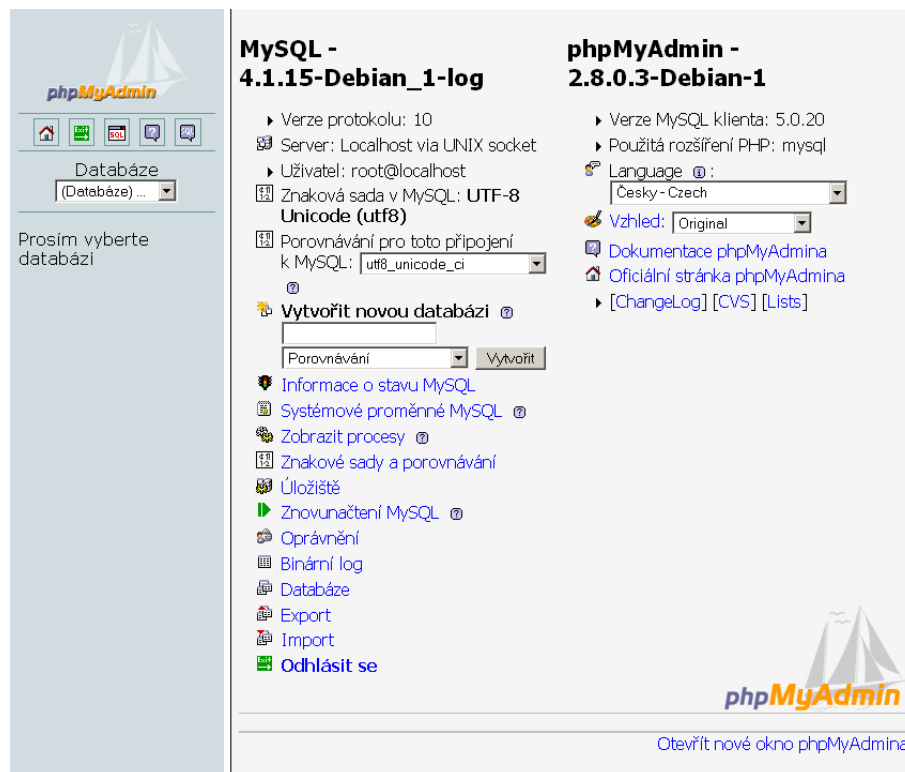
Nyní následuje propojení Apache a MySQL nainstalováním těchto balíčků:  
**apt-get install libapache2-mod-auth-mysql php5-mysql phpmyadmin**

Pro spolupráci PHP a MySQL je nutné už jen odkomentovat v souboru `/etc/php5/apache2/php.ini` řádek **extension=mysql.so**.

## 9.4.2 phpMyAdmin

Pro jednoduchou a přehlednou správu nad databázemi lze použít rozhraní phpMyAdmin. Jeho prostředí nabízí snadnou manipulaci s databázemi pomocí SQL příkazů, ať už těch, které se vykonají po kliknutí na některou položku, nebo těch, které uživatel sám vepíše do přednastaveného pole.

Jeho instalace se provádí klasickým příkazem **apt-get install phpmyadmin** a nepotřebuje žádnou konfiguraci, protože si všechny závislosti najde sám. Ukázka rozhraní je na obr.9.1



Obr. 9.1: Rozhraní phpMyAdmin

## 10 WEBMIN MONITORING PLUGIN

### 10.1 Úvod

Jak již bylo řečeno v kap.6.5.3, program Webmin slouží ke kompletní správě serveru. Podle nainstalovaných rozšíření dokáže pracovat s uživatelskými účty, spravovat firewall, DHCP server i Samba server. Z výčtu těchto možností vyplývá velké potenciální riziko, je tedy dobré povolit přístup do této aplikace jen z vnitřní sítě a nejlépe jen z několika konkrétních adres.

### 10.2 Instalace

Webmin je také, stejně jako většina monitorovacích programů, obsažen v distribučních balíčcích. Jeho instalace se tedy provádí příkazem:

```
apt-get install webmin
```

Podle požadavků na další správu je možné analogicky nainstalovat i apache, dhcpd, firewall, inetd, samba, sshd a mnoho dalších.

Neoficiálně pak vyšel i plugin sloužící k monitoringu částí routeru. Ten se jmenuje **webmin-sysstats** a je možné ho stáhnout ze stránky autora:

<http://www.gallet.info.free.fr>. Instalace se pak provádí příkazem:

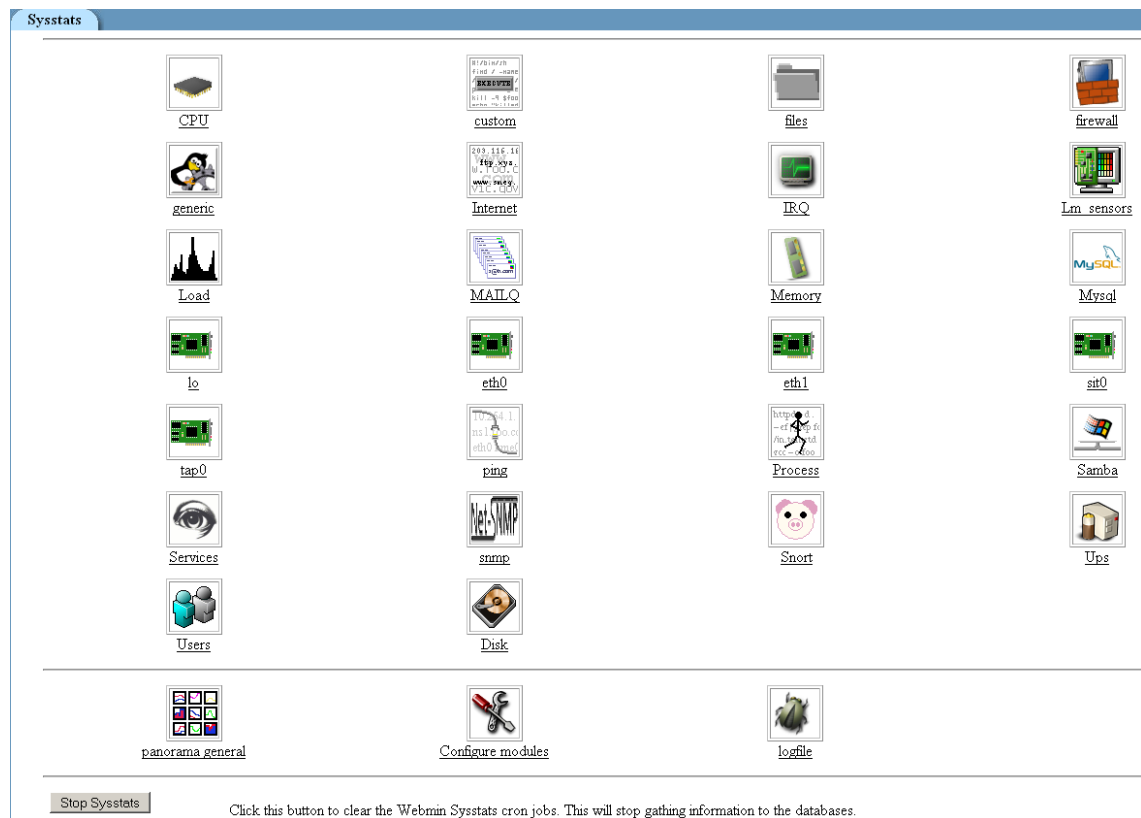
```
dpkg -i /home/webmin-sysstats-0.10.0-1-all.deb.
```

### 10.3 Konfigurace

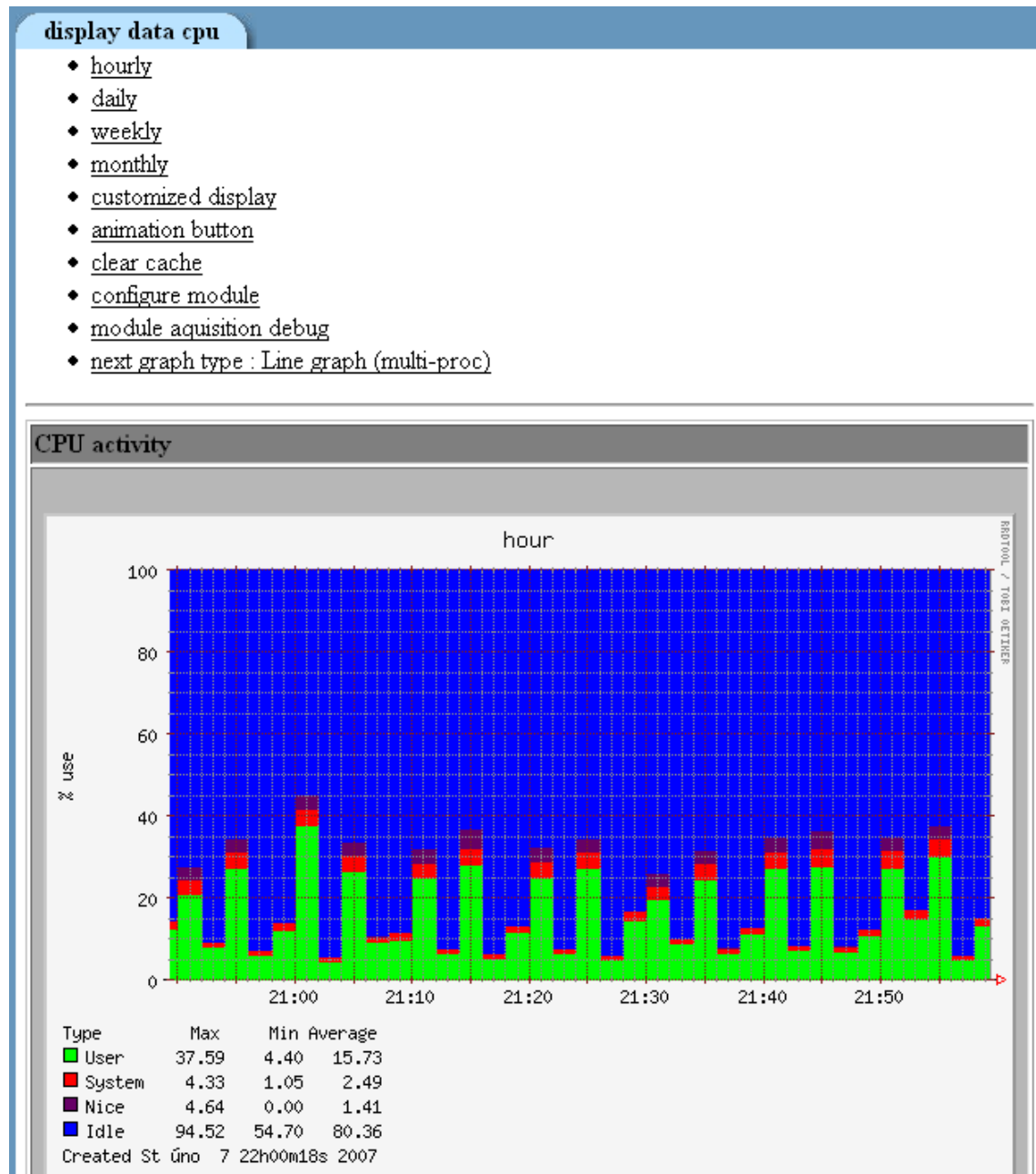
Základní konfigurace se provádí v souboru `/etc/webmin/miniserv.conf`, kde se provádí specifikace portu, na kterém Webmin poběží, cesta k hlavnímu adresáři, nastavení logování a povolení konkrétních IP adres. Ostatní konfigurace jednotlivých rozšíření probíhá přehledně přes webový prohlížeč.

### 10.4 Výsledek

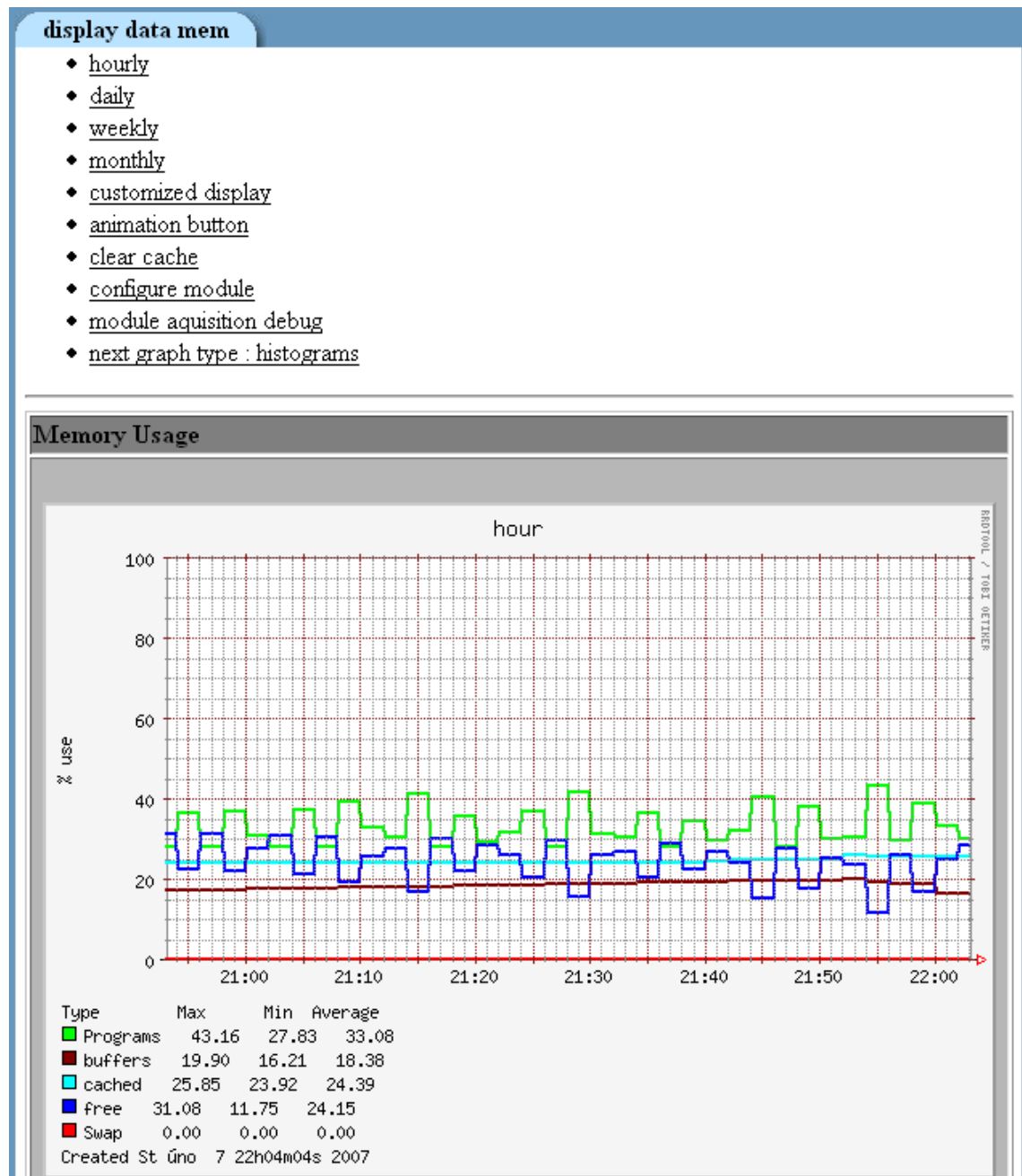
Ukázky monitoringu v tomto programu jsou na následujících obrázcích:



Obr. 10.1: Základní okno Webmin-sysstats pluginu



Obr. 10.2: Sledování vytížení procesoru pomocí Webminu



Obr. 10.3: Sledování využití paměti pomocí Webminu

## 11 IPAC-NG

### 11.1 Úvod

Jedním z nejlepších programů sloužících ke sledování přenosu v menší lokální síti je ipac-ng, který pracuje s pravidly pro iptables nebo ipchains. Je to novější verze programu ipac, který podporuje jádro 2.4. a vyšší. Mezi jeho hlavní přednosti patří malá náročnost na systémové prostředky, jednoduchá správa a možnost propojení s databázovým serverem.

### 11.2 Instalace

Instalace se v distribuci Debian provádí klasicky:

**apt-get install ipac-ng**

Pokud si chcete zkompilevat vlastní instalaci, zdrojové soubory je možné stáhnout například zde: (<http://ipac-ng.sourceforge.net/>).

### 11.3 Konfigurace

Konfigurace se provádí v souborech `/etc/ipac-ng/rules.conf` a `/etc/ipac-ng/ipac.conf`. První uvedený obsahuje data o IP adresách, které budou sledovány a vypadá například takto:

```
Incoming Total System|ipac~o|eth1|all|
Incoming Total System|ipac~fo|eth1|all|
Outgoing Total System|ipac~i|eth1|all|
Outgoing Total System|ipac~fi|eth1|all|
WWW Incoming|ipac~fi|eth1|tcp|0/0 80|
WWW Outgoing|ipac~fo|eth1|tcp||0/0 80|
SMTP Incoming|ipac~fi|eth1|tcp|0/0 25|
SMTP Outgoing|ipac~fo|eth1|tcp||0/0 25|

##### 1 --- 192.168.1.1 #####
1_download|ipac~o|eth0|all||192.168.1.1|
1_download|ipac~fo|eth0|all||192.168.1.1|
1_upload|ipac~i|eth0|all|192.168.1.1|
1_upload|ipac~fi|eth0|all|192.168.1.1|
```

Kde eth1 označuje měřený síťový adaptér, tedy výstup do sítě internet.

Soubor **ipac.conf** obsahuje informace o správci paketů (iptables/ipchains), typu výstupního souboru (podporuje i databázové servery) a cestu k souboru **rules.conf**. Vypadá tedy například následovně:

```
account agent = iptables
storage = mysql
rules file = /etc/ipac-ng/rules.conf
db name = "ipac"
db user = ""
db pass = ""
```

### 11.3.1 Vytváření grafů

Aby bylo možné vytvářet grafy a statistiky, je nutné mít k dispozici dostatek hodnot. K tomu bude použito opět Cronu, kde vytvoříme v adresáři **/etc/cron.d** soubor **ipac** a ten bude obsahovat:

```
@reboot          root /usr/local/sbin/fetchipac -S
*/5 * * * * root (/usr/local/sbin/fetchipac >>/var/log/fetchipac.log)
```

kde první část zajišťuje zapnutí ipac-ng po restartování počítače a druhá vytváří statistiky každých 5 minut.

Pro generování grafů je dobré vytvořit další soubor, například **grafy**, a do něj nahrát následující:

```
*/5 * * * * root /usr/local/sbin/ipacsum -s 23h59m55s --gif \
/var/www/ipac/day --gif-normalize 8 --gif-average-curve 20 \
--gif-width 900 --gif-height 300 >/dev/null
1 * * * * root /usr/local/sbin/ipacsum -s 6D23h59m55s --gif \
/var/www/ipac/week --gif-normalize 8 --gif-average-curve 20 \
--gif-width 900 --gif-height 300 >/dev/null
@daily root /usr/local/sbin/ipacsum -s 30D23h59m55s --gif \
/var/www/ipac/month --gif-normalize 8 --gif-average-curve 10 \
--gif-width 900 --gif-height 300 >/dev/null
```

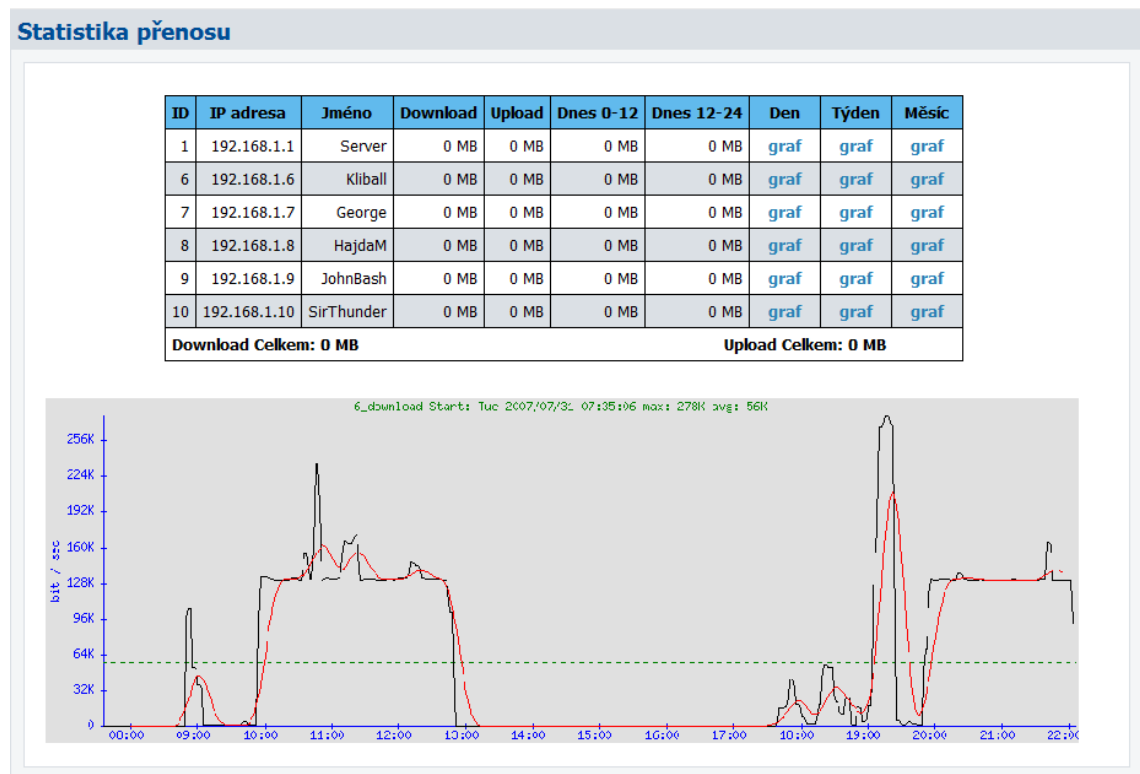
První část skriptu generuje graf pro den, druhá pro týden a třetí pro měsíc.

### 11.3.2 Propojení s MySQL

Jak již bylo uvedeno dříve, ipac-ng dokáže data ukládat na databázový server. Následná manipulace s daty je už jen v režii SQL příkazů.

## 11.4 Výsledek

Výsledná konfigurace může vypadat například jako obr.11.1.



Obr. 11.1: Statistika přenosu a grafy v ipac-ng

Bohužel v době psaní diplomové práce nebyl vydán žádný patch pro tento program, který by spolupracoval s verzí jádra 2.6.18-4-686 SMP a verzí iptables v1.3.8. Program tedy nepracoval korektně.

## 12 MRTG

### 12.1 Úvod

Jedním z nejsnadnějších a zároveň nejpoužívanějších programů určených ke sledování zatížení routeru je MRTG (Multi Router Traffic Grapher). Program je poskytován zcela zdarma a používá skriptovací jazyk PERL. Potřebné informace program získává přes SNMP (Simple Network Management Protocol) z monitorovaných stanic a umožňuje je zobrazovat za určité období (den, týden, měsíc, rok). Jeho výhodou je funkčnost na OS Linux, Windows i v síti Netware.

### 12.2 Instalace

Program pro svůj běh potřebuje nainstalovaný webserver, kterému se věnovala kapitola 9 a protokol SNMP. Instalaci samotného programu provádí příkaz:

```
apt-get install mrtg
```

Při instalaci se v adresáři, který obsahuje webový prostor, vytvoří adresář mrtg (v debianu je to `/var/www/mrtg`).

### 12.3 Konfigurace

Parametry, specifikující daný server se nastavují v `/etc/mrtg.cfg`. Ten v základním nastavení obsahuje jen informace o trafficu na síti, proto je nutné ho upravit a rozšířit o další části. Ty pak budou zapsány do adresáře `/etc/mrtg`.

#### 12.3.1 Vytváření souborů

Nejprve je nutné vytáhnout ze základního nastavení hodnoty pro soubor `traffic.cfg`, což se provádí příkazem:

```
/usr/bin/cfgmaker \
--output=/etc/mrtg/traffic.cfg \
--ifdesc=ip \
--ifref=descr \
--global "WorkDir: /var/www/localhost/htdocs/mrtg" \
--global "Options[_]: bits,growright" \
    public@localhost}
```

Nyní následuje vytvoření konfiguračních souborů pro monitorování procesoru a operační paměti.

**Monitorování vytížení procesoru**

```

WorkDir: /var/www/localhost/htdocs/mrtg
LoadMIBs: /usr/share/snmp/mibs/UCD-SNMP-MIB.txt
Target[localhost.cpu]:ssCpuRawUser.0&ssCpuRawUser.0:public@127.0.0.1
+ ssCpuRawSRouterUptime[localhost.cpu]: public@127.0.0.1
MaxBytes[localhost.cpu]: 100
Title[localhost.cpu]: CPU Load
PageTop[localhost.cpu]: <H1>Active CPU Load %</H1>
Unscaled[localhost.cpu]: ymwd
ShortLegend[localhost.cpu]: %
YLegend[localhost.cpu]: CPU Utilization
Legend1[localhost.cpu]: Active CPU in % (Load)
Legend2[localhost.cpu]:
Legend3[localhost.cpu]:
Legend4[localhost.cpu]:
LegendI[localhost.cpu]: Active
LegendO[localhost.cpu]:
Options[localhost.cpu]: growright,nopercent

```

**Monitorování vytížení paměti**

```

LoadMIBs: /usr/share/snmp/mibs/HOST-RESOURCES-MIB.txt
Target[localhost.mem]: .1.3.6.1.4.1.2021.4.6.0&.1.3.6.1.4.1.2021.4.6.0 \
:public@localhost
PageTop[localhost.mem]: <H1>Free Memory</H1>
WorkDir: /var/www/localhost/htdocs/mrtg
Options[localhost.mem]: nopercent,growright,gauge,noinfo
Title[localhost.mem]: Free Memory
MaxBytes[localhost.mem]: 1000000
kMG[localhost.mem]: k,M,G,T,P,X
YLegend[localhost.mem]: bytes
ShortLegend[localhost.mem]: bytes
LegendI[localhost.mem]: Free Memory:
LegendO[localhost.mem]:
Legend1[localhost.mem]: Free memory, not including swap, in bytes

```

**12.3.2 Generování mrtg.cfg**

Nyní následuje vytvoření zástupných souborů, které budou použity jako spouštěcí skripty.

Soubory budou vytvořeny v adresáři `/etc/cron.mrtg/` a příklad souboru `traffic`, zastupující `traffic.cfg`:

```
#!/bin/sh
/usr/bin/mrtg /etc/mrtg/traffic.cfg
```

Pro ostatní je vytvoření analogické.

Nyní je nutné souborům přiřadit atribut spustitelnosti příkazem `/bin/chmod +x /etc/cron.mrtg/*` a následně každý z nich 3x spustit, aby se program korektně zinicizoval.

Posledním krokem je vygenerování souborů `index.html` a `mrtg.cfg` příkazy:

```
/usr/bin/indexmaker --output=/var/www/localhost/htdocs/mrtg/index.html \
--title="MRTG správa pro 84.242.66.121" \
--sort=name \
--enumerate \
/etc/mrtg/traffic.cfg \
/etc/mrtg/cpu.cfg \
/etc/mrtg/mem.cfg

cfgmaker --global "WorkDir: /var/www/localhost/htdocs/mrtg/" \
--global "Options[_]: growright,bits" \
--ifref=ip \
public@localhost > /etc/mrtg/mrtg.cfg
```

### 12.3.3 Aktualizace dat přes Cron

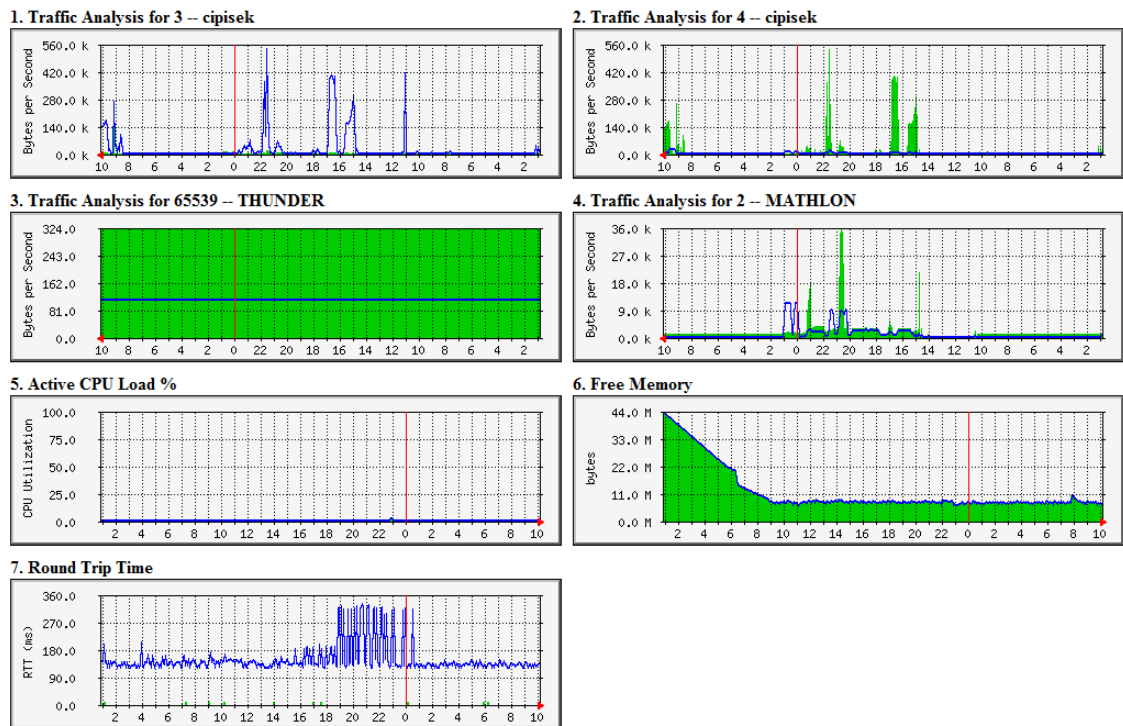
V tuto chvíli už by mělo být vše funkční. Stačí už jen zautomatizovat aktualizaci dat. K tomu využijeme Cron a pro něj vytvořený skript:

```
* /5 * * * * root /etc/cron.mrtg/cpu > /dev/null
* /5 * * * * root /etc/cron.mrtg/mem > /dev/null
* /5 * * * * root /etc/cron.mrtg/traffic > /dev/null
```

## 12.4 Výsledek

Ukázka takto nakonfigurovaného monitoringu je na obr.12.1

### Cipisek stats



**MRTG** MULTI ROUTER TRAFFIC GRAPHER  
 version 2.14.7  
 Tobias Oetiker <toebi@oetiker.ch>  
 and Dave Rand <dlr@bungli.com>

Obr. 12.1: Základní správa MRTG

## 13 KVALITA SLUŽEB

### 13.1 Úvod

Velkým problémem, se kterým se může potkat každý síťový administrátor je zahlcení šířky pásma jedním uživatelem a následná nedostupnost internetového připojení tam, kde je potřeba.

Vzorovým případem budiž firma prodávající počítače, ve které technik v service potřebuje stáhnout ovladače, zatímco prodejce musí obsluhovat klienty. V případě, že by stahování ovladačů vytížilo celou linku dané firmy, obchodník by nemohl vykonávat svou profesi. Pokud však zavedeme jednoduché pravidla kvality služeb, které zvýhodní obchodní oddělení (zajistí vyšší prioritu paketů), bude současně obchodník prodávat a zaměstnanec v servisu stahovat ovladače. Tímto způsobem můžeme upravovat prioritu protokolů i jednotlivých IP adres.

### 13.2 Prerekvizity

Kolekce programů zajišťující řízení provozu v TCP/IP sítích je poskytována pod souhrnným názvem **Iproute2**. Zvolit můžete stažení balíčků z domovských stránek projektu [linux-foundation.org](http://linux-foundation.org), či v debianu zadat příkaz:

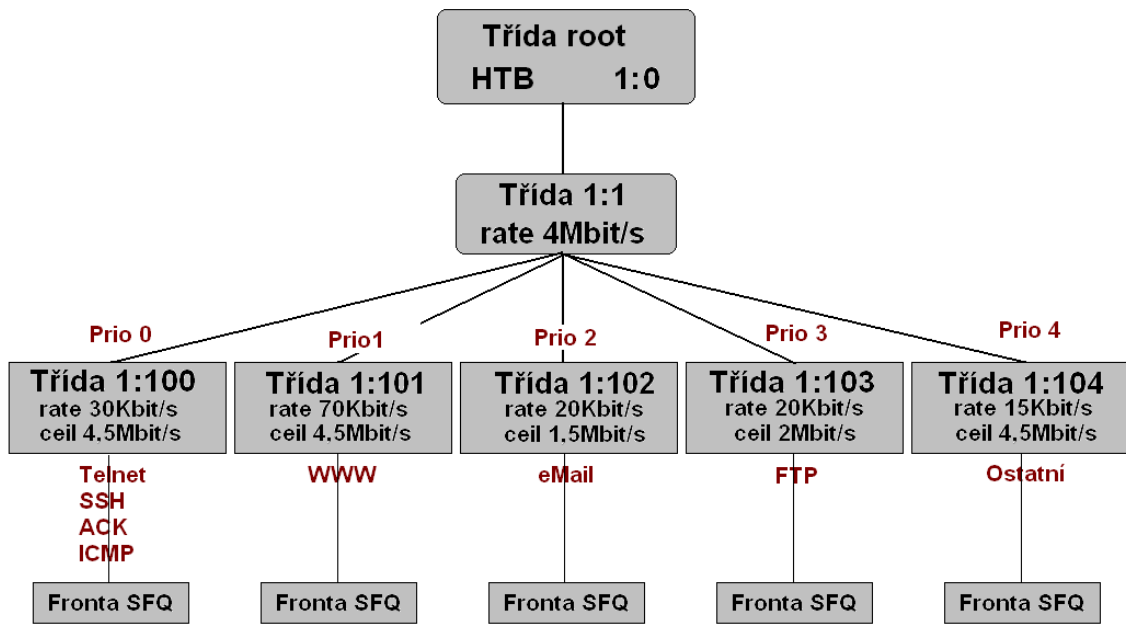
```
apt-get install iproute
```

Stejně jako v případě Iptables se zadávají příkazy ve formě jednotlivých příkazů. Pro maximální přehlednost a zachování konfigurace je proto výhodné vytvořit spustitelný skript.

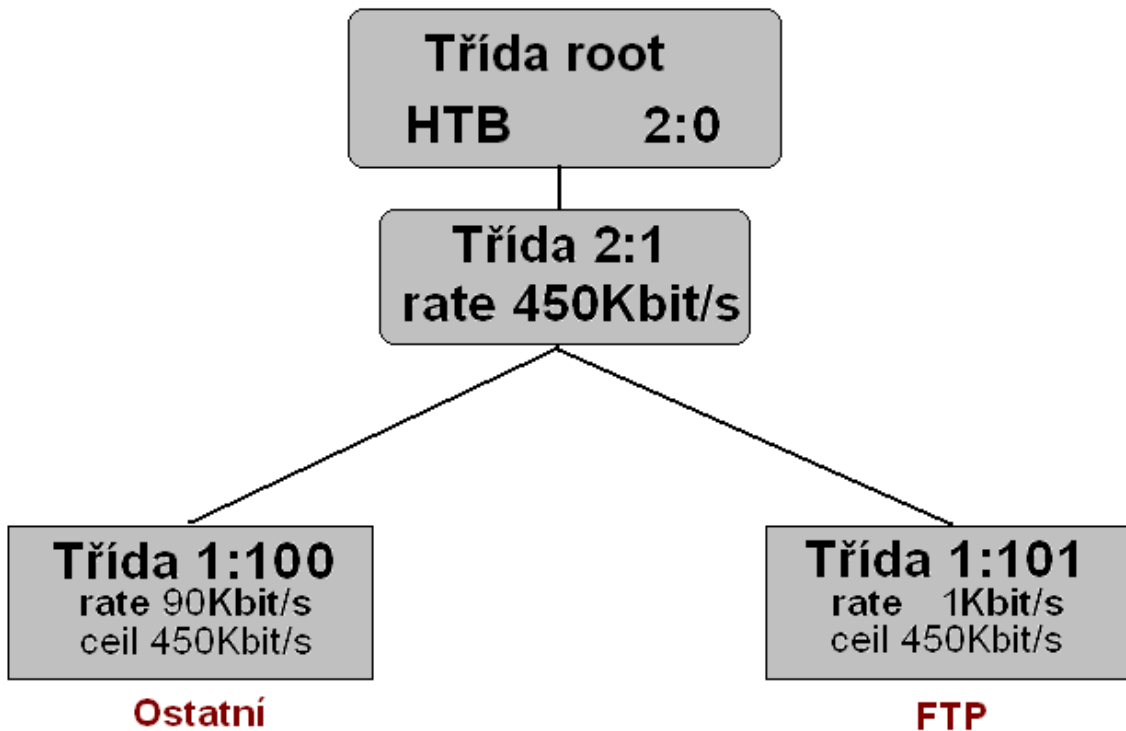
V `/etc/init.d/` vytvoříme soubor „htb“, do kterého vepíšeme potřebné příkazy. Soubor zavedeme do spouštěcí části Linuxu (`rc.d`) na 30 pozici příkazem **update-rc.d htb defaults 30**.

### 13.3 Konfigurace

Jak již bylo řečeno, pravidla se zapisují do řádků v souboru „htb“. Celý soubor se nachází v příloze C. Vzhledem k připojení k internetu přes rozvody kabelové televize, které zajišťují nesymetrický provoz (Download 5Mbit/s, Upload 512Kbit/s), bylo nutné řídit přenos jak pro příchozí, tak i odchozí komunikační kanál. Výsledná schémata jsou na obr.13.1 a obr.13.2.



Obr. 13.1: Rozdělení šířky pásma pro příchozí spojení (download)



Obr. 13.2: Rozdělení šířky pásma pro odchozí spojení (upload)

### 13.3.1 Popis skriptu

Klasifikace provozu je založena na principu kořenové struktury, kde úplně nejvýše se nachází celá linka **root**, která je dále dělena na třídy (class). Vzájemná hierarchie je zajišťována atributem **parent** (rodič) odkazujícím na výše položenou vrstvu.

#### Základní struktura

Prvním krokem je vytvoření základní struktury. Jednotlivým třídám a disciplínám je přiřazován atributem **handle** identifikátor ve tvaru rodič:potomek.

```
tc qdisc add dev eth1 root handle 1:0 htb default 15
tc class add dev eth1 parent 1:0 classid 1:1 htb rate 4600kbit
tc class add dev eth1 parent 1:1 classid 1:100 htb rate 30kbit
ceil 4600kbit prio 0 burst 50k quantum 19300
```

Vnižších třídách se setkáme s dalšími specifickými pravidly:

- Parametr **rate** udává garantovanou propustnost, zatímco **ceil** vyjadřuje strop pro danou třídu.
- Dalším volitelným parametrem je **prio**, který mění pořadí vykonávání filtrů.
- Hodnota **burst** udává počet bytů, které je možné jednorázově odeslat, pokud máme dostatečný počet volných tokenů.
- Pokud více tříd sdílí jedno pásmo, je potřeba zadefinovat hodnotu **quantum**, která udává počet bytů, po jejichž vyčerpání přichází na řadu další třída.

#### Řazení do front

Dalším krokem je definice fronty, se kterou bude daná třída pracovat. Disciplíny front jsou následující:

##### Classless - bez řazení do tříd

- SFQ (Stochastic Fairness Queueing) - rozdělení paketů až do 256 front, které jsou algoritmem cyklicky obsluhovány.
- FIFO (First In First Out) - implicitně nastavená fronta. Funguje na jednoduchém principu: kdo první přijde, první také odchází.

- TBF (Token Bucket Filter) - algoritmus pracující na principu kyblíčku. Příchozí data jsou řazena do fronty - pakety jsou propouštěny pouze pokud jsou volné tokeny. Po přetečení fronty jsou další příchozí pakety zahazovány. Algoritmus používaný k omezení přenosové rychlosti.
- RED (Random Early Detect, Random Early Drop) - algoritmus pracující se statistikami provozu. Čím více se linka blíží stavu zahlcení, tím více je zahozených paketů. Příslušný kanál je tak donucen ke snížení přenosové rychlosti.

### Classful - s rozřazením do tříd

- CBQ (Class Based Queueing) - funguje na principu rozdělení přenosového pásma do tříd. V případě potřeby je možné „vypůjčení“ rychlosti z jiné třídy či od rodičovské třídy. Nechybí ani podpora priorit.
- HTB (Hierarchical Token Bucket) - pracuje podobně jako CBQ. Pakety jsou rozděleny do toků a podle hodnoty **rate** odebíráme. Až je dosaženo hodnoty rate, přejdeme na další tok. Pokud mají všechny toky vyčerpanou hodnotu rate, testujeme, jestli si nemohou „vypůjčit“ z vyšších tříd (předků, ...).
- PRIO - pakety jsou rozděleny do tříd s různými prioritami, které jsou spravovány jako FIFO dle priorit.

Výsledný příklad může vypadat např. takto:

```
tc qdisc add dev dev1 parent 1:100 handle 100:0 sfq perturb 10
```

Hodnota perturb udává čas v sekundách, za který se změní hashovací funkce.

### Značkování paketů

Značkování paketů může být prováděno buď samotným Traffic Control filtrem, či v případě složitějšího značkování programem Iptables.

### Označení pomocí tc

Ke značkování Traffic Control filtrem se používá nástroj U32 classifier, který dokáže pakety řadit na základě zdrojové/cílové IP adresy, portu a mnoha dalších pravidel.

```
tc filter add dev eth1 parent 1:0 protocol ip u32
match ip dst 192.168.1.1 match ip src 192.168.1.15 classid 1:11
```

### Označení pomocí Iptables

Značkování pomocí Iptables je výhodné použít při potřebě složitějších pravidel (např. na základě MAC adres). Použití mangle tabulky pro jakékoliv filtrování však není doporučeno vzhledem ke zvýšené časové náročnosti.

```
iptables -A PREROUTING -t mangle -d 192.168.1.1 -j MARK --set-mark 11  
tc filter add dev eth1 parent 1:0 protocol ip handle 100 fw flowid 1:11
```

## 14 PRAKTICKÁ ČÁST

V následující části se práce bude zabývat popisem zvoleného řešení návrhu sítě pro malou a střední firmu. Následně bude proveden rozbor síťových služeb, které bude směrovač nabízet. Posledním bodem je otestování funkčnosti jednotlivých serverů a praktický přínos celého navrženého řešení.

### 14.1 Konfigurace routeru

Funkci routeru spolehlivě zastal standardní počítač s procesorem Pentium4 2,4GHz se 512MB operační paměti. Za operační systém byl zvolen Debian verze 4.0 r3 „Etch“ release s jádrem 2.6.18-4-686.

### 14.2 Zadání

V rámci práce prostudujte a popište možná linuxová řešení počítačové sítě pro malé a střední firmy. Na tomto základě navrhnete vlastní řešení sítě. Kromě poskytnutí základních služeb a přiměřené bezpečnosti se požaduje jednoduchá správa sítě. Navržené řešení zdůvodněte, prakticky ověřte a zhodnoťte.

### 14.3 Postup řešení

Klíčovým elementem této práce bylo vytvoření bezpečného serveru, který zajistí poskytování služeb uživatelům vnitřní sítě. Službami jsou myšleny DHCP, web (intranet), FTP či samba server. Důraz byl také kladen na zajištění rovnoměrného rozdělení šířky pásma mezi účastníky, což bývá často opomíjeno ať už u menších, či středních firem.

Nejdůležitějším úkolem bylo zabezpečení sítě pomocí firewall. Pro její konfiguraci bylo použito programu IPTables a zavedení nastavení do spouštěcího skriptu pro opakované nastavení. Nastavení bylo provedeno podle ukázkového skriptu uvedeného v příloze C. Spuštění skriptu proběhlo bez chyb a výsledek zabezpečení je popsán v kap.14.3.1.

Dalším v pořadí byl DHCP server, který se stará o přiřazování IP adres uživatelům. Při konfiguraci tohoto serveru jsem zvolil poměrně benevolentní politiku, kdy server přidělí adresu prakticky komukoliv. Tento typ nastavení je pro firemní použití velmi nebezpečný a proto byla do firewall přidána pravidla, která umožní přístup do sítě pouze uživatelům s platnou kombinací IP adresa - MAC adresa.

Samba server nastavený jako v kap.6 usnadňuje klientům práci v síti zejména možností sdílení souborů mezi různými platformami. Firemní sektor jistě využije možnost síťového tisku - ušetří se náklady spojené s dražší síťovou tiskárnou či modulem umožňujícím síťový tisk.

Někdo by mohl namítnout, že použití FTP serveru může být potenciálním rizikem. V případě zakázání standardní nešifrované komunikace však odpadá starost o možnost odchyty uživatelského jména a hesla. Spojení ProFTPd a Mysql databáze se ukázalo jako velmi praktické, zejména díky přehlednosti a snadné duplikaci či odstranění povolených uživatelů.

Při použití webserveru také doporučuji jistou obezřetnost a pokud server opravdu nebude sloužit k prezentaci firmy, je lepší stránky omezit pouze na intranetový provoz.

Posledním cílem byla snadná správa a jednoduchý dohled nad sítí. K obojímu je možné využít Webmin, avšak vzhledem k jeho komplexnosti a možnosti měnit vše od hesla správce systému (root) až po změnu nastavení firewall, je velmi nasnadě povolit přístup pouze z několika málo IP adres a určitě jen z lokální sítě. Ideálním se mi pro monitorování stažených dat jevil program Ipac-ng, který však zejména kvůli pomalému vývoji přestal na novějších verzích jádra a IPTables fungovat. Pro monitorování provozu na síti jsem tedy nakonec zvolil velmi jednoduchý a přehledný program MRTG, který byl popsán v kap.12.

Co se týče funkčnosti skriptu, rozdělujícího datový tok podle protokolu na několik tříd, funguje již velmi dlouhou dobu bez problémů. Původní záměr jeho vytvoření bylo pouze omezení odchozího provozu z FTP, aby uživatelé vnitřní sítě měli stále svižný přístup na internet. Postupem času byl rozdělen do kategorií i provoz při stahování a to zejména z důvodu pomalé odezvy SSH spojení při maximálním vytížení linky.

### 14.3.1 Výsledky scanování portů

Pro scanování portů jsem použil dvě metody. První metoda se nachází na stránce <https://www.grc.com> a nechává po odsouhlasení požadavků a výběru testu testovat uzavřenost portů. Druhou možností je použití programu Nmap. Tento program je volně ke stažení na domovské stránce programu: <http://www.insecure.org/nmap/>. Popis instalace včetně možných parametrů skenování je uveden přímo na uvedené stránce.

**Internetovou stránku Shields UP!**

GRC Port Authority Report created on UTC: 2008-03-21 at 16:56:21

Results from scan of ports: 0-1055

3 Ports Open

0 Ports Closed

1053 Ports Stealth

-----

1056 Ports Tested

Ports found to be OPEN were: 21, 80, 443

Other than what is listed above, all ports are STEALTH.

TruStealth: FAILED - NOT all tested ports were STEALTH,  
- NO unsolicited packets were received,  
- A PING REPLY (ICMP Echo) WAS RECEIVED.

**Program Nmap v4.00**

```
cipisek:~# nmap -sT 84.242.66.121
```

```
Starting Nmap 4.53 ( http://insecure.org ) at 2008-03-21 18:05 CET
Interesting ports on bno-84-242-66-121.karneval.cz (84.242.66.121):
Not shown: 1709 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
443/tcp   open  https
Nmap done: 1 IP address (1 host up) scanned in 0.138 seconds
```

**Proč skenování portů?**

Skenování portů serveru je použito jako ukázka nejnižší úrovně zabezpečení serveru. Z uvedených výsledků je patrné, že na dané adrese nějaký server je. Pokud by však byly přesměrovány porty pro Web a FTP a zakázána odezva na příkaz ping, nebyl by server z venku jakkoliv viditelný při skenování základního rozsahu portů (0-1023).

## 15 ZÁVĚR

V rámci této práce jsem se zabýval problematikou síťového zabezpečení malé a střední firemní sítě a snažil se také minimalizovat vstupní náklady na její vybudování. Výsledkem je centralizovaný router, který zároveň plní roli většiny serverů, které jsou ve firemním prostředí používány. Výhodou takto nakonfigurované sítě jsou velmi malé vstupní náklady pohybující se v řádu několika tisíc korun. Nevýhodou je nízká možnost rozšíření sítě o větší počet klientů.

Jako ekvivalent sítě pro střední firmu jsem použil stejného routeru, jaký byl použit pro malou firmu s tím rozdílem, že těchto serverů by bylo v celé síti použito hned několik. Nadřazen by jim byl pouze jeden centrální server.

Hlavním cílem bylo navržení sítě, jejíž správu bude obstarávat linuxový router a bude současně plnit i funkci serverů. K realizaci tohoto úkolu jsem použil již vybudovanou domácí síť s pěti uživateli, kterou jsem rozšířil o Samba server. Jeho úkolem je zajištění sdílení souborů mezi uživatelskými stanicemi s operačními systémy Windows a Linux. Spolehlivě se také postaral o sdílení síťové obvyčejné tiskárny mezi uživatele v síti.

Další funkcí, kterou by měl každý server mít, je přiřazování IP adres klientským stanicím pomocí DHCP serveru. Tímto serverem se prakticky minimalizuje možnost kolize dvou stejných adres v síti. Samozřejmě nově přichozí uživatelé mohou ihned přistupovat do lokální sítě či do sítě internet. Důležité je však zabezpečit síť proti neoprávněnému vstupu nepovolané osoby. Řešením by mohla být například autentizace pomocí VPN sítě.

Pro snadný přenos dat i mimo lokální síť jsem zvolil FTP server. Jelikož se počítá s tím, že jsou firemní data tím nejdůležitějším, bylo nutné nezabezpečený FTP přenos šifrovat pomocí SSL protokolu. Z důvodu podpory šifrování a takřka neomezených možností nastavení jsem z mnoha FTP serverů vybral ProFTPD.

Dalším nezbytným krokem pro „život“ firmy je její prezentace. V dnešní době komunikačních technologií se takřka žádná firma neobejde bez webových stránek a tudíž nemohl chybět webový a databázový server. Velmi oblíbenou kombinací bývá Apache s MySQL. Zvolil jsem tedy totéž zejména kvůli perfektní dokumentaci a v případě problémů možnost nalezení jejich řešení na diskuzních serverech. Webové rozhraní se však netýká pouze firemních webových stránek; bylo využito i programem Webmin, který monitoruje celou síť včetně vytížení serveru a nabízí velmi přehledné a detailní statistiky veškerého provozu.

Posledním, avšak nejdůležitějším krokem bylo zabezpečení celé sítě. V tomto bodě jsem vycházel z firewall, který jsem nakonfiguroval v rámci bakalářské práce a rozšířil její použití o filtraci uživatelů a optimalizoval její činnost. Firewall je tedy nakonfigurován pomocí programu IPtables, který je součástí každé distribuce

Linuxu. Nastavování pomocí jednotlivých příkazů, definujících akce s pakety se mi jeví jako nejlepší a nejkvalitnější řešení, protože je celá síť pod naprostým dohledem, na rozdíl od komplexních grafických nástaveb, u kterých není zajištěna absolutní bezpečnost.

Aby uživatelům v síti byla rozdělována šířka pásma spravedlivě, bylo použito HTB skriptu, který byl vázán na jednotlivé porty. Pokud tedy jeden uživatel stahoval data, druhý mohl prakticky bez navýšení odezvy otevírat webové stránky. Volná kapacita pásma je dynamicky přidělována. Jeden uživatel tedy může využívat plné rychlosti do té chvíle, než bude vyžadována dalším klientem.

Takto nakonfigurovaný server poskytuje rozsáhlou nabídku služeb. Vyšší úroveň bezpečnosti by mohla být realizována dodatečným Radius serverem umístěným v lokální části sítě, pomocí kterého by se autentizovali uživatelé při vstupu.

Porovnání s dalšími síťovými řešeními není příliš snadné, neboť v mém případě byl kladen velký důraz na nízké pořizovací náklady. Pro malou, stále se rozvíjející firmu by byl tento server ideálním řešením. Cenové náklady na pořízení tohoto serveru se pohybují kolem 6 000 Kč (vč. gigabitového switchu). Za tuto cenu je možné pořídit miniaturizované řešení v podobě „inteligentního“ routeru, který nabízí obdobné funkce. Jeho nevýhodou je však omezený výkon procesoru, často špatná rozšiřitelnost a nedostatek operační paměti.

Pro střední firmy je už využití tohoto serveru otázkou požadavků a také topologie sítě. Soutěžit např. s pokročilými směrovacími funkcemi Cisco routerů není vůbec snadné, přesto by mnou realizované řešení mohlo najít místo v nižších úrovních takto vybudované sítě, kde by zaujmul místo hraničního routeru mezi jednotlivými odděleními firmy. Využití by ale určitě našlo i při realizaci intranetových stránek firemního oddělení, nebo při budování zálohovacího serveru. Rychlostí a stabilitou může ostatním řešením směle konkurovat.

## LITERATURA

- [1] GRAHAM, Steven - SHAH, Steve. *Administrace systému Linux*. 3.vydání Praha: Grada, 2003. 552 s. ISBN 80-247-0641-5
- [2] DOSTÁLEK, Libor - KABELOVÁ, Alena. *Velký průvodce protokoly TCP/IP a systémem DNS*. 3.vydání Praha: Computer Press, 2002. 552 s. ISBN 80-7226-675-6
- [3] DOSTÁLEK, Libor. *Velký průvodce protokoly TCP/IP: Bezpečnost*. 3.vydání Praha: Computer Press, 2002. 550 s. ISBN 80-7226-849-X
- [4] Linuxsoft Poslední pevností na dobývání pro Linux je desktop [online]. 2004  
URL:<<http://www.linuxsoft.cz>>.
- [5] OpenSSL. OpenSSL HOWTO [online]. 2005  
URL:<<http://www.openssl.org>>.
- [6] PETŘÍČEK, Miroslav. Stavíme firewall [online]. 2001  
URL:<<http://www.root.cz/clanky/stavime-firewall-1/>>.
- [7] DAWSON, Terry - RUBINI, Alessandro Linux Networking-HOWTO [online]. 2004  
URL:< <http://www.tldp.org/HOWTO/NET3-4-HOWTO.html>>.
- [8] PETERKA, Jiří. Rodina protokolů TCP/IP, verze 2.3 [online]. 2005  
URL:< <http://www.earchiv.cz/l215/index.php3>>.
- [9] RFC Linux - HOWTO [online]. 2006  
URL:< <http://www.ietf.org/rfc.html>>.
- [10] System Administration Tips and Resources. LAMP on Sarge (Apache2, PHP5, MySQL5, phpMyAdmin, Smarty, ADODB) [online]. 2006  
URL:< <http://www.debian-administration.org/articles/357>>.

# A PŘÍLOHA - SKRIPT IPTABLES

```
#!/bin/bash
#set -x

# Vnejsi IP adresa a vnejsi rozhrani
INET_IP="84.242.66.121"
INET_IFACE="eth2"

# IP a broadcast adresa a rozhrani vnitřni site
LAN_IP="192.168.1.1/32"
LAN_BCAST="192.168.1.255"
LAN_IFACE="eth1"

# Lokalni loopback rozhrani
LO_IFACE="lo"
LO_IP="127.0.0.1/32"

# Cesta k programu iptables
IPTABLES="/sbin/iptables"

echo "1" > /proc/sys/net/ipv4/ip_forward

modprobe ip_nat_ftp
modprobe ip_conntrack_ftp

$IPTABLES -F INPUT
$IPTABLES -F OUTPUT
$IPTABLES -F FORWARD
$IPTABLES -t nat -F PREROUTING
$IPTABLES -t nat -F POSTROUTING

# Implicitni politikou je zahazovat nepovolene pakety
$IPTABLES -P INPUT DROP
$IPTABLES -P OUTPUT DROP
$IPTABLES -P FORWARD DROP

#$IPTABLES -P INPUT ACCEPT
#$IPTABLES -P OUTPUT ACCEPT
```

```
#$IPTABLES -P FORWARD ACCEPT

#
# Retezec POSTROUTING v NAT tabulce
#

# IP maskarada - SNAT
# NATujeme
$IPTABLES -A POSTROUTING -t nat -o $INET_IFACE -j SNAT --to $INET_IP

#
# Retezec FORWARD
#

# Vazba IP a MAC adres pro forward
$IPTABLES -A FORWARD -i $LAN_IFACE -s 192.168.1.6 -m mac
--mac-source 00:0C:76:59:5A:24 -j ACCEPT
$IPTABLES -A FORWARD -i $LAN_IFACE -s 192.168.1.7 -m mac
--mac-source 00:04:61:5B:9A:C8 -j ACCEPT

# Zakázán veškerý rozsah krom zmíněných IP/MAC kombinací
$IPTABLES -A FORWARD -i $LAN_IFACE -m iprange
--src-range 192.168.1.2-192.168.1.254 -j REJECT

# Routing zevnitř sítě ven neomezujeme
$IPTABLES -A FORWARD -i $LAN_IFACE -o $INET_IFACE -j ACCEPT

# Routing zvenku dovnitř pouze pro navazana spojeni (stavovy firewall)
$IPTABLES -A FORWARD -i $INET_IFACE -o $LAN_IFACE -m state
--state ESTABLISHED,RELATED -j ACCEPT

#
# Retezec INPUT
#

# Pravidla pro povolené služby
$IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 21 -j ACCEPT #FTP server
$IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 22 -j DROP #SSH server
$IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 80 -j ACCEPT #web server
$IPTABLES -A INPUT -i $INET_IFACE -p TCP --dport 443 -j ACCEPT #https
```

```
# Propouštíme ICMP
$IPTABLES -A INPUT -i $INET_IFACE -p ICMP -j ACCEPT

# Loopback omezovat
$IPTABLES -A INPUT -i $LO_IFACE -j ACCEPT

# Zamezení měnění IP adres přes mac adresy
$IPTABLES -A INPUT -m mac --mac-source 00:0C:76:59:5A:24
-i $LAN_IFACE -s 192.168.1.6 -j ACCEPT
#$IPTABLES -A INPUT -m mac --mac-source 00:04:61:5B:9A:C8
-i $LAN_IFACE -s 192.168.1.7 -j ACCEPT

# Zahazování paketů ze sítě, která není potřebná
$IPTABLES -A INPUT -i $LAN_IFACE -m iprange
--src-range 192.168.1.2-192.168.1.254 -j DROP

# Stejně jako pakety z lokální sítě, jsou-li určeny nám
$IPTABLES -A INPUT -i $LAN_IFACE -d $LAN_IP -j ACCEPT
$IPTABLES -A INPUT -i $LAN_IFACE -d $INET_IP -j ACCEPT

# Broadcasty na lokálním rozhraní jsou také naše
$IPTABLES -A INPUT -i $LAN_IFACE -d $LAN_BCAST -j ACCEPT

# Pakety od navázaných spojení jsou v pořádku
$IPTABLES -A INPUT -d $INET_IP -m state --state ESTABLISHED,RELATED -j ACCEPT

#
# Retezec OUTPUT
#

# Povolíme odchozí pakety, které mají naše IP adresy
$IPTABLES -A OUTPUT -s $LO_IP -j ACCEPT
$IPTABLES -A OUTPUT -s $LAN_IP -j ACCEPT
$IPTABLES -A OUTPUT -s $INET_IP -j ACCEPT

# Povolíme DHCP broadcasty na LAN rozhraní
$IPTABLES -A OUTPUT -o $LAN_IFACE -p UDP --dport 68 --sport 67 -j ACCEPT
```

```
case "$1" in

    stop)
        echo "Stopping firewall: "
        echo 0 > /proc/sys/net/ipv4/ip_forward
        $IPTABLES -F INPUT
        $IPTABLES -F OUTPUT
        $IPTABLES -F FORWARD
        $IPTABLES -t nat -F PREROUTING
        $IPTABLES -t nat -F POSTROUTING
    echo
        ;;

    restart)
        echo "Restarting firewall: "
        $0 stop
        $0 start
    # /usr/local/sbin/fetchipac -S

        echo
        ;;

    status)
        $IPTABLES -L
        ;;

    *)
        echo "Usage: $0 {start|stop|restart|status}"

esac
exit 0
```

## B PŘÍLOHA - KONFIGURAČNÍ SOUBOR FTP SERVERU

```
#Zakladni informace o serveru
ServerName      "Proftpd Server"
ServerType      Standalone
ServerAdmin     aaaa@bb.cc

# Skryti co nejvíce informací o serveru pro příchozí uživatele
ServerIdent on "Welcome to the FTP server. Please login..."
DeferWelcome    on
DefaultServer   on

# Povolení navázování souborů při uploadu
AllowStoreRestart on

# Defaultně používaný port
Port 21

# Umask definuje, jaké práva budou mít defaultní uživatelé.
# Zabraňuje nežádoucímu vytváření souborů a složek
Umask 022

# Vypnutí kontroly IP adresy přes DNS servery. Urychluje přihlázení.
IdentLookups off

# Pro zabránění DoS útokům nastavíme hodnotu podprocesů na 30
MaxInstances 30

# Nastavení skupiny a uživatele, pod kterým bude spuštěn FTP server.
User nobody
Group nogroup

# Logovací soubory
TransferLog     /var/log/ftp
ExtendedLog     /var/log/ftp.login AUTH auth

# Formát zápisu logu
```

```
LogFormat      default "%h %l %u %t \"%r\" %s %b"
LogFormat      auth    "%v [%P] %h %t \"%r\" %s"
LogFormat      write   "%h %l %u %t \"%r\" %s %b"
```

```
# Zabraňujeme brouzdání po serveru uživatelům a ukazuje jim jen
# jejich domovský adresář.
DefaultRoot ~
```

```
# Povoluje přepisování souborů
AllowOverwrite on
```

```
# Přidá povolení k SSH spojení
AllowForeignAddress on
```

```
# Hesla v MySQL jsou kryptovány algoritmem CRYPT
SQLAuthTypes Plaintext Crypt
SQLAuthenticate users* groups*
```

```
# Ukazuje, v jakém tvaru se připojuje k DB serveru
SQLConnectInfo ftpdb@localhost proftpd password
```

```
# Zde říkáme FTP serveru názvy jednotlivých sloupců tabulky
SQLGroupInfo ftpgroup groupname gid members
```

```
# Nastavení SQL minimálního UID a GID
SQLMinID 5000
```

```
# Vytváří k účtům domovské adresáře, pokud neexistují.
SQLHomedirOnDemand on
```

```
# Zvýší hodnotu Count při každém přihlášení
SQLLog PASS updatecount
SQLNamedQuery updatecount UPDATE "count=count+1,
accessed=now() WHERE userid='%u'" ftpuser
```

```
# Aktualizuj záznam modified pokaždé, když uživatel nahraje nebo smaže soubor
SQLLog STOR,DELE modified
SQLNamedQuery modified UPDATE "modified=now() WHERE
userid='%u'" ftpuser
```

```
RootLogin off
RequireValidShell off

# Nastavení TLS (zabezpečení)
<IfModule mod_tls.c>
TLSEngine on
TLSLog /var/log/proftpd/tls.log
#TLSProtocol TLSv1
TLSProtocol SSLv3

# Pokud nevyžadujeme zabezpečený přenos a stačí i standardní
TLSRequired Off

# Certifikáty serveru
TLRSACertificateFile /etc/proftpd/ftpd-rsa.pem
TLRSACertificateKeyFile /etc/proftpd/ftpd-rsa-key.pem

# Autentizace klientů, kteří chtějí zabezpečený přenos
TLSVerifyClient off
TLSOptions NoCertRequest
```

## C PŘÍLOHA - SKRIPT ZAJIŠŤUJÍCÍ QOS

```
#!/bin/sh
#####
#   Deklarace promennych
#####
LOCAL="eth1"           #the internal interface - LAN
NET="eth2"             #the external interface - Internet

RateDownload=4000     #maximum download speed
CeilDownload=4600     #maximum download speed
EmailDownload=1500    #maximum email download speed
FTPDDownload=2000     #maximum ftp download speed
CeilDownloadP2P=200   #maximum download speed for P2P

CeilUpload=450        #maximum upload speed
FTPUUpload=256        #maximum upload speed for P2P
#####
#   Start HTB skriptu
#####
#   Download
#####

tc qdisc add dev $LOCAL root handle 1:0 htb default 15

tc class add dev $LOCAL parent 1:0 classid 1:1 htb rate
${RateDownload}kbit ceil ${CeilDownload}kbit

#####
#   Definice jednotlivych trid
#####
#11-->Fastest - telnet, ssh, acks, ping
tc class add dev $LOCAL parent 1:1 classid 1:100 htb rate 30kbit
ceil ${CeilDownload}kbit prio 0 burst 50k quantum 19300
#12-->Fast - WWW
tc class add dev $LOCAL parent 1:1 classid 1:101 htb rate 70kbit
ceil ${CeilDownload}kbit prio 1 burst 50k quantum 19300
#13-->Medium - eMail
```

```
tc class add dev $LOCAL parent 1:1 classid 1:102 htb rate 20kbit
ceil ${EmailDownload}kbit prio 2 burst 50k quantum 6400
```

```
#14-->Slow - FTP
```

```
tc class add dev $LOCAL parent 1:1 classid 1:103 htb rate 20kbit
ceil ${FTPDownload}kbit prio 3 burst 50k quantum 3200
```

```
#15-->Other - Default
```

```
tc class add dev $LOCAL parent 1:1 classid 1:104 htb rate 15kbit
ceil ${CeilDownload}kbit prio 4 burst 10k quantum 3200
```

```
#filter
```

```
tc filter add dev $LOCAL parent 1:0 protocol ip prio 1 handle 100
fw classid 1:100
```

```
tc filter add dev $LOCAL parent 1:0 protocol ip prio 2 handle 101
fw classid 1:101
```

```
tc filter add dev $LOCAL parent 1:0 protocol ip prio 3 handle 102
fw classid 1:102
```

```
tc filter add dev $LOCAL parent 1:0 protocol ip prio 4 handle 103
fw classid 1:103
```

```
tc filter add dev $LOCAL parent 1:0 protocol ip prio 5 handle 104
fw classid 1:104
```

```
#sfq
```

```
tc qdisc add dev $LOCAL parent 1:100 handle 100:0 sfq perturb 10
```

```
tc qdisc add dev $LOCAL parent 1:101 handle 101:0 sfq perturb 10
```

```
tc qdisc add dev $LOCAL parent 1:102 handle 102:0 sfq perturb 10
```

```
tc qdisc add dev $LOCAL parent 1:103 handle 103:0 sfq perturb 10
```

```
tc qdisc add dev $LOCAL parent 1:104 handle 104:0 sfq perturb 10
```

```
#IPTABLES
```

```
iptables -A PREROUTING -t mangle -p tcp -j CONNMARK --restore-mark
```

```
iptables -A PREROUTING -t mangle -p tcp -m mark ! --mark 0 -j RETURN
```

```
#11
```

```
#telnet
```

```
iptables -A PREROUTING -t mangle -p tcp --sport 23 -j MARK
--set-mark 100
```

```
iptables -A PREROUTING -t mangle -p udp --sport 23 -j MARK
--set-mark 100
```

```
#ssh
```

```
iptables -A PREROUTING -t mangle -p tcp --sport 22 -j MARK
```

```
--set-mark 100
  iptables -A PREROUTING -t mangle -p udp --sport 22 -j MARK
--set-mark 100
#icmp
  iptables -A PREROUTING -t mangle -p icmp -j MARK --set-mark 100
#dns
  iptables -A PREROUTING -t mangle -p tcp --sport 53 -j MARK
--set-mark 100
  iptables -A PREROUTING -t mangle -p udp --sport 53 -j MARK
--set-mark 100
#acks and small packets
  iptables -A PREROUTING -t mangle -p tcp -m tcp
--tcp-flags SYN,RST,ACK SYN -j MARK --set-mark 100
  iptables -A PREROUTING -t mangle -p tcp -m length
--length :64 -j MARK --set-mark 100

#12
#WWW
  iptables -A PREROUTING -t mangle -p tcp --sport 80 -j MARK
--set-mark 101
  iptables -A PREROUTING -t mangle -p tcp --sport 443 -j MARK
--set-mark 101
  iptables -A PREROUTING -t mangle -p tcp -s 192.168.1.1 -j MARK
--set-mark 101

#13
#pop3
  iptables -A PREROUTING -t mangle -p tcp --sport 110 -j MARK
--set-mark 102
  iptables -A PREROUTING -t mangle -p udp --sport 110 -j MARK
--set-mark 102
#smtp
  iptables -A PREROUTING -t mangle -p tcp --sport 25 -j MARK
--set-mark 102
  iptables -A PREROUTING -t mangle -p udp --sport 25 -j MARK
--set-mark 102
#imap
  iptables -A PREROUTING -t mangle -p tcp --sport 143 -j MARK
--set-mark 102
```

```
iptables -A PREROUTING -t mangle -p udp --sport 143 -j MARK
--set-mark 102
```

```
#14
```

```
#ftp
```

```
iptables -A PREROUTING -t mangle -p tcp -m tcp --sport 20:21
-j MARK --set-mark 103
```

```
#####
```

```
# Upload
```

```
#####
```

```
tc qdisc add dev $NET root handle 2: htb default 11
```

```
tc class add dev $NET parent 2: classid 2:1 htb rate
${CeilUpload}kbit ceil ${CeilUpload}kbit
```

```
#####
```

```
# Definice jednotlivych trid
```

```
#####
```

```
#11-->Nic z uploadu neomezovat
```

```
tc class add dev $NET parent 2:1 classid 2:100 htb rate 90kbit
ceil ${CeilUpload}kbit prio 7 burst 15k quantum 19300
```

```
#12-->FTP upload
```

```
tc class add dev $NET parent 2:1 classid 2:101 htb rate 1kbit
ceil ${FTPUpload}kbit prio 8 burst 5k quantum 1600
```

```
tc filter add dev $NET parent 2:0 protocol ip prio 1 handle
100 fw classid 2:100
```

```
tc filter add dev $NET parent 2:0 protocol ip prio 6 handle
101 fw classid 2:101
```

```
#14
```

```
#ftp
```

```
iptables -A POSTROUTING -t mangle -o eth2 -p tcp
--sport 20:21 -j MARK --set-mark 101
```