

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

SOFTWARE PRE PODPORU WEBHOSTINGU

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

MIROSLAV HULA

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

SOFTWARE PRO PODPORU WEBHOSTINGU

WEBHOSTING MANAGEMENT SOFTWARE

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MIROSLAV HULA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. PETR WEISS

BRNO 2009

Abstrakt

Cieľom práce je vytvoriť software pre podporu webhostingu, čo znamená, vytvoriť systém umožňujúci užívateľom vytváranie si vlastných subdomén v doméne, využívanie emailovej služby, zabezpečenie prístupu k FTP serveru a k databázovému serveru. Obsluha služieb užívateľom musí prebiehať pomocou grafického užívateľského rozhrania podobne, ako aj samotná registrácia do tohto systému.

Kľúčové slová

GNU/Linux, Apache HTTP Server, MySQL Server, PHP: Hypertext Preprocessor, SMTP server, POP3/IMAP Server, FTP Server, webhosting, virtuálni užívatelia

Abstract

The main goal of this thesis is to create a software tool to support webhosting, it means to create a system that provides email services, services of an FTP server and database server, and allows users to create their own subdomains within the domain. Moreover, the system must provide a graphical interface for its users.

Keywords

GNU/Linux, Apache HTTP Server, MySQL Server, PHP: Hypertext Preprocessor, SMTP server, POP3/IMAP Server, FTP Server, webhosting, virtual users

Citácia

Miroslav Hula: Software pre podporu webhostingu, bakalárska práca, Brno, FIT VUT v Brně, 2009

Software pre podporu webhostingu

Prehlásenie

Čestne prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána Ing. Petra Weissa.

Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....
Miroslav Hula
18. mája 2009

Pod'akovanie

Rád by som poďakoval vedúcemu mojej bakalárskej práce Ing. Petrovi Weissovi za jeho pripomienky, odborné rady a čas, ktorý mi pri tvorbe práce venoval. Pod'akovanie patrí aj mojej rodine, za jej trpezlivosť a pomoc.

© Miroslav Hula, 2009.

Táto práca vznikla ako školské dielo na Vysokom učení technickom v Brne, Fakulte informačných technológií. Práca je chránená autorským zákonom a jej použitie bez udelenia oprávnenia autorom je nezákonné, s výnimkou zákonom definovaných prípadov.

Obsah

1	Úvod.....	2
2	Prehľad platforiem pre podporu webhostingu.....	4
2.1	Windows.....	4
2.2	Unix.....	4
2.3	Sun Solaris.....	5
2.4	Zhrnutie.....	5
3	Prostriedky pre vytváranie webových aplikácií.....	6
4	Open Source projekty a free software.....	7
4.1	GNU/Linux.....	7
4.2	Apache HTTP Server.....	8
4.3	MySQL Server.....	9
4.4	PHP: Hypertext Preprocessor	10
4.5	Mailserver.....	11
4.5.1	Porovnanie SMTP serverov.....	11
4.5.2	POP3/IMAP server	13
4.6	FTP server.....	13
4.6.1	Porovnanie FTP serverov.....	14
5	Návrh riešenia.....	16
5.1	Návrh aplikácie.....	16
5.2	Návrh užívateľského rozhrania.....	17
5.3	Diagram modulov aplikácie.....	18
6	Implementácia.....	19
6.1	Nastavenia webservera Apache.....	19
6.2	Nastavenia MySQL servera.....	20
6.3	Nastavenia PHP.....	22
6.4	PhpMyAdmin.....	23
6.5	Inštalácia a nastavenia servera Postfix.....	23
6.5.1	Nastavenia POP3/IMAP servera Dovecot.....	26
6.6	Nastavenia FTP servera Pure-ftpd.....	27
7	Práca so softwarom.....	30
8	Záver.....	33

1 Úvod

Používanie internetu a využívanie služieb ponúkaných na internete je v dnešnej dobe stále častejšie a internet patrí v súčasnosti bezpochyby medzi najpoužívanejšie zdroje informácií. So zvyšujúcim sa počtom užívateľov internetu, prikladajú internetu a svojej vlastnej prezentácii na internete, čoraz väčší význam aj firmy, či jednotlivci. Množstvo ľudí chce mať na internete zverejnenú vlastnú internetovú stránku, na ktorej by prezentovali svoje produkty či nápady a ktorá by im robila reklamu. Aby zvýšili svoju popularitu či zisky, firmy a jednotlivci často umožňujú užívateľom internetu a návštevníkom ich stránok, kontaktovať ich. Aby sa však zrýchlila spätná väzba medzi užívateľmi a firmami, už nepostačuje fyzická adresa a komunikácia napríklad prostredníctvom listových zásielok. V dnešnej dobe sa využíva predovšetkým elektronická adresa a komunikácia prostredníctvom emailov.

Medzi najpoužívanejšie služby na internete patria v dnešnej dobe práve dve vyššie spomenuté. Avšak nie každý, kto ich chce využívať, si ich vie sám aj zabezpečiť a následne ďalej spravovať. Tieto dôvody boli impulzom pre vznik firiem ponúkajúcich obe tieto služby bežným zákazníkom pričom od zákazníkov nie sú požadované žiadne vedomosti o daných službách. Firmám, ktoré sa zaoberajú poskytovaním takýchto služieb sa hovorí webhostingové firmy a tieto služby sa dajú nazvať súhrnným názvom webhosting.

Cieľom tejto bakalárskej práce je implementovať software pre podporu webhostingu, čiže software, ktorý využívajú aj vyššie spomenuté webhostingové firmy pre poskytovanie webhostingových služieb zákazníkom. Tento software by mal užívateľom poskytovať možnosť využívania elektronickej pošty a poskytovať diskový priestor pre dáta, ktoré si užívateľ bude želať zverejniť na internete. Dáta v tomto diskovom priestore si bude môcť užívateľ aj sám spravovať a to prostredníctvom protokolu FTP, čiže svoj diskový priestor, bude môcť využívať aj ako úložisko dát bez potreby zverejnenia dát na internete. Medzi štandardné služby ponúkané webhostingovými firmami v dnešnej dobe patri aj možnosť podpory skriptov PHP a využívanie databázového systému. Preto by náš software mal zahŕňať aj tieto služby. Okrem toho je samozrejme nutné implementovať registráciu nových užívateľov v systéme a zautomatizovať vytváranie ich účtov pre konkrétne služby. Pre správu systému administrátorom, je vhodné implementovať rozhranie pre administrátora systému, čo bude v našom softwari takisto implementované.

Práca je rozdelená do siedmych kapitol. V prvom rade bude čitateľ zoznámený s platformami, ktoré sa využívajú pre podporu webhostingu. Ďalej sa dozvie niečo o technológiách tvorby webových aplikácií, open source projektoch a nástrojoch ako aj o ich histórii. V ďalšej kapitole sa čitateľ zoznámi s návrhom riešenia aplikácie a diagram modulov aplikácie mu ponúkne približnú

predstavu fungovania aplikácie. Predposledná, najrozsiahlejšia kapitola, je venovaná samotnej implementácii softwaru a porovnaniu použitých nástrojov. V poslednej, záverečnej kapitole sú zhrnuté výsledky práce a poznatky získané pri jej tvorbe. Čitateľ v nej nájde spomenuté aj problémy, ktoré sa pri implementácii softwaru vyskytli a taktiež možné vylepšenia softwaru.

2 Prehľad platforiem pre podporu webhostingu

Zo začiatku väčšina serverov pre podporu webhostingu pracovala na platforme UNIX, ktorá disponovala výkonným a flexibilným systémom. Napriek tomu, že UNIX je stále populárny a často používaný, porovnateľnú kvalitu služieb ponúkajú aj ďalšie systémy, z ktorých najdôležitejšie budú stručne priblížené v nasledujúcom texte.

2.1 Windows

Hlavným rozdielom medzi serverovými produktami Windows a Unix je cena. Windows je komerčným produktom, zatiaľčo unixové systémy sú voľne šíriteľné. Nezanedbateľným rozdielom je však bezpečnosť Windows serverov. Keďže tieto sú orientované skôr na jednoduchosť ovládania a prijateľné užívateľské rozhranie, z bezpečnostného hľadiska za Unix systémami zaostávajú.

Na druhej strane, serverové produkty firmy Microsoft sú navrhnuté okrem iného pre podporu Microsoft aplikácií. Čo prakticky znamená bezproblémovú podporu Microsoft aplikácií ako sú napríklad Microsoft SQL Server, Microsoft Access, Microsoft FrontPage, technológia Active Server Pages (ASP), skriptovacieho jazyka Visual Basic a jazyka ColdFusion. Windows server však neobsahuje nástroje postačujúce pre administráciu týchto serverov vzdialene a preto je pre administráciu potrebné využiť ďalšie aplikácie ako napríklad PCAnywhere.¹

2.2 Unix

UNIX (Linux, FreeBSD a ich ďalšie distribúcie) je mnohostranná platforma, obzvlášť vhodná pre splnenie požiadaviek na zasielanie emailov, webové činnosti a činnosti so súbormi. Hlavnou výhodou týchto systémov okrem už vyššie spomenutého faktu, že sú voľne šíriteľné, je ich bezpečnosť, stabilita a ovládanie cez interaktívny shell v podstate z hociakého vzdialeného počítača s prístupom na internet bez nutnosti doinštalovania si ďalšej aplikácie. Unix systémy sú tiež kompatibilné s niektorými rozšíreniami a aplikáciami Microsoftu ako napríklad, MS SQL alebo Microsoft Front Page. Kompatibilita s Microsoft aplikáciami však býva často chybová a celková podpora týchto aplikácií nie je bezproblémová. Okrem vyššie spomínaných výhod je výhodou unixových systémov

¹ Viac informácií dostupných na: http://www.webdevelopersnotes.com/hosting/windows_web_hosting.php3

bezproblémová podpora PHP skriptov a skriptov jazyka Perl. Tieto technológie sú podporované aj platformou Windows, avšak ich bezchybná funkčnosť nie je zabezpečená.²

2.3 Sun Solaris

Servery Sun Solaris ponúkajú najvyššiu úroveň zdrojov a kapacity a patria k najrobustnejším serverom. Je to rozvinutá platforma a existuje veľké množstvo aplikácií a vývojových nástrojov, ktoré sú pre ňu vhodné. Vďaka svojej kapacite a stabilite je Sun ideálny pre funkcie, ktoré súvisia s vysokou návštevnosťou ako databázové servery a často navštevované webové servery. Ich použitie však nie je časté a v porovnaní s vyššie spomenutými systémami je v podstate zanedbateľné. Viac informácií možno získať na [1].

2.4 Zhrnutie

Pri použití technológií Microsoftu sa odporúča použiť pre server systém podporujúci tieto aplikácie, a teda Windows server. Avšak v prípade ak tieto technológie využité nie sú a ani nie je predpoklad ich použitia, mal by byť jasnou voľbou unixový systém. A to hlavne z už spomenutých dôvodov väčšej stability, vyššej úrovne bezpečnosti a tiež vysokých hodnôt času bezporuchovosti. Táto hodnota určuje, koľko percent času za istý meraný úsek bol server k dispozícii a teda bol schopný obslúžiť požiadavky užívateľa. Podľa štatistik³ pri unixových systémoch dosahuje hodnota času bezporuchovosti až 99,99%.

2 Zdroj: <http://www.tm4yhosting.co.za/webhosting-info/webhosting-the-right-operating-system.html>

3 Zdroj: <http://www.iaps.com/2008-server-reliability-survey.html>

3 **Prostriedky pre vytváranie webových aplikácií**

Webová aplikácia je aplikácia typu klient-server, vytvorená pre prostredie internetu alebo intranetu, ku ktorej sa pristupuje cez internetový prehliadač. Hlavným dôvodom rozšírenia webových aplikácií je prítomnosť prehliadača v každom operačnom systéme a tiež teoretická funkčnosť týchto aplikácií nezávisle na type operačného systému, na ktorom daný prehliadač (klient) beží.

Štruktúra webových aplikácií sa skladá väčšinou z troch vrstiev: prezentačnej, aplikačnej a dátovej. Prezentačnú vrstvu predstavuje internetový prehliadač. Nástroje pre generovanie dynamických stránok ako napríklad ASP, ASP.NET, CGI, PHP, Perl, Python, Java/JSP predstavujú aplikačnú vrstvu a treťou, dátovou vrstvou, sú databázy.

Princípom webových aplikácií je postupné generovanie dynamických stránok, ktoré sú podporované štandardnými prehliadačmi, a to vo formáte HTML alebo XHTML. Generovanie týchto stránok prebieha na základe dotazov poslaných prehliadačom, prvou vrstvou, vrstve druhej. Druhá, aplikačná vrstva, tieto dotazy vyhodnotí a na ich základe vykoná operácie v tretej vrstve, v databáze. Následne sa pre prehliadač vygeneruje nový obsah zobrazovaný užívateľovi. Napriek tomu, že prehliadač prijíma každú stránku ako statický dokument, postupnosť takýchto stránok meniacich sa na základe vstupov prijatých od užívateľa, pôsobí dojemom interaktivity. Pre vyššiu úroveň dynamiky stránok a zlepšenie interaktívneho dojmu z webovej aplikácie, sa používa skriptovací jazyk JavaScript alebo najnovšia technológia, ktorá využíva okrem JavaScriptu rozhranie XMLHttpRequest, takzvaná technológia Ajax. Táto technológia využíva skriptovanie na klientskej strane a umožňuje teda načítanie informácií zo servera klientským skriptom, bez nutnosti obnovenia celej stránky. Okrem spôsobu zisťovania dát zo servera prináša táto technológia aj novinku v spôsobe skriptovania. Skriptovanie v doteraz používaných a stále veľmi rozšírených technológiách ako PHP, CGI či ASP totiž prebiehalo vždy na strane servera a úlohou klienta bolo len zobrazovanie vygenerovanej stránky užívateľovi.

4 Open Source projekty a free software

4.1 GNU/Linux

Projekt GNU bol v roku 1984 založený programátorom Richardom Stallmanom a jeho cieľom bolo vytvoriť operačný systém so slobodnou licenciou, ktorý by neobsahoval žiadny zo zdrojových kódov UNIXu. Z tohto cieľa vzniklo aj samotné meno projektu, ktoré je rekurzívnou skratkou pre *GNU's Not Unix*. Cieľ projektu bol splnený v roku 1992, keď bol dokončený slobodný operačný systém, GNU/Linux, ktorý sa hovorovo označuje iba menom Linux.

Pojmom slobodný, v zmysle slobodný software alebo slobodný operačný systém, je označená hlavne myšlienka a nie cena softweru samotného. Za slobodný GNU software je možné zaplatiť, ale taktiež to nie je nutné. Podstata projektu GNU spočíva hlavne v spôsobe používania slobodného softwaru a nakladania s ním. Toto používanie možno definovať tromi hlavnými bodmi, ktoré sú zároveň aj podstatou licencie GNU General Public License (GPL). Prvým je možnosť slobodného kopírovania tohto softwaru, druhým je právo meniť program podľa vlastných prianí a potrieb za prístupu ku všetkým zdrojovým kódom programu a posledným bodom je právo rozširovať vylepšené verzie a tým pomôcť budovať komunitu.

Podobným druhom licencie je licencia public domain. Rozdiel je však v tom, že ak niekto public domain program vylepší, môže z neho spraviť proprietárny program a ďalej ho takto distribuovať. Tým pádom ďalší užívateľ tohto programu stráca možnosť program ďalej modifikovať, pretože ten už nie je public domain. Tomuto javu sa snaží zabrániť právny mechanizmus „copyleft“ vynájdený R.Stallmanom, ktorý vlastne používa „copyright“ práva, ale v obrátenom zmysle a to tak, že ktokoľvek kto distribuuje a redistribuuje zmenený software, musí ďalšiemu užívateľovi ponechať slobodu k ďalšiemu kopírovaniu a zmenám. A teda ak niekto šíri dielo s GPL licenciou a so svojimi vlastnými modifikáciami, požiadavky na distribúciu celého diela nemôžu byť akokoľvek zväčšené oproti požiadavkám, ktoré sú zahrnuté v GPL.

Medzi najvýznamnejší softvér šírený pod licenciou GPL patrí GCC (GNU Compiler Collection), Linux (jadro operačného systému) a skriptovací jazyk Perl. Licenciu GPL však využíva aj množstvo iných programov, ako o tom svedčia aj štatistiky – napríklad v roku 2009 približne 66% zo všetkých projektov slobodného softweru bolo licencovaných pod licenciami GNU⁴ a približne 71% projektov na portáli SourceForge.net v roku 2003 spadalo taktiež pod licenciu GNU GPL.⁵

4 Zdroj štatistík: <http://www.blackducksoftware.com/oss>

5 Zdroj: <http://www.dwheeler.com/essays/gpl-compatible.html>

Aktuálnou verziu licencie GPL je jej tretia verzia vydaná 29. júna 2007. Okrem tejto licencie je GNU autorom aj licencií AGPL, GFDL a LGPL (LesserGPL), ktorej použitie sa však dnes už nedoporučuje. Podrobnejšie informácie o projekte GNU a konkrétnych licenciách sú dostupné na [2].

4.2 Apache HTTP Server

Apache HTTP Server je softwerový webový server s otvoreným kódom pre Unix, FreeBSD, Linux, Windows, Solaris, Mac OS X a ďalšie platformy. Webový server, ako taký, sa stará o vybavovanie HTTP požiadaviek od klienta, ktorým je internetový prehliadač. Vybavením požiadavky sa rozumie zaslanie obvykle HTML dokumentu webovým serverom prehliadaču.

Hlavným dôvodom rozšírenia webového servera Apache a jeho použitia takmer až na 67%⁶ zo všetkých serverov a to aj napriek tomu, že súčasťou niektorých operačných systémov je iný webový server (ako napríklad webový server IIS v operačnom systéme Microsoft Windows), je okrem iného cena. Kým na to, aby bolo možné použiť webový server IIS, je nutné si operačný systém Windows zakúpiť, pre použitie webového servera Apache, nie je potrebné zakúpiť si nič, keďže ho je možné prevádzkovať na rôznych platformách a to aj na takých, ktoré možno získať bezplatne (ako napríklad Linux). Z toho vyplýva jeho ďalšia výhoda, ktorou je multiplatformnosť. Ďalšou nespornou výhodou je možnosť rozširovania jeho funkčnosti pridávaním modulov podporujúcich množstvo ďalších funkcií a tiež jeho vhodnosť pre tvorbu dynamicky generovaných stránok. Podporuje rozhranie CGI skriptov a vďaka rozšíriteľným modulom aj ďalšie skriptovacie jazyky.

Architektúra webového servera Apache je tvorená niekoľkými vrstvami. Tou základnou je jadro, ktoré zodpovedá za komunikáciu s operačným systémom, komunikáciu so sieťovým prostredím a vytvára aplikačné rozhranie pre moduly. V prípade pripojenia rozširujúcich modulov musí jadro zabezpečiť programové prepojenie týchto modulov a výmenu informácií medzi nimi.

Vlastná funkčnosť Apache servera je daná modulmi, ktoré môžu byť jeho pevnou súčasťou alebo môžu byť dynamicky pridané alebo odobrané počas prevádzky. Moduly obsahujú množstvo direktív, ktoré nastavujú konkrétne vlastnosti servera. Pre základnú funkčnosť Apache však moduly požadované nie sú. Pričom pod pojmom základná funkčnosť sa rozumie schopnosť jadra zabezpečovať výmenu zdrojov, podporovať model procesov, počúvať na komunikačných portoch protokolu TCP/IP, prenášať požiadavky HTTP a pod. Viac informácií je možné získať z [3].

Apache od verzie 1.3 podporuje okrem iného aj takzvaný „virtual hosting“, čo v podstate znamená, že jedna inštalácia Apache servera je schopná naraz obsluhovať viacero internetových stránok. Okrem toho je Apache súčasťou rôznych inštalačných balíčkov pre webový server ako LAMP alebo XAMPP, čo taktiež zvyšuje jeho popularitu. Jeho aktuálnou verziou je verzia 2.2.

6 Štatistiky na stránke: <http://news.netcraft.com>

Apache sever je licencovaný pod špeciálnou Apache licenciou, ktorá je takmer totožná s licenciou GNU GPL, avšak obsahuje body, kvôli ktorým nemohol byť Apache zahrnutý pod licenciu GPL.⁷

4.3 MySQL Server

MySQL je slobodný a otvorený viacvláknový, viacúčítateľský SQL relačný databázový server. MySQL možno prevádzkovať na viacerých platformách (napríklad Windows, Linux, Solaris) a je implementovaný vo viacerých programovacích jazykoch (ako PHP, C++, Perl). Databázový systém je relačný, typu DBMS (database management system). Každú databázu v MySQL tvorí jedna alebo viacero tabuliek, ktoré sa skladajú z riadkov a stĺpcov. V riadkoch sú uložené jednotlivé záznamy a stĺpce určujú ich dátové typy, pracuje sa s nimi ako s poľami. Práca s MySQL databázou je vykonávaná pomocou takzvaných dotazov, ktoré vychádzajú z programovacieho jazyka SQL (Structured Query Language).

Knižnice pre prácu s MySQL databázami sú dostupné vo všetkých hlavných programovacích jazykoch pomocou príslušných API funkcií. Okrem toho ODBC rozhranie s názvom MyODBC povoľuje prídavné programovacie jazyky, ktoré pomáhajú ODBC rozhraniu komunikovať s MySQL databázou, napríklad ASP alebo ColFusion.

Pre administráciu MySQL databázy je možné použiť príkazový riadok alebo si možno stiahnuť zo stránky výrobcu administrátorské nástroje ako napríklad MySQL Administrator alebo MySQL Query Browser. Oba tieto nástroje sú aktuálne zhrnuté v jednom balíčku s názvom MySQL GUI Tools. Medzi najčastejšie používané administratívne nástroje však patrí phpMyAdmin, ktorý je voľne šíriteľným webovým administrátorským rozhraním realizovaným v PHP.

O zvýšenie popularity MySQL sa stará aj jeho použitie vo webových aplikáciách a databázových komponentoch LAMP, MAMP, XAMPP a taktiež jeho nástroje na hľadanie chýb ako Bugzilla, ktoré používajú otvorené zdrojové kódy. Jeho obľúbenosť vo vzťahu k tvorbe webových aplikácií je úzko zviazaná s popularitou programovacích jazykov PHP a Ruby on Rails, ktoré sú často kombinované s MySQL serverom.

Rozdiel medzi MySQL a niektorými inými systémami spočíva v tom, že databázovému systému MySQL chýba určitá podpora dotazovania a má tiež obmedzené možnosti paralelného spracovávanía a ďalšie obmedzenia vedľajšieho charakteru. Čo sa týka paralelného prístupu k databáze MySQL, je schopná súčasne sprístupňovať desiatky procesov, ale už nie stovky. Obmedzenia MySQL však majú obvykle len malé dopady na vývoj webových databázových aplikácií. Ak však systém vyžaduje vysokú priepustnosť, vysoký počet súčasne prístupujúcich

⁷ Licencia Apache v.2.0: <http://www.apache.org/licenses/LICENSE-2.0>

užívateľov alebo je databáza veľmi často modifikovaná, je možné namiesto MySQL použiť iný databázový systém.

Tak ako mnoho ďalších softwarov, je aj MySQL distribuovaný pod dvojakou licenciou. Jednou je licencia GNU GPL, vďaka ktorej je MySQL server dostupný ako slobodný software a druhá slúži pre použitie iné než uvedené v GPL. Najnovšou verziou MySQL je verzia 5.1.30, ktorá podporuje aj predtým postrádané uložené procedúry, trigger a pohľady. Viac informácií o MySQL je dostupných v [4].

4.4 PHP: Hypertext Preprocessor

PHP (rekurzívna skratka PHP: Hypertext Preprocessor, „PHP: Hypertextový procesor“, pôvodne Personal Home Page) je skriptovací programovací jazyk, určený predovšetkým pre programovanie dynamických internetových stránok. Najčastejšie sa začleňuje priamo do štruktúry jazyka HTML, XHTML alebo WML. PHP možno použiť aj pri tvorbe konzolových a desktopových aplikácií.

PHP patrí do skupiny skriptovacích jazykov, ktoré sa vykonávajú na strane servera. PHP je na serveri závislé, pretože na ňom beží jeho interpreter, ktorý skripty vykonáva. PHP sa týmto odlišuje napríklad od JavaScriptu, ktorého skripty se sťahujú priamo s HTML stránkou a sú vykonávané na strane klienta jeho webovým prehliadačom. Výhodou PHP v tomto prípade je, že sa k zdrojovým kódom skriptov nedostane nikto, iný než autor, kým ku zdrojovému kódu JavaScriptu sa dostane každý, kto si stiahne HTML stránku, v ktorej je skript uložený. PHP však nedokáže dynamicky reagovať na udalosti vykonané klientom, pretože k vykonaniu každej novej udalosti musí byť prehliadačom znovu odoslaná požiadavka na server. Medzi výhody jazyka v PHP však patrí aj schopnosť získavať dáta z formulárov na HTML stránkach a ďalej s týmito dátami pracovať. Dokáže čítať a ukladať cookies, má veľké množstvo vstavaných funkcií a jeho dôležitou schopnosťou je aj široká spolupráca s databázami, či už staršími alebo novšími.

PHP sa stalo veľmi obľúbeným predovšetkým vďaka jednoduchosti použitia a skombinovaniu vlastností viacerých programovacích jazykov. V kombinácii s operačným systémom Linux, databázovým systémom a webovým serverom je často využívaný k tvorbe webových aplikácií. Podrobnejšie informácie o PHP, jeho nastaveniach, moduloch a direktívach možno získať v [5].

Aktuálnou verziou PHP je verzia 5.2.9, viac informácií o PHP, zoznam a popis funkcií možno nájsť na domovskej stránke projektu⁸.

8 Domovská stránka: <http://www.php.net>

4.5 Mailserver

Úlohou mailservera, inak aj SMTP servera, je zabezpečiť doručenie emailu od momentu, ako ho získal od klientskeho programu. Pre emaily vo všeobecnosti platí, že sú uchovávané v dočasných súboroch na disku, až do chvíle, kým ich SMTP server nezačne spracovávať. Následným pripojením sa na cieľový SMTP server sa SMTP server pokúsi email doručiť. Ak doručenie neprebehlo úspešne, obdrží odosielateľ email s chybovou správou.

Toto je zjednodušený popis fungovania SMTP servera. Každý SMTP server má však svoje špecifické vlastnosti, výhody aj nevýhody. Preto je dôležité vybrať si ten najvhodnejší server pre systém v ktorom ho chceme využívať. V nasledujúcej časti si preto zhrnieme dôležité informácie o najpoužívanějších SMTP serveroch, porovnáme ich vlastnosti a vyberieme si ten, ktorý nám bude najviac vyhovovať a ktorý budeme ďalej používať.

4.5.1 Porovnanie SMTP serverov

4.5.1.1 Qmail

SMTP server Qmail vznikol v roku 1996 ako moderný SMTP server ponúkajúci vysokú úroveň zabezpečenia a širokú podporu doplnkov a vylepšení tretích strán. Okrem toho vyriešil problémy súvisiace s ukladaním správ v mailbox formáte, zavedením formátu Maildir, bol rýchly, flexibilný a zahŕňal podporu virtuálnych domén. Ďalej zahŕňa POP3 server, implicitne nepodporuje TLS, SSL, spoluprácu s LDAP a ani s databázovým systémom MySQL.

Problémom Qmailu však bolo to, že od roku 1997 keď bol vydaný, nebol ďalej udržiavaný a vyvíjaný a dokonca nebolo povolené jeho vydávanie nikým iným, než autorom. To znamenalo, že každý užívateľ si síce mohol upravovať vlastnú verziu Qmailu, ale nemohol ju zverejňovať. Toto malo za následok zastavenie ďalšieho rozvoja Qmailu. Až v roku 2004 bol vydaný balíček patchov pre Qmail pod menom netqmail. Okrem neho sa objavil aj ďalší balíček pod menom qmailrocks. Kto však chce aktuálne používať Qmail ako svoj SMTP server, musí mať dobrý prehľad a vedomosti o tom, ktoré patche použiť na vyše 10 rokov starý program.

Napriek tomu však Qmail stále patrí medzi napoužívanéjšie poštové serveri, približne so 17% zastúpením na trhu. Viac informácií je dostupných na domovskej stránke projektu⁹.

4.5.1.2 Sendmail

Jeho prvá verzia bola vydaná v roku 1982, čiže v období, keď bezpečnosť nebola prioritou, čo bolo dôvodom rôznych útokov na systémy využívajúce tento SMTP server. Časom však boli tieto nedostatky odstránené a aktuálne je bezpečnosť Sendmailu na vysokej úrovni.

⁹ Domovská stránka projektu Qmail: <http://www.qmail.org/>

Sendmail podobne ako Qmail zahŕňa podporu virtuálnych domén, avšak narozdiel od Qmailu umožňuje ukladanie správ len vo formáte mailbox. Podporuje však spoluprácu s LDAP, kryptografický protokol TLS a zahŕňa aj podporu IPv6. Okrem toho dokáže Sendmail pri doručovaní pošty vyvolávať aj ďalšie programy, vďaka čomu ho možno použiť pre vytvorenie poštovej brány do iného prenosového systému ako napríklad UUCP.

Napriek tomu, že Sendmail je jedným z najpoužívanejších SMTP servrov - približne 24% podiel, má aj značné nevýhody. Najväčšou z nich je pomerne obtiažná konfigurácia servera a orientácia v konfiguračnom súbore pre neskúseného administrátora. Okrem toho je Sendmail považovaný za nedostatočne výkonný a pomalý server, v porovnaní s inými a jeho vysoké percento na trhu sa často pripisuje aj faktu, že je štandardnou súčasťou mnohých unixových systémov. Viac informácií je dostupných na domovskej stránke projektu¹⁰.

4.5.1.3 Exim

Exim bol vytvorený v roku 1995 ako reakcia na problémy Sendmailu a bol inšpirovaný programom Smail. Svojou stavbou sa Exim podobá na Sendmail, avšak jeho nástroje sú obcenejšie.

Medzi veľké výhody Eximu patrí flexibilnejšie smerovanie pošty, zahŕňa podporu LDAP a MySQL, ukladania správ vo formáte mailbox a maildir, IPv6, virtuálnych domén, TLS a SSL. Zabezpečenie Eximu dosahuje vysokú úroveň, avšak nižšiu než pri ďalšom SMTP serveri, Postfixe. Nevýhodou toho sú možné útoky na systémy využívajúce Exim, naopak výhodou je dobrá spolupráca s inými externými aplikáciami. Je veľmi kvalitne zdokumentovaný a pomerne jednoducho konfigurovateľný. Cieľom tvorcov Eximu je spraviť z neho čo najvšestrannejšie použiteľný SMTP server.

Približne s 13% podielom patrí Exim taktiež medzi najpoužívanejšie SMTP serveri. Aj keď pri porovnaní rýchlostí so servermi Qmail a Postfix zaostáva, je schopný znášať vysokú záťaž a je považovaný za spoľahlivejší a kvalitnejší server než sú servery Qmail a Sendmail. Viac informácií je dostupných na domovskej stránke projektu¹¹.

4.5.1.4 Postfix

Postfix sa používa od roku 1997, kedy vznikol a odborníci ho zaraďujú niekde medzi Exim a Qmail.

Podobne ako Exim zahŕňa podporu LDAP a MySQL, virtuálnych domén, ukladanie správ vo formátoch mailbox i maildir, TLS a SSL, no na rozdiel od Eximu nepodporuje IPv6¹². Medzi jeho výhody bezpochyby patrí aj vlastnosť, ktorou sa typicky vyznačuje a to je veľmi vysoká úroveň zabezpečenia. Tou sa podobá Qmailu, ale bezpečnosť Postfixu je ešte vyššia. To však má aj svoje

10 Domovská stránka projektu Qmail: <http://www.sendmail.org/>

11 Domovská stránka projektu Exim: <http://www.exim.org/>

12 Zdroj: http://shearer.org/MTA_Comparison

nevýhody a tou je napríklad fakt, že Postfix je menej všestranný než Exim. Kým Exim je na úkor nižšieho zabezpečenia schopný komunikovať s externými aplikáciami, Postfix práve kvoli vyššej úrovni zabezpečenia takéto možnosti neposkytuje. Medzi výhody Postfixu však možno určite zaradiť veľmi prehľadné konfiguračné súbory a jednoduchú konfiguráciu. Vo všeobecnosti je Postfix považovaný za veľmi rýchly, ak nie celkom najrýchlejší SMTP server, a tiež veľmi výkonný. Výkonnejší než napríklad Exim, no tieto rozdiely sa pri modernom hardwari a vysokom zaťažení zmazávajú.

Podieľ zastúpenia serveru Postfix na trhu je 20%, čo ho radí na druhé miesto zo všetkých používaných SMTP serverov¹³. Viac informácií je dostupných na domovskej stránke projektu¹⁴.

4.5.2 POP3/IMAP server

Ak na serveri, ktorý maily prijíma, majú užívatelia konto umožňujúce prístup k shellu, čítanie mailov nie je problémom, pretože prostredníctvom shellu je možné svoju schránku bez problémov spravovať. Náš webhostingový systém je však založený na virtuálnych užívateľoch, ktorý k shellu prístup nemajú. Prezeranie mailov a správu schránky je však aj tak potrebné zabezpečiť. Pre tento účel slúžia protokoly POP3 a IMAP a možnosť ich používania zabezpečuje POP3/IMAP server.

Vybrať POP3/IMAP server si možno z viacerých možností. Medzi najznámejšie patria Dovecot, Courier Mail Server a Cyrus IMAP. Kým druhý menovaný zahŕňa aj podporu SMTP protokolu a v našom prípade sme pre SMTP server použili server Postfix, na výber zostávajú zvyšné dva, ktoré zahŕňajú len podporu POP3 a IMAP protokolov. Pre náš systém zvolili server Dovecot¹⁵.

4.6 FTP server

Základnou funkciou, ktorú zabezpečuje FTP server, je poskytovanie dát a dátového priestoru ostatným počítačom prostredníctvom FTP protokolu. Tento protokol je založený na princípe klient-server, kde sa klient pripája k serveru a následne môže na serveri vykonávať rôzne operácie (napríklad prenos dát, výpis obsahu adresára) s dátami, ktoré sú na ňom uložené.

Z veľkého množstva FTP servrov, ktoré existujú¹⁶, zúžime náš výber na tie, ktoré spĺňajú nasledujúce kritériá: - sú voľne šíriteľné,

- bežia na platforme Linux,
- sú rozšírené a teda je predpoklad, že budú najmenej chybové.

13 Zdroj štatistík zastúpenia serverov na trhu: <http://www.mailradar.com/mailstat>

14 Domovská stránka projektu Qmail: <http://www.postfix.org/>

15 Domovská stránka projektu Dovecot: <http://www.dovecot.org/>

16 Zoznam FTP serverov: http://en.wikipedia.org/wiki/Comparison_of_FTP_server_software

4.6.1 Porovnanie FTP serverov

4.6.1.1 Proftpd

ProFTPD je multiplatformný FTP server pre Linux, BSD systémy, Cygwin, Mac OS či SunOS alebo systémy Solaris.

Je jednoducho konfigurovateľný pomocou jedného súboru, ktorý sa svojou štruktúrou podobá konfiguračnému súboru webservera Apache. Zahŕňa podporu SSL/TLS, Radius, LDAP, SQL ako aj IPv6, užívateľských kvót a obmedzení napríklad maximálnu priepustnosť. Tento FTP server predstavuje veľmi kvalitnú kombináciu bezpečnosti, stability a jednoduchej konfigurovateľnosti.

ProFTPD je veľmi kvalitný a rozšírený FTP server a na jeho vývoji a odstraňovaní chýb sa neustále pracuje. V minulosti bola jednou z jeho hlavných nevýhod bezpečnosť, tento problém však už nepretrváva, aj keď ProFTPD nie je najbezpečnejším FTP serverom na trhu. Aktuálne sa za jeho nedostatok považuje rýchlosť v porovnaní so serverom Pure-FTPd a to hlavne pri autentizácii užívateľov. Na rozdiel od ostatných FTP serverov ponúka možnosť logovania využívania šírky prenosového pásma. Viac informácií o tomto serveri ponúka jeho domovská stránka¹⁷.

4.6.1.2 Pure-ftp

Podobne ako server ProFTPD je aj tento server multiplatformný, podporujú ho BSD systémy, Linux, Irix, Mac OS či Solaris. Jeho predchodcom bol server Troll-FTPd z roku 1995.

Pure-ftp je spoľahlivý server, ktorý kladie dôraz na bezpečnosť, rýchlosť a zároveň na jednoduchosť. Jeho konfigurácia s veľmi prehľadnou dokumentáciou, ktorá je k dispozícii, je veľmi jednoduchá, takisto ako spravovanie virtuálnych užívateľov. Informácie o nich môžu byť uložené v MySQL, PostgreSQL, LDAP alebo v databáze PureDB, ktorá je vstavaná priamo do servera. Okrem toho Pure-ftp podporuje zahŕňa podporu SSL/TLS a IPv6.

Rozdiely medzi Pure-ftp a ProFTPD sú však zanedbateľné a ich funkčnosť je veľmi podobná. Hlavným faktorom pri ich porovnávaní by mali byť zrejme požiadavky klienta. Ak chce monitorovať používanie šírky prenosového pásma, bude pre neho správnou voľbou ProFTPD, pretože Pure-ftp takúto možnosť nepodporuje. Ak však preferuje vyššiu úroveň zabezpečenia a vyššiu rýchlosť, server Pure-ftp je tou správnou voľbou. Oficiálna stránka projektu poskytuje podrobnejšie informácie¹⁸.

4.6.1.3 Vsftpd

Server Vsftpd nie je podporovaný toľkými platformami ako predchádzajúce dva serveri, podporujú ho len Linux a BSD systémy.

17 Domovská stránka servera ProFTPD: <http://www.proftpd.org/>

18 Domovská stránka servera Pure-Ftpd: <http://www.pureftpd.org/>

Užívatelia tohto servera zaraďujú medzi jeho hlavné výhody bezpečnosť, rýchlosť a jednoduchosť konfigurácie. Zabezpečenie tohto servera je naozaj na veľmi vysokej úrovni a z voľne šíriteľných FTP serverov je zrejme tým najbezpečnejším. Jeho rýchlosť je takisto veľmi vysoká, no či je vyššia než rýchlosť ostatných serverov, nebolo dokázané. Jednoduchosť konfigurácie je spochybiteľná. Aj keď konfigurácia servera pre obyčajné zdieľanie je jednoduchá, ak chceme nakonfigurovať niečo zložitejšie, môžeme natrafiť na problémy, ktoré sa v niektorých prípadoch nespomínajú ani v dokumentácii tohto servera.

Napriek vyššie uvedeným faktom a skutočnosti, že vsftpd nepodporuje MySQL, je tento server taktiež pomerne rozšíreným FTP serverom. Zahŕňa taktiež podporu SSL/TLS a IPv6 a v prípade, ak máme záujem o veľmi bezpečný FTP server, bude tento server tou správnou voľbou pre nás. Viac informácií o tomto serveri je k dispozícii na jeho domovskej stránke¹⁹.

19 Domovská stránka servera Vsftpd: <http://vsftpd.beasts.org/>

5 Návrh riešenia

5.1 Návrh aplikácie

Software pre podporu webhostingu bude využívať nasledujúci software:

Webserver: Apache (viac informácií je uvedených v kapitole 6.1)

Mailserver: Postfix (viac informácií je uvedených v kapitole 6.5)

FTP server: Pure-ftp (viac informácií je uvedených v kapitole 6.7)

POP3/IMAP Server: Dovecot (viac informácií je uvedených v kapitole 6.6)

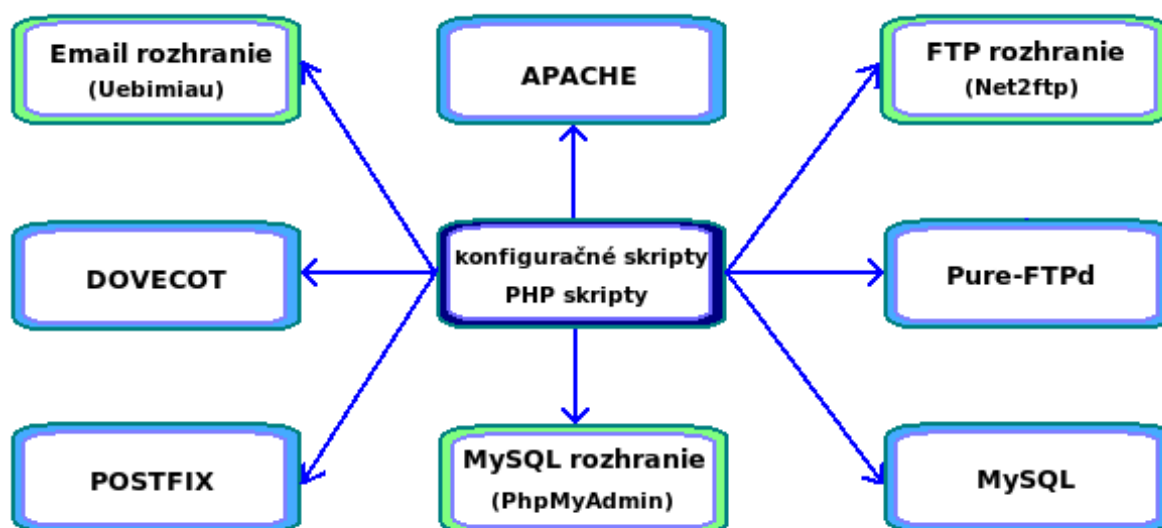
Databázový server: MySQL (viac informácií je uvedených v kapitole 6.2)

Skriptovací jazyk: PHP (viac informácií je uvedených v kapitole 6.3)

Rozhranie pre email: Uebimiau (viac informácií je uvedených v kapitole 6.6)

Rozhranie pre FTP: Net2ftp (viac informácií je uvedených v kapitole 6.7)

Rozhranie pre MySQL: PhpMyAdmin (viac informácií je uvedených v kapitole 6.4)



Obrázok 5.1: Súčasti softwaru pre podporu webhostingu

Aplikácia bude implementovaná a riadená dvoma druhmi skriptov: - PHP skriptami

- konfiguračnými skriptami

PHP skripty budú implementovať hlavne užívateľské rozhranie. Okrem toho budú sprostredkovať informácie konfiguračným skriptom. Napríklad pri registrácii účtu, budú využité PHP skripty, ktoré túto registráciu sprostredkujú. Uložia záznamy do databázy, ak sa v nej záznam

s rovnakými údajmi ešte nenachádza a zašlú užívateľovi email, na emailovú adresu povinne zadanú pri registrácii, s odkazom pre potvrdenie. Keď užívateľ na túto linku klikne a potvrdí tým registráciu, PHP skript predá údaje o ňom konfiguračnému skriptu. Popritom sa samozrejme kontroluje iným PHP skriptom korektnosť zadaných údajov a to či boli zadané všetky povinné údaje.

Konfiguračné skripty na základe informácií, ktoré obdržali (napríklad login atď.), vytvoria na serveroch účty pre daného užívateľa – vytvoria pre neho emailovú schránku, účet na FTP serveri, priestor v MySQL databáze a podobne. Následne budú všetky údaje o emailovej adrese, adrese na FTP server, MySQL server a heslo vygenerované PHP skriptom, užívateľovi prehľadne zobrazené.

Po úspešnej registrácii a vytvorení všetkých účtov, sa bude môcť užívateľ prihlásiť do systému. Po prihlásení by mal možnosť vidieť svoj emailový a svoj FTP adresár. Mali by mu byť ponúknuté prehľadne zobrazené informácie o obsadenom diskovom priestore, informácie o súboroch, prípadné obmedzenia a podobne. Okrem toho by mal vedieť svoje FTP konto jednoducho spravovať. Nahrávať a vymazávať súbory prípadne s nimi inak manipulovať. Toto všetko by malo byť taktiež implementované pomocou PHP skriptov.

Ďalšou úlohou konfiguračných skriptov okrem vytvorenia účtov – emailového, FTP, MySQL; bude aj nastavenie veľkosti diskového priestoru – pre všetkých užívateľov o rovnakej veľkosti, nastavenie kvót a práv pre užívateľské účty – aby užívatelia nemali prístup k súborom a adresárom iných užívateľov, prípadne k adresárom samotného systému.

5.2 Návrh užívateľského rozhrania

Rozhranie aplikácie by sa malo skladať z troch častí.

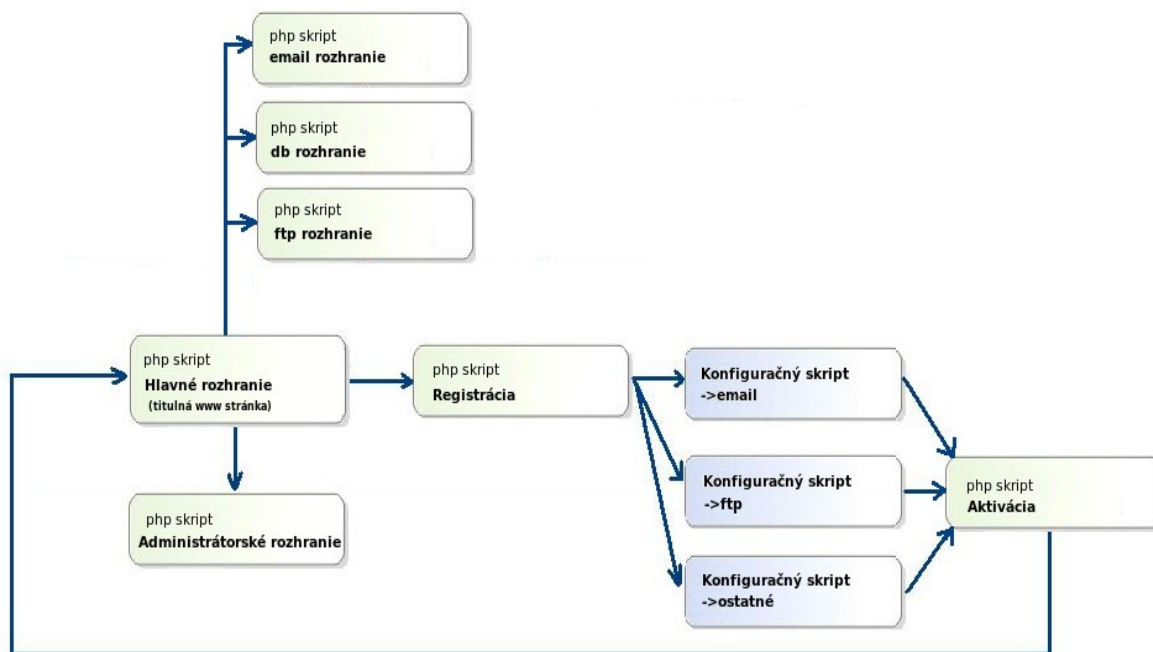
Prvou časťou bude samotné rozhranie pre registráciu užívateľov a ich prihlasovanie. Nemalo by sa jednať o graficky zložené prostredie, ale skôr jednoduché a prehľadné, poskytujúce všetky potrebné informácie, možnosť prihlásenia sa k jednotlivým službám na základe loginu a hesla, a tiež možnosť registrácie nového užívateľa prostredníctvom zadania údajov do registračného formulára.

Ďalšou časťou bude webové užívateľské rozhranie, ktoré sa bude zobrazovať po prihlásení. Každá zo služieb bude mať zrejme svoje vlastné rozhranie oddelené od rozhrania inej služby. Pôjde teda o celkovo tri rôzne rozhrania, ktorých hlavnou úlohou bude zobrazovať užívateľovi potrebné informácie. Pre prípad využívania služieb FTP servera, by správa súborov mala predstavovať zoznam súborov, ktoré užívateľ na daný server nahral, informácie o nich - ich veľkosť, čas kedy boli užívateľom nahraté, ich názvy a podobne. Ďalej by tu mala byť možnosť súbory nahrávať alebo vybrať niektoré a tie vymazať, prípadne ich modifikovať. Rozhranie pre prístup k emailovému kontu užívateľa by malo poskytovať hlavne prehľad správ a možnosť písania a odosielania nových.

Rozhranie pre správu databázy užívateľa by malo byť schopné obslúžiť všetky základné operácie týkajúce sa databázy. Pre zjednodušenie využijeme rozhrania, ktoré sú k dispozícii na internete a sú voľne šíriteľné. Tieto rozhrania sú popísané v kapitolách 6.4, 6.6, 6.7.

Tretou časťou rozhrania bude rozhranie pre administrátora serveru. Administrátor by sa ku službám mohol prihlasovať podobným spôsobom ako všetci ostatní užívatelia. Prihlásenie k správe užívateľov by prebiehalo cez odkaz na hlavnej stránke alebo iným spôsobom, napríklad pomocou URL. Následne by sa zobrazilo rozhranie pre správu užívateľov. Toto rozhranie by pozostávalo zo zoznamu užívateľov a informácií o nich ako napríklad ich emailové adresy, názvy subdomén atď. Taktiež by tu bola možnosť užívateľa vymazať prípadne upraviť informácie o ňom. Okrem toho by tu mohli byť celkové informácie o aktivite na FTP serveri, počte registrovaných užívateľov a podobne.

5.3 Diagram modulov aplikácie



Obrázok 5.2: Diagram modulov aplikácie

6 Implementácia

Táto kapitola popisuje implementáciu a konfiguráciu jednotlivých služieb, ktoré spoločne vytvoria samotný systém pre podporu webhostingu. Postupne budú popísané všetky komponenty v postupnosti, ktorú považujeme za zmyslupnú a správnu vzhľadom na ďalšiu bezproblémovú funkčnosť tohto systému. Všetky skripty, ktoré budú spomenuté v tejto kapitole, sa nachádzajú v prílohe B.

6.1 Nastavenia webservera Apache

Z dôvodov uvedených v kapitole 4.2 sme si pre vytváranie systému vybrali webserver Apache, konkrétne jeho najnovšiu verziu 2.2 prístupnú na domovskej stránke projektu.²⁰

Pred samotnou inštaláciou je potrebné zdrojové súbory skonfigurovať s modulmi pre rozšírenie základnej funkčnosti servera. To, ktorými modulmi jeho funkčnosť rozšírime, je uvedené v konfiguračnom skripte `apache_install`. Najdôležitejšie z nich si v krátkosti popíšeme:

- `mod_so`: modul poskytujúci serveru direktívu `LoadModule`, ktorá bude potrebná pre spúšťanie PHP skriptov v prehliadači. Bez tejto direktívy, by PHP skripty boli namiesto načítania sťahované ako hociaké iné súbory.
- `mod_env`: zabezpečuje predanie premenných prostredia do CGI a SSI skriptov
- `mod_setenvif`: zabezpečuje nastavenie premenných na základe informácií od klienta
- `mod_mime`: zabezpečí nastavenie MIME typu súboru pomocou jeho prípony
- `mod_mime_magic`: podobne ako `mod_mime`, nastavenie však prebieha na základe obsahu súboru
- `mod_headers`: zabezpečuje správu a úpravu hlavičiek protokolu HTTP
- `mod_access`: zaistí obmedzený prístup klientov k určitému hostiteľskému názvu alebo IP adrese
- `mod_alias`: prostredníctvom tohto modulu je možné používať presmerovanie URL k mapovaniu rôznych súčastí hostiteľského názvu k odlišným umiestneniam v strome dokumentov
- `mod_vhost_alias`: zabezpečuje podporu dynamickej konfigurácie virtuálnych hostov

Táto časť bola prevzatá z [5], kde sa možno dočítať o ostatných nastaveniach.

Po konfigurácii a následnej inštalácii by mal byť Apache webserver pripravený k použitiu. Kedže však chceme poskytovať prostredníctvom nášho systému služby viacerým užívateľom, bude potrebné zabezpečiť, aby bol náš webserver schopný hostiť viac doménových názvov. Čo zjednodušene znamená, že keď do prehliadača zadáme rôzne adresy, pre každú sa nám môže zobrazit' iný obsah stránky. Toto je umožnené aj vďaka tomu, že každý hostiteľ môže mať svoj vlastný

²⁰ Domovská stránka projektu Apache: <http://www.apache.org/>

takzvaný DocumentRoot, čo je v podstate adresár, v ktorom sú uložené súbory zobrazujúce sa prostredníctvom klienta po zadaní adresy daného hostiteľa. Informácie o tom, ktorý hostiteľ má mať svoj vlastný DocumentRoot, kde ho má webserver hľadať a ďalšie doplňujúce nastavenia pre každého hostiteľa, patria ku konfiguračným nastaveniam každého virtuálneho hostiteľa osobitne. Môžu sa nachádzať v konfiguračnom súbore httpd-vhosts.conf alebo pre prehľadnejšiu správu a administráciu v špeciálnom adresári, kde budú informácie o každom virtuálnom hostiteľovi vo zvláštnom súbore a v súbore httpd-vhosts.conf bude uvedená len cesta k adresáru, v ktorom sa majú konfiguračné súbory hostiteľov hľadať. Po týchto nastaveniach by virtuálni hostitelia už mohli byť teoreticky funkční. Prakticky je však potrebné urobiť ešte jednu úpravu systémového súboru. Keďže je tento systém zhotovovaný pre využívanie na localhoste, teda na stanici, ktorá nie je registrovaná v žiadnom DNS systéme, virtuálni hostitelia nebudú použiteľní. Ak by náš systém bol registrovaný v DNS systéme a mal na ňom svoje A-záznamy, v ktorých by boli doménové mená mapované na IP adresu nášho systému, nebolo by problémom rozlišovať doménové mená, pre ktoré je náš systém serverom. Keďže však takúto možnosť nemáme, pre testovacie účely si vystačíme so súborom /etc/hosts, v ktorom budeme doménové mená mapovať na IP adresu (alebo na viac IP adries, ak máme k dispozícii viac sieťových rozhraní). Táto alternatíva je však pre iné ako testovacie účely nevhodná. Dôvodom je fakt, že právo modifikácie tohto súboru má len správca operačného systému. Ak by sa nám teda aj podarilo zautomatizovať modifikáciu tohto súboru pomocou skriptov spúšťaných automaticky pri registrácii do systému, vznikla by nebezpečná bezpečnostná diera. Ak by niekto prostredníctvom týchto skriptov, získal práva správcu operačného systému, bolo by to veľmi nebezpečné. Z tohto dôvodu budeme v našom systéme pre testovacie účely modifikovať súbor /etc/hosts prostredníctvom manuálneho spustenia skriptu správcou operačného systému. Tento skript bude pre každého užívateľa špecifický, vytvorí sa na základe informácií o užívateľovi zadaných pri registrácii do webhostingového systému.

Vyššie uvedený spôsob vytvárania virtuálnych hostiteľov samozrejme predpokladá nastavenie virtuálneho hostiteľa podľa mena. Čiže mapovanie viacerých mien na jednu IP adresu. Ďalšou možnosťou je nastavenie virtuálneho hostiteľa podľa IP adresy. Táto varianta si však vyžaduje mapovanie každého doménového mena na osobitnú IP adresu, preto ju v našom systéme využívať nebudeme.

6.2 Nastavenia MySQL servera

Databázový server MySQL nám umožňuje ukladanie dát spôsobom, vďaka ktorému je možné mať k týmto dátam rýchly prístup pre následné opätovné používanie. Túto vlastnosť pri tvorbe nášho

systému využijeme a v tabuľkách databáze budú uložené údaje potrebné pre korektné fungovanie ďalších služieb, ktoré má náš webhostingový systém podporovať.

Z domovskej stránky²¹ si stiahneme najnovšiu verziu MySQL servera, ktorou je aktuálne verzia 5.0 a podobne ako pri webserveri, bude pred inštaláciou potrebné skonfigurovať zdrojové súbory. Nato nám poslúži už pripravený konfiguračný súbor `conf_mysql`. Najskôr sa v tomto súbore nastavujú premenné prostredia `CFLAGS` a `CXXFLAGS` kvôli práci s kompilátorom GCC:

```
CFLAGS="-O3" CXX=gcc CXXFLAGS="-O3 -mpentiumpro -mstack-align-double -felide-constructors -fno-exceptions -fno-rtti"
```

Vďaka týmto nastaveniam kompilátor GCC použije iba GCC a nebude využívať napojenie na `libg++` a `libstdc++`. Toto nastavenie nie je nevyhnutné, avšak zabráni prípadným náhodným zlyháním serveru MySQL, ku ktorým môže dôjsť, ako sa uvádza v [5]. Okrem týchto nastavení obsahuje tento súbor konfiguračné direktívy inštalácie:

- `with-client-ldflags=-all-static` a `with-mysqld-ldflags=-all-static`: použitím oboch direktív súčasne sa dosiahne skompilovanie klienta a servera MySQL ako staticky prepojených programov, čím sa dosiahne vyššia rýchlosť systému. Tieto dve direktívy sú teda pomerne dôležité a ich použitie sa odporúča.
- `enable-thread-safe-client`: zabezpečí skompilovanie klienta s vláknami, čo je dôležité hlavne ak klient, ktorého využívame používa vlákna
- `enable-assembler`: zabezpečí používanie assemblerovských verzií niektorých reťazcových funkcií, ak tie sú k dispozícii
- `disable-shared`: zabráni zostaveniu zdieľaných knižníc

Viac informácií o konfiguračných direktívach inštalácie servera MySQL verzie 5.0 možno nájsť na [6].

Po skonfigurovaní zdrojových súborov a pred samotnou inštaláciou je potrebné vykonať ešte jeden krok. Tým krokom je vytvorenie užívateľa a skupiny, do ktorej bude tento užívateľ patriť. Vytvorenie užívateľa je potrebné z bezpečnostných dôvodov. `Mysqld`, čo je démon MySQL, potrebuje užívateľa a skupinu, pod ktorou bude pracovať. Keby sme ho nechali pracovať pod užívateľom správca systému, bolo by to nebezpečné z dôvodu možnosti napadnutia celého nášho systému. Preto užívateľ a skupina vytvorení pre démona MySQL bude spĺňať len túto funkciu a z bezpečnostných dôvodov ho inde využívať nebudeme. Z pochopiteľných dôvodov následne po inštalácii bude potrebné zmeniť vlastníka adresárov, do ktorých sa MySQL nainštaloval. Umiestnenie, kde sa tieto adresáre budú po inštalácii nachádzať, sme si taktiež určili v našom konfiguračnom skripte.

Splnením predchádzajúcich krokov získame nainštalovaný MySQL server. Pre jeho využívanie je potrebné vytvoriť si užívateľa s plnými právami pre celý server. Tento užívateľ bude

21 Domovská stránka: <http://dev.mysql.com>

mať právo spravovať všetkých užívateľov a ich databázy na serveri. Keďže náš webhostingový systém bude čerpať informácie o užívateľoch z databázy MySQL a bude musieť byť schopný do nej údaje pridávať, meniť ich, mazať ich a pridávať užívateľov, ktorí budú vlastníelmi svojich databáz, budeme na tieto funkcie využívať práve užívateľa, ktorý vlastní práva pre celú správu MySQL servera. V našom systéme si vystačíme s dvomi hlavnými tabuľkami pre správu užívateľov a s ďalšími poskytujúcimi štatistické údaje. Tabuľky slúžiace pre správu resp. evidenciu užívateľov a ich overovanie pri prihlasovaní sa k danej službe budú, ako už bolo spomenuté, dve. Jedna pre mailserver a druhá pre ftpserver. Aj keď tieto tabuľky obsahujú niektoré údaje duplicitne, je to z toho dôvodu, že ako pri mailserveri, tak pri FTP serveri možno využívať overovanie užívateľov na základe údajov, ktoré nemusia byť nevyhnutne uložené v databáze MySQL. A teda aj keby sme sa pri niektorom z daných serverov rozhodli pre iné umiestnenie autentizačných údajov, tabuľka v databáze MySQL slúžiaca pre overovanie užívateľov druhého servera/služby, by sa dala naďalej využívať. Tieto tabuľky ako aj tabuľky štatistík, aj s obsahom testovacích dát, sú uložené v priloženom súbore db.sql. (viď príloha B)

6.3 Nastavenia PHP

Preto aby na našom systéme mohli byť interpretované PHP skripty, je na náš server potrebné nainštalovať podporu PHP.

Samotnej inštalácii bude predchádzať inštalácia doplnkových knižníc, ktoré sú potrebné pre správne fungovanie nášho systému. Knižnica Mcrypt a s ňou súvisiaca knižnica libmcrypt sú náhradou starého balíčka crypt(), ktorý slúžil pre podporu šifrovania unixových súborov. Knižnica Mcrypt je teda potrebná pre šifrovanie a dešifrovanie citlivých dát rôznymi kryptografickými metódami ako napríklad DES, Blowfish a iné. Ďalšou knižnicou, ktorú bude potrebné nainštalovať, je knižnica Mhash. Používa sa pri vytváraní kontrolných súčtov, autentizačných kódov správ a podobne. Pre tento účel ponúka veľké množstvo hashovacích algoritmov ako napríklad MD5, SHA1. Viac informácií o týchto knižniciach možno získať na [7] a [8].

Po nainštalovaní týchto knižníc skonfigurujeme zdrojové súbory balíčka PHP. K správne skonfigurovaniu nám pomôže priložený konfiguračný skript php_install, ktorý obsahuje nasledujúce prepínače, ktoré sú dôležité pre správnu funkčnosť nášho systému:

- with-apxs2 : tento prepínač využíva rozšírenie Apache SAPI, zaisťujúce interakciu s webovým serverom, a zabezpečuje kompiláciu PHP ako zdieľaného objektu pre Apache verzie 2.0
- with-mysql : určuje umiestnenie inštalácie MySQL
- with-mcrypt : určuje umiestnenie inštalácie mcrypt, ktorá nahrádza balíček crypt()

- enable-ftp : aktivuje klientské funkcie použiteľné pre komunikáciu s FTP servermi, čo pre nás bude dôležité z hľadiska komunikácie užívateľov s FTP serverom cez internetový prehliadač prostredníctvom webového rozhrania
- enable-trans-sid : povoľuje podporu transparentného identifikátora sedenia; táto vlastnosť je od verzie 4.2 podporovaná implicitne, čiže prepínač nie je potrebné uvádzať; my ho uvádzame z dôvodu možnosti použitia tohto konfiguračného skriptu aj pre staršie verzie PHP

S podrobnejšími informáciami o konfiguračných voľbách sa možno oboznámiť na stránke venovanej tejto problematike²².

Po následnej inštalácii bude webserver schopný spolupracovať s PHP a v tom prípade by sme v prehliadači mohli sledovať obsahy stránok vygenerované našimi php skriptami. Z bezpečnostného hľadiska však ešte pracovať nebude. Aj keď už bude Apache schopný spolupracovať s PHP, nebude schopný zobrazovať stránky. Dôvodom je blokovanie službou SELinux, ktorá tomu zabráni. Konkrétne zabraňuje spolupráci webservera s knižnicou libphp5.so, ktorá je však pre korektnú funkčnosť potrebná. Zmenou bezpečnostného kontextu spomínanej knižnice príkazom „chcon -t texrel_shlib_t /cesta/k/suboru/libphp5.so“ dosiahneme želaný efekt a plnú funkčnosť servera Apache s podporou PHP.

6.4 PhpMyAdmin

Kedže náš systém pre podporu webhostingu má užívateľom umožňovať vytváranie a administráciu databáz, bude potrebné vyriešiť prístup užívateľov k MySQL serveru. Pretože sa v našom systéme nebude jednať o systémových užívateľov, ale o užívateľov virtuálnych, nebudú mať prístup k príkazovému riadku a z toho dôvodu ani možnosť administrácie svojej databázy pomocou neho. Okrem toho by správa databáz prostredníctvom príkazového riadku nebola užívateľsky prijateľným riešením.

Riešením tohto problému je rozhranie pre správu MySQL databáz prístupné cez internetový prehliadač, ktoré je voľne prístupné na internete. Tento nástroj sa nazýva PhpMyAdmin a pre svoje spustenie a funkčnosť vyžaduje nainštalovaný Apache server s podporou PHP a MySQL server.

6.5 Inštalácia a nastavenia servera Postfix

Na základe vyššie uvedeného prehľadu a vzhľadom na vyššiu úroveň zabezpečenia serveru Postfix a s prihliadnutím na to, že neplánujeme využívať žiadne externé aplikácie, sme sa rozhodli pre využitie serveru Postfix ako mailserveru pre náš systém.

²² Konfiguračné voľby PHP: <http://php.biz.net.id/manual/sk/configure.php>

Pred procesom samotnej inštalácie poštového serveru Postfix, je potrebné zostaviť takzvaný súbor Makefile, ktorý je využívaný nástrojom make, pre určenie podmienok, závislostí a volieb používaných pri zostavovaní balíčka. Keďže vieme, že informácie slúžiace pre overovanie užívateľov nášho systému pri prihlasovaní budú uložené v databáze MySQL, bude potrebné Postfix nainštalovať s podporou MySQL systému. Nasledujúcim príkazom :

```
make -f Makefile.init makefiles 'CCARGS=-DHAS_MYSQL -I/cesta/k/hlavičkovým/súborom/mysql'
'AUXLIBS=-L/cesta/k/inštalácii/mysql -lmysqlclient -lz -lm'
```

, zostavíme Makefile, ktorý bude obsahovať voľby pre inštaláciu Postfixu s podporou MySQL. Premenná AUXLIBS v príkaze určuje umiestnenie, kde má linker vyhľadávať knižnice, ktoré sa nenachádzajú na štandardnom mieste. Premenná CCARGS predáva kompilátoru informácie o tom, kde má hľadať hlavičkové súbory (v našom prípade systému MySQL), ak sa nenachádzajú na štandardnom mieste, ktorým je adresár /usr/include. Preto, aby bol Postfix s podporou MySQL nainštalovaný správne, je potrebné pre všetky prídavné balíčky nadefinovať konkrétne makro (pre systém MySQL to bude makro HAS_MYSQL), na základe ktorého Postfix vie, že má toto makro použiť pri zostavovaní. Voľba -D umožňuje definovanie makra už v čase kompilácie samotného Postfixu.

Podobne ako pri systéme MySQL, je aj teraz pred samotnou inštaláciou potrebné vytvoriť vyhradený účet pre užívateľa a skupinu. Užívateľský účet bude slúžiť serveru Postfix, bude vlastníkom jeho *fronty* a väčšiny jeho procesov. Z bezpečnostných dôvodov nebude tento účet umožňovať prihlásenie, nebude mať možnosť využívať shell a taktiež nebude mať svoj domovský adresár. Účet pre skupinu bude špecifický tým, že do tejto skupiny nebude patriť žiadny užívateľský účet, dokonca ani účet vyhradený pre server Postfix.

Po vykonaní týchto krokov, je možné spustiť samotnú inštaláciu serveru Postfix. V rámci inštalácie sa inštalačný proces bude dotazovať na informácie o nastaveniach Postfixu v systéme. V našom prípade bude pre každý z týchto dotazov postačujúce jeho potvrdenie, pretože implicitné nastavenia ich hodnôt budú nášmu systému vyhovovať. Viac informácií o tomto procese možno získať na oficiálnej stránke projektu v časti venovanej inštalácii [9].

Ak inštalácia prebehla úspešne, nasledujúcim krokom bude konfigurácia serveru Postfix. Pre správnu funkčnosť SMTP servera Postfix v našom systéme, bude dôležité upraviť niekoľko hlavných nastavení v konfiguračných súboroch inštalácie servera, ktoré si v krátkosti popíšeme:

- myhostname: tento konfiguračný parameter nastavuje plne kvalifikovaný názov hostiteľa systému, ak tento parameter nastavený nebude, Postfix použije názov hostiteľa hlásený systémom. Niektoré systémy však nemusia byť dobre nastavené alebo nemusia poskytovať plne kvalifikovanú verziu názvu hostiteľa, preto je dobré tento parameter zadávať.

- mydomain : nastavuje názov domény systému, ak tento parameter ostane prázdny, bude automaticky nastavený systémom Postfix na hodnotu myhostname bez prvého komponentu
 - myorigin : určuje, aký doménový názov sa má pridať pri odosielaní alebo prijímaní emailov; štandardne sa používa hodnota uvedená v parametri myhostname; priradením mydomain do parametra myorigin tomuto však zamedzíme a adresa bude obsahovať, ako to býva štandardom, názov domény a nie plne kvalifikovaný názov hostiteľa
 - virtual_transport : určuje typ transportu, ktorý má byť používaný pre doručovanie správ do virtuálnych schránok
 - virtual_uid_maps a virtual_gid_maps: určuje vlastníka a skupinu, ktoré používa Postfix pri doručovaní do súborov virtuálnych schránok, zadáním druhu mapy static (static:hodnota_uid resp. static:hodnota_gid), zaistíme, že ten istý užívateľ patriaci do istej skupiny vyhradenej pre neho, bude vlastníkom všetkých virtuálnych schránok
 - virtual_mailbox_base : určuje základný adresár úložiska správ virtuálnych účtov; všetky súbory schránok sú uložené relatívne voči tomuto adresáru
 - virtual_mailbox_maps : vyhľadávací súbor, ktorý mapuje emailové adresy virtuálnych schránok na zodpovedajúce súbory schránok; mapovanie bude uložené v databáze MySQL, úlohou tohto súboru bude poskytnúť Postfixu informácie pre naviazanie spojenia so systémom MySQL a následné zisťovanie potrebných dát o schránkach užívateľov
 - virtual_mailbox_domains : obsahuje názvy domén s virtuálnymi schránkami
 - luser_relay : v tomto parametri nastavíme, meno užívateľa, ktorý bude prijímať poštu, ktorá je adresovaná neznámym užívateľom. Tento a nasledujúci parameter bude v konfiguračnom súbore zahrnutý iba v prípade, ak v našom systéme chceme zachytávať správy adresované neexistujúcim užívateľom. Implicitne však Postfix takéto správy odmieta, v tom prípade tieto dva parametre neuvádzame.
 - local_recipient_maps : aby server poštu adresovanú neznámym užívateľom neodmietal, je potrebné tento parameter nechať prázdny
 - relayhost : tento konfiguračný parameter bude opísaný podrobnejšie v nasledujúcej časti
- Táto časť bola čiastočne prevzatá z [10].

Spracovanie správy serverom Postfix sa skladá z viacerých častí a zapája sa do neho viacero démonov, ktorí sú súčasťou Postfixu. Po zostavení správy užívateľom a jej následnom odoslaní, zavolá emailový klient postfixový príkaz sendmail. Následne démon pickup túto správu vyzdvihne a po preverení ju predá ďalej démonovi cleanup. Ten zaistí jej finálne spracovanie a ak nebol zadáný odosielateľ správy alebo v adrese nebol použitý kvalifikovaný názov hostiteľa, cleanup démon tieto údaje doplní. Nasleduje predanie správy správcovi fronty, démonovi qmgr, a upozornenie správcu na

to, že je k doručeniu pripravená nová správa. Ak je správca fronty pripravený na spracovanie novej správy, presunie túto správu do aktívnej *fronty*, *active*. Ak je daná správa určená pre virtuálneho užívateľa na lokálnom systéme, bude pomocou démona *virtual* doručená do úložiska správ na danom systéme. Ak je však táto správa určená pre užívateľa z vonkajšieho systému, správca *fronty* vyzve démona *smtp* k jej doručeniu. Náš systém však v tomto prípade nebude schopný doručiť správu adresátovi, pretože nie je adresátom nachádzajúcim sa na lokálnom systéme a zároveň náš server nie je registrovaným serverom v systéme DNS a preto nemá oprávnenie pre odosielanie emailov adresátom umiestneným na vonkajších systémoch. Z dôvodu, ak z nášho systému chceme odosielať emaily adresátom na vonkajších systémoch, je nutné zadať konfiguračný parameter *relayhost*, ktorý bude obsahovať názov nadradeného SMTP servera, prostredníctvom ktorého sa budú emaily preposielať. Takýmto nadradeným mailovým serverom väčšinou býva server poskytovateľa internetu. Konkrétne ním môže byť napríklad server *out.smtp.cz*, ktorý však vyžaduje registráciu a poplatky za danú službu. V sieti KolejNet túto úlohu spĺňa server *smtp.kn.vutbr.cz*, ktorý však funguje iba v danej sieti. Pre správne fungovanie nášho systému pre podporu webhostingu teda bude potrebné kontaktovať poskytovateľa internetu a zistiť možnosti týkajúce sa serveru určeného pre odosielanie emailov alebo využiť spoplatnené služby ponúkané na internete.

6.5.1 Nastavenia POP3/IMAP servera Dovecot

Aby boli dodržané už spomenuté skutočnosti, že informácie o užívateľoch systému sa budú čerpať z databázy MySQL, je potrebné Dovecot nainštalovať s podporou MySQL. Táto podpora nie je zabudovaná implicitne, avšak Dovecot ju umožňuje.

Po úspešnej inštalácii bude potrebné vytvoriť konfiguračný súbor, ktorý bude obsahovať informácie, na základe ktorých sa bude POP3/IMAP server vedieť spojiť s MySQL serverom, bude schopný vybrať tabuľku, z ktorej má čerpať informácie a následne bude tieto informácie, ktorými budú prihlasovacie meno a heslo užívateľa, z tabuľky databázy získať. Ďalší konfiguračný súbor, ktorý vznikne po inštalácii, bude potrebné upraviť. Najdôležitejšie bude určiť spôsob šifrovania dát v direktíve *mechanisms* a určiť, na základe čoho má server Dovecot hľadať informácie o užívateľoch, čiže mu zadať cestu k vyššie spomínanému súboru, na základe ktorého bude môcť čerpať informácie z databázy MySQL. Okrem toho bude dôležité upraviť aj niekoľko nasledujúcich direktív:

- *protocols* : určí, ktoré protokoly bude možné využívať pre prístup k mailovým schránkam
- *disable_plaintext_auth* : z dôvodu bezpečnosti je potrebné zakázať autentifikáciu prostredníctvom informácií v *plaintexte*
- *mail_location* : určuje umiestnenie mailových schránok v adresárovej štruktúre systému
- *mail_privileged_group* : určuje skupinu, ktorá má prístup ku schránkam; bude sa jednať o tú istú skupinu, ktorá je určená konfiguračným parametrom *virtual_gid_maps* serveru Postfix

- `first_valid_uid`, `last_valid_uid`, `first_valid_gid`, `last_valid_gid` : vymedzuje rozmedzie UID/GID, pre ktoré bude umožnené prihlásenie sa k prezeraniu mailových schránok; v našom prípade pre testovacie účely postačí, ak prvé dve direktívy nastavíme na hodnotu `virtual_uid_maps` serveru Postfix a ďalšie dve na hodnotu `virtual_gid_maps`

Po vykonaní týchto zmien by mal byť náš POP3/IMAP server plne funkčný. Vzhľadom na spôsob, akým budú užívatelia s celým našim webhostingovým systémom komunikovať, čiže cez internetový prehliadač, je potrebné zabezpečiť prístup k mailovým schránkam užívateľov, taktiež cez prehliadač. Na interne te je k dispozícii viacero webových rozhraní pre prístup k mailom, avšak kvalitatívne je medzi nimi markantný rozdiel. V tejto práci bolo použité rozhranie Uebimiau, ktoré je voľne prístupné a viac informácií o ňom možno získať na jeho domovskej stránke²³.

6.6 Nastavenia FTP servera Pure-ftpd

Vzhľadom nato, že informácie o užívateľoch nášho systému budú uložené v databáze MySQL a informácie o využívaní šírky prenosového pásma pre nás dôležité nebudú, naopak rýchlosť áno, zvolili sme si na základe vyššie uvedeného prehľadu pre FTP server v našom systéme server Pure-ftpd.

Pred inštaláciou Pure-ftpd servera bude potrebné zdrojové súbory skonfigurovať s rozšíreniami, ktoré si zvolíme. Pre náš webhostingový systém budú nevyhnutné rozšírenia, zabezpečujúce podporu MySQL systému, virtuálnych užívateľov, rozšírenia ftpwho a takzvaného chroot-u. Rozšírenie ftpwho nie je nevyhnutne potrebné, avšak administrátorovi systému umožní sledovať, čo sa na FTP serveri deje, ktorí užívatelia sú prihlásení a akú činnosť aktuálne vykonávajú. Podpora nastavenia chroot taktiež nie je nutne nevyhnutná. V našom systéme chceme zabezpečiť, aby mal každý z užívateľov prístup iba k vlastnému adresáru na FTP serveri, z dôvodu aby nemohol zasahovať a ani si prezerať adresáre ostatných užívateľov alebo adresáre operačného systému. Pre splnenie tejto požiadavky je chroot nevyhnutný. Toto nastavenie totiž zabezpečí, že pre každého užívateľa FTP servera, bude jeho domovský adresár zároveň aj koreňovým adresárom celého systému. A keďže nad koreňovým adresárom v adresárovej štruktúre sa už nič nenachádza, užívateľ nebude mať prístup k žiadnemu inému adresáru.

Nasledovať bude samotná inštalácia servera, po ktorej bude potrebné tento server nakonfigurovať. Konfiguračný súbor zahŕňa veľké množstvo konfiguračných možností, nám však pre náš systém postačí zlomok z nich, ktoré si stručne popíšeme:

23 Domovská stránka projektu: <http://www.uebimiau.org/>

- ChrootEveryone : aktivuje/deaktivuje vyššie popísané nastavenie chroot
- MaxClientsNumber : definuje maximálny počet súčasne prihlásených užívateľov
- MaxClientsPerIP : definuje maximálny počet pripojení z jednej IP adresy
- AnonymousOnly : pri voľbe "yes" bude prihlásenie sa na FTP server povolené len anonymným užívateľom, kým registrovaní užívatelia sa prihlásiť nebudú môcť
- NoAnonymous : povolí/zakáže prihlásenie anonymných užívateľov na FTP server
- MySQLConfigFile : určuje cestu k súboru, v ktorom má FTP server hľadať informácie pre prácu so systémom MySQL
- UserBandwidth : maximálna priepustnosť (šírka prenosového pásma) pre všetkých užívateľov udávaná v KB/s
- ProhibitDotFilesWrite : zakáže/povolí vytváranie a odstraňovanie súborov s bodkou na začiatku pre ich vlastníkov; ak bude povolený prepínač TrustedGID, bude umožnený týmto užívateľom prístup aj k týmto súborom
- ProhibitDotFilesRead : zakáže/povolí čítanie súborov s bodkou na začiatku; tento, podobne ako aj vyššie uvedený prepínač, by mali byť povolené, ak chceme užívateľom povoliť vytváranie a editovanie napríklad dôležitého súboru .htaccess
- CreateHomeDir : veľmi dôležitý prepínač zaručujúci vytváranie domovských adresárov pre užívateľov, ktorí sú zaregistrovaní, ale svoj adresár ešte nemajú; tento adresár sa vytvorí po prvom prihlásení na FTP server; šetrí prácu pri vytváraní adresárov iným spôsobom (napríklad pomocou skriptu) , ktorý by mohol byť potenciálnou bezpečnostnou dierou v systéme
- Quota : určuje maximálny počet súborov, ktoré možno nahrať na FTP server a ich a maximálnu veľkosť

Po vykonaní týchto nastavení, bude potrebné vytvoriť užívateľa, ktorý bude môcť pristupovať k FTP serveru, a skupinu, do ktorej bude tento užívateľ patriť. Prístup virtuálnych užívateľov k FTP serveru potom možno zabezpečiť tým spôsobom, že všetci virtuálni užívatelia, ktorí nevlastnia systémový účet, budú k serveru pristupovať prostredníctvom privilegovaného užívateľa, ktorý tento prístup má. Budú na to využívať jeho UID. Ďalšou možnosťou je každému užívateľovi prideliť jeho vlastné UID a priradiť mu ID skupiny, do ktorej patrí vyššie spomínaný užívateľ. V našom prípade, z dôvodu prehľadnosti, využívame druhé spomenuté riešenie, ktoré môže byť lepšie v prípade, ak musí mať každý užívateľ vlastné UID. Napríklad z dôvodu využívania iných webových aplikácií, či služieb, ktoré vyžadujú identifikáciu aj na základe UID.

Dôležitou časťou konfigurácie bude ďalej vytvorenie súboru obsahujúceho informácie potrebné pre autentifikáciu užívateľov FTP servera. Tento súbor bude obsahovať informácie umožňujúce spojenie s databázou MySQL a príkazy, na základe ktorých bude prebiehať výber dát z databázy,

samotné prihlasovanie sa užívateľov k FTP serveru a zisťovanie ďalších informácií. Medzi ne bude patriť napríklad zisťovanie FTP serverom, do ktorého adresára má prihlasovaný užívateľ právo vstúpiť, prípadne tu možno každému užívateľovi individuálne určiť šírku prenosového pásma pre upload a download alebo mu vymedziť diskový priestor pomocou určenia kvót vyjadrujúcich maximálny počet súborov na serveri a maximálnu veľkosť súboru.

Ako bolo už v úvode kapitoly o FTP serveri spomínané, FTP protokol je založený na princípe klient-server. Doteraz sme sa zaoberali časťou servera, avšak pretože všetky služby, ktoré bude náš systém pre podporu webhostingu poskytovať, musia byť prístupné cez internetový prehliadač. Podobne ako pri mailovom serveri, je aj pri FTP serveri potrebné zabezpečiť internetové rozhranie, pomocou ktorého budú užívatelia s FTP serverom komunikovať. Toto rozhranie bude druhou časťou, v modeli klient-server protokolu FTP, a teda klientom. Na internete možno nájsť viacero takýchto webových FTP klientov. Medzi všetkými však svojou kvalitou vyčnieva klient s názvom Net2ftp, ktorý si je možné stiahnuť z jeho domovskej stránky²⁴. Tento webový klient ponúka veľké množstvo funkcií zjednodušujúcich komunikáciu s FTP serverom, ako bežným užívateľom FTP servera, tak aj administrátorovi tohto servera. Po úprave jednoduchých nastavení, je možné využívať aj systém MySQL. V ňom si administrátor webservera jednoducho cez Net2ftp rozhranie môže nechať vytvoriť tabuľky v databáze, do ktorých sa bude zaznamenávať aktivita na FTP serveri a mnoho iných informácií. Tieto funkcie využívame aj v našom webhostingovom systéme, konkrétne v administrátorskej časti webového rozhrania systému.

24 Domovská stránka projektu: <http://www.net2ftp.com/>

7 Práca so softwarom

Aby užívateľ mohol využívať služby podporované softwarom pre podporu webhostingu, je nutné sa do tohto systému zaregistrovať.

Pri registrácii je potrebné zadať meno užívateľa, login, na základe ktorého bude užívateľ pristupovať k jednotlivým službám, heslo a už existujúcu emailovú adresu, na ktorú bude užívateľovi doručený potvrdzujúci email. Okrem toho si užívateľ bude môcť vybrať, či chce využívať služby databázového systému a napokon bude musieť súhlasiť s podmienkami využívania softwaru pre podporu webhostingu, ako ukazuje obrázok 7.1.

The screenshot shows a web registration interface for a web hosting service. At the top, there is a navigation bar with links: Home, Registrácia, Email, FTP, and Databáza. The main heading is 'Registrácia nového užívateľa'. Below this, there are four input fields: 'Zadaj meno :', 'Zadaj login, pre prístup k účtom :', 'Zadaj heslo :', and 'Zadaj email pre potvrdenie :'. The values entered are 'usertest', 'testuser', '*****', and 'test@gmail.com' respectively. Below the email field, there are two radio buttons: 'Mám záujem o databázu' (selected) and 'Nemám záujem o databázu'. A scrollable window displays the 'GNU GENERAL PUBLIC LICENSE Version 3, 29 June 2007'. Below the license text, there is a checkbox with a checkmark and the text 'Súhlasim s licenčnými podmienkami.'. At the bottom, there is a 'Registrovať' button.

Obrázok 7.1: Priebeh registrácie nového užívateľa

Po registrácii zašle systém užívateľovi na emailovú adresu zadanú pri registrácii automaticky email obsahujúci odkaz. Po kliknutí na tento odkaz bude užívateľ presmerovaný na stránku, ktorá automaticky na základe loginu a identifikačného reťazca, ktoré sú súčasťou tohto odkazu, aktivuje

užívateľovi prístup k jednotlivým službám, prehľadne mu zobrazí odkazy a prihlasovacie údaje, prostredníctvom ktorých sa bude k službám prihlasovať. Počas procesu aktivácie sa užívateľovi zároveň vytvorí účet v databázovom systéme, ak o neho prejavil pri registrácii záujem.

K jednotlivým službám možno pristupovať okrem vyššie spomenutého spôsobu aj pomocou odkazov v menu webových stránok softwaru. Pre prihlásenie sa k emailovej službe a službe správy databáz, je potrebné zadať iba login a heslo. Pre využívanie služieb FTP servera, poskytuje titulná stránka rozhrania viac možností znázornených na obrázku 7.2.

net2ftp
a web based FTP client

net2ftp - A web based FTP client

FTP server: domena.sk port: 21
Example: ftp.server.com, 192.123.45.67

Username: testuser

Password:

Language: English

Skin: Blue

FTP mode: ☐ Binary ☒ Automatic

Anonymous ☐ Passive mode ☐

Login

Clear cookies Admin

Help Guide | Forums | License

Obrázok 7.2: Titulná stránka webového rozhrania FTP

Implicitné nastavenia ako meno FTP servera, port a mód sú však pre prácu s FTP serverom vyhovujúce a tak po zadaní loginu a hesla užívateľa a prípadnej zmene vzhľadu či jazyka, sa užívateľ môže prihlásiť k FTP serveru.

net2ftp
a web based FTP client

domena.sk

New dir New file Upload Advanced

Transform selected entries: Copy Move Delete Rename Chmod Download Zip Unzip Size Search

All	Name	Type	Size	Owner	Group	Perms	Mod Time	Actions
<input type="checkbox"/>	Up... .ftpquota	FTPQUOTA File	0	5009	5001	rw-----	May 5 17:51	View Edit Open

Directories: 0
Files: 1 / 0 B
Symlinks: 0
Unrecognized FTP output: 0
Data transferred from this IP address today: 0 B
Data transferred to this FTP server today: 0 B

Help Guide | Forums | License

Obrázok 7.3: Webové rozhranie pre správu dát na FTP serveri

Po prihlásení môže užívateľ využívať služby ponúkané FTP serverom – môže súbory vytvárať, mazať, meniť, stahovať či nahrávať na FTP server. V jeho koreňovom adresári sa od prvého prihlásenia bude nachádzať súbor .ftpquota, s ktorým však užívateľ nebude vedieť manipulovať, pretože na to nebude mať oprávnenie. Tento súbor je však dôležitý pre správne fungovanie FTP účtu daného užívateľa, pretože obsahuje konfiguračné informácie tohto FTP účtu.

8 Záver

V tejto bakalárskej práci sme sa zoznámili s problematikou vytvárania softwaru pre podporu webhostingu prostredníctvom konfigurácie serverov zabezpečujúcich dané služby. Využité serveri boli starostlivo vybrané na základe ich funkčnosti a licencie, pod ktorou sú registrované. Podmienkou totiž bolo využiť voľne šíriteľný software a taktiež samotný software, ktorého implementácia je popísaná v tejto práci, zaregistrovať pod licenciou GNU GPL.

Výsledná aplikácia poskytuje užívateľom jednoduché webové rozhranie umožňujúce registráciu v systéme. Registráciu je potrebné potvrdiť kliknutím na odkaz, ktorý užívateľ obdrží v emaili, ktorý mu bude zaslaný po vykonaní registrácie. Následne bude môcť užívateľ pristupovať ku svojim účtom a využívať všetky služby ponúkané softwarom pre podporu webhostingu. Prístup k službám je umožnený využitím webových rozhraní, ktoré ponúkajú veľké množstvo funkcií pre správu účtov daných služieb.

Pri implementácii tohto projektu sa vyskytli dva problémy. Prvý z týchto problémov sa týka nemožnosti odosielať emaily na vonkajšie domény, čiže na iné domény, než na ktorej beží náš SMTP server. Druhým problémom je nemožnosť zautomatizovať registráciu do systému. Konkrétne sa jedná o nemožnosť automaticky vytvoriť doménu pre užívateľa, ktorý sa do systému zaregistruje. Tieto problémy však nie sú problémami implementačnými a ich riešením by bolo zaregistrovanie do DNS systému. Obidva tieto problémy sú presnejšie popísané v samotnej práci.

Vhodnými rozšíreniami tohto softwaru by mohli byť hlavne rozšírenia týkajúce sa bezpečnosti. Konkrétne by mohlo ísť o bezpečnostné rozšírenie webového servera Apache, ktoré možno zabezpečiť inštaláciou tohto servera s podporou SSL. Pre podporu SSL (Secure Socket Layer – vrstva zabezpečených soketov), je potrebné využiť knižnicu OpenSSL. Ďalším vhodným rozšírením by mohlo byť rozšírenie týkajúce sa SMTP servera. Najužitočnejšími rozšíreniami by bolo rozšírenie o spamovú a antivírusovú kontrolu. Spamovú kontrolu možno zabezpečiť inštaláciou balíčka SpamAssassin a antivírusovú pomocou balíčka Clamav alebo Amavisd.

Literatúra

- [1] *Aký operačný systém je najlepší pre váš web hosting?* [online]. 2009 [cit. 2009-02-18]. Dostupný z WWW: <<http://www.blog-webhosting.sk/aky-operacny-system-je-najlepsi-pre-vas-web-hosting/>>.
- [2] Free Software Foundation, Inc.. *The GNU Operating System* [online]. 2009 , 2009/04/15 17:08:26 [cit. 2009-02-19]. Dostupný z WWW: <<http://www.gnu.org/>>
- [3] POŠMURA, Vlastimil. *Apache : Příručka správce WWW serveru*. 1. vyd. Praha : Computer Press, 2002. 311 s. ISBN 80-7226-696-9
- [4] LEE, James, WARE, Brent. *OpenSource – vývoj webových aplikací : Linux, Apache, MySQL, PHP a Perl*. Praha : Mobil Media, 2003. 448 s. ISBN 80-86593-43-6
- [5] ROSEBROCK, Eric, FILSON, Eric. *Linux, Apache, MySQL a PHP : Instalace a konfigurace prostředí pro pokročilé webové aplikace*. Karel Voráček. Praha : Grada Publishing, 2005. 344 s. ISBN 80-247-1260-1
- [6] Sun Microsystems, Inc.. *MySQL 5.0 Reference Manual :: 2.16.2 Typical configure Options* [online]. 2008-2009 [cit. 2009-04-03]. Dostupný z WWW: <<http://dev.mysql.com/doc/refman/5.0/en/configure-options.html>>
- [7] *MCrypt* [online]. 19.09.2003 [cit. 2009-03-15]. Dostupný z WWW: <<http://mcrypt.sourceforge.net/>>.
- [8] MAVROYANOPOULOS, Nikos, SCHUMANN, Sascha. *Mhash* [online]. 23.09.2005 [cit. 2009-03-19]. Dostupný z WWW: <<http://mhash.sourceforge.net/>>.
- [9] VENEMA, Wietse. *Postfix Installation From Source Code* [online]. 01.09.2008 [cit. 2009-03-07]. Dostupný z WWW: <<http://www.postfix.org/INSTALL.html>>.
- [10] DENT, Kyle. *Postfix : Kompletní průvodce*. Odpovědný redaktor Martin Kysela; přeložil Ludvík Roubíček. 1. vyd. Praha : Grada Publishing, a.s., 2005. 252 s. ISBN 80-247-1029-3

Zoznam príloh

Príloha A: Návod na inštaláciu a konfiguráciu softwaru pre podporu webhostingu

Príloha B: CD-ROM so zdrojovými, konfiguračnými a testovacím súborom

Príloha A

Táto príloha predstavuje návod ako sprevádzkovať software pre podporu webhostingu. Tento postup bol testovaný na operačnom systéme Linux distribúcia Fedora 7 za použitia verzií softwaru uvedených v návode .

APACHE

1. Zistíme, či je nainštalovaný nejaký starý balíček Apache:

rpm -q httpd

2. Ak áno, výstupom tohto príkazu, bude napríklad:

httpd-2.0.48-1.2

3. Tento balík odinštalujeme, predtým však je nutné vyhľadať závislosti tohto balíčka a tie odstrániť ako prvé.

rpm -q --whatrequires httpd

4. Výstupom môže byť nič alebo hocijaký iný balíček, napríklad:

squirrelmail-1.4.10a-1.fc7

5. Pri odinštalovávaní začíname balíčkami, ktoré sú závislé na balíčku, ktorý sme chceli pôvodne odinštalovať. Až potom odinštalujeme pôvodný - v tomto prípade Apache.

rpm -e squirrelmail-1.4.10a-1.fc7

rpm -e httpd-2.0.48-1.2

6. Následne je potrebné zaistiť, aby v systéme nebežal žiadny Apache server.

Zistíme, či nejaké bežia:

ps aux | grep httpd

Ak áno a výpis nebude prázdny, vypneme tieto procesy:

killall httpd

7. Na adrese httpd.apache.org si vyberieme verziu Apache, ktorú chceme nainštalovať alebo priamo z adresy <http://public.picvi.com/apache/httpd/httpd-2.2.11.tar.gz> stiahneme balíček zdrojových

súborov. Tento balíček umiestnime na miesto, odkiaľ ho budeme inštalovať. Napríklad do novovytvoreného adresára **webserver**.

mkdir /usrlocal/src/webserver

8. Balíček zdrojových súborov umiestnený v adresári **/usrlocal/src/webserver** rozbalíme:

tar xzf httpd-2.2.11.tar.gz

9. Následne nastavíme skriptu **apache_install** oprávnenie k vykonaniu príkazom

chmod +x apache_install

a príkazom

./apache_install

z adresára, v ktorom je tento skript umiestnený, spustíme konfiguráciu zdrojových súborov.

POZOR! Prvý riadok uvedeného skriptu predpokladá rozbalenie balíčka zdrojových súborov v adresári **/usrlocal/src/webserver**. V prípade, že ste si balíček rozbalili inam, je nutné tento riadok zmeniť podľa potreby.

Uvedený skript pri prvom spustení môže vygenerovať chybu, ktorú spôsobí príkaz "make distclean". Túto chybu však možno ignorovať, tento príkaz len vymazáva všetky predchádzajúce kompilácie tohto balíčka.

Tento skript nakonfiguruje zostavovacie skripty tak, aby bol Apache nainštalovaný v adresári /www. Toto je možné zmeniť. Takisto možno meniť moduly, ktoré sa prostredníctvom tohto skriptu aktivujú. Viac informácií o moduloch možno získať na stránkach <httpd.apache.org>. Informácie o moduloch ku konkrétnej, vyššie zvolenej verzii webservera, možno získať na stránkach <http://httpd.apache.org/docs/2.2/mod/>.

10. Po vykonaní skriptu, prejdeme do zdrojového adresára Apache webservera a spustíme príkaz "make":

cd /usr/local/src/webserver/httpd-2.2.11

make

11. Po dokončení vykonávania príkazu "make" a skontrolovaní výstupu, v prípade ak tento výstup neobsahuje chyby, možno nainštalovať server príkazom:

make install

12. Po nainštalovaní Apache, je vhodné si vytvoriť odkaz na binárny súbor riadiaci Apache. Vďaka tomu, bude možné jednoduchým príkazom "apachectl" ovládať Apache, odhliadnuc od pozície v súborovom systéme:

```
ln -s /www/bin/apachectl /usr/bin/apachectl
```

```
ln -s /www/bin/apachectl /usr/sbin/apachectl
```

13. Spustenie servera:

```
apachectl start
```

Následne po zadaní adresy "localhost" do prehliadača, sa zobrazí stránka oznamujúca korektné nainštalovanie webservra Apache.

Konfigurácia servera APACHE:

Konfiguračný súbor Apache webservra sa nazýva **httpd.conf** a nachádza sa v adresári "/conf" inštalácie Apache (v našom prípade teda v adresári /www/conf).

Pre nás je najdôležitejšie na koniec tohto súboru pridať nasledujúce 2 riadky:

```
AddType application/x-httpd-php .php .php3
```

```
AddType application/x-httpd-php-source .phps
```

Tieto riadky zabezpečia, že Apache bude schopný spracovávať aj súbory s koncovkami .php, .php3, .phps.

Toto nastavenie spolu so štandardnými nastaveniami servera Apache uvedenými vo vyššie spomínanom súbore, sú nastaveniami postačujúcimi pre vykonávanie bežných úloh.

Ďalej je potrebné si v súbore vyhľadať direktívu "<IfModule dir_module>" a za "DirectoryIndex index.html" doplniť aj ďalšie typy súborov: Riadok by mal následne vyzeráť takto:

```
DirectoryIndex index.php default.php index.html default.html index.htm default.htm
```

Virtuálni hostitelia

Pod pojmom virtuálni hostitelia rozumieme viac webových serverov, ktoré bežia na jednom počítači. Tieto serveri sú súčasťou konfigurácie jedného servera Apache a môžu byť odlišené IP adresou alebo

doménovým menom. Táto vlastnosť servera Apache v podstate umožňuje hostiť viac doménových názvov na jednom serveri, hovorí sa tomu **webhosting**.

V prípade, ak chceme na našom webovom serveri poskytovať webový priestor užívateľom s vlastnými doménami/subdoménami, je ešte v súbore **httpd.conf** nevyhnutné vyhľadať a odkomentovať direktívu:

```
Include conf/extra/httpd-vhosts.conf
```

Následne do súboru **httpd-vhosts.conf** uloženého v adresári **conf/extra/** inštalácie Apache, je potrebné pridať nasledujúce riadky:

```
<Directory /usr/home/ftpusers>
```

```
Order Deny,Allow
```

```
Allow from all
```

```
</Directory>
```

Tieto riadky nastaví korektne práva adresára, ktorý obsahuje dáta, ku ktorým chceme zabezpečiť prístup prostredníctvom webservera (čiže napríklad k stránkam, ktoré chceme na internete zverejňovať). V tomto prípade predpokladáme, že súbory a adresáre virtuálnych užívateľov budú uložené v adresári **/usr/home/ftpusers**. Toto nastavenie je však možné podľa potreby zmeniť.

Ďalej v tomto súbore upravíme direktívu "NameVirtualHost *:80" na nasledovný tvar:

```
NameVirtualHost *
```

Táto direktíva môže namiesto "*" obsahovať aj konkrétnu IP adresu, na ktorej sa majú očakávať prichádzajúce požiadavky. Ak však obsahuje "*", umožní to konfiguráciu virtuálneho hostiteľa pre akúkoľvek IP adresu.

Po týchto všeobecných nastaveniach už možno vykonávať nastavenia pre konkrétneho virtuálneho hostiteľa. Pre tento účel slúži direktíva "<VirtualHost>".

Ako príklad si uvedieme obsah konfiguračného súboru pre doménu s menom "domena.sk":

```
<VirtualHost *>
```

```
ServerName www.domena.sk
```

```
ServerAlias domena.sk
```

```
DocumentRoot /www/htdocs/
```

```
ServerAdmin admin@domena.sk
```

```
</VirtualHost>
```

Znak "*" v direktíve "<VirtualHost *>" znamená, že daný virtuálny hostiteľ, bude reagovať na požiadavku na daný doménový názov, na akejkol'vek IP adrese. Toto možno samozrejme obmedziť zadáním konkrétnej adresy.

Direktíva "ServerName www.domena.sk" ma jednoduchú funkciu, určuje názov servera.

Nasledujúca direktíva "ServerAlias domena.sk" umožňuje nasmerovať požiadavky poslané na "domena.sk" na "www.domena.sk". Čo má praktické využitie hlavne z dôvodu toho, že užívatelia niekedy môžu chcieť k nášmu webovému sídlu pristupovať aj bez zadania "www".

Ďalšou veľmi dôležitou direktívou v konfigurácii virtuálneho hostiteľa je "DocumentRoot /www/htdocs/". Táto direktíva označuje cestu k adresáru, ktorý obsahuje súbory webového sídla.

Ďalšou direktívou môže byť "DirectoryIndex index.php index.html index.htm index.shtml", ktorá určuje, aké typy stránok majú byť načítané ako hlavné stránky.

Direktíva "CustomLog /www/htdocs/logs/access_log combined" určuje cestu k logovaciemu súboru, ktorý obsahuje prístupové informácie o danom virtuálnom hostiteľovi.

Podobnou je aj direktíva "ErrorLog /www/htdocs/logs/error_log" , ktorá obsahuje chybové informácie týkajúce sa danej domény.

V prípade vynechania posledných dvoch direktív, sa dané informácie budú zapisovať do logovacích súborov samotného webservera.

Vyššie uvedený príklad konfigurácie virtuálneho hostiteľa, by mal byť uložený v súbore **httpd-vhosts.conf**. Z praktického hľadiska jednoduchšej editácie, je však výhodnejšie, keď budú informácie o konfigurácii jednotlivých virtuálnych hostiteľov v jednotlivých zvláštnych súboroch. Aby boli na jednom mieste, vytvoríme pre ne špeciálny adresár:

mkdir /www/conf/vhosts

A do súboru **httpd-vhosts.conf** pridáme direktívu "Include conf/vhosts", aby server vedel, kde má informácie o virtuálnych hostiteľoch hľadať.

Ďalej je v súbore **httpd-vhosts.conf** potrebné zakomentovať informácie uvedené za nadpisom "VirtualHost example:", prípadne ich vymazať, aby nám zbytočne nevyhadzovali chyby pri spúšťaní servera.

Pre dokončenie konfigurácie virtuálneho hostiteľa je potrebné editovať súbor **/etc/hosts**.

V našom prípade pre hostiteľa "www.domena.sk", ktorý má byť prístupný z "localhostu", zabezpečíme prístup k tejto doméne pridaním riadka:

127.0.0.1 domena.sk www.domena.sk

Teraz bude náš systém prevádzať názvy "domena.sk" a "www.domena.sk" na IP adresu "localhostu", 127.0.0.1. Po následnom reštartovaní servera Apache, vytvorení stránky, ktorá bude uložená v adresári **/www/htdocs/**, ako je zadané direktívou "DocumentRoot" a zadaní adresy "www.domena.sk" do nášho prehliadača, by sme mali vidieť nami vytvorenú stránku.

POZNÁMKA: Z testovacích dôvodov tejto práce, bude potrebné nastaviť práva adresára **vhosts** nasledovne:

chmod 747 /www/conf/vhosts/

MySQL

1. Zistíme, či je nainštalovaný nejaký starý balíček MySQL prípadne aplikácie s ním súvisiace:

```
rpm -qa | grep -i mysql
```

2. Ak máme nainštalované nejaké aplikácie MySQL, výstupom tohto príkazu, bude napríklad:

```
MySQL-client-4.0.18
```

```
MySQL-server-4.0.18
```

```
MySQL-devel-4.0.18
```

3. Pred odinštalovaním týchto aplikácií, je potrebné vyhľadať ich závislosti na iných aplikáciách.

```
rpm -q --whatrequires MySQL-client
```

4. Výstupom môže byť nič alebo hociaká ďalšia aplikácia, napríklad:

```
MySQL-devel-4.0.18
```

5. Pri odinštalovávaní začíname aplikáciami závislými na pôvodných aplikáciách.

```
rpm -e MySQL-devel-4.0.18
```

6. Následne, pred odinštalovaním pôvodných aplikácií, je potrebné zastaviť procesy MySQL, ktoré by mohli bežať.

Zistíme, či nejaké bežia:

```
ps aux | grep mysqld
```

Ak áno a výpis nebude prázdny, zastavíme tieto procesy:

```
killall mysqld
```

Teraz môžeme pôvodné aplikácie odinštalovať:

```
rpm -e MySQL-client-4.0.18
```

```
rpm -e MySQL-server-4.0.18
```

7. Zo stránky <http://dev.mysql.com/downloads> stiahneme verziu MySQL, ktorú chceme používať.

V našom prípade je to verzia MySQL 5.0, ktorú budeme inštalovať z balíčka mysql-5.0.67.tar.

8. Balíček zdrojových súborov rozbalíme:

```
tar xf mysql-5.0.67.tar
```

9. Pred inštaláciou je potrebné vytvoriť užívateľa, pod ktorým bude "mysqld", čo je démon MySQL, bežať. Vytvoríme teda napríklad užívateľa "mysql" patriaceho do skupiny "mysql".

groupadd mysql

useradd -g mysql mysql

10. Pre kompiláciu MySQL použijeme skript **conf_mysql**. Najskôr mu nastavíme oprávnenie k vykonaniu príkazom:

chmod 775 conf_mysql

11. Potom, v adresári MySQL, kam uložíme aj náš skript, tento skript spustíme:

./conf_mysql

12. Ak všetko prebehlo v poriadku, na konci konfigurácie sa zobrazí poďakovanie "Thank you for choosing MySQL!". Potom môžeme zadať príkaz:

make

13. Ak aj toto prebehlo v poriadku, zadáme príkaz pre inštaláciu, ktorý nám MySQL nainštaluje do adresára špecifikovaného v našom skripte (v našom prípade: /usr/local/mysql):

make install

14. Ak MySQL inštalujeme prvýkrát, bude potrebné vytvoriť tabuľku oprávnení MySQL príkazom:

scripts/mysql_install_db

15. Potom, z bezpečnostných dôvodov, treba zmeniť vlastníka súborov. Zmeníme ho nasledovne:

chown -R root /usr/local/mysql

chown -R mysql /usr/local/mysql/var

chgrp -R mysql /usr/local/mysql

16. Po vykonaní týchto krokov by MySQL server mal byť správne nainštalovaný. Pre overenie si ho spustíme a necháme si vypísať informácie o verzii MySQL:

spustenie:

/usr/local/mysql/bin/mysqld_safe&

výpis informácií:

/usr/local/mysql/bin/mysqladmin version

17. Je dôležité zadať heslo root-a. V našom prípade zadáme heslo "bakalarka". Možno ho zadať dvoma spôsobmi:

```
/usr/local/mysql/bin/mysqladmin -u root password 'bakalarka'
```

alebo

po zadaní nasledovného príkazu, ktorý od nás bude žiadať heslo, sa prihlásime k MySQL a zadáme heslo:

```
/usr/local/mysql/bin/mysql -u root mysql -p
```

```
SET password FOR 'root'@'localhost' = password('bakalarka');
```

```
FLUSH PRIVILEGES;
```

18. Ďalším príkazom, si do MySQL importujeme databázy uložené v priloženom súbore **db.sql**, pričom je potrebné zadať cestu k tomuto súboru:

```
/usr/local/mysql/bin/mysql -u root -p < /cesta/k/súboru/db.sql
```

19. Následne sa prihlásime k MySQL, vyberieme si databázu a necháme si vypísať niektorú z tabuliek, uvidíme výstup, čo znamená, že je všetko korektne nastavené:

```
/usr/local/mysql/bin/mysql -u root -p
```

```
use virtual
```

```
select* from mailusers;
```


PHP

1. Do adresára **/usr/local/src/webserver** vytvoreného pri inštalácii servera Apache zo stránky www.php.net stiahneme súbory pre inštaláciu PHP. V našom prípade sme si stiahli balíček zdrojových kódov **php-5.2.8.tar.gz** a rozbalíme ho:

```
tar xzf php-5.2.8.tar.gz
```

2. Na kompiláciu využijeme skript **php_install**. predtým mu však nastavíme práva, aby sa mohol vykonať a taktiež musíme doinštalovať pomocné rozšírenia a knižnice, aby mohlo byť PHP skonfigurované s rozšírením **mcrypt**:

```
chmod 755 php_install
```

3. Doinštalovať bude potrebné nasledujúce nástroje: **mcrypt**, **libmcrypt**, **mhash**.

Najskôr si ich stiahneme. Mcrypt a libmcrypt možno stiahnuť z toho istého zdroja, ktorým je stránka <http://sourceforge.net/projects/mcrypt> a mhash zase zo stránky <http://mhash.sourceforge.net>.

4. V našom prípade sa jedná o verzie: **mcrypt-2.6.8.tar.gz**, **libmcrypt-2.5.8.tar.gz** a **mhash-0.9.9.9.tar.gz**. Následne tieto rozšírenia nainštalujeme. Postup je nasledovný:

Libmcrypt:

```
tar xzf libmcrypt-2.5.8.tar.gz
```

```
cd libmcrypt-2.5.8
```

```
./configure --disable-posix-threads
```

```
make
```

```
make install
```

```
echo "/usr/local/lib/libmcrypt" >>/etc/ld.so.conf
```

```
ldconfig
```

Mhash:

```
tar xzf mhash-0.9.9.9.tar.gz
```

```
cd mhash-0.9.9.9
```

```
./configure
```

```
make
```

```
make install
```

Mcrypt:

```
tar xzf mcrypt-2.6.8.tar.gz
```

```
cd mcrypt-2.6.8
```

```
./configure
```

```
make
```

```
make install
```

```
cp /usr/local/lib/libmcrypt.* /usr/lib
```

```
ln -s /usr/local/lib/libmhash.so.2 /usr/lib/libmhash.so.2
```

```
ldconfig
```

5. Potom, keď už je všetko potrebné nainštalované, spustíme samotný inštalačný skript PHP:

```
./php_install
```

6. Ak na konci výstupu vidíme správu "Thank you for using PHP", prejdeme do adresára so zdrojovými súborami a spustíme príkaz:

```
cd php-5.2.8
```

```
make
```

7. Ak sa na výstupe neobjavili žiadne chyby, zadáme:

```
make install
```

8. Následne, je potrebné prekopírovať súbor **php.ini** na správne miesto:

```
cp php.ini-dist /usr/local/lib/php.ini
```

9. Aby Apache mohol využívať knižnicu **libphp5.so** a zároveň aby mohol bežať SELinux, zadáme nižšie uvedený príkaz. Ak by sme tento príkaz nevykonali, SELinux by z bezpečnostných dôvodov blokoval prístup k tejto knižnici, čo by obmedzilo funkčnosť nášho servera. Predpokladáme, že Apache je nainštalovaný podľa vyššie uvedeného návodu v adresári **/www**.

```
chcon -t texrel_shlib_t /www/modules/libphp5.so
```

10. Napokon, ak reštartujeme server Apache:

```
apachectl stop
```

```
apachectl start
```

A v adresári **/www/htdocs** vytvoríme súbor **index.php**, ktorý bude obsahovať:

```
<?php phpinfo(); ?>
```

v prehliadači zadáme adresu: "www.domena.sk", mali by sme vidieť stránku zobrazujúcu informácie o PHP.

PHPMYADMIN

1. Pre jednoduchšiu správu databáz administrátorom a hlavne pre prístup užívateľov ku svojim databázam, si nainštalujeme rozhranie, pre prácu s MySQL, nazývané PhpMyAdmin.

Inštalačný balíček si možno stiahnuť zo stránky www.phpmyadmin.net. V našom prípade sme si stiahli balíček phpMyAdmin-3.1.3.1-all-languages.tar.gz. Rozbalíme ho:

```
tar xzf phpMyAdmin-3.1.3.1-all-languages.tar.gz
```

2. Následne presunieme obsah vzniknutého adresára do adresára, ktorý nazveme "phpmyadmin" a bude sa nachádzať v koreňovom adresári nášho webservera.

```
mv phpMyAdmin-3.1.3.1-all-languages /www/htdocs/phpmyadmin
```

3. Ďalej bude potrebné v súbore **config.sample.inc.php** kvôli správne nastaveniu zmeniť nasledujúce riadky:

Kvôli autentizácii je potrebné doplniť do riadka začínajúceho direktívou \$cfg['blowfish_secret'] heslo.

Napríklad:

```
$cfg['blowfish_secret'] = 'bakalarka';
```

Orem toho je nutné určiť názov *hosta*, ten sa bude v našom prípade volať "domena.sk":

```
$cfg['Servers'][$i]['host'] = 'domena.sk';
```

Po týchto krokoch súbor uložíme a premenujeme na **config.inc.php**.

4. Reštartujeme Apache a po zadaní adresy "www.domena.sk/phpmyadmin" sa budeme vedieť prihlásiť k správe databáz ako "root" zadaním hesla, ktoré sme použili pri konfigurácii MySQL.

SMTP server POSTFIX

1. Pred inštaláciou nového SMTP servera, je potrebné odstrániť starý server. Najskôr ho zastavíme:

/etc/init.d/sendmail stop

2. Ak predchádzajúci príkaz skončil chybou, zrejme daný server nemáme nainštalovaný a nasledujúce dva kroky môžeme preskočiť. V opačnom prípade pokračujeme príkazmi, ktoré zistia, aké servery máme nainštalované:

rpm -q sendmail

rpm -q qmail

3. Ak niektorý z vyššie uvedených príkazov zobrazí balíček a verziu, je potrebné ho odinštalovať, napríklad:

rpm -e sendmail

4. Ako ďalší krok, si zo stránky www.postfix.org stiahneme súbory potrebné k inštalácii SMTP servera. V našom prípade ide o balíček zdrojových súborov postfix-2.4.10.tar.gz. Následne ho rozbalíme:

tar xzf postfix-2.4.10.tar.gz

5. Presunieme sa do vzniknutého adresára:

cd postfix-2.4.10

6. Pred samotnou inštaláciou, je potrebné zostaviť **Makefile**. Keďže dopredu vieme, že informácie o užívateľoch nášho systému chceme mať uložené v databáze MySQL, zostavíme Makefile s podporou tohto databázového systému. Z tohoto dôvodu je potrebné definovať makro **HAS_MYSQL** a zadať adresáre knižníc MySQL a hlavičkových súborov. Treba taktiež vykonať prepojenie s matematickou knižnicou, ktorá sa v unixových systémoch nachádza štandardne, a s knižnicami **libmysqlclient.so** a **libz.so**. Náš spôsob zostavenia Makefile s podporou MySQL predpokladá, že knižnica MySQL sa nachádza v adresári **/usr/local/lib/mysql** a hlavičkové súbory v adresári **/usr/local/include/mysql**. Takto by to malo byť v prípade, že sme MySQL inštalovali podľa vyššie uvedeného návodu. Ak je to vo vašom prípade ináč, alebo máte nejaké nejasnosti, v adresári **/README_FILES** distribúcie Postfixu, ktorú ste si v kroku 4. rozbalili, sa nachádza súbor **MYSQL_README**, ktorý obsahuje viac informácií. Následne po zostavení Makefile, spustíme aj samotnú konfiguráciu zdrojových súborov:

```
make -f Makefile.init makefiles 'CCARGS=-DHAS_MYSQL -I/usr/local/include/mysql'  
'AUXLIBS=-L/usr/local/lib/mysql -lmysqlclient -lz -lm'
```

7. Ak všetko prebehlo úspešne zadáme príkaz kompilácie pre preklad Postfixu:

```
make
```

8. Pre bezpečnú prevádzku servera Postfix, je potrebné pre neho vytvoriť účet, ktorý bude vlastníkom fronty a väčšiny jeho procesov. Tento účet by nemal umožňovať prihlásenie, nepotrebuje shell a ani žiadny domovský adresár. Položka v súbore **/etc/passwd** by potom mohla vyzerat' napríklad takto:

```
postfix:*:1001:1001:/no/where:/bin/false
```

9. Taktiež je potrebné vytvoriť vyhradenú skupinu, ktorá však nebude používaná žiadnym užívateľským účtom. Podľa konvencie sa táto skupina zvykne nazývať **postdrop**. Vytvoríme ju pridaním nasledujúceho riadka do súboru **/etc/group**:

```
postdrop:*:1007:
```

10. Teraz už môžeme zadať príkaz pre inštaláciu:

```
make install
```

11. Pri inštalácii sa inštalačný skript bude dotazovať na informácie pre nastavenie Postfixu v našom systéme. Za každým z týchto dotazov, stačí stlačiť "Enter", pretože štandardné nastavenia hodnôt, nám budú vyhovovať.

12. Postfix by už mal byť nainštalovaný. Teraz ho bude potrebné nakonfigurovať.

Konfigurácia servera POSTFIX:

1. Konfiguračné súbory by sa štandardne mali nachádzať v adresári **/etc/postfix**.

Najdôležitejší z konfiguračných súborov bude súbor **main.cf**. Ten zeditujeme pridaním nasledujúcich riadkov:

```
myhostname = mail.domena.sk
```

```
mydomain = domena.sk
```

```
myorigin = $mydomain
```

relayhost = smtp.kn.vutbr.cz

virtual_transport = virtual

virtual_uid_maps = static:1003

virtual_gid_maps = static:1005

virtual_mailbox_base = /home/vmail

virtual_mailbox_maps = mysql:/etc/postfix/virtual_mailboxes.cf

virtual_mailbox_domains = domena.sk

luser_relay = unknown

local_recipient_maps =

Vysvetlenie konfiguračných parametrov:

myhostname - nastavuje plne kvalifikovaný názov hostiteľa systému, v našom prípade sme si zvolili názov "mail.domena.sk"; ak tento parameter nastavený nebude, Postfix využije názov hostiteľa hlásený samotným systémom.

mydomain - nastavuje názov domény systému

myorigin - určuje, aký doménový názov sa má pridať pri odosielaní alebo prijímaní emailov; štandardne sa používa hodnota uvedená v parametri myhostname (niekto@mail.domena.sk), priradením mydomain do parametra myorigin tomuto zamedzíme a adresa bude obsahovať názov domény (niekto@domena.sk)

relayhost - adresa servera slúžiaceho pre rozposielanie emailov na ďalšie (vonkajšie) domény (napríklad na gmail.com atd.)

virtual_transport - určuje typ transportu, ktorý má byť používaný pre doručovanie správ do virtuálnych schránok

virtual_uid_maps - určuje vlastníka, ktorého používa Postfix pri doručovaní do súborov virtuálnych schránok

virtual_gid_maps - určuje skupinu, ktorú používa Postfix pri doručovaní do súborov virtuálnych schránok

virtual_mailbox_base - určuje základný adresár úložiska správ virtuálnych účtov

virtual_mailbox_maps - vyhľadávací súbor, ktorý mapuje emailové adresy na súbory schránok

virtual_mailbox_domains - obsahuje názvy domén s virtuálnymi schránkami

luser_relay - v tomto parametri môžeme nastaviť, kto má prijímať poštu, ktorá je adresovaná neznámym užívateľom (napríklad administrátor)

local_recipient_maps - aby server poštu adresovanú neznámym užívateľom neodmietal, je potrebné tento parameter nechať prázdny

2. Ak chceme využívať doručovanie do virtuálnych schránok, je potrebné si vytvoriť ďalšieho užívateľa, ktorý bude vlastníkom súborov týchto schránok.

Do súboru **/etc/group** pridáme riadok:

vmail:x:1005:

A do súboru **/etc/passwd** pridáme riadok:

vmail:x:1003:1003::/home/vmail:/bin/bash

3. Následne je potrebné vytvoriť súbor **/etc/postfix/virtual_mailboxes.cf**, v ktorom budú definované údaje umožňujúce spojenie s databázou MySQL. Do tohto súboru pridáme nasledujúce riadky:

hosts = 127.0.0.1

user = root

password = bakalarka

dbname = virtual

table = mailusers

select_field = mailbox

where_field = email

4. Súbor uložíme a za predpokladu že sa nachádzame v adresári **/etc/postfix**, zadáme príkaz:

postmap virtual_mailboxes.cf

5. Potom server Postfix spustíme:

postfix start

6. SMTP server by mal byť funkčný, môžeme ho vyskúšať príkazom:

telnet domena.sk smtp

Poštu je teda možné odosielať, avšak je potrebné zabezpečiť, aby si ju užívatelia mohli aj čítať a spravovať. Nato nám posluží **IMAP/POP3 server**.

IMAP/POP3 server DOVECOT

1. Ako náš IMAP/POP3 server sme si zvolili server DOVECOT, no možností je viac, ako napríklad CourierImap, Cyrus apod. Pre našu distribúciu operačného systému **Fedora 7**, existujú RPM balíčky pre inštaláciu serveru Dovecot. Možno ich stiahnuť napríklad zo stránky:

<http://www.city-fan.org/ftp/contrib/yum-repo/fc7/i386/>. Podobné balíčky existujú aj pre ďalšie distribúcie Linuxu.

2. Okrem balíčka **dovecot** je potrebné stiahnuť si aj balíček **dovecot-mysql**, pretože informácie o užívateľoch chceme mať v databáze MySQL. Nainštalujeme ich príkazmi:

```
rpm -Uvh dovecot-1.1.11-2.0.cf.fc7.i386.rpm
```

```
rpm -Uvh dovecot-mysql-1.1.11-2.0.cf.fc7.i386.rpm
```

3. Ak chceme otestovať, či sa nám Dovecot naozaj nainštaloval s podporou MySQL, zadáme príkaz:

```
dovecot --build-options
```

4. Po inštalácii budú konfiguračné súbory uložené v adresári **/etc**. Budeme upravovať súbor **dovecot.conf**. Nájde v ňom riadok začínajúci direktívou "auth default" a doplníme ho takto:

```
auth default {
```

```
mechanisms = plain login
```

```
passdb sql {
```

```
args = /etc/dovecot-mysql.conf
```

```
}
```

```
userdb sql {
```

```
args = /etc/dovecot-mysql.conf
```

```
}
```

Toto nastavenie určuje mechanizmus autentifikácie užívateľov a to, kde budú uložené informácie o užívateľoch a heslách. Tie budú uložené v databáze MySQL. Server Dovecot však musí vedieť kde presne, a preto je potrebné editovať ďalší súbor.

Predtým je však potrebné v tomto súbore nájsť a upraviť, prípadne doplniť, nasledujúce riadky:

```
protocols = pop3 pop3s imap imaps
```

```
disable_plaintext_auth = yes
```

```
mail_location = maildir:/home/vmail/%u
```

```
mail_privileged_group = vmail
```



```
first_valid_uid = 1003  
last_valid_uid = 1003  
first_valid_gid = 1005  
last_valid_gid = 1005  
pop3_uidl_format = %08Xu%08Xv  
auth_verbose = yes  
auth_debug = yes  
auth_debug_passwords = yes
```

5. Vytvoríme súbor **dovecot-mysql.conf**, v tom istom adresári, kde sa nachádza predchádzajúci konfiguračný súbor. Tento súbor bude obsahovať nasledujúce informácie, ktoré mu umožnia pracovať s databázou MySQL a overovať prostredníctvom nej užívateľov:

```
driver = mysql  
connect = host=localhost dbname=virtual user=root password=bakalarka  
password_query = SELECT password FROM mailusers WHERE login = '%u'  
user_query = SELECT mailbox, 1003 AS uid, 1005 AS gid FROM mailusers WHERE login = '%u'
```

6. Následne je možné server spustiť a vyskúšať ho príkazom:

```
telnet domena.sk pop3  
alebo  
telnet domena.sk imap
```

UEBIMIAU

Pre komfortnejší prístup užívateľov k svojim schránkam, je dobré používať grafické rozhranie. My sme sa rozhodli pre rozhranie **Uebimiau**, ktoré si možno stiahnuť zo stránky www.uebimiau.org.

1. Po stiahnutí balíčka, je potrebné ho rozbalit' a umiestniť celý adresár do koreňového adresára webservera Apache, v našom prípade sa bude jednať o adresár **/www/htdocs**.

```
mv /usr/local/src/webmail /www/htdocs/mail
```

2. Adresár **/docs** tohto webového rozhrania obsahuje súbor **INSTALL.txt**, v ktorom sú uvedené kroky potrebné pre inštaláciu.

Po umiestnení do koreňového adresára servera Apache, je potrebné vytvoriť pomocný adresár pre toto rozhranie. Napríklad:

```
mkdir /tmp/uebimiau
```

Vlastníkom musí byť užívateľ, pod ktorým beží webserver, v našom prípade užívateľ **daemon**:

```
chown daemon /tmp/uebimiau/
```

Tento vlastník musí vedieť do tohto adresára zapisovať:

```
chmod 700 /tmp/uebimiau/
```

3. Okrem toho, bude treba v adresári **/inc** upraviť súbor **config.php**, ktorý obsahuje informácie o type protokolu, názve servera, meno adresára pre dočasné súbory a podobné upresnenia a nastavenia. Najdôležitejšie je, aby súbor obsahoval nasledujúce riadky:

```
$temporary_directory = "/tmp/uebimiau/";  
$smtp_server = "domena.sk";  
$quota_limit = 4096;  
$mail_detect_login_type = "%user%";  
$mail_servers[] = Array(  
    "domain" => "domena.sk",  
    "server" => "domena.sk",  
    "login_type" => "%user%",  
    "protocol" => "imap",  
    "port" => "143",  
$default_mail_server = "mail.domena.sk";  
$one_for_all_login_type = "%user%";  
$mailer_type = "sendmail";
```

V týchto nastaveniach sa tiež možno rozhodnúť pre používanie protokolu POP3 namiesto protokolu IMAP. Nastavenia potom budú obdobné, až na názov protokolu, ktorý sa zmení na POP3 a na port ktorý sa zmení na číslo 110.

Okrem toho možno v súboroch adresára **/inc** meniť čiastočne vzhľad tohto rozhrania a nastavenia určujúce čo sa bude zobrazovať a čo nie.

Napríklad v súbore **config.languages.php** nastavením nasledujúceho riadku:

```
$default_language = 'en_UK';
```

zabezpečíme, aby implicitným jazykom bola angličtina.

Nasledujúcim:

\$allow_user_change_theme = no;

zakážeme užívateľom meniť grafické motívy tohto rozhrania.

4. Následne po zadaní adresy "www.domena.sk/mail" do prehliadača, sa spustí automaticky skript, testujúci, či je nastavené všetko správne.

5. Pred prvým prihlásením bude potrebné v súbore **/mail/inc/inc.php** vyhľadať príkaz "Header" a odstrániť z jeho parametrov znaky "\r\n".

6. Po vykonaní týchto krokov, by mal byť prístup k schránkam možný cez prehliadač po zadaní adresy: "www.domena.sk/mail" a následnom prihlásení.

FTP server PURE-FTPD

1. Zo stránok projektu Pure-FTPd www.pureftpd.org/project si stiahneme verziu servera, ktorú chceme nainštalovať. Pri ďalšom postupe budeme predpokladať jej stiahnutie do adresára **/usr/local/src/**. V našom prípade sme si stiahli balíček **pure-ftpd-1.0.21.tar.gz**, ktorý rozbalíme:

```
tar xzf pure-ftpd-1.0.21.tar.gz
```

2. Prejdeme do adresára, ktorý vznikol rozbalením. Predpokladáme, ako je aj vyššie spomenuté, jeho umiestnenie v adresári **/usr/local/src/**:

```
cd pure-ftpd-1.0.21
```

3. Spustíme kompiláciu zdrojových súborov.

```
./configure --prefix=/usr/local/pureftpd --without-inetd --without-capabilities --with-mysql=/usr/local/ --with-paranoidmsg --with-altlog --with-pam --with-puredb --with-cookie --with-throttling --with-ratios --with-quotas --with-ftpwho --with-welcomemsg --with-extauth --with-uploadscript --with-virtualhosts --with-virtualchroot --with-diraliases --with-peruserlimits --with-language=english --with-boring
```

Rozšírenia, s ktorými kompiláciu spúšťame, možno samozrejme zmeniť. Viac o tom je možné sa dočítať v adresári **pure-ftpd-1.0.21** v súbore **README**.

Pri prepínači "**--with-mysql**" je však vždy nutné uviesť cestu, kde je MySQL nainštalované.

4. Napokon zadáme príkazy pre kompiláciu a inštaláciu:

```
make
```

```
make install
```

5. Po úspešnom nainštalovaní bude potrebné vykonať nasledujúce príkazy vytvárajúce pomocné adresáre a kopírujúce súbory zo zdrojového adresára na správne miesto:

```
mkdir /var/log/pure-ftpd
```

```
mkdir /usr/local/pureftpd/etc
```

```
mkdir /var/lock/subsys
```

```
mkdir /var/run/pure-ftpd
```

```
cd configuration-file/
```

```
cp pure-config.pl /usr/local/pureftpd/sbin/
```

```
cp pure-ftpd.conf /usr/local/pureftpd/etc/
```

```
cp /usr/local/src/pure-ftpd-1.0.21/pam/ftpusers /etc/
```

```
cp /usr/local/src/pure-ftpd-1.0.21/pam/pure-ftpd/etc/pam.d/  
chmod +x /usr/local/pureftpd/sbin/pure-config.pl
```

6. Ďalším krokom bude konfigurovanie Pure-FTPd servera. Štandardne by sme ho aktuálne mali mať nainštalovaný v adresári **/usr/local/pureftpd/**. Tu nájdeme adresár **/etc**, ktorý obsahuje konfiguračný súbor. Ako prvý budeme upravovať **súbor pure-ftpd.conf**. Najdôležitejšie pre nás budú riadky s nasledujúcimi konfiguračnými parametrami, ktorých hodnoty by mali byť takéto:

```
ChrootEveryone      yes  
NoAnonymous         yes  
MySQLConfigFile     /usr/local/pureftpd/etc/pureftpd-mysql.conf  
CreateHomeDir       yes  
Quota                100:5
```

Vysvetlenie konfiguračných parametrov:

ChrootEveryone - zabezpečiť to, že pre každého virtuálneho užívateľa bude jeho adresár akoby adresárom koreňovým a teda, nebude mať prístup k adresárom iných užívateľov, ani k adresárom systémovým

NoAnonymous - zamedzí anonymnému prístupu k serveru

MySQLConfigFile - určuje, v ktorom súbore hľadať informácie pre prácu s MySQL

CreateHomeDir - ak užívateľ nemá domovský adresár, pri prvom prihlásení sa mu vytvorí

Quota 100:5 - tento limit povoľuje užívateľom uložiť na server maximálne 100 súborov o maximálnej veľkosti 5MB

7. Ďalším konfiguračným súborom, ktorý však budeme musieť aj vytvoriť, bude súbor **pureftpd-mysql.conf**, ktorý bude obsahovať informácie umožňujúce pripojenie k databáze MySQL a o tom, ktoré informácie sa majú hľadať v akých tabuľkách. Mal by vyzeráť nasledovne:

```
#MYSQLServer  localhost  
#MYSQLPort    3306  
MYSQLSocket   /tmp/mysql.sock  
MYSQLUser     root  
MYSQLPassword bakalarka  
MYSQLDatabase virtual  
MYSQLCrypt    md5  
MYSQLGetPW    SELECT password FROM ftpusers WHERE login="\L"  
MYSQLGetUID    SELECT uid FROM ftpusers WHERE login="\L"
```

MYSQLGetGID *SELECT gid FROM ftpusers WHERE login="\L"*

MYSQLGetDir *SELECT dir FROM ftpusers WHERE login="\L"*

8. Po týchto zmenách, ako aj po každej inej zmene konfigurácie, bude potrebné zadať príkaz:

/usr/local/pureftpd/sbin/pure-config.pl /usr/local/pureftpd/etc/pure-ftpd.conf

9. Potom už možno server spustiť, napríklad nasledujúcim príkazom:

/usr/local/pureftpd/sbin/pure-config.pl /usr/local/pureftpd/etc/pure-ftpd.conf start

10. Vzhľadom na to, že chceme využívať virtuálnych užívateľov, bude vhodné vytvoriť systémového užívateľa a skupinu pre ftp server. Tento užívateľ sa k systému nebude môcť prihlásiť a nepotrebuje ani domovský adresár. Bude patriť do skupiny, do ktorej budú patriť aj všetci ostatní virtuálni užívatelia, pričom systémový účet vlastniť nebudú. Pridáme teda do súboru **/etc/group** riadok:

ftpgroup:x:5001:

A do súboru **/etc/passwd** riadok:

ftpusers:x:5001:5001::/dev/null/./:/sbin/nologin

11. Pre jednoduchšiu prácu s FTP serverom, je podobne ako pri mailserveri, dobre použiť webové rozhranie. V našom prípade sme sa rozhodli použiť **Net2ftp**, ktorý si je možné voľne stiahnuť zo stránky www.net2ftp.com a možno ho ľubovoľne upravovať.

NET2FTP

1. Po stiahnutí niektorej z verzií, v našom prípade net2ftp_v0.97.zip, je potrebné tento balíček rozbaliť. Predpokladáme umiestnenie v **/usr/local/src**.

2. Po rozbalení nám vznikne adresár, ktorý obsahuje ďalšie 2 adresáre. Obsah adresára **files_to_upload** bude potrebné skopírovať do adresára nachádzajúceho sa v koreňovom adresári servera Apache, v našom prípade teda, do adresára **/www/htdocs**. A cieľový adresár sa bude volať **ftp**.

mv /usr/local/src/net2ftp_v0.97/files_to_upload/ /www/htdocs/ftp

2. Pre používanie tohto rozhrania bude potrebné niektoré súbory editovať. Najdôležitejší bude súbor **settings.inc** nachádzajúci sa aktuálne v adresári **/ftp** (prípadne sa môže volať ináč, v závislosti od toho, ako sme ho nazvali v kroku 1. . V ňom je potrebné nastaviť:

Prihlasovacie meno a heslo administrátora:

```
$net2ftp_settings["admin_username"] = "admin";  
$net2ftp_settings["admin_password"] = "bakalarka";
```

Informácie pre pripojenie k databáze MySQL:

```
$net2ftp_settings["dbusername"] = "root";  
$net2ftp_settings["dbpassword"] = "bakalarka";  
$net2ftp_settings["dbname"] = "virtual";  
$net2ftp_settings["dbserver"] = "localhost";
```

Čas ako dlho sa budú archivovať *logy* FTP servera:

```
$net2ftp_settings["log_length_days"] = 90;
```

Okrem toho možno zmeniť mnoho iných vecí, ako napríklad aj vzhľad daného rozhrania. Najmä v súboroch **settings_screens.inc**, **browse.inc** a podobne.

3. Dôležitou časťou nastavení Net2ftp rozhrania je nastavenie práv adresára **/ftp/temp** nasledovne:

```
chmod 777 temp
```

4. Následne po zadaní adresy "www.domena.sk/ftp" budeme môcť k nášmu FTP serveru pristupovať pomocou internetového prehliadača.

5. Keďže nahrávanie súborov na FTP server by bolo obmedzené kvôli obmedzeniu o maximálnej veľkosti súboru spracováanej pomocou PHP, ktoré je implicitne nastavené na hodnotu 2MB, bude dobré toto obmedzenie zvýšiť aspoň na takú hodnotu, aké najväčšie súbory sme povolili užívateľom na FTP server nahrávať, a to 5MB. Toto nastavenie sa vykonáva v súbore **php.ini**, ktorý by sa mal nachádzať pri nami vykonanej inštalácii PHP v adresári **/usr/local/lib**. V tomto súbore nájdeme a zmeníme direktívu:

```
upload_max_filesize = 5M
```

Pri dodržaní všetkých vyššie uvedených krokov inštalácií a konfigurácií, dostaneme plne funkčný software pre podporu webhostingu prístupný cez webový prehliadač na adrese www.domena.sk, poskytujúci emailovú službu, služby FTP servera a služby databázového systému. Na adrese www.domena.sk/admin sa možno prihlásiť k administrátorskému rozhraniu systému.