Electronic Payment and Encouraged Cooperation in a Secure and Privacy-Preserving P2P Content Distribution System

Amna Qureshi, Helena Rifà-Pous and David Megías
Estudis d'Informàtica, Multimèdia i Telecomunicació, Internet Interdisciplinary Institute (IN3)
Universitat Oberta de Catalunya (UOC), Barcelona, Spain
E-mail:{aqureshi,hrifa,dmegias}@uoc.edu

***Abstract*—In this paper, we propose a peer-to-peer (P2P) content distribution system that allows the efficient distribution of large-sized multimedia contents while preserving the security and privacy of content providers (merchants) and end users (buyers), respectively. However, the privacy of a buyer can be revoked as soon as he/she is found guilty of copyright violation. A payment protocol is also proposed that provides a secure payment mechanism, where personal information and order information cannot be exposed to an unauthorized third party. In addition, a reputation-based system is introduced for the selection of the proxy peers required for secure delivery of the fingerprinted content from the merchant to the buyer. The paper presents a thorough security analysis of the system against several security compromising attacks.**

***Keywords*—privacy; security; collusion-resistant fingerprinting; permutation; peer-to-peer; e-payment; reputation.**

1 Introduction

The low-cost, scalability and ease of content dissemination provide a lucrative opportunity for content providers to generate revenues through P2P systems. However, the content providers have been reluctant in adopting P2P systems as a distribution vehicle to monetize digital content, since these systems are plagued with piracy. The ability to make perfect copies and the ease with which these copies can then be distributed has given rise to significant problems regarding the misuse, illegal copying and re-distribution. The content providers apparently fear losing control of content ownership in the sense that they are no longer in control of the content distribution and worry about the promotion of illegal activity. Also, tracing a copyright violator in a P2P system with millions of connected users is an immense task. Therefore, ensuring the appropriate use of copyrighted multimedia content in P2P systems has become increasingly critical. This copyright infringement problem motivates the development of content protection techniques. Among various content protection techniques, digital fingerprinting addresses the problems of copyright protection and traitor tracing.

Digital fingerprinting gives merchants more options to control the distribution of their content. Fingerprinting techniques involve the generation of a fingerprint (a buyer-specific identification mark), the embedding operation and the realization of traceability from re-distributed copies. In traditional fingerprinting schemes, it is assumed that the merchants are trustworthy and always perform embedding honestly [1]. Thus, a dishonest merchant could frame an innocent buyer, while a cheating buyer would be able to deny his/her responsibility for a copyright violation act. Asymmetric fingerprinting schemes [2] were introduced to overcome this problem. In these schemes, only the buyer obtains the exact fingerprinted content, and hence the buyer cannot claim that a pirated copy was originated from the merchant. However, most of the asymmetric fingerprinting schemes in the literature incur high computational and communicational burdens at the merchant's and/or at the buyer's end, due to the use of cryptographic protocols such as homomorphic encryption or committed oblivious transfer.

Though the content protection techniques enable the merchants to enforce copyrights in the content, these techniques are often criticized for breaking buyers' privacy by collecting information about the buyers, such as the transaction history or the purchasing behavior. A priori, copyright protection places the buyer into an adversarial relation with the merchant. Hence, the incorporation of a content protection mechanism in a P2P system can have serious effects on the privacy interests of the buyers. Recent years have drawn increasing attention from the research community towards the preservation of the merchants' ownership property and buyers' privacy in P2P content distribution systems. To date, very few P2P distribution systems have been proposed that provide both copyright protection and privacy preservation.

Megías and Domingo-Ferrer [3] introduced a novel concept of a recombination fingerprinting mechanism for P2P content distribution. The proposed scheme provides copyright protection, collusion resistance and traitor tracing. However, this system is implemented with a two-layer anti-collusion code (segment level and fingerprint level), that results in a longer codeword. Furthermore, honest and committed proxies are required for the generation of valid fingerprints at the buyer's end. Megías [4] proposed an improved version of [3], in which a four-party anonymous communication protocol is proposed to prevent malicious proxies to access clear-text fingerprinted contents. However, the system still requires a two-layer anti-collusion code. Domingo-Ferrer and Megías [5] proposed a P2P protocol for distributed multicast of fingerprinted content in which each receiver obtains a different fingerprinted copy of the content, which allows the provider to trace re-distributors without affecting the privacy of honest buyers. However, an implementation of a secure multi-party protocol results in increased computational and communication costs at the buyer end. Qureshi, Megías and Rifà-Pous [6] proposed a P2P content distribution framework for preserving privacy and security of the user and the merchant based on homomorphic encryption. In the framework, some discrete wavelet transform (DWT) low-frequency (approximation) coefficients are selected according to a secret key for embedding an encrypted fingerprint to prevent data expansion due to homomorphic encryption. Although the selective public-key encryption of the multimedia content results in lesser data expansion, it imposes computational burden on a merchant and an increased complexity in file reconstruction at the buyer's end.

In this paper, we present a P2P content distribution system that provides copyright protection and conditional privacy to the merchant and the buyer, respectively. In the proposed system, the original multimedia file is partitioned by the merchant into a small-sized base file and a large-sized supplementary file. This enables to reduce the communication bandwidth and the computation power required by the merchant in delivering the large-sized multimedia file. The base file contains the most important information and is transmitted in a semi-centralized way. The supplementary file is unusable without the base file and is distributed through a P2P network. A merchant forms a base file by using a pre-computation-based secure embedding mechanism in which the DWT approximation coefficients are embedded in parallel with all 1s and all 0s bit streams. An asymmetric fingerprinting protocol based on collusion-resistant codes and a robust embedding scheme is performed between a merchant, a buyer and a set of proxies in the presence of a third party (monitor), in such a way that the merchant does not know the fingerprint or the fingerprinted content, and the proxies are unable to frame honest buyers by combining their assigned permuted fingerprint bits. A reward and punishment mechanism is also proposed to ensure that each proxy peer's best strategy is to loyally follow the prescribed fingerprinting protocol. The system also enables buyers to purchase digital contents anonymously by using dynamic pseudonyms based on a one-way hash function instead of their real IDs.

The paper is organized as follows. In Section 2, the building blocks of the system are introduced. In Section 3, the proposed P2P content distribution system is described in detail. In Section 4, we discuss the security analysis of the system's protocols through a number of attack scenarios. Section 5 presents the comparative analysis of the proposed system with related P2P content distribution systems. Finally, Section 6 summarizes the conclusions.

2 Building Blocks

In this section, a brief overview of the building blocks (embedding domain and algorithm, collusion-resistant fingerprinting codes, PseudoTrust model and permutation) of the system is presented.

A. Embedding domain

In the signal processing research area, the wavelet transform has gained widespread acceptance in recent years. The DWT is used in the system to embed the collusion resistant fingerprint into a multimedia content. The DWT of a signal results into approximation and detail coefficients. Since the low frequency coefficients can effectively resist various signal processing attacks, the fingerprint bits are typically embedded into the approximation coefficients of the signal after the DWT. Moreover, the original signal can be reconstructed from the approximation and detail coefficients through the inverse discrete wavelet transform (IDWT).

B. Embedding algorithm

An embedding algorithm is used to embed a fingerprint into different copies of the same content. Quantization index modulation (QIM) [7] is a relatively recent embedding technique that has become popular because of the high watermarking capacity and the ease of implementation. The basic QIM scheme embeds a fingerprint bit f by quantizing a DWT coefficient W by choosing between a quantizer with even or odd values, depending on the binary value of f. The proposed system employs a QIM-based watermarking technique to embed the collusion-resistant fingerprint into the content.

C. Collusion-resistant fingerprinting codes

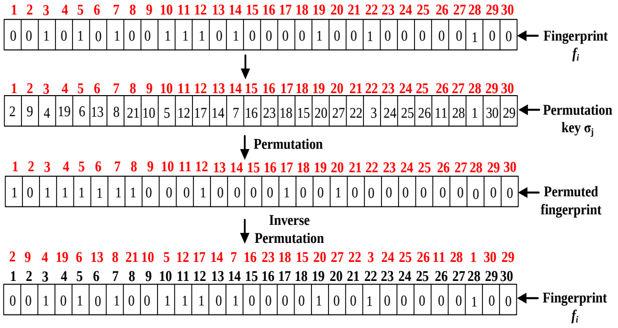
Nuida *et al.*'s c_0-secure codes [8] are used in the system for the generation of the collusion-resistant code. Nuida *et al.* proposed a discrete distribution of state-of-the-art collusion-resistant Tardos codes with a δ-marking assumption (the number of undetectable bits that are either erased or flipped is bounded by δ-fraction of the total code length m) to reduce the length of the codewords and the required memory amount without degrading the traceability. The tracing algorithm of Nuida *et al.* outputs one user with the highest accusation score. The details of Nuida *et al.*'s fingerprint generation and traitor-tracing algorithms can be found in [8].

D. PseudoTrust model

The PseudoTrust model proposed by Lu *et al.* [9], based on a zero-knowledge proof-of-identity, is used in the system to provide revocable anonymity and unlinkability properties. The PseudoTrust model enables pseudonym-based trust management such that the real identities of the peers are protected during the authentication. In addition, the communication between two peers is anonymized using onion routing within the system. In the PseudoTrust model, the pseudo-identities are generated by the peers without any trusted third party, which leads to an accountability problem in the system. Thus, to add accountability to our system, an internal certificate authority (CA_R) is incorporated in the PseudoTrust model. Each peer is authenticated by CA_R before he/she joins the network. Hence, each peer has a private key, a public key and a public-key certificate signed by CA_R. The details of generation of pseudo-identities and anonymous authentication process are provided in [6].

E. Permutation

In the proposed system, the buyer's security and non-repudiation (merchant's security) are provided by using the concept of permutation. The permuted fingerprint generated by the monitor is permuted using different permutation keys and is then assigned to a set of proxy peers Pr_j in such a way that the merchant cannot predict about the fingerprint and the fingerprinted content, and Pr_j are unable to frame honest buyers by combining their information bits.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Fingerprint f_i	0	0	1	0	1	0	1	0	0	1	1	1	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Permutation key σ_j	2	9	4	19	6	13	8	21	10	5	12	17	14	7	16	23	18	15	20	27	22	3	24	25	26	11	28	1	30	29

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Permuted fingerprint	1	0	1	1	1	1	1	1	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0

	2	9	4	19	6	13	8	21	10	5	12	17	14	7	16	23	18	15	20	27	22	3	24	25	26	11	28	1	30	29
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Fingerprint f_i	0	0	1	0	1	0	1	0	0	1	1	1	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0

Figure. 1: Permutation of a fingerprint

Figure. 1 illustrates the permutation concept of a fingerprint in the system. Figure. 1 shows a fingerprint f_i of 30 bits, and a random permutation key σ_j of 30 elements. σ_j is applied to f_i such that the bit position 1 of the fingerprint corresponds to the bit position 2 of a permuted fingerprint ($1 \rightarrow 2$), the second bit position corresponds to the bit position 9 of the permuted fingerprint ($2 \rightarrow 9$), and so on. On applying the inverse permutation key σ_j^{-1} to a permuted fingerprint, the original fingerprint f_i is obtained.

3 Proposed System

This section describes the design and functionality of the system. In Section 3-A, we define the role of each entity. Section 3-B defines the functionality requirements and the security assumptions.

A. System entities

The system involves seven entities and the function of each entity is defined as follows:

- A merchant M is an entity that distributes the copyrighted content to the buyers in the P2P system. It is involved in the fingerprint generation, the file partitioning, the distribution of base and supplementary files, the traitor tracing and the dispute resolution protocols.
- A buyer B_i is an entity that can either play the role of data requester or provider. B_i is involved in the registration protocol, acquisition of a base file (*BF*) from the merchant, the distribution of a supplementary file (*SF*) through the system, the file reconstruction protocol and a dispute resolution, in case he/she is found guilty of copyright violation.
- A super peer *SP* acts as a coordinator for a small portion of the group of peers (buyers). However, instead of peers' addresses, their pseudonyms are stored. *SP* facilitates B_i's acquisition of *BF* from M, and *SF* from the buyers present in the system.
- A Certification Authority CA_R is a trusted party that is responsible of issuing certificates to the buyer for the acquisition of *BF* from M, and *SF* from other buyers.
- A monitor *MO* functions as a trusted party, which is responsible for the registration of buyers and merchants, the generation of collusion-resistant fingerprint codes, the distribution of *BF*, the file reconstruction, the traitor tracing and the dispute resolution protocol. *MO* also acts as a bank that assists B_i to download *BF* from M after making a payment. In addition, *MO* manages the rewards and punishments mechanism in the system.
- A proxy peer Pr is responsible for querying content of *BF* available at M's end with the pre-assigned bits of a fingerprint codeword and transferring the retrieved content to B_i.
- A judge J is assumed to be a trusted party, which resolves the disputes between M and B_i with the cooperation of *MO* and CA_R.

B. Design requirements and assumptions

In this section, the design requirements and security assumptions of the system are described.

- **Design Requirements:**
  - M should be able to trace and identify an illegal re-distributor in case of finding a pirated copy with the help of *MO*, J and CA_R.
  - The scheme should be collusion-resistant against a given number of colluders c_0 as specified by Nuida *et al.* codes [8].
  - The possible collusion of Pr_j should be unable to frame an honest B_i. Also M should not be able to frame an honest B_i of illegal re-distribution.
  - A B_i accused of re-distributing an unauthorized copy should not be able to claim that the copy was created by M or a collusion of the proxies Pr_j.
  - The real identity of a buyer should remain anonymous during transactions unless he/she is proven guilty of copyright violation.
  - J, with the help of *MO*, should be able to resolve the disputes without involving B_i in the process.

- The reconstruction of the original file from *BF* and *SF* should be performed at the buyer's end. *BF* cannot be shared within the buyers of the system.
- The buyers should register to *MO* with a subscription fee at a system start-up.
- The coin generated by *MO* should be revocable, thus enabling *MO* to refund the money to B_i in case of incomplete *BF* delivery to B_i.

• **Security Assumptions:**

- M and B_i do not trust each other but they both trust *MO*.
- In order to deliver *BF* from M to B_i, *MO* selects a fixed number (n) of proxy peers. These proxy peers follow each other in a sequential manner to transfer *BF* to B_i from M.
- The permutation keys σ_j (for $j = 1, \ldots, n$) are generated by B_i to perform permutation of a fingerprint codeword to be assigned to the proxy peers (Pr_j).
- Pr_j are not trusted and the content transferred through them is encrypted.
- Each entity (M, *MO*, Pr_j, B_i, CA_R, J) is supposed to have a public key K_p, a private key K_s. Public-key cryptography is restricted to the encryption of small-length binary strings, such as symmetric session and permutation keys.
- Before joining the system, B_i is authenticated by CA_R of the system. Once authenticated, B_i obtains a private key and a public key certified by CA_R. CA_R generates a random number r and shares it with an authenticated B_i for the generation of a pseudo-identity. Each buyer can have multiple pseudo-identities.
- M is assumed to be registered with *MO* at a system start-up.

4 Model

In this section, we detail the system designing and how to motivate the proxy peers in the base file distribution protocol to rationally play their corresponding roles.

A. Registration

Before joining the system, each buyer is assumed to be authenticated by CA_R and also the pseudo-identity of each buyer is assumed to be generated (Section 3-B). On joining the system, B_i sends a registration request to *MO* with his/her pseudo-identity. On receiving the request, *MO* verifies the pseudo-identity of B_i from CA_R. On verification, *MO* opens up an account of B_i and sends him/her the details of the subscription fee payment. B_i deposits the subscription fee and sends the signed payment receipt to *MO*. *MO* acknowledges the payment, creates a transaction identity TID in his/her database and generates a digital coin C_{B_i}. Then, *MO* signs C_{B_i} and sends it to B_i. M is also assumed to be registered with *MO*. Once registered with *MO*, the buyers connect with *SP* to obtain the multimedia content. In case the same buyer B_i joins the system with another pseudo-identity, he/she must send the old pseudo-identity to *MO* along with the new pseudo-identity in the registration request. Figure. 2 illustrates the registration protocol between MO and B_i.

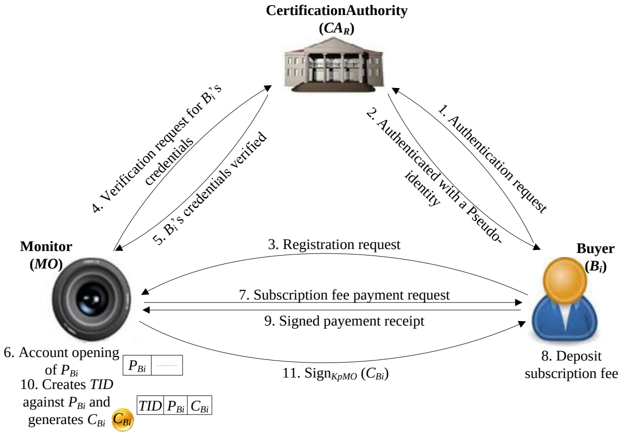

CertificationAuthority
(CA_R)
1. Authentication request
2. Authenticated with a Pseudo-identity
3. Registration request
4. Verification request for B_i's credentials
5. B_i's credentials verified
Monitor
(*MO*)
Buyer
(B_i)
7. Subscription fee payment request
9. Signed payement receipt
6. Account opening of P_{Bi}
P_{Bi}
10. Creates *TID* against P_{Bi} and generates C_{Bi}
C_{Bi}
TID
P_{Bi}
C_{Bi}
11. Sign$_{KpMO}$ (C_{Bi})
8. Deposit subscription fee

Figure. 2: Registration protocol

B. Fingerprint generation

The algorithm for fingerprint generation takes a parameter ε for error probability, the total number N of users and c_0 colluders as inputs, and outputs a collection $F = (f_1, \ldots, f_N)$ of binary codewords (f_i) of size m and a secret bias vector p. The details of the fingerprint generation algorithm can be found in [8].

C. File partitioning

The DWT decomposition on a file results in approximation (a) and detail (d) coefficients. The 3-level approximation coefficients (a_3) are used to imperceptibly embed f_i using a blind, robust and secure QIM-based watermarking scheme. M uses a_3 twice to create *BF* in such a way that it employs an embedding algorithm to insert a codeword of all ones into a_3 and simultaneously using the same embedding scheme embeds a codeword of all zeros into a_3. The two variants of a'_3 form *BF* in a binary form. The detail coefficients d are used to form *SF*. Figure. 3 shows the partitioning of a multimedia file into *BF* and *SF*.

D. Base file distribution

When B_i requests *SP* for a particular content, *SP* provides B_i all the details of M having a requested content. Before the transaction, B_i generates a one-time anonymous key pair $(K^*_{pB_i}, K^*_{sB_i})$ and sends an anonymous certificate request to CA_R. On receiving an anonymous certificate $\text{Cert}_{CA_R}(K^*_{pB_i}, P_{B_i})$ from CA_R, B_i negotiates with M to set-up an agreement (*AGR*) that explicitly states the rights and obligations of both parties and specifies the price and the multimedia content (X). During *AGR* set-up, B_i uses his/her pseudonym P_{B_i} and $\text{Cert}_{CA_R}(K^*_{pB_i}, P_{B_i})$. M verifies the received certificate from CA_R and, on verification, generates a transaction ID (*TID*) for keeping a record of

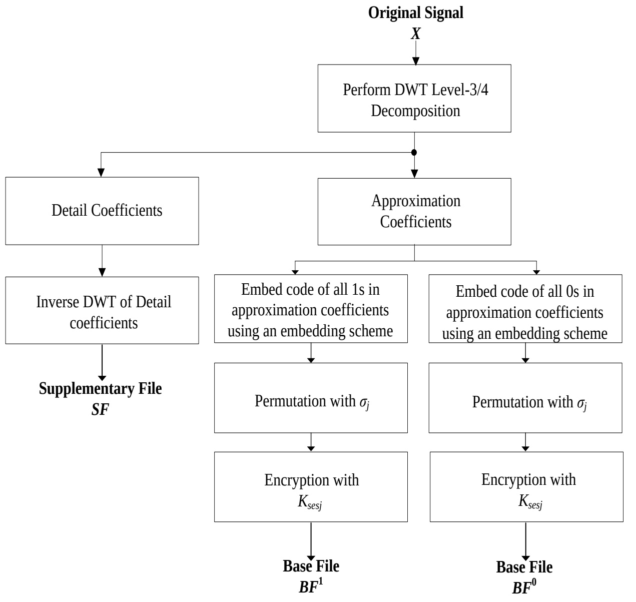

Original Signal
X
Perform DWT Level-3/4 Decomposition
Detail Coefficients
Approximation Coefficients
Inverse DWT of Detail coefficients
Embed code of all 1s in approximation coefficients using an embedding scheme
Embed code of all 0s in approximation coefficients using an embedding scheme
Supplementary File
SF
Permutation with σ_j
Permutation with σ_j
Encryption with K_{sesj}
Encryption with K_{sesj}
Base File
BF^1
Base File
BF^0

Figure. 3: File partitioning

the transaction between him/her and B_i. Then, M sends a request for f_i to *MO* by sending $\text{Cert}_{CA_R}(K^*_{pB_i}, P_{B_i})$, $\text{Cert}_{CA_R}(M)$, *AGR*, P_{B_i} and $\text{Sign}_{K^*_{pB_i}}(AGR)$. *MO* validates the certificates and signatures of M and B_i from CA_R. After verification, *MO* generates a Nuida's c_0-secure codeword f_i of length m and randomly selects n proxy peers Pr_j for the delivery of a fingerprinted *BF* from M to B_i. *MO* then sends a request of permutation keys σ_j to B_i. B_i then generates n random σ_j of length $l = \lfloor m/n \rfloor$. B_i sends $E_{K_{p_{MO}}}(\sigma_j)$ to *MO*. *MO* decrypts $E_{K_{p_{MO}}}(\sigma_j)$ with $K_{s_{MO}}$ and obtains σ_j. *MO* generates n session keys K_{ses_j} and divides f_i into n segments (s_j) of length l and permutes s_j using σ_j in the same order as received by B_i. *MO* then sends $E_{K_{p_M}}(\sigma_j)|E_{K_{p_M}}(K_{ses_j})$ to M. M performs permutation on both pre-computed variants of *BF* with σ_j. It then encrypts the permuted variants of *BF* with K_{ses_j}. *MO* assigns contiguous permuted fingerprint segments to Pr_j, who then contact M in a sequential manner to obtain the fragments of the encrypted and permuted approximation coefficients fa_j. M sends a set of encrypted and permuted fragments of pre-computed coefficients to Pr_j. Pr_j selects the correct pre-computed approximation coefficients from the received coefficients using the assigned permuted fingerprint segments.

E. Supplementary file distribution

Initially, *SP* is fed with *SF* by M. On joining the system, a buyer constructs an onion path with existing peers, which points to it and adds this path to *SP* of its group. By doing so, a content requesting peer R can use this onion path to contact the content-providing peer P while knowing nothing about the provider's identity. The peer requests for a particular file to *SP* of its group. If found, it displays the list of the peers having that particular file; else it sends a request for the file to other connected *SP*s. The other *SP*s, on finding the particular content provider, send the response to the requesting *SP*. *SP* then establishes a path between R and P. After receiving a positive reply from P, R initiates a two-party authenticated key exchange (AKE) protocol to authenticate each other identities and exchange the content of *SF* anonymously. The details of *SF* distribution can be found in [6].

F. File reconstruction protocol

On delivering fa_j to B_i, Pr_j generates a one-time hash of fa_j, encrypts it with the public key of *MO* $(h(fa_j))$ and sends $E_{K_{p_{MO}}}(h(fa_j))$ to *MO*. When B_i receives fa_j from Pr_j, he/she also generates a one-time hash of fa_j, encrypts it with the public key of *MO* $(h(fa_j))$ and sends $E_{K_{p_{MO}}}(h(fa_j))$ to *MO*. *MO* stores $h(fa_j)$ in his/her database against TID that includes date, time, *AGR* and pseudo-identities of B_i and M. On receiving all the fragments of the *BF* from Pr_j, B_i sends a request for the session keys from *MO* by sending him/her a signed digital coin $\text{Sign}_{K_{p_{MO}}}(C_{B_i})$. *MO* charges B_i for *BF* and sends the signed receipt and encrypted session keys $E_{K*_{pB_i}}(K_{ses_j})$ to B_i. *MO* puts C_{B_i} in spent-transaction database, credits M's account and sends the payment confirmation to M. B_i decrypts $E_{K*_{pB_i}}(K_{ses_j})$ with his/her $K^*_{sB_i}$, then decrypts the received fragments of *BF* with K_{ses_j}, and finally applies the inverse σ_j^{-1} on the decrypted fragments of *BF*. B_i recombines all the un-permuted and decrypted fragments to form a single *BF*. B_i receives *SF* in parallel to *BF* through P2P network. Once both files are available at B_i's end, an inverse L-level DWT is performed on the approximation (embedded *BF*) and detail (*SF*) coefficients to form a fingerprinted multimedia content X'.

G. Traitor tracing

Once a pirate copy Y of content X is found, M extracts the pirated codeword pc by decomposing Y with the same wavelet basis used in the fingerprint embedding protocol. This gives the approximation coefficient matrix in which pc is embedded. The watermark detection technique is applied on the approximation coefficient matrix to extract pc. Then M sends pc to *MO*, which performs the tracing algorithm of Nuida's *et al.* codes to identify the colluder(s). The output of this tracing algorithm is the buyer with the highest score. The details of the tracing algorithm can be found in [8].

H. Dispute resolution

The goal of the dispute resolution protocol, performed between M, *MO*, CA_R and J, is to reveal the real identity of the traitor or reject the claims made by M. In order to reveal the real identity of the traitor, *MO* sends (Y, pc, $K_{p_{MO}}(f_i)$) and M sends $\text{Cert}_{CA_R}(K^*_{pB_i}, P_{B_i})$, $\text{Cert}_{K_{pB_i}}(K^*_{pB_i})$, AGR, $K^*_{PB_i}$ and $\text{Sign}_{K^*_{pB_i}}(AGR)$ to J. J verifies the validity of all the certificates and the signatures. If valid, it asks *MO* to decrypt $E_{K_{p_{MO}}}(f_i)$. If pc and f_i match with a high correlation, it requests CA_R to provide the real identity of the buyer. Otherwise, the buyer is proved innocent.

I. Rewards and punishments

In an attempt to induce Pr_j to correctly follow the *BF* distribution protocol, a reputation-based mechanism is introduced for a proxy peer who delivers the content correctly and honestly to the buyer or behaves maliciously and deviates from his/her course of the *BF* distribution protocol. *MO* is responsible for awarding or punishing a proxy peer. The reputation of a proxy peer is calculated using the following data: the collection of feedback about Pr_j from each buyer after reconstruction of his/her multimedia file, the collection of feedback about Pr_j from the merchant after completion of the *BF* distribution protocol, the collection of feedback about Pr_j from other peers selected by *MO* for an anonymous *BF* delivery to a buyer and the evaluation of the transaction history of each proxy peer maintained at *MO*'s end.

Based on above parameters, *MO* calculates a score of each proxy peer over a period of a time, e.g., one month, in terms of positive and negative values. A proxy peer with a positive score is rewarded with a discount coupon for his/her future content purchases, whereas, a proxy peer with a negative score is punished by *MO* in terms of money deduction from his/her account and other penalties (e.g., black listing of proxy peer's pseudo-identity). Thus, in terms of game theory, the dominant strategy solution for each proxy peer is to honestly and correctly follow the *BF* distribution protocol.

5 Security Analysis

In this section, possible security and privacy attacks on the protocols are discussed.

- **Buyer's security:** The possible collusion of Pr_j cannot frame an honest B_i and held him/her responsible for illegal re-distribution due to the fact that Pr_j would need to compute $l!$ combinations each on the colluded fingerprint. Thus, with more m-bits in f_i, Pr_j would need to carry out an increased number of permutations, which would be computationally infeasible. Also, if all Pr_j combine their fa_j, they cannot decrypt these fragments since the fragments can only be decrypted with K_{ses_j}, which are known only to M and *MO* and finally to B_i after making the payment.

  In another scenario, if B_i is unable to obtain all the fragments from M through Pr_j, he/she can request *MO* for digital coin's revocability. Since *MO* keeps the details of all the signed fragments sent by B_i, he/she can accept or deny the request of B_i.
- **Merchant's security:** From the perspective of M, the system is secure because B_i has no idea about the original digital content and the embedded f_i in the purchased copy. Also, B_i cannot claim that Y is created by M since f_i is generated by *MO*, which is trusted by both B_i and M. Also a possible B_i and Pr_j collusion is prevented by assigning the task of selecting Pr_j to *MO* using a reputation-based mechanism. Moreover, a claim made by B_i about receiving invalid fragments from M is repudiated by *MO*. *MO* could deny this claim since he/she stores the hashes of fa_j sent by Pr_j and B_i in the file reconstruction protocol. Thus, in case of a piracy claim made by B_i, *MO* could compare the hashes received from Pr_j with the hashes received from B_i. If the hashes are not equal, *MO* can investigate to determine the cheating party (either Pr_j or B_i).
- **Unlinkability:** Despite the fact that anonymous certificates provide anonymity to B_i, the transactions carried out by the same pseudo ID can be linked to one another. The solution to this problem is to allow a buyer to apply for multiple pseudonyms and anonymous certificates.
- **Coin integrity:** The integrity of C_{B_i} is guaranteed due to the signature of *MO* that generated that coin. Such a signature cannot be computed by anybody else, as the private key of *MO* is never disclosed.
- ***BF* security:** In case a malicious buyer E steals *BF* from another buyer's machine and requests his/her *SP* for *SF* only, this security attack is withstood by our system. After Pr_j deliver fa_j to B_i, both B_i and Pr_j generate a one-time hash of fa_j, encrypt it with $K_{p_{MO}}$ and send $E_{K_{p_{MO}}}(h(fa_j))$ to *MO*. *MO* saves the received $E_{K_{p_{MO}}}(h(fa_j))$ in his/her database along with other transaction details. When E sends a request to *SP* for *SF* only, then *SP* asks E to send the chain of the encrypted hashes of the fragments of the *BF* that he/she had sent to *MO*. In this scenario, E has the *BF* but he/she does not have the chain of the encrypted hashes of the *BF* fragments. In case E generates fake hashes and sends it to *SP*, the *SF* request from E would be denied due to verification of the hashes stored in *MO*'s database.
- **Buyer's privacy:** The attempt of de-anonymization attack by E is withstood by the collusion resistance of the hash function that is used for generation of a pseudo-identity of a buyer. Moreover, E cannot use the pseudo-identity of another buyer because he/she does not know the secret number r shared by the buyer with CA_R. Also, in the *BF* distribution protocol, an attempt by M to find an identity of the buyer by relating proxies to each buyer is withstood by considering a fixed number n of Pr_j for *BF* delivery. Moreover, to ensure anonymous *BF* delivery, *MO* selects random peers and creates an anonymous path in such a way that Pr_j are unable to predict that the next peer in the path is the buyer or some other peer.

6 Comparative Analysis

This section presents a comparative analysis of the proposed system with [3]–[6] in terms of security, privacy and performance. Table I presents the functionality comparison among our proposed system and related P2P content distribution systems.

TABLE I: Comparison of the proposed system with related P2P content distribution systems

Properties	[3]	[4]	[5]	[6]	Our Scheme
Buyer's security	Yes	Yes	Yes	Yes	Yes
Merchant's security	Yes	Yes	Yes	Yes	Yes
Buyer's privacy	Yes	Yes	Yes	Yes	Yes
Traceability	Yes	Yes	Yes	Yes	Yes
Unlinkability	Yes	Yes	Yes	Yes	Yes
Payment mechanism	Yes	Yes	No	No	Yes
Length of anti-collusion codeword	Large	Large	N/A	Small	Small
Computational complexity	Low	Low	High	High	Low

From Table I, it can be seen that the proposed system and the systems in [3]–[6] provide security against customer's rights problem (buyer's security), non-repudiation (merchant's security), piracy tracing, unlinkability and anonymity to a buyer. Our system and the systems in [3], [4] provide an electronic payment protocol between a buyer, a trusted monitor and a merchant in a centralized manner. The systems in [5], [6] do not explicitly consider payment by the buyers to the merchant. While the fingerprinting protocol in our proposed system and the system in [6] are based on Nuida's *et al.* [8] collusion-resistant fingerprinting codewords that result in small length fingerprint codewords, the systems in [3], [4] are implemented with a two-layer anti-collusion code, which results in a longer codeword. Authors in [5] have not considered the collusion resistance of the scheme against collusion attacks. The lower computational complexity of our system and systems in [3], [4] is due to the fact that these systems do not require highly demanding technology (public-key encryption of the content and secure multi-party protocols, among others) unlike the systems in [5], [6]. The proposed system utilizes the idea of permutation and file partitioning to avoid an increased computational costs at the merchant's end, whereas the systems proposed by [3], [4] provide recombined automatic fingerprints, which are generated as contents are downloaded by the buyers from other peers of the system.

7 Conclusions

In this paper, we have proposed a P2P content distribution system, which provides security and privacy to the merchant and the buyer, respectively. The newly proposed scheme can benefit merchants to distribute their contents such as video files, without fear of copyright violation, using the convenience of P2P networks. This scheme reduces the burden of the merchant by only sending a small-sized base file and making use of the P2P network to support the majority of the file transfer process. For distribution of a base file, an asymmetric fingerprinting protocol is performed between the merchant, the proxy peers and the buyer in the presence of a trusted monitor. The buyer's privacy is preserved until he/she is found guilty of illegal re-distribution. The buyer can access the received base file for file reconstruction once he/she makes a payment of the requested content to the monitor. The reputation-based mechanism enables the monitor to select the reputed proxy peers for secure delivery of the fingerprinted content from the merchant to the buyer.

Acknowledgment

This work was partly funded by the Spanish Government through a project TIN2011-27076-C03-02 "CO-PRIVACY".

References

[1] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure Spread Spectrum Watermarking for Multimedia," IEEE Transactions on Image Processing, vol. 6, no. 12, 1997, pp. 1673-1687.

[2] B. Pfitzmann and M. Schunter, "Asymmetric Fingerprinting," Proc. 15th Annual International Conference on Theory and Application of Cryptographic Techniques, EUROCRYPT96, Springer, 1996, pp. 84-95.

[3] D. Megías and J. Domingo-Ferrer, "Privacy-aware Peer-to-Peer Content Distribution using Automatically Recombined Fingerprints," Multimedia Systems, vol. 20, no. 2, 2013, pp. 105-125.

[4] D. Megías, "Improved Privacy-Preserving P2P Multimedia Distribution based on Recombined Fingerprints," IEEE Transactions on Dependable and Secure Computing, vol. PP, no. 99, 2014, pp. 1.

[5] J. Domingo-Ferrer and D. Megías, "Distributed Multicast of Fingerprinted Content Based on a Rational Peer-to-Peer Community," Computer Communications, vol. 36, no. 5, 2013, pp. 542-550.

[6] A. Qureshi, D. Megías, and H. Rifà-Pous, "Framework for Preserving Security and Privacy in Peer-to-Peer Content Distribution Systems," Expert Systems with Applications, vol. 42, 2015, pp. 1391-1408.

[7] B. Chen and G. W. Wornell, "Quantization Index Modulation: A Class of Provably Good Methods for Digital Watermarking and Information Embedding," IEEE Transactions on Information Theory, vol. 47, no. 4, pp. 1423-1443, 2001.

[8] K. Nuida, "Short Collusion-secure Fingerprint Codes against Three Pirates," International Journal of Information Security, vol. 11, 2012, pp. 85-102.

[9] L. Lu *et al.*, "Pseudo Trust: Zero-knowledge Authentication in Anonymous P2Ps," IEEE Transactions on Parallel and Distributed Systems, vol. 19, no. 10, 2007, pp. 1-10.