Potato System and Signed Media Format - an Alternative Approach to Online Music Business

Jürgen Nützel
Technische Universität Ilmenau
D-98684 Ilmenau, Germany
Juergen.Nuetzel@TU-Ilmenau.de

Rüdiger Grimm
Technische Universität Ilmenau
D-98684 Ilmenau, Germany
Ruediger.Grimm@TU-Ilmenau.de

Abstract

Thanks to modern compression techniques and increased bandwidth, the distribution of digital music via Internet has become affordable and easy. Many peer-to-peer (P2P) systems show this effect spectacularly. Therefore music publishers rely on so-called strong Digital Rights Management (DRM) systems which restrict and control the usage of their content. In this paper, we want to discuss a different approach. We introduce a new business model as well as a new file format. The system that we propose is called Potato. In Potato System the users play an active distribution part. Our approach motivates the users to re-distribute content they have paid for and earn money with it. The Potato System pays for any re-distributed file a defined percentage on commission. This allows a fast distribution of new content. The Potato System provides its own P2P clients which contact a central web-service. In the standard Potato System the identity of the last buyer is simply added to the file name. This is sufficient to reward re-distributing users. For well known major music we provide the so called Signed Media Format (SMF). In SMF files the user identity is signcrypted into the media content.

1. Motivation and Introduction

The distribution of music and other virtual goods over the Internet is simple and cheap. Even the consumers act as distributors. In fact, it has become simple enough to allow anyone to act as a distributor. The traditional and centralized view of the publishers makes them believe that a free usage of digital content out of their control would undermine their business models. Music publishers therefore rely on so-called strong Digital Rights Management (DRM) systems which restrict and control the usage of content that has been legally downloaded and paid for [1],[2]. Many potential customers would pay for digital content without usage control. But providers restrict the usage of their content and treat their customers as enemies [3]. They ignore decentralist architecture of the Internet and the strong position of the users within. This conflict blocks the development of a growing business on the Internet.

What can we do in order to put online music business back on sound feet? Our idea is to keep the honest people honest. We propose an alternative approach which motivates the honest user to cooperate with the interests of the publishers and artists. We call this approach Potato System [4].

Content providers want to sell their products. And selling means distribution. Therefore, content providers do have high interest in the distribution of their products. And as we now from peer-to-peer (P2P) systems like KaZaA re-distribution is obviously in the interest of the end-users. However, the content providers want to realize a profit. In the Potato System the users do not pay for the (almost public) content itself, they pay for a (song-based) re-distribution license and service (e.g. user matching [5] and access to P2P network) concerning this license. A customer, who pays for that service, gets the right to re-distribute the song and earn money with it.

Why should a user pay for such a license? He wants to get a reward, a percentage of the payment, which the next recipients pay for a similar license. If he does not pay for a license, he will have no chance to get any reward later [6].

In the next chapter we describe the standard version of Potato System which is applicable to every file type. This file format independency underlines the open and fair-use idea of the Potato System. In the third chapter we introduce a special version of Potato System which uses a new file type. The new file format is used to store detailed license information.

2. The Potato System

In this section we describe the Potato System, which was invented at Fraunhofer AEMT and the 4FO AG (www.4fo.de).

The system description is divided into two subsections. In the first subsection we explain traditional download uses-cases without any need for a P2P network. The second subsection describes the role of the P2P functionality within the Potato System.

2.1 The Provider or Artist Sells the Content Using a Music Portal with integrated Payment System

First we describe how a content owner or artist brings new music into the Potato System community. The first use-case is called "content registration". In this use-case three actors are involved. Let Fred, George and Potato play different roles in the content registration use-case. Fred is an artist or music producer. He produced a song (*mysong.mp3*) which is ready to publish. The file is located at Fred's own web-server in a subdirectory which is unknown to the public. In the next step Fred contacts George. George runs a music portal with integrated payment. George co-operates with the Potato System. Fred tells George's server where to find the song. Fred defines a price (e.g. 1.10 Euro) and a price model. The price model defines the algorithm to calculate re-distributors commissions. Let us suppose Fred defines a commission rate of 50%. To complete the content registration George's server contacts (via XML) the accounting server of the Potato System. The accounting server is realized as a web-service [7]. George's server transfers the registration information and a calculated SHA1 hash from Fred's file to this web-service.

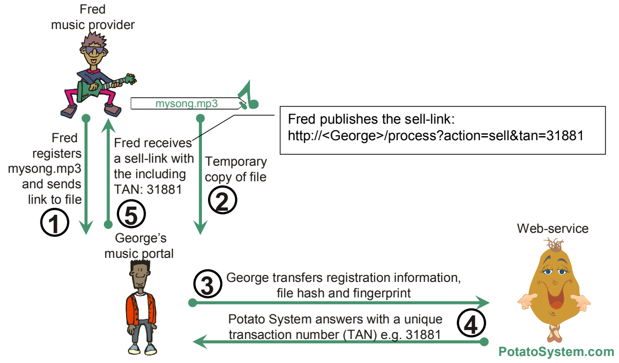
Fred
music provider
mysong.mp3
Fred publishes the sell-link:
http://<George>/process?action=sell&tan=31881
Fred registers mysong.mp3 and sends link to file
1
Fred receives a sell-link with the including TAN: 31881
5
Temporary copy of file
2
George's music portal
Web-service
3 George transfers registration information, file hash and fingerprint
Potato System answers with a unique transaction number (TAN) e.g. 31881
4
PotatoSystem.com

Figure 1 Fred registers mysong.mp3 on the music portal. George co-operates with Potato System.

The hash allows the Potato System web-service in later use-cases to check integrity of the file. If the file includes audio content a robust fingerprint like AudioID [8] is applicable. Such a fingerprint allows Potato System to identify *mysong.mp3* even after down-sampling or other modifications. Potato System stores all these information and answers with a unique transaction number (TAN). *31881* is the TAN in figure 1.

A TAN is the receipt for the registration. Every TAN in the Potato System starts with a customer number (here "3188") followed by a user specific transaction number.

George uses the TAN to build a sell-link for Fred's song. George publishes this sell-link *http://<George>/process?action=sell&tan=31881* on his portal. *<George>* stands for George's address. The sell-like could be published additionally on Fred's own webpage or any other page in the web.

Figure 2 shows the "pay before download" use-case. To simplify its description we suppose that Ginny already has a login and customer number ("3712") from the Potato System. Let Ginny play the role of a fan who wants to buy the newest song of Fred. Ginny enters (e.g.) Fred's web-site and clicks the sell-link. The link leads Ginny to George's payment service. After successful payment George contacts the Potato System to register Ginny's purchase. After this registration process Ginny is an official re-distributor of Fred's song. The Potato System answers with a new TAN ("37121"). The TAN is the receipt and proof for Ginny's (license) purchase.

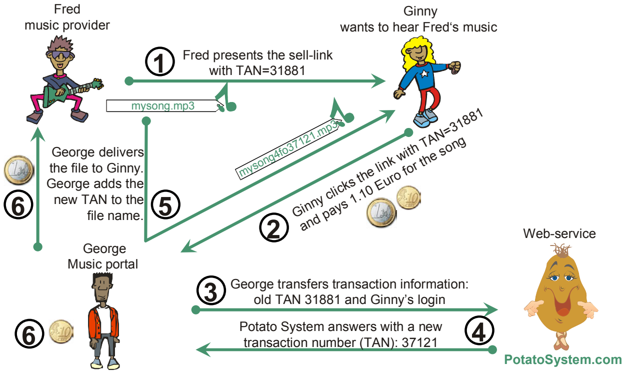
Fred
music provider
Ginny
wants to hear Fred's music
1 Fred presents the sell-link with TAN=31881
mysong.mp3
mysong4fo37121.mp3
George delivers the file to Ginny. George adds the new TAN to the file name.
5
6
Ginny clicks the link with TAN=31881 and pays 1.10 Euro for the song
2
George
Music portal
Web-service
3 George transfers transaction information: old TAN 31881 and Ginny's login
Potato System answers with a new transaction number (TAN): 37121
4
6
PotatoSystem.com

Figure 2 Ginny pays for Fred's song using George's payment service. Ginny becomes a re-distributor

While Ginny downloads the file via George's server George adds the new TAN to file name. The new file name is *mysong4fo37121.mp3*. File renaming is the easiest way to add a receipt (for a license) to a file. The complete license information is stored on the accounting server. Later we describe a more elegant way to add license information to media content. Beside the file Ginny receives her own sell-link. Ginny can publish this link like Fred or George. If a new user follows her link and pays, Ginny will receive her commission.

2.2 Potato Users Share Files and pay for Licenses in the Peer-to-Peer System

Ginny has several motivations to pay for Fred's song. The song was brand new and there was no other way to find the file. A second motivation was that Ginny wants to become a re-distributor. As a re-distributor Ginny sends her sell-link to her friend Harry. Or Harry finds this link on Ginny's home-page. If Harry buys the song using this link, Ginny receives 50 Cents from Fred's revenue. This is already known as affiliated marketing. Amazon does this with books. Affiliated programs are not fully sufficient to promote new content to new customers. This was the reason to provide Ginny a special P2P client, which allows her to transfer freely registered Potato System content. We call this client P2P Potato Messenger. The client is a signed Java applet, which uses the open source peer-to-peer framework JXTA [9].

Using the P2P client Ginny gets free access to content other Potato System users have paid for. But Ginny is only able to transfer content of limited value. The limit is (e.g.) 20-times of the amount she has paid for Fred's song before. If Ginny wants to "test" more new music she has to purchase a re-distribution license for one of the songs she already has transferred. We are also discussing to demand a small base fee for the P2P access. But Ginny has a second opportunity. She could pay with the credits on her Potato System account. To earn more credits Ginny has to provide on her computer songs she has a re-distribution license for.

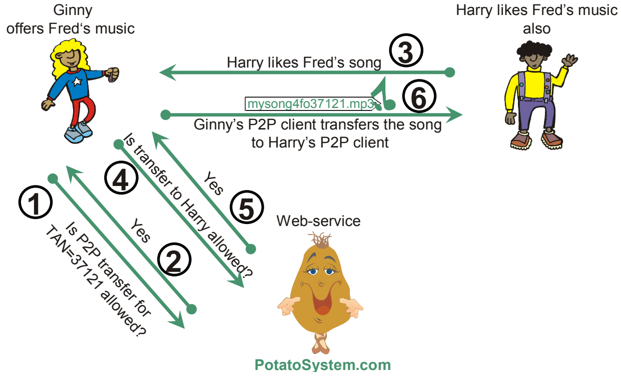

Figure 3 Ginny transfers (P2P) Fred's song to Harry. Harry receives a free copy of the song.

Figure 3 shows the peer-to-peer use-case with Ginny, Harry and the central Potato System web-service. Ginny uses the Potato Messenger to offer Fred's song. To setup, the Potato Messenger contact the web-service to check if the file was registered in the Potato System (white list check). The messenger sends for every file Ginny wants to offer the TAN and the SHA1 hash. Optionally the messenger could send the AudioID [8]. Harry also uses the P2P client. He found Ginny's offer (using the matching service of Potato). But before Ginny's client is allowed to transfer a file to Harry, it has to ask the web-service. Potato System checks if Harry is still allowed to make file transfers.

3. Signed Media Format

A great drawback of the Potato System transaction number (TAN) is that Harry has to contact the accounting server to verify the licence information of Ginny. In [3] we gave a solution for this problem. We defined a special data structure for a digital receipt (license). This detailed XML data structure contains beside authors name and re-distributor's name additionally information. The license is digitally signed by the accounting server. We used a Java-like (JAR) archive format to put the unmodified content file (e.g. mp3), the license information and the signature together (unsealed envelope). The JAR file type is based on ZIP. JAR provides structures for multiple signatures.

A similar approach to combine a license and the media content is called "light weighted DRM" [2],[10]. In this system a new file format which is called signed media format (SMF) is introduced. The core idea behind of SMF is a closer combination of content (license and media content) and user identity. The media content is symmetrically encrypted with a randomly generated secret key. The advanced encryption standard (AES) [11] is used for encryption. The secret key is signcrypted with the buyer's private key. The signcryption public key scheme was first proposed by Zheng [12]. It combines the functionality of a digital signature scheme with that of an encryption scheme in a more efficient way. Every SMF consists of several components:

- AES encrypted media content (MPEG4/AAC or MP3)
- Signcrypted secret AES key.
- X.509 certificate of buyer (Includes public key).
- License from the accounting server (XML).
- Signature of the license (by the accounting server)

The main advantage of signcryption and SMF compared to the JAR-approach is the close binding of buyer's identity and the content. A SMF compatible player has to decrypt the content using the certificate which is included. This forces the player to read and verify the digital signature of the last buyer.

The license is a XML data structure which is similar to one in described in [3]. It is generated and signed by the accounting server. It includes the following information:

- Date of purchase
- Name of content owner (e.g. Fred)
- Content Description
- AudioID of content (or SHA1 hash)
- Name of last buyer (e.g. Ginny)
- Price and price model
- Sell-link for Ginny (includes TAN)
- Additional meta data and further information

Before we explain the purchase of SMF via George's server, we have to tell what Ginny needs to do before. First she has to register at George service using her personal digital certificate. We suppose Ginny already has such a X.509 certificate. If not, she has to contact a so called certification authority (CA). In the registration process Ginny sends the public part of her certificate to George. George forwards this certificate to the Potato System web-service. The Potato System stores this certificate in Ginny's account. Now Ginny is ready to download a signed media file.

Figure 4 shows the download use-case for SMF. Fred provides on his server a SMF file of his song. He has signcrypted this file. Fred received from George a license for his file registration. In figure 4 steps 4, 5 and 7 mainly differ from figure 2. Step 4: Potato System provides a new license, which was signed by the web-service. Step 5: The signed license and the SMF file from Fred's server is delivered to Ginny. Step 7: Ginny's SMF player replaces Fred's signcryption and the old license with her signcryption and the new license (*license..xml*) A P2P functionality was added to make SMF more attractive.

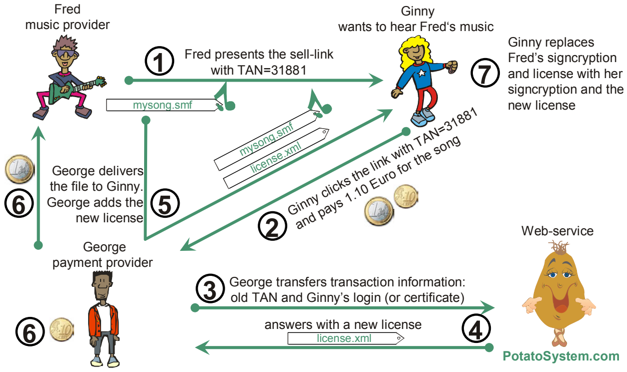

Figure 4 Ginny pays for Fred's song and a license.

A P2P transfer of SMF works similar to the transfers shown in figure 3. In figure 5 we see what happens after the P2P transfer (step 1). Step 2: Harry opens the SMF file using his player. Harry's SMF compatible player (with P2P functionality) compares the certificate found in the file (Ginny's) with his own (stored on disk). Because the certificates are different, a message box appears: "*Ginny is the owner and re-distributor of Fred's song, if you pay 1.10 Euro you become the owner and re-distributor. Ginny receives 50% of your payment*." Every time Harry plays Ginny's song, this box will pop up.

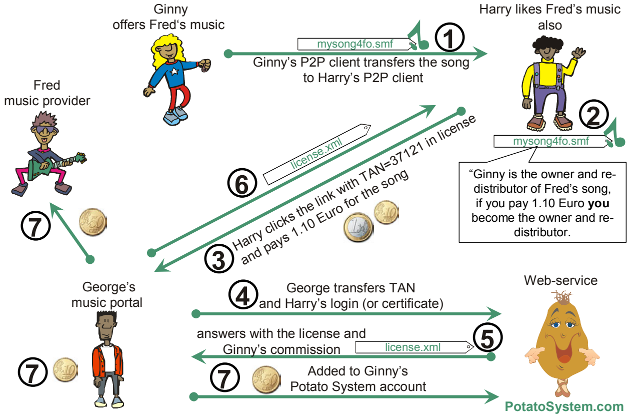
Ginny
offers Fred's music
Harry likes Fred's music
also
mysong4fo.smf
1
Ginny's P2P client transfers the song
to Harry's P2P client
Fred
music provider
2
mysong4fo.smf
"Ginny is the owner and re-distributor of Fred's song, if you pay 1.10 Euro **you** become the owner and re-distributor.
license.xml
6
Harry clicks the link with TAN=37121 in license
and pays 1.10 Euro for the song
3
7
George's
music portal
Web-service
4
George transfers TAN
and Harry's login (or certificate)
answers with the license and
Ginny's commission
license.xml
5
7
7
Added to Ginny's
Potato System account
PotatoSystem.com

Figure 5 Ginny uses the P2P system to transfer Fred's song to Harry. Later Harry pays for a license.

Step 3: Harry purchases his own license, because he wants to earn money and he wants to make more P2P transfers. Because Harry has no credits on his Potato System account, he has to pay via George's payment service. Step 3 to 5 is similar to the corresponding steps in figure 4. Step 6: A new license is transferred to Harry.

The major difference is shown in step 7. Fred shares his profit according to the price model. George transfers 50 Cents to Fred and 50 Cents to the Potato System. Potato System credits this money Ginny's account. Next time Ginny wants to purchase a song she can use her Potato System account.

4. Conclusion and Further Work

In the Potato System, the user pays not for content; he pays for a re-distribution license. This allows a fast and distribution of new music. Re-distribution of content without a license is still possible. However, it is not so attractive. The signed media format allows to pop-up special information while the song is rendered. Such pop-ups remember new users to pay for a license. This is the maximum applicable handicap. More handicaps or even local access restriction would undermine the idea of the fair-use in the Potato System.

More information is publicly available on [4] and [10]. We just started the standard version of Potato System. We use a P2P Java applet using JXTA [9]. We have implemented different user matching algorithms to make the purchase of license more attractive [5]. We provide further use-cases for the Potato System web-service. One of these use-cases works face-to-face using mobile devices. In a European research project we will try to apply Potato System to home HIFI systems.

References

[1] 3GPP TS 22.242 V6.0.0 (2002-06). Digital Rights Management (DRM) Stage 1, Release 6, June 2002.

[2] Neubauer, Ch.; Brandenburg, K.; Siebenhaar, F.: Technical Aspects of Digital Rights Management Systems, In 113th AES-Convention, LA, October 2002. Paper 5688

[3] Grimm, R., Nützel, J.: A Friendly Peer-to-Peer File Sharing System with Profit but Without Copy Protection. Intelligent Internet Computing Systems I2CS, Kühlungsborn, June 2002, Springer, LNCS 2346.

[4] Homepage of Potato System, http://www.4friendsonly.org

[5] Nützel, J.: Matching Algorithms in File-sharing Systems to find new Users having new Content. I2CS, Leipzig, June 2003, Springer, LNCS

[6] Grimm, R., Nützel, J.: Peer-to-Peer Music-Sharing with Profit but Without Copy Protection, Wedelmusic 2002, Darmstadt, 9th - 11th December 2002. IEEE Computer Society, ISBN 0-7695-1623-8, p. 17 ff. http://www.4friendsonly.org/eng/papers.htm

[7] W3C: Web Services Architecture, 2002. Working Draft. http://www.w3.org/TR/2002/WD-ws-arch-20021114/

[8] Allamanche, E.; Herre, J.; Hellmuth, O.; Fröba, B.; Cremer, M.: AudioID: Towards Content-Based Identification of Audio Material. In 110th AES-Convention, Amsterdam, 2001. Convention Paper 5380.

[9] Homepage of JXTA: http://www.jxta.org

[10] Homepage of LWDRM, http://www.lwdrm.com

[11] AES Home Page: http://csrc.nist.gov/CryptoToolkit/aes/

[12] Zheng, Y.: Digital Signcryption or How to Achieve Cost (Signature & Encryption) << Cost(Signature) + Cost(Encryption), Advances in Cryptology - CRYPTO '97, Springer, LNCS 1294, p. 16