

System pro správu autorských práv u souborů MP3

Juraj Kočar

Bakalářská práce
2005



Univerzita Tomáše Bati ve Zlíně
Fakulta technologická
Institut řízení procesů a aplikované informatiky

Súhlasím s tým, že s výsledkami mojej práce môže byť naložené podľa uváženia vedúceho bakalárskej práce, riaditeľom ústavu a inštitútu. V prípade publikácie budem uvedený ako spoluautor.

Prehlasujem, že som na celej bakalárskej práci pracoval samostatne a použitú literatúru som citoval.

V Zlíně, 01. 09. 2005

.....

podpis

RESUMÉ

Hlavným cieľom tejto práce bolo vytvorenie systému pre správu autorských práv u súborov MP3. Na úspešné dokončenie tejto práce som musel najskôr bližšie pochopiť problematiku formátu mp3 a ID3 Tagov. Pri realizácii tohoto systému sa stal jazyk PHP jeho hlavnou súčasťou. To hlavne vďaka funkcii `getID3()`, ktorá priamo umožňuje prácu jak s ID3 Tagy, tak s MP3 súbormi. Systém obsahuje kompletnú správu užívateľských účtov, štatistiky o zťahovaní a vlastníctve MP3 súborov užívateľmi, umožňuje vyhľadávanie podľa rôznych filtrov, alebo znovaautorizáciu MP3 súboru v prípade, že užívateľ prišiel o svoju nahrávku.

Konštrukcia systému je realizovaná v jazyku PHP s návaznosťou na databázu MySQL.

SUMMARY

The main goal of this work was to create a system for administration of copyrighted mp3 audio files. To get this work to the successful finish first I had to know something more about the theory of mp3 files and ID3 Tags. When constructing this system the PHP language became its main part. That is because of the `getID3()` function that allows to work with ID3 Tags and MP3 files. The system is capable of complete administration of user accounts, viewing users statistics about download and ownership of mp3 files, searching with using different kinds of filters or re-authorization of mp3 files in case that the user has lost his file.

The construction of this system is based on PHP language with connection to MySQL database.

OBSAH

ÚVOD	7
1 TEORETICKÁ ČASŤ	9
1.1 PHP	9
1.1.1 Regulárne výrazy	10
1.2 MySQL	10
1.3 HTML a CSS	11
1.4 MP3	12
1.5 Metadata (Tags)	14
1.5.1 ID3v1	14
1.5.2 ID3v1.1	15
1.5.3 ID3v2	17
1.6 Watermarking	18
1.6.1 Úvod	18
1.6.2 Teória	18
2 INŠTALAČNÁ ČASŤ	19
2.1 Easy PHP	19
2.2 PsPad	19
2.3 MySQL Control Center	20
3 VÝVOJOVÁ ČASŤ	21
3.1 Štruktúra getID3	21
3.1.1 Analýza a výpis informácií MP3	23
3.1.2 Analýza adresára s MP3 súbormy	24
3.1.3 Zápis ID3 tagu do súboru	25
3.1.4 Napojenie getID3 na MySQL databázu	26
3.2 Registrácia užívateľov	27
3.3 Autorizácia nahrávky a digitálny podpis	28
3.4 Identifikácia MP3 súboru v databázi	30
3.5 Celková schéma autorizácie	30
3.6 Prístupové práva	30
3.7 Skripty systému	31
3.7.1 skript login.php	31
3.7.2 skript authorize.php	32

3.7.3 skript browse.php.....	32
3.7.4 skript re_authorize.php.....	32
ZÁVER.....	34
ZOZNAM POUŽITÉJ LITERATURY.....	35
ZOZNAM OBRÁZKOV	36
ZOZNAM PRÍLOH	37

ÚVOD

V súčasnej dobe dochádza k veľkému rozvoju internetu. S rýchlim rozširovaním širokopásmového internetu na celom svete, narastá aj problém autorských práv dát na ňom uverejnených. Nelegálne kopírovanie a distribúcia digitálnych médií nebolo nikdy jednoduchšie. Príčinou toho sa stalo, že len v hudobnom priemysle odhadli minuloročné straty kôli vzrastajúcim počtom užívateľov používajúcim P2P software, alebo systém bittorrent na zľahovanie a distribúciu nelegálnych kópií mp3 súborov na hodnotu 5 miliárd dolárov. Právne pokusy o vyriešenie tohoto problému ukázali doteraz omedzený úspech vzhľadom na komplexnosť problému ilegálneho šírenia súborov s copyrightom.

Veľké medzinárodné firmy sa snažia zabrániť tomuto trendu, a preto investujú do rozvoja technológií a šifer, ktoré by umožňovali ochranu autorských prav a obsahu či už na DVD alebo CD nosičoch. Preto začali hudobné spoločnosti distribuovať svoje nahrávky na CD-čkách ktoré mali integrovanú ochranu. Každá spoločnosť používala svoj vlastný spôsob ochrany datových médií, ale všetky mali niečo spoločné, určitý druh obmedzenia, buď sa nedala spraviť záloha CD a ak áno, tak nie v pomere 1:1 čiže len s určitou stratou kvality, CD nedalo prehrať v počítači, alebo iný druh obmedzenia, ktorý sa ale užívateľom nepáčil. Tento postup je ale vo väčšine krajín nelegálny, pretože každý kto si kúpi originálne CD, alebo DVD má právo si spraviť zálohu pre svoje osobné potreby.

Vždy je to len otázka času kedy bude prelomená ďalšia šifra, ktorá by mala chrániť médium s obsahom copyrightových dát. Keďže programy ktoré dokážu prehrať médium s chráneným obsahom musia obsahovať šifru, ktorá je použitá na ochranu obsahu média, sústreďujú sa hackery na tieto programy, z ktorých môžu šifru dostať. Ak je táto šifra raz prelomená, na všetkých nosičoch ktoré používajú túto šifru sa stáva ich obsah nechránený. Budeme si musieť ešte asi počkať, kým spoločnosti zistia, že ochrana CD a DVD nosičov súčasnými šiframi nemá budúcnosť.

Systém ktorý prezentujem požíva odlišný spôsob ochrany autorských práv a distribúcie (client-server) audio súborov. Má veľa výhod oproti súčasným spôsobom ochrany. Napríklad ak by si zákazník kúpil nejakú skladbu cez tento systém, do súboru je uložený jeho digitálny podpis a spolu s digitálnym podpisom sa uloží aj informácia o tom, že mp3

súbor bol aktualizovaný práve týmto systémom. Popis a princíp tohoto systému vysvetlím na nasledujúcich stranách.

1 TEORETICKÁ ČASŤ

V tejto časti sú popísané všetky technológie, ktoré boli využité pri tvorbe systému pre správu autorských práv u súborov MP3

1.1 PHP

Princip PHP je podobný jako u SSJS (Server Side Java Script) a ASP (Active Server Pages). Na rozdiel od spomenutých systémov je celý produkt pod licenciou GNU GPL. Ďalšiou výhodou je nezávislosť na platforme. Dnes sú k dispozícii verzie PHP pre UNIX a MS Windows. PHP nieje zviazané so žiadnym konkrétnym HTTP serverom, môže bežať na ľubovoľnom servery (Apache, IIS, PWS). PHP bolo vytvorené presne pre písanie skriptov začleňovaných do HTML stránok. Veľká výhoda týchto jazykov je že ak sa chce klient pozrieť na svoje stránky odošle dotaz na server a stránky s koncovkou (PHP, PHP3, ASP, CSS...) si obsah týchto stránok nahrá a vyhodnotí, potom prevedie všetky príkazy do formátu HTML a až potom odošle do prehliadača klienta. Takto síce klient vidí HTML stránku, ale nikdy neuvidí zdrojový kód tejto stránky, pretože sa mu zobrazí až výsledok vyhodnocovania.

Pôvodná zkratka a v podstate aj význam, systému PHP bola Personal Home Page. Na začiatku zrodu tohoto systému stál Rasmus Lerdort, ktorý si vo svojom voľnom čase vylepšoval svoj systém pre evidovanie prístupu k jeho stránkam. Tento systém sa zapáčil aj ostatným užívateľom serverov a začali ich používať. Systém sa stal obľúbený a stále viac používaným, užívatelia prichádzali so stále viac požiadavkami na vylepšenie. Autor preto systém rozšíril, doplnil o dokumentáciu a uvoľnil ho pod názvom Personál Home Page Tools.

Okrem spomenutého systému pre evidovanie prístupu ku stránkam vytvoril Lerforf aj nástroj, ktorý umožňuje začleňovanie SQL dotazov do stránok. Projekt sa rozšíril a na jeho vývoji pracuje Rasmus Lerdofr s niekoľkými ďalšími vývojármi. Obsah zkratky PHP už teraz stráca na svojom pôvodnom význame a doporučené označenie celého systému je *hypertextový preprocesor PHP* (PHP:Hypertext Preprocessor).

Poslednou verziou tohoto obľúbeného jazyka je PHP 5.0, ktorá býva označovaná pod pojmom ZEND. Bola vo veľa veciach vylepšená, rozšírená o niektoré príkazy a na strane

serveru vylepšená štruktúra spracovania skriptu. Často sa tiež v súvislosti s PHP hovorí o tzv. hypertextovom preprocesoru.

Systém PHP sa tiež čoskoro dočká jeho grafickej nadstavby, ktorá umožní vytvárať priamo Win32 a Unix grafické aplikácie.

Pri realizácii systému pre správu autorských práv u MP3 súborov se stal jazyk PHP jeho hlavnou súčasťou. To hlavne vďaka funkcii `getID3()`, ktorá priamo umožňuje prácu jak s ID3 Tagy, tak so MP3 súbormi.

1.1.1 Regulárne výrazy

Regulárne výrazy slúžia k porovnávaniu a spracovaniu textu. Existuje niekoľko špecifikácií syntaxe regulárnych výrazov. PHP podporuje syntaxu POSIX-Extended.

Funkcia regulárnych výrazov spočíva v prehľadávaní určitej časti textu. Prostredníctvom regulárnych výrazov sa definujú vzory, ktoré sa PHP potom snaží rozpoznať v určitom texte. Najjednoduchšími vzormi sú znaky: čísla, písmená, symboly. Ak vyhovuje daný text vyhladávanému vzoru, je na výstupe funkcie hodnota *true*, v opačnom prípade hodnotu *false*.

Funkcia regulárnych výrazov nezpočíva len v prehľadávaní textu, ale aj v jeho prípadnom nahradení textom iným při splnení definovaném podmienky.

1.2 MySQL

Databázový systém MySQL vyvinula švédská firma TcX. Hlavnými užívateľmi toho systému sú univerzity, tzv. ISP (*internet service provider*) a rôzne neziskové organizácie. Dôvodom jeho obľuby je okrem iného hlavne to, že je za určitých podmienok distribuovaný zadarmo, práve pro neziskovú činnost.

Ďalší dôvod stúpajúcej popularity systému MySQL je tiež nástup architektúry tzv. open-source systémov a voľne šíriteľných implementácií.

Databázový systém MySQL je často zamieňaný s SQL, čo je štrukturovaný dotazový jazyk (*Structured Query Language*) vyvinutý spoločnosťou IBM. Ide teda o databázový systém využívajúci tento jazyk pre prácu s dátami, prípadne k ich zobrazeniu. MySQL je program pre správu databáze, SQL je programovací jazyk, ktorý je v MySQL využívaný k vykonávaniu rôznych úloh v rámci používanej databáze. Medzi programami určenými pre

správu databází patria aj známe produkty ako Microsoft SQL Server alebo Sybase Adaptive Server. Systém MySQL je plnohodnotným systémom správy relačných databází. MySQL je viac vláknová (multithreaded) aplikácia, čo v tomto zmysle *viacvláknová* znamená, že s vytvorením každého nového spojenia k serveru sa vytvorí nový proces, alebo podproces, ktorý obsluži klientské požiadavky.

Systém MySQL plne odpovedá štandardu kódovania ANSI SQL92, čo je súbor štandardov pre SQL jazyk, dohodnutý v roku 1992 v Americkom inštitúte pre národné štandardy (ANSI, *American National Standards Institute*).

V súčasnej dobe je MySQL adaptovaný pre všetky dostupné platformy a využíva časť rozhrania pre aplikačné programy, napr. pre Perl, C/C++, Java a ODBC.

1.3 HTML a CSS

Jazyk HTML (*HyperText Markup language*) je stavebným kamením pre písanie webových stránok. Prvý krát bol špecifikovaný v roku 1991. Pre vytváranie HTML stránok existuje celá rada nástrojov od jednoduchých textových editorov k prepracovaným WYSIWYG (*What you see is what you get*) aplikáciám. HTML stránky sa prenášajú po internete pomocou protokolu HTTP (hyper text transfer protokol), ktorý bol k tomu účelu vytvorený.

Každý typografický systém využíva sadzbu textu tzv. štruktúrne značky, ktorými definuje napr. vzhľad prostredia (text), ktoré tieto značky uzatvárajú. HTML používa pre formátovanie textu tzv. (tagy) tagy, napr. tag ktorý definuje nadpis prvej úrovne <h1> nadpis </h1>. Text medzi týmto tagom má potom iné atributy, napr. iný stupeň, rez alebo farbu. Tagy je možné do seba vnorovať takže napr. text medzi tagmi <u> slovo </u> bude tučné a podčiarknuté → **slovo**. Tagy by sa nemali krížiť.

Vytváranie HTML dokumentov je veľmi jednoduché a to bol aj pôvodný zámer tvorcov HTML. Na druhú stranu sa pri tvorbe väčšej webovej prezentácie HTML javí ako dosť neefektívne. Napr. pri zmene farebného schématu webovej prezentácie, ktorá má napr. 20 HTML dokumentov musel by som prejsť všetkých 20 súborov a v každom upraviť staré farebné schéma na nové.

Preto boli vyvinuté tzv. CSS (*Cascade style sheet*) – kaskádové štýly. CSS je jednoduchý štylizáčny mechanizmus, ktorý umožňuje tvorcom, ale aj čitateľom pripojiť štýly (napr. fonty, farby a rozmiestnení) k HTML dokumentom. Tieto štýly môžu byť zviazané s HTML

dokumentom trojakým spôsobom. Buď pomocou externého súboru, uvedeného v definičnej časti HTML súboru (sekcia `<head></head>`) alebo priamo v HTML tagu. CSS používaný obecný atribut `style`, ktorému sa priradzuje nejaká definícia. Je to zložitešie ale viac obecné.

1.4 MP3

Vývoj algoritmu MP3 začal v roku 1987 vzájomnou spoluprácou Fraunhoferovho inštitútu a Erlangskej Univerzity. Bol štandardizovaný organizáciou ISO (International Organization for Standardization) ako MPEG (Moving Pictures Expert Group) Audio Layer III (ISO/IEC 11172-3 and ISO/IEC 13818-3)

Formát MP3 využíva nedokonalosti ľudského sluchu a pri kompresii sa jednoducho vypúšťajú zvuky, ktoré sú prehlúsené hlasnejšími, alebo nie sú počuť vôbec. Čoskoro sa stal de-facto štandard pre stratové kódovanie audia vďaka vysokému kompresnému pomeru (1/12 originalnej veľkosti, stále zachovávajúcu značnú kvalitu, formálne a informálne testy ukázali že MP3 súbor pri dátovom toku 160-224kbps má nerozoznateľnú kvalitu od originálu vo väčšine prípadov), vysokej dostupnosti dekodérov a nízkej CPU náročnosti na prehrávanie (486 DX2-66 je dostačujúci na real-timeové dekódovanie). Preto sa MP3 rýchlo rozšíril a dnes okrem počítačov a MP3 prehrávačov môžete poslúchať MP3 aj z mobilného telefónu alebo USB kľúča.

Formát MP3 podporuje aj viackanalové (multichannel) súbory (aj keď doteraz nie je žiadna implementácia verejne dostupná), vzorkovaciu frekvenciu od 16kHz do 24kHz (MPEG-2 Layer III) a od 32kHz do 48kHz (MPEG-1 Layer III).

Výhody

- Široká podpora skoro vo všetkých audio prehrávačoch a zariadeniach
- Je to ISO štandard, časť špecifikácie MPEG
- Rýchle dekódovanie, menej komplexné než AAC alebo Vorbis
- Ktokoľvek si môže spraviť svoje vlastné implementácie (dostupnosť zdrojov aj špecifikácii od organizácie ISO)
- Voľná licenčná politika

- Vysoká dostupnosť obsahu v MP3

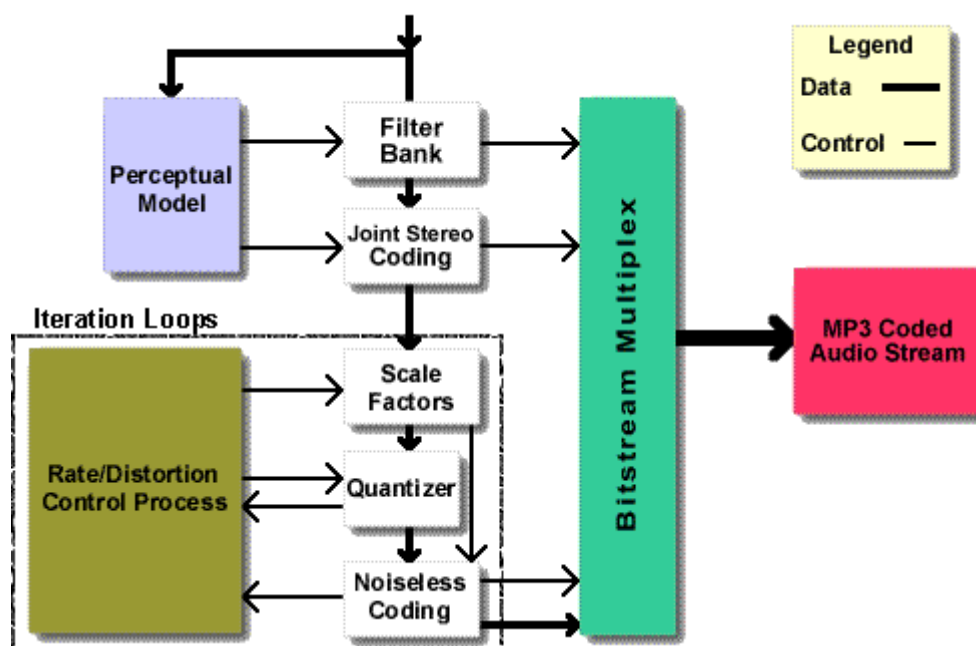
Nevýhody

- Nižšia výkonnosť/efektívnosť ako moderné kodeky
- Niekedy maximálny dátový tok (320kbps) nieje dostačujúci
- Žiadne viackanálové implementácie
- Nevhodné pre high definition audio, napr. DVD-A (vzorkovacia frekvencia vyššia než 48kHz)

Techniky používané pri kompresii

- Huffmanovo kódovanie
- Nelineárna quantizácia
- Intensity stereo
- Modifikovaná diskretná cosínusová transformácia

Funkčná schéma bloku audio kodeku MPEG1 layer 3 používaného v MP3



Obr. 1 Funkčná schéma bloku audio kodeku MPEG1 layer 3

Existuje aj neštandardizovaný formát MP3 nazývaný MP3Pro, ktorý používa SBR (Spectral Band Replication) kódovanie, zabezpečujúce lepšiu kvalitu pri súboroch s nízkym dátovým tokom (nižším než 96kbps). Je spätne kompatibilný s prehrávačmi klasických MP3 súborov, ale SBR časť tam nie je prehraná, takže kvalita na starších prehrávačoch, ktoré nepodporujú MP3Pro nebude až tak dobrá. Ďalšou nevýhodou tohoto formátu je fakt, že je to uzavretý formát bez dokumentácie a zdrojového kódu.

1.5 Metadata (Tags)

1.5.1 ID3v1

Audio formát MPEG Layer I, II a Layer III (MP3) nemá žiadny natívny spôsob ako ukladať informácie o obsahu, okrem pár jednoduchých ano/nie parametrov ako “private”, “copyrighted” a “original home” (čo znamená, že toto je originálny súbor a nie kópia). Riešenie na tento problém bolo predstavené s programom “Studio3” ktorý spravil Eric Kemp v roku 1996. Pridaním malého bloku dát na koniec súboru dosiahol že MP3 súbor dokázal niest’ aj informáciu o audio a nie len audio samotné.

Umiestnenie tagu, ako tie dáta nazval, bolo vybrané kôli malej šanci rušenia dekóderov. Aby bolo jednoduché nájsť presne určenú veľkosť 128 bytov má ID3v1 Tag nasledujúce usporiadanie



Obr. 2 Príklad ID3v1 Tagu

- Song title: 30 znakov
- Artist: 30 znakov
- Album: 30 znakov
- Year: 4 znaky
- Comment: 30 znakov
- Genre (žáner): 1 byte

Ak spočítame veľkosti všetkých týchto polí vidíme že $30+30+30+4+30+1$ sa rovná 125 bytov a nie 128. Chýbajúce 3 byty môžeme nájsť na úplnom začiatku Tagu, pred Song title (názvom pesničky). Tieto 3 byty su vždy "TAG" a to je identifikácia že je to ID3 Tag.

Keďže málokto má meno dlhšie ako 30 znakov, je dané, že zvyšné byty v poli sú vyplnené binárnou hodnotou 0. Aj keď do pola Genre (žáner) môžeš vložiť len 1 byt, čo sa môže zdať nedostačujúce, hodnota vloženého bytu korešponduje k preddefinovanému zoznamu. Zoznam, ktorý Eric Kemp vytvoril má 80 vstupov v rozmedzí 0 až 79.

1.5.2 ID3v1.1

ID3v1 je určite ľahko implementovateľná pre programátorov, ale je to určite frustrujúce pre tých s vlastnými, kreatívnymi nápadmi. Keďže ID3v1 má fixnú konkrétnu veľkosť a žiadne miesto rezervované pre budúce možné rozšírenie, nezostáva veľa miesta pre vylepšovanie ak sa chce zachovať kompatibilita s existujúcim softwarom.



Obr. 3 Příklad ID3v1.1 Tagu

- Song title: 30 znakov
- Artist: 30 znakov
- Album: 30 znakov
- Year: 4 znaky
- Comment: 28 znakov
- Album track: 2 znaky
- Genre (žáner): 1 byte

Michael Mutschler prišiel s rozumným riešením ako vylepšiť ID3v1. Keďže do všetkých nevyplnených polí musia byť dosadené 0 byty, všetky programy ktoré čítajú ID3v1 Tagy prestanú načítavať pole ak narazia na 0 byte. Ak predposledný byt poľa je 0 a posledný nie, máme extra byt na vyplnenie informáciou. Pretože pole Comment je veľmi krátke na to, aby sme tam napísali niečo užitočné, ID3v1.1 štandard deklaruje aby toto pole malo 28 znakov, nasledujúci znak má byť vždy 0 a posledný byte pred Genre má udávať číslo skladby na ktorej je uložená pesnička na CD.

1.5.3 ID3v2

ID3v2 je nový tagovací systém, ktorý umožňuje vkladať obohacujúce a relevantné informácie o audio súboroch. Zjednodušene povedané je ID3v2 skupina dát dopĺňujúcich binárne audio dáta. Každý ID3v2 Tag obsahuje jeden alebo viacero menších kúskov informácie nazývané rámce. Tieto rámce môžu obsahovať akýkoľvek druh informácie ako napr. názov skladby, album, meno umelca, web stránku, text piesne, prednastavenia equalizeru, obrázky atď.



Obr. 4 Príklad ID3v2 Tagu

Jednou z hlavných úloh nového designu bola flexibilita a rozširovateľnosť. Je veľmi jednoduché pridať nové vlastnosti do ID3v Tagu, pretože presne ako v HTML, všetky prekladače budú ignorovať akúkoľvek informáciu, ktorú nedokážu rozpoznať. Pretože každý rámec môže mať až 16MB a celý Tag môže mať až 256MB už nenastane situácia ako vo verzii ID3v1.1, že by sa nedal napísať nejaký užitočný text do poľa Comment pretože neexistuje obmedzenie na 28 znakov. ID3v2 tiež podporuje Unicode takže každý môže do neho písať svojím rodným jazykom.

1.6 Watermarking

1.6.1 Úvod

Ako som už spomenul na začiatku, technológie ktoré sa používajú dnes nie sú dost' účinné na zakódovanie, alebo zašifrovanie súborov s obsahom dát, ktoré sú autorsky chránené. Hlavným dôvodom je fakt, že zašifrované súbory sú prehrávané v originálnej forme, čím môžu byť znovu nahrané, a potom už voľne distribuované. Nádej na distribúciu súborov s copyrightom leží na technologickom zlepšení, ktoré by zaistilo druh kódovania na systéme client-server. Jeden zo sľubných spôsobov do budúcnosti je označenie média s tajnou, robustnou a vystopovateľnou digitálnou vodotlačou tzv. Watermarking. Ale na úspešné zavedenie tejto technológie bude potreba globálnej adopcie veľa štandardov a medzinárodné spoločnosti musia byť odhodlané do viesť tento systém do úspešného konca.

1.6.2 Teória

Watermarking je vlastne proces, ktorý funguje na princípe "vliatia" informácie nazývanej watermark do multimediálnych signálov na zabezpečenie aplikácií s pridanou hodnotou ako napr. monitorovanie prenosu, označovanie signálu a kontrole kopírovania. Proces digitálneho vodoznaku spočíva v tom, že sa nepatrne modifikuje media signal použitím nejakého tajného kľúča. Tomuto signálu sa hovorí krycí signál, pretože vlastne zapúzdruje obidva signály - pôvodný media signal + vodoznak. Media signál môžu byť dáta v podobe obrázku, audio súboru, alebo elektrického signálu. Modifikácie sa robia tak spoľahlivé (robustné), že extrakcia vodotlače za použitia tejného kľúča je možná aj pri značnom poškodení aplikovaného vodotlačného signálu. Tieto poškodenia či už úmyselné alebo nie sú známe pod názvom attack (útok). Watermarking sa používa na ochranu proti nelegálnemu šíreniu elektronického copyrightového obsahu.

2 INŠTALAČNÁ ČASŤ

Táto časť popisuje inštaláciu vývojového prostredia, programovacieho jazyka spolu s inštaláciou všetkých súčasti systému pre správu autorských práv u súborov MP3.

2.1 Easy PHP

Interpretáciu systému som robil v PHP s podporou MySQL database. Takže museli byť na počítač na ktorom som pracoval nainštalované príslušne služby.

Pretože všetky potrebné služby (interpret PHP, webový server Apache, databázový systém MySQL a administračný systém MySQL – PhpMyAdmin) sú náročné na vedomosti, zvolil som open-source produkt Easy PHP, ktorý obsahuje v sebe všetky tieto služby a ich inštalácia prebieha v jednom kroku.

Po inštalácii je už len treba nastaviť v konfiguračnom súbore httpd.conf Apache serveru parametr DocumentRoot, ktorá určuje v ktorom adresári na fyzickom disku serveru sa nachádzajú dokumenty pre spracovanie serverom Apache, tak interpretom PHP. Easy PHP pracuje len pod rozhraním Win32. Jednotlivé komponenty tohoto produktu pracujú ale aj pod systémom Unix (Linux), sám výrobca doporučuje používať Apache server pod platformou Win32 len na testovanie.

Tento produkt je k dispozícii na internetovej adrese www.easyphp.org

2.2 PsPad

Na písanie zdrojového kódu som požil český freewareový editor PsPad, ktorý dokáže zobrazíť syntaxu asi väčšiny zdrojových kódov, dokáže automaticky dopĺňovať tagy, zátvorky atď. Jeho nevýhodou ale je, že pri testovaní skriptov sa nedá program krokovať, alebo použiť debug. Našiel som aj iné programy, ale neboli voľne k stiahnutiu v plnej verzii tak som zostal u PsPad.

Tento program je možné stiahnuť na adrese www.pspad.com

2.3 MySQL Control Center

MySQL Control Center je nástroj pre zpravu databází. Nemá sice toľko možností a funkcií ako veľmi rozšírený phpMyAdmin, ale pre moju prácu bol dostačujúci. Jeho veľká výhoda je fakt, že je samostatne spustiteľný program na rozdiel od phpMyAdmin, ktorý pracuje v okne WWW prehliadača. Na tomto programe sa stále pracuje, a preto verzia, ktorú som požíval ja bola 0.9.4-beta.

3 VÝVOJOVÁ ČASŤ

V tejto časti popisujem postup mojej práce pri vytváraní systému, ktorý by úplne pokryl požiadavky pre správcovanie a distribúciu MP3 súborov v sieti internet.

3.1 Štruktúra getID3

Štruktúra getID3 sa stala stavebným kameňom celého systému. Systém pre prácu s ID3 tagy v prostredí PHP sa skladá z niekoľkých desiatok php skriptov, ktoré umožňujú kompletné pokrytie všetkých dôležitých a mysliteľných požiadaviek. V prílohe č.1 popisujem význam všetkých skriptov štruktúry getID3.

getID3 vyžaduje k svojmu chodu PHP verzie 4.1.0 a vyššiu.

Najdôležitejším skriptom je `getID3.php`, je to vlastne knižnica, ktorá vytvára funkciu `getID3()`. Táto funkcia potom slúži pre získanie kompletných informácií o MP3 súbore (uplný výpis je tiež popísaný v prílohe č. 2). Je tu aj možnosť konfigurácie chovania tejto funkcie, ktorá je nasledujúca:

```
// public: Settings
var $encoding = 'ISO-8859-1';
nastavuje spôsob všeobecného kodovania jazyka

var $encoding_id3v1 = 'ISO-8859-1';
nastavuje spôsob kódovania id3 tagu verzie 1

// public: Optional tag checks - disable for speed.
var $option_tag_id3v1 = true;
prepínač pre povolenie k čítaniu a práci s ID3 Tagy verzie 1

var $option_tag_id3v2 = true;
prepínač pre povolenie k čítaniu a práci s ID3 Tagy verzie 2

var $option_tag_lyrics3 = true;
prepínač pre povolenie k čítaniu a práci s Lyrics3
```

```
var $option_tag_apetag = true
```

prepínač pre povolenie k čítaniu a práci s APE Tagy

```
// public: Optional tag/comment calculations
```

```
var $option_extra_info = true;
```

Zistí dodatočné info, ako je bitrate atď.

Čo všetko funkcia getID3() dokáže.

Čítanie a rozbor:

Tagy:

APE (v1 a v2), ID3v1 (& ID3v1.1), ID3v2 (v2.4, v2.3, v2.2), Lyrics3 (v1 & v2)

Formáty audia (ztrátové):

MP3/MP2/MP1, MPC / Musepack, Ogg (Vorbis, OggFLAC, Speex), RealAudio, Speex, VQF, AIFF, AU, Bonk, CD-audio (*.cda), FLAC, LA (Lossless Audio), LPAC, MIDI, Monkey's Audio, OptimFROG, RKAU, VOC, WAV (RIFF), WavPack

Audio – video formáty:

ASF: ASF, Windows Media Audio (WMA), Windows Media Video (WMV), AVI (RIFF), Flash, MPEG-1 / MPEG-2, NSV (Nullsoft Streaming Video), Quicktime, RealVideo

Obrazové formáty:

BMP, GIF, JPEG, PNG

Datové formáty:

ISO-9660 CD-ROM image (directory structure), SZIP (limited support), ZIP (directory structure)

Zápis:

ID3v1 (& ID3v1.1), ID3v2 (v2.3 & v2.4), VorbisComment on OggVorbis, VorbisComment on FLAC (not OggFLAC), APE v2, Lyrics3 (delete only). To bolo k základnému popisu štruktúry getID3. Ako už z popisu a nasledujúcich príloh vyplýva, táto štruktúra obsahuje všetky potrebné nástroje pre prácu s MP3 formátmy.

Teraz sa od obecného popisu dostávam k tomu, ako som použil getID3.

3.1.1 Analýza a výpis informácií MP3

Prvým krokom je nalinkovanie getid3 knižnice a to nasledovne:

```
require_once('/get_id3_folder/getid3.php');
```

Kde zložka get_id3_folder je zložkou kde sú uložené súbory štruktúry getID3 (ich úplný popis je v prílohe č. 1)

```
$getID3 = new getID3;
```

Inicializácia getID3 enginu

```
$ThisFileInfo = getID3->analyze($filename);
```

Analizuje súbor a uloží dáta do premennej \$ThisFileInfo (\$filename v tomto prípade znamená meno a cestu k súboru)

```
getid3_lib::CopyTagsToComments($ThisFileInfo);
```

tento príkaz zkopíruje dáta zo všetkých podpolí [tags] do poľa [comments] a tak sú všetky metadáta k dispozícii pre všetky Tag formáty.

```
echo @$ThisFileInfo['comments_html']['artist'][0];  
echo @$ThisFileInfo['tags']['id3v2']['title'][0];  
echo @$ThisFileInfo['audio']['bitrate'];  
echo @$ThisFileInfo['playtime_string'];
```

tu už prebieha výpis informácií. Tieto informácie už nie sú závislé a druhu tagu

3.1.2 Analýza adresára s MP3 súborom

Takto to funguje v prípade analýzy jedného súboru. V prípade že chceme analyzovať celé adresáre či podadresáre, môže skript pre túto úlohu vyzerať takto:

```
require_once('/get_id3_folder/getid3.php');  
$getID3 = new getID3;
```

Prvé dva riadky sú zhodné tak ako v predchádzajúcom príklade. A sú vlastne rovnaké pre všetky operácie.

```
$DirectoryToScan = '/example_folder';
```

Nastavíme adresár, ktorý chceme preskúmať

```
$dir = opendir($DirectoryToScan);  
while (($file = readdir($dir)) !== false)  
{  
    $FullFileName = realpath($DirectoryToScan.'/'.$file);  
    if (is_file($FullFileName)) {  
        $ThisFileInfo = $getID3->analyze($FullFileName);  
        getid3_lib::CopyTagsToComments($ThisFileInfo);  
        echo ...; }  
}
```

Tento cyklus vykoná analýzu všetkých súborov v zadanom adresári. Za príkazom echo nasleduje už len výpis týchto informácií, napr. do tabuľky, zápis môže byť obdobný ako v predchádzajúcom príklade.

[illegible]

Obr. 5 Výpis súboru v adresáry po jednoduchom naformátovaní do tabuľky

3.1.3 Zápis ID3 tagu do súboru

Nalinkujemo knjižnice a aktivujemo getid3 engine.

```
require_once('/get_id3_folder/getid3.php');
```

```
$getID3 = new getID3;
```

```
$getID3->encoding = $TaggingFormat;
```

```
nequire_once('../getid3/write.php');
```

nalinkujeme knižnicu ktorá sa stará o zápis tagu

```
$tagwriter = new getid3_writetags;
```

```
$tagwriter->filename = '/cesta_k_suboru.mp3';
```

```
$tagwriter->tagformats = array('id3v1', 'id3v2.3');
```

Nadefinujeme cestu k súboru a zvolíme aké informácie chceme do mp3 súboru zapisovať.

```
$tagwriter->overwrite_tags = true;
```

```
$tagwriter->tag_encoding = $TaggingFormat;
```

```
$tagwriter->remove_other_tags = true;
```

Toto je voľiteľné nastavenie, jedná sa o formát tagu, možnosť prepisu atď.

```
$TagData['title'][] = 'Moj song';
```

```
$TagData['artist'][] = 'Autor';
$TagData['album'][] = 'Nazov albumu';
$TagData['year'][] = '2004';
$TagData['genre'][] = 'Rock';
$TagData['comment'][] = 'komentár';
$TagData['track'][] = '04/16';
nastavíme údaje ktoré chceme do tagu zapísať
if ($tagwriter->WriteTags())
{
    echo 'Tag zapísaný';
    if (!empty($tagwriter->warnings))
    {
        echo 'Varovanie:<br>'.implode('<br><br>',
            $tagwriter->warnings);
    }
}
else
{
    echo 'Chyba zapisu<br>'.implode('<br><br>',
        $tagwriter->errors);
}
```

Funkcia `WriteTags()` vykoná zápis tagu a následne vypíše varovanie alebo chybovú hlášku.

3.1.4 Napojenie `getID3` na MySQL databázu

Pre moju prácu bolo tiež nutné uložiť si informácie o mp3 súboroch do databáze a túto databázu ďalej aktualizovať.

```
$getid3_demo_mysql_encoding = 'ISO-8859-1';
```

Nastavíme kódovanie mysql databáze

```
define('GETID3_DB_HOST', 'localhost');
```

```
define('GETID3_DB_USER', 'uživatel');
```

```
define('GETID3_DB_PASS', 'heslo');  
define('GETID3_DB_DB', 'databaza_mp3');  
define('GETID3_DB_TABLE', 'subory');  
nadefinujeme prístupové údaje, názov databáze a databázových tabuliek.
```

```
if (!@mysql_connect(GETID3_DB_HOST, GETID3_DB_USER,  
GETID3_DB_PASS))  
{  
die('Nedá sa pripojiť k databáze');  
}  
if (!@mysql_select_db(GETID3_DB_DB))  
{  
die('Nedá sa vybrať databáza');  
}  
if (!@include_once('../getid3/getid3.php'))  
{  
die('Nedá sa otvoriť getid3.php');  
}
```

Tieto zdrojové riadky slúžia k pripojeniu a vybratiu databáze, pokiaľ sa táto činnosť nepodarí, php vypíše chybové hlášky.

```
$getID3 = new getID3;  
$getID3->option_md5_data = $getid3_demo_mysql_md5_data;  
$getID3->encoding = $getid3_demo_mysql_encoding;  
Znovu nasleduje inicializácia getID3 enginu.
```

To je asi všetko k ukladaniu informácií do databáze. Ešte sa k tejto téme vrátim v jednej z príloh, kde uvediem formát tabuľky s ktorou štruktúra getID3 pracuje.

3.2 Registrácia užívateľov

Tento systém má rozlišovať dve úrovne užívateľov ktorí sú rozdelení do dvoch skupín, a to do skupiny admin a user. Skupina user sa ešte delí na užívateľov ako fyzické a právnické osoby.

System registrácie užívateľov sa skladá zo skriptov `register_p.php` a `register_f.php`, tieto skripty sú defakto formuláre, ktoré sa jeden od druhého líšia len vkladacími údajmi, ktoré sú pre fyzické a právnické osoby rozdielne. Každý z týchto formulárov má však svoju databázovú tabuľku.

Na tieto formuláre potom naväzuje skript, je to jediný skript `add_user.php` ktorý prevádza kontrolu a správnosť vložených údajov.

Užívateľ je definovaný svojím prihlasovacím menom a heslom. Prihlasovacie meno je v systéme jedinečné, a kontrola jedinečnosti sa vykonáva už pri registrácii. Tak ako je kontrolované číslo poistenia, ktoré je samozrejme jedinečné. Preto sa nedá vytvoriť na jednu fyzickú osobu viac registrácií. Číslo poistenia je tu použité ako jedinečný identifikátor registrovanej osoby. V Slovenskej republike je číslo poistenia rovné rodnému číslu, ale v iných štátoch sa rodné číslo nepoužíva a používa sa práve len číslo poistenia. U právnických osôb je ako jedinečný identifikátor použité identifikačné číslo, tzv. IČO.

Registrace nového fyzického uživatele			
uživatelské jméno:		jméno:	
heslo:		příjmení:	
heslo znovu:		adresa:	
		číslo pojištění :	
registrovat			

Obr. 6 Registračný formulár

3.3 Autorizácia nahrávky a digitálny podpis

V prvej fázi vývoja systému pre správu autorských práv u mp3 súborov bude nahrávka autorizovaná pomocou digitálneho podpisu. Digitálny podpis je v tomto prípade vytvorený z informácií o užívateľovi. To znamená, že k vytvoreniu podpisu sa použije meno, priezvisko a číslo poistenia registrovaného užívateľa. K tomuto reťazcu sa pridá špeciálne heslo, ktoré si užívateľ zvolí sám. Toto platí u registrovaných fyzických osôb. U právnických osôb sa ako zdroj pre vytvorenie digitálneho podpisu použije meno organizácie,

identifikačné číslo, a tak ako v predchádzajúcom prípade, k celému reťazcu sa pridá užívateľsky definované kontrolné heslo.

Na výsledný reťazec použijeme funkciu `md5()` v php, tá nám vytvorí tzv. hash vstupného reťazca. Toto bude náš výsledný digitálny podpis, ktorý sa uloží do databáze k tabuľke užívateľov.

MD5, celým názvom **Message-Digest Algorithm**, je produkt spoločnosti RSA Data security, Inc. Funguje samozrejme len jedným smerom. Prelomenie tohoto algoritmu si vyžaduje nepomerne vyššie prostriedky než je samotný profit z prelomenia tohoto kódu v prípade nášho systému.

Schéma digitálneho podpisu:

Meno+priezvisko+číslo_poistenia+užívateľsky_voliteľný_kód

➤ použitie md5 kódovania

=

dc9b7cad30e9826543a0c74e8c33be56

➤ uloženie do databáze

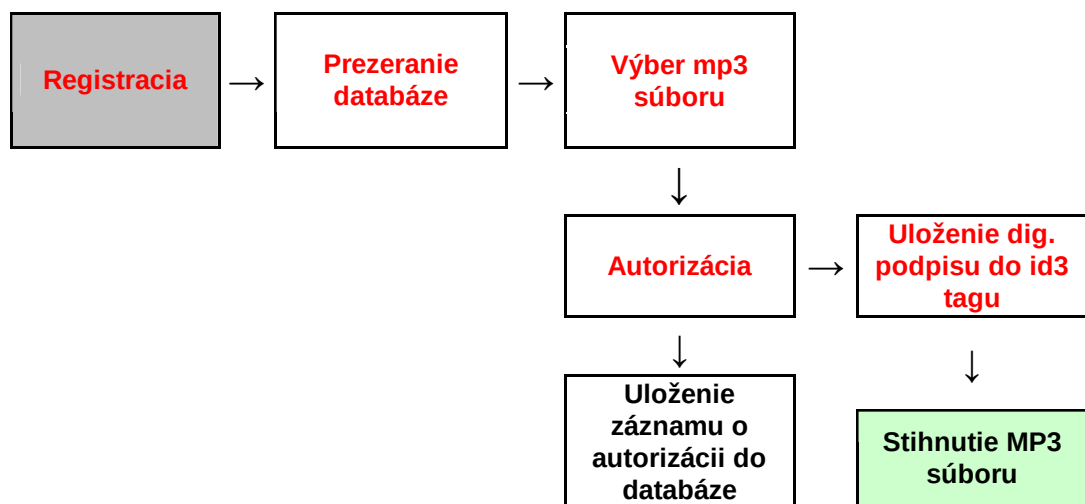
Skript ktorý zaisťuje autorizáciu sa nazýva `authorize.php` a jeho detailnejší popis nie je nutný. Tento digitálny podpis sa potom použije k vlastnej autorizácii mp3 súborov a to tým spôsobom, že sa vloží pri autorizácii do id3 tagu a stane sa jeho neoddeliteľnou súčasťou. Záznam o tejto autorizácii bude uložený na strane serveru, spolu s ďalšími údajmi, ako je dátum autorizácie a hlavne hash vlastného mp3 súboru. O tom ale až v nasledujúcej kapitole.

3.4 Identifikácia MP3 súboru v databázi

Aby bolo možné jednoznačne identifikovať a porovnať mp3 súbor, ktorý bude uložený na strane serveru a bude mať svoj záznam v databáze, vytvoríme tiež hash, alebo kontrolný súčet, ktorý bude vychádzať z vlastností súboru. Vlastnosťami súboru sa rozumie jeho veľkosť, bitrate, rozsah, kódovanie a údaje vložené v id3 tagu.

Tento kontrolný súčet bude súčasťou záznamu o mp3 súbore v databázi.

3.5 Celková schéma autorizácie



3.6 Prístupové práva

Ako bolo napísané v kapitole 3.2, systém rozlišuje dve kategórie užívateľov, a to užívateľov a administrátorov. Jak je už z názvu viditeľné, tieto dve skupiny majú úplne odlišné práva.

Chcel by som tu popísať aspoň stručne, akým spôsobom prebieha registrácia a autorizácia vlastného mp3 súboru.

Užívateľ vyplní registračný formulár, pokiaľ ho vyplní správne, prebehne jeho automatické zalogovanie do systému. Tu má možnosť si prezerať jednotlivé nahrávky, prípadne ich autorizovať a stiahnuť. Informácie o jednotlivých autorizáciách sú tam uložené tiež.

Možnosti administrátora sú rozdielne. Administrátor má k dispozícii niekoľko nástrojov pre správu mp3 databáze. Tieto nástroje zabezpečujú prácu s id3 tagy atď. Ďalej tu má k náhľadu štatistiky prístupu jednotlivých užívateľov a autorizácií.

Všetky možnosti jednotlivých skupín užívateľov popisuje nasledujúca tabuľka.

user	administrator
prehliadanie katalogu mp3 suboru filtrácia zobrazených skladieb autorizácia mp3 súboru	pridavanie a odoberanie mp3 skladieb z databáze uprava id3 priamo na fyzickom súbore synchronizácia verzií id3 tagu filtrácia súboru v databáze podľa: identického md5 súčtu autora a názvu album audio bitrate VBR (Variable bitrate) Metody (Lamé..) podľa verzie ID3 tagu žánru Kódovania audia rozpoznanie chybného formátu suboru

Obr. 7 Rozdelenie práv

3.7 Skripty systému

Tu popíšem stručne význam jednotlivých skriptov v systéme.

3.7.1 skript login.php

Tento skript slúži pre zalogovanie užívateľa do systému. Vykoná overenie prístupových údajov, zkontroluje či nie je užívateľ deaktivovaný administratorom a vytvorí užívateľskú session. (Session štandardne funguje len so zapnutými cookies vo webovom prehliadači).

3.7.2 skript `authorize.php`

Skript `authorize.php` obsahuje štrukturu `case` s dvomi položkami, jedna zaisťuje vytvorenie digitálneho podpisu po registrácii užívateľa, druhá položka je potom autorizácia mp3 súboru týmto digitálnym podpisom.

3.7.3 skript `browse.php`

Skript slúži pre prehliadanie skladieb ktoré sú uložené v databázi. Obsahuje aj jednoduché procedúry pre filtráciu výpisu.

3.7.4 skript `re_authorize.php`

Ako už z názvu vyplýva, úlohou tohoto skriptu je znova autorizácia mp3 nahrávky v prípade, že užívateľ o túto nahrávku nejakým spôsobom príde. Tento skript najskôr vyhladá záznam o autorizácii nahrávky, zkontroluje ho a následne vyhladá fyzické umiestnenie mp3 súboru na serveri a pripraví súbor pre autorizáciu a ziahnutie užívateľom.

ZÁVER

Výsledkom tejto bakalárskej práce je funkčný systém pre správu autorských práv u súborov MP3. Štruktúra `getID3` ktorá bola v ňom použitá sa stala stavebným kameňom celého systému. Systém pre prácu s ID3 tagy v prostredí PHP sa skladá z niekoľkých desiatok php skriptov, ktoré umožňujú kompletné pokrytie všetkých dôležitých a mysliteľných požiadaviek. Najdôležitejším skriptom je `getID3.php`, je to vlastne knižnica, ktorá vytvára funkciu `getID3()`. Táto funkcia potom slúži pre získanie kompletných informácií o MP3 súbore (uplný výpis je tiež popísaný v prílohe č. 2). Je tu aj možnosť konfigurácie chovania tejto funkcie

K napísaniu tejto bakalárky som bol inšpirovaný aj kôli tomu že nepoznám žiadny takýto funkčný systém na slovenskom alebo českom internete. Pokiaľ bude pokračovať ďalší vývoj toho systému má šancu že sa jeho určitá odboda uchyť a začne používať vo väčšej miere. Taktiež problematika okolo dodržiavania autorských práv je v účastnosti veľmi aktuálna a stretávame sa s ňou už skoro všade. Na ďalšom vývoji toho systému sa dá ešte zapracovať pretože je relatívne ľahko napadnutelný, keďže id3 tag ide relatívne ľahko prepísať a nieje stopercentné že nahrávka je totožná s týmto ID3 Tagom. Optimálnym riešením by bola integrácia už spomínaného systému na ochranu súborov tzv. Watermarking. Celý systém ešte nebol poriadne otestovaný v reálnom prostredí a zatiaľ ani nebola vybrána platforma na ktoré bude pracovať. To by ale nemal byť problém, pretože jak skriptovací jazyk PHP, tak databázový systém MySQL sú multiplatformový.

ZOZNAM POUŽITEJ LITERATURY

- [1] Bakken, S. S. a kol: Dokumentace systému PHP, online dokument dostupný na adrese <http://www.php.net/manual/cs/>
- [2] MASLAKOWSKI, M. *Naučte se MySQL*. Praha, ComputerPress, 2001.
- [3] PHP Group, The, *PHP* [online]. C2001, poslední revise 30.05.2004. Dostupné z [<http://www.php.net>](http://www.php.net).
- [4] TcX, *MySQL* [online]. C2001, poslední revise 24.05.2004. Dostupné z [<http://www.mysql.com>](http://www.mysql.com).
- [5] MP3–Hydrogenaudio Knowledgebase. Dostupné z [< http://wiki.hydrogenaudio.org/index.php?title=MP3>](http://wiki.hydrogenaudio.org/index.php?title=MP3)

ZOZNAM OBRÁZKOV

Obr. 1	Funkčná schéma bloku audio kodeku MPEG1 layer 3	13
Obr. 2	Príklad ID3v1 Tagu	14
Obr. 3	Príklad ID3v1.1 Tagu	16
Obr. 4	Príklad ID3v2 Tagu	17
Obr. 5	Výpis súboru v adresáry po jednoduchom naformátovaní do tabuľky	25
Obr. 6	Registračný formulár	28
Obr. 7	Rozdelenie práv	31

ZOZNAM PRÍLOH

P 1 : SÚBORY ŠTRUKTÚRY GETID3

P 2 : TABUĽKA POPISUJÚCA MP3 SÚBOR V GETID3