IP-Based DRM - A Fair and Privacy Preserving DRM Framework

[1]Hung-Min Sun, [2]King-Hang Wang, [3]Yih-Sien Kao
[1,2]*Department of Computer Science, National Tsing Hua University, Taiwan, R.O.C.*
[3]*Industrial Technology Research Institute, Taiwan, R.O.C.*
[1]*hmsun@cs.nthu.edu.tw*,[2]*khwang0@is.cs.nthu.edu.tw*, [3]*yskao@itri.org.tw*

ABSTRACT

DRM systems can be categorized as Identity-based and Device-based DRM systems. Identity-based DRM systems invade user privacy by strongly bundling users' information with the content. Device-based DRM systems authenticate users via the machines they used to play which also violate the privacy principle. No matter how the protocols are designed, these two systems will loss balance between usage fairness and unauthorized usage of copyright contents. In this paper, we propose the concept of IP-Based DRM which authenticates user via IP address. IP-based DRM system has several significant advantages over those existing systems. It not only achieves a better privacy goal, but also provides convenience to the consumers. This system enables a free-trial within a geographic bound, makes licensing for home network easier, and avoids the problems brought by hardware cloning. We also address the problem for those devices are connection limited (needs synchronizing with PC irregularly), the problem of a device having multiple network address, and the problem of roaming service. We will end up with comparing our system to the existing systems with performance and fairness issue.

1: INTRODUCTION

DIGITAL RIGHT MANAGEMENT (DRM) systems [12] can be divided into two categories by the ways of authentication: Device-based DRM and Identity-based DRM systems. Basically, the security of a Device-based DRM system [1][13][15][17] founds on compliant devices. A consumer purchase a digital content from a content server will be given a license. This license will explicitly specify which machine is allowed to play this content.

An Identity-based DRM system [3][7][20] differs from Device-base DRM systems by the license description. When a consumer plays his content, he has to prove his identity to the compliant device. He can either key in password (what he remember), perform a fingerprint scan (who he is), or insert a smart card (what he has).

Currently, most of the existing implementations are of Device-based DRM. Motivated by the insufficiency of Device-based and Identity-based DRM systems, we propose IP-based DRM system. An IP-based DRM system differs from Device-based and IP-based DRM system by the license description, which specifies the IP address of a compliant device. A compliant device will play the digital content when its IP-address matches with the license description.

There are many advantages in IP-based DRM system. We will explain them very detail in the later sections. Briefly, it achieves the followings

1. Better privacy preserving
2. Simple home network DRM
3. Avoid hardware cloning
4. Suitable for portable device
5. No extra authentication device needed

Readers may doubt that what happens if a device having multiple IP addresses, or it changes IP frequently, or it has a private IP (start with 192.x.x.x in IPv4). Also, what if adversary mimics IP addresses? We will also address those in the later section of the paper.

The paper is organized as follows: We will first introduce our key technology - Identity-Based Encryption in section 2. It will be followed by the framework of IP-based DRM in section 3. In section 4 and 5, we will suggest some possible functions and analysis some security issues of the proposed system. We will compare our system with device-based DRM and ID-based DRM systems in section 6. Finally, we draw our conclusion in section 7.

2: IDENTITY-BASED ENCRYPTION

The concept of Identity-Based Encryption (IBE) was proposed by Shamir [19] in 1985. Any entity is allowed to encrypt a message using the recipient's identity as the public key. In this setting, senders do not need to know receivers' public key, but their name or index is enough. The private key of the corresponding entity will be generated by a trusted third party called a Private Key Generator (PKG). The PKG has a master key which can derive the private key of every entity. The PKG is trusted to give the private key to the valid entity but not the others. A detail survey can be found in [5].

Using IP address as public key is not new. Aura has proposed the concept cryptographically generated address (CGA) [2]. His solution does not need any

infrastructure to manage with. In our application, a managed infrastructure IP address IBE which support broadcast encryption is more suitable.

3: IP-BASED DRM SYSTEM

Briefly, an IP-based DRM system authenticates user through IP address. We assume each IP is owned by a single machine (although this assumption is not true in real life, we will resolve that in section 5), this machine processes a private key of an identity-based encryption (IBE) scheme with the identity be the encoding of IP address. Licenses retrieved from the license server are encrypted using IBE with the encoding of the users' IP address as public key. An authorized machine is able to decrypt the encrypted license to play the media content.

3.1: ARCHITECTURE

Take a closer look to the basic IP-based DRM system architecture, there are four roles in total: a *Content Server*, a *License Server*, a *Public Key Generator* (PKG), and a set of *Users*. Depends on the implementations, it is also possible to split the content server as *Content Provider* and *Content Distributor* to separate the jobs of creating content and distributing content, or to introduce *Usage Clearing House* [7] to collect information about the usage of content, or adding a *Roaming Server* which described in later section, but the IP-based system structure is merely the same.

A *content server* creates and distributes content to users as usual. That piece of content is packaged and encrypted using content key.

A *license server* is responsible to issue licenses to the users who purchase content. On the license, there are a content key to decrypt the content. To avoid unauthorized distribution of content, this content key is further encrypted by another key. Here, we use the IP address of the user machine as the identity in IBE to encrypt the content key. The license server also needs to handle payment and license delivery.

A user who employs this system needs to register to PKG. This registration will not record any user information except for his IP address. A secret key corresponding to the user's IP address will be delivered to the user via secure channel. This secret key is stored in the compliant machine and is not extractable. Then the user can download content from the content server and purchase license from the license server.

The PKG is a trusted organization who will only generate one secret key per IP address in a fixed period. For several reasons (e.g. owners of IP addresses changes, revocation of incompliant devices, etc), the secret key will be updated regularly. The updated secret key will be delivered to the corresponding users. It is possible to merge the role PKG into license server. It is also possible that we do not use PKG and IBE to encrypt the content key. But we clarify that no matter how the system being implemented, it should not allow the license server or the content server to gain knowledge about his client.

3.2: A SIMPLE IP-BASED DRM SYSTEM

To better illustrate the idea of IP-based DRM, we propose a simple system here. Since this system is for explaining idea, it will not capable to cover all the aspect discussed in DRM, especially in security.

After a creation a piece of content M, the content server packages the digital content and encrypts it using a content key K. Then it sends this content key to the license server L via secure channel. The content server will be responsible for promoting and distributing this content as well.

A user U who processes the IP address A wants to buy the content M will first download the content from the content server. The metafile of the content M will direct him to the particular license server L. The user will be promoted to register in the corresponding *PKG*.

PKG registration

1) User U (here we refer U be the compliant DRM software but not the user himself, thus things are done automatically and trustworthy) retrieves the public key P of *PKG* from the web or license server. He picks a long enough symmetric random key r and calculates the following terms

U -> *PKG*: $E_P(r \mid A)$

which represents the public key encryption of r concatenated by the IP address A with the key P. We assume this public key encryption is secure, say indistinguishable against chosen ciphertext adversary.

2) On receiving this message, *PKG* will check if this IP address is registered before. If this IP address has not yet been registered, or the previous owner of this IP address has changed his IP and abandons this IP already, it will encode A with a collision free function and takes this encoding as identity to generate a random number K

PKG -> U: K

and sends this back to U through this IP address in a particular port. This port will be specified by all the operating system that can only be listened by compliant DRM software.

3) U is required to encrypt the K to prove he owns the IP.

U -> *PKG*: $E_P(r \mid A \mid K)$

4) Then *PKG* generates a private key R respects to the IP address A and replies him the encryption cipher of R with key r.

PKG -> U: $r(R)$

During the above protocol, the *PKG* will release a private key to the holder of that IP address. The IP holder can check the received private key is correct respect to that IP. In the appendix, we will show that if the private is correctly received by the user, then the probability for any adversary to learn information from

the IP address is negligible.

Now, a customer *U'* who may be the user *U* himself, or someone else like to paid for this content will purchase this content. He connects to the license server web page, key in all the related information for payment and the IP address *A* of the user (This can also be done via anonymous channel for further privacy protection for the customer *U*', but it is out of our scope). If this transaction is valid, the license server will add a record in its database <*A*, *M*> to indicate user from IP address *A* has the right of use of content *M*.

When the user *U* plays the content *M* in his compliant machine, he will connect to the license server to retrieve the license.

License Retrieval

1) User *U* sends the license server *L* a request message in the follow format

U -> PKG: $E_L(M \mid A)$

which indicates the encryption of *M* concatenated by *A* with the license server *L*'s public key.

2) *L* looks up its database and retrieves the record <*A*, *M*>. Then it will issue a license specifying the rights of use *W* and the content key *K*. The content key *K* will be further encrypted using IBE with the encoding of *A*. To ensure the integrity of the license, the license server will sign on the license too. In addition, to avoid third person observing the license, the license will be encrypted. Overall, the license is in the following format.

PKG -> U: $E_A((W, E_A(K), Sign_L(W, E_A(K))))$

This license will be delivered to the user *U* through the IP address *A*.

3) After receiving the license, the compliant device will check if the license is received from the network adaptor having the IP address *A*. This avoids the user filling in invalid IP address in some unused network adaptors. This process is optional, and even is not feasible for some applications.

Since the IP address does not permanently stick to a device, *PKG* will update the private key periodically. This update can be weekly based or daily based, depends on the scales of the network. On each update, users are required to retrieve the new private key from *PKG* using the registration protocol described above.

3.3: DISCUSSION

The greatest advantage brought by IP-based DRM architecture is privacy preservation. There is only one piece of information is known to the license server and the content server about the user – his IP address. This piece information will also be known in Device-based and Identity-based DRM. However, the other two DRM architectures require some additional information which is sensitive to the user himself.

Secondly, if the device is upgraded, stolen, sold, or damaged, user can easily replace the device with a new compliant device and register the IP with the *PKG*. It is also secure against hardware cloning attack. These are two significant advantages over Device-based DRM.

Thirdly, unlike Identity-based DRM, authentication in IP-based DRM relies on IP address only. It does not require addition device to authenticate the user. After IPv6 is employed, there should be sufficient for each electronic device on earth to have an IP address. For connection limited devices (say portable MP3 player) which update content by synchronizing with PC will also enjoy this advantage. These devices can save the licenses in its database and record the last IP address it used for synchronization.

As the uses of IP address are complicated, some points worth to be discussed:

1. A machine may have several IP addresses

It is not a problem if all of its IP addresses are able to connect to the internet. For which IP address is registered in license server, the media player chooses that network adaptor to connect the license server, in which there is not a problem. But for some reasons, which IP address cannot connect to the internet, we can still retrieve the license from the license server if the PKG have not released a new private key. However, after PKG renewing the private key for the IP address, users are not able to listen to the music until that IP address resumes connectivity. A permanent change of IP address is referred to the third point in below.

2. A machine that have limited internet connection

Although IP address is rich in IPv6, it does not mean every device has immediate internet connection. In Java micro edition, it defines a class of portable device call *Connected Limited Device Configuration (CLDC)* [8]. This class of devices has limited connection from the internet, may be once a day or a week. It is not possible for those devices to retrieve license for each time it plays media content. It is not the only problem in IP-based DRM, it also happens in Device-based DRM and Identity-based DRM. The real problem in IP-based DRM is when this device is synchronized with another machine and proceeding with another IP address. Shall we let this device having multiple IBE private key as it held different IP addresses? Or shall we only keep the most recent IBE private key?

Here, we have no biases in choosing either way. The former one would incur unauthorized redistribution of digital content, while the later may not be fair enough. Similar problem does appear in Device-based DRM too.

There is an alternating framework to solve this problem. We may allow merge the IP-based DRM to Device-based DRM as a hybrid type DRM. In this framework, a device-based DRM gateway is installed in a household. This gateway will act like a PKG in a home network. It is a variation from the work of [16] while inducing the IP-based DRM concept in it. Owing to limited space, we would not discuss it in detail here.

3. Infrequent changes of IP address.

Suppose you are moving out from a household and change your ISP. It is very likely that all of your electronic device will be given another IP addresses. In this case, all the digital content bought will not be able to play anymore. In this case, you have to inform the

license server to perform an IP updating procedure. During the update procedure, the user must provide the knowledge of the IBE secret key. We may apply Identity-based Identification schemes [10][11] to achieve this goal.

We must notice that a too frequent IP update may imply an unauthorized content redistribution. Also, if a user IP addresses is redistributed without prior notification, or a hacker is successful to spoof a valid IP address to obtain an IBE private key from the PKG, a legitimate user's contents would be "transferred" out of his account by IP updating mechanism. Hence, this procedure should be done carefully. May be during the transition period, say three days, we allow the old IP-address and the new-address to play the content simultaneously while warning the old IP-address owner that the IP address is going to be updated.

We also notice that IP-address updating policy allows peer-to-peer content transfer. A legitimated user can sell his content to other peer by filling in the buyer IP-address. In [21], the authors proposed a complex content transfer protocol to achieve this goal. However, it is very easy to achieve process transfer of content in this system.

4. A portable machine change IP addresses time to time

A portable machine, like a laptop, can have different IP addresses from time to time. Unlike the case of CLDC, this portable machine has network ready. It should be able to purchase new album from the net anytime, anywhere. As we have discussed in the third point above, it is not suitable to update a user IP address in the license server very frequently. Instead, we introduce a new role in the architecture, called *Roaming Server*.

When the customer purchases a digital content, he checks the option that user's machine will change IP address frequently. Then this portable machine will have software installed inside to regularly report its IP address to the roaming server. The Roaming Server acts like a DNS server which translate a domain name to an IP address. Then the license server will encrypt the content key using the domain name of the portable machine instead of the IP address. The private key of IBE will also generate based on the domain name, not the IP address.

The concept of roaming server is not new. Some companies [14] have launched this service to customers who do not process a fixed IP address. We extend this idea in IP-based DRM.

We admit that this is not a good option in IP-based DRM system. Since those machines need to install a software on it. Not every portable device, for example, cell phone, can install software on it. Secondly, to avoid adversary stealing your domain name, this machine must have shared a secret key with the roaming server. However, this violates the philosophy of IP-based DRM in which has absolute no share with servers.

5. A private network with private IP only

For security reason, some networks have only private IP for machines behind the firewall. Apparently, if we allow these machines to use the gateway IP address as their IP address, then only one compliant device behind this gateway can enjoy the service. Since there is only one machine will receive private key from the PKG in a fixed period and this private key cannot be transferred out. Besides, by using roaming server mentioned above also cannot solve this problem.

A possible solution for this network is to further specify a port along with the IP address. Suppose each machine behind this gateway has the right to preserve a fixed port, he can connect to the PKG with specifying his gateway IP and the fixed port to generate an IBE private key. Then the license server issues license using the gateway IP along with the port.

The security this system is depends on how the gateway manager allocates those ports. If those ports are not permanently allocated for users, this scheme would not work.

4: POSSIBLE FUNCTIONS

To highlight the advantage of IP-based DRM, in this section we introduce some possible functions that are simpler to be implemented in IP-based DRM.

4.1: HOME NETWORK DRM

It is ironic to have customers to purchase a same piece of content several times for playing it on different machines. However, without suitable restrictions on transferring mechanism, it will bring economic damage to content provider from illegally redistribution of content. DRM on home networks was addressed in [9][16] which inherited the idea from [6][8][18]. They proposed the concept "Authorized Domain" in which compliant devices placed within a same household are grouped to form an authorized domain. Within this domain, content can be transferred freely with the only restriction that not crosses over the domain border.

In [16], the authors solve the problem by setting up a Domain Manager which is responsible for checking device compliance, distributing key, revoking incompliance device, etc.

But in IP-based DRM architecture, we can rewrite the license and make thing become very simple. First, we assume that within a home network, devices have adjacent IP addresses. This assumption is strong but reasonable. Now, to issue a license for a home network, we simple write the license with an IP address in home network with subnet mask. Then, with subnet mask, the compliant device will know if it is within the authorized range. The content key is a little tricky. Originally, we would encrypt the content key using IBE with the IP address specified. Now we have a group of IP address, if we encrypt the content key with a particular device's IP address, we need to re-encrypt the content from that device and redistribute the key. It is a possible solution,

but we would do it in a more advance way.

IBE can be used in broadcast encryption. Let's us put the home network devices as a group. We use broadcast encryption to encrypt the content key with the group public key. Since the devices employ adjacent IP address, it is very suitable to apply hierarchical scheme like hierarchical IBE (HIBE). IP addresses are already in hierarchical form which is very suitable to apply HIBE and broadcast encryption. Broadcast encryption is already well developed in the literature. Many works support adding new members and removing old members. It seems that HIBE broadcast encryption fulfills our needs in home network DRM.

4.2: GEOGRAPHICAL LICENSE

Imagine this scenario. Suppose you are taking a distant train, or plane travel. You may like to download and watch a movie with your portable MPEG 4 player. For license problem, you are not allowed to play the movie you get off from the transport. It is not suitable to deploy Identity-based DRM and Device-based DRM without pre-registering. Let's see why it is not suitable. If the transport connects to the internet, any users left the transportation may require license update or download another new movie. It requires an additional pre-register mechanism to determine an user is on the transportation or not. Also, logging user information without properly handle may raise privacy problem.

Using IP-based DRM is very easy. After obtaining an IP address from the transport (surely it does give you an IP for downloading the media), you can immediate register to the mini-PKG on the transport after you obtain a private IP address. Then, you may watch the movie on the transport. After you get off from the transport, you will be asked to return your IP address, the license will therefore be disabled. Neither license update nor newly download will be liable for users who get off from the transport.

Such a function provides a license with geographical usage which is not achievable in Identity-based DRM and Device-based DRM. We call this feature as a *Geographic License*. A geographic license can also be used in concert and stadium (close up shot of the performer), album shop (free trial on some pop song), book store (free trial of books).

A geographical license is a closed network, therefore the PKG (or mini-PKG) is a micro version of PKG. Its master key is independent from the other PKG or mini-PKG. Therefore, if someone migrates from a sub-network to another sub-network and accidentally uses the same private IP address, he will not be able to decrypt the previous content using the new IBE private key, even if he is able to backup the old license and encrypted content. Since the key from PKG is not the same.

This idea can further extend to ad hoc network. Suppose three friends are gathering with their wireless network enabled PDA. They can share digital content using geographical license. The content owner can be the master in this ad hoc pico network. He distributes IP addresses as a DHCP server and then setting a pico-PKG and distributes the content. Of course, after they finish this gathering, the license will also expire as well.

5: SECURITY ANALYSIS

Authenticating users with IP addresses may look not sound to some readers. We must clarify that the license delivery from license server to users is secure. Since it employs IBE as a cryptographic primitive. The only risky part is when users apply for their IBE private key from the PKG. In this section, we give a detail security analysis on the risk of IP spoofing or related problems and their corresponding countermoves.

5.1: IP SPOOFING

IP packet has a fixed format. Everyone can write a fake IP packet based on the format, including filling up a fake source IP address. It is the gateway responsibility to check the source IP belongs to its own subnet. However, not every gateway is responsible to do this, especially for those controlled by malicious user. A real challenge of IP spoofing is how the attacker can receive the packet from the PKG. A tough hacker can control some routers near PKG to eavesdrop those packets. Some secure options like SSL or IPSec can be included to tackle the problem. However, there must exist some security holes for hacker to spoofing an IP address and receive packet from the PKG no matter what we do. This is owing to the proof of IP address ownership is not strong. No one can proof the ownership of an IP address using cryptographic techniques (an identification scheme).

Fortunately, the PKG will only release a private key per time period. If someone signs up with the PKG already, even the hacker can spoof the IP address, he cannot retrieve the IBE private key from the PKG. In the next update phase, the IP owner will be request to show his knowledge of the IBE secret key. Otherwise, his request will be pended until it reaches timeout. This situation implies a change of ownership of IP address. It may happen to a user upgrade his machine or the IP address is reallocated. If an IP address is re-registered by new entity, the owner of old IP address provides his proof on knowledge of the private key, then we will inform the manager of the PKG and acquire for human intervention.

Therefore, even hacker spoof an IP address, he will be fail when attempting to obtain the private key from the PKG. With negligible chance if he obtain the key from PKG during the new key update period and pass the pending timeout, a legitimate user can also report to the PKG manager about his IP address being stolen.

A possible solution to tackle IP spoofing is to apply the similar concept of Domain Key [4]. Domain Key is proposed to combat spam mail by Yahoo. A domain

owner generates a public key pair and registers its public key in DNS. Every mail sent from this domain will be signed with the domain key. Applying the Domain Key concept, every application to PKG will be signed with the domain key. This reduces the chance for attacker to spoof IP address.

5.2: DENIAL OF SERVICE

A consumer will confirm his IP is registered in PKG and receive the private key from it before purchasing content from license server. However, if his IP is pre-registered by somebody already, he will not be able to enjoy the service.

A powerful adversary may register many IP addresses in PKG and process a Denial of Service (DoS) attack. Unlike DoS in web service, an IP addressed is registered if the applicant replies a confirmation message. If the adversary spoofs an IP address by simply mimicking the IP header, the registration will be halt on the third flow and the PKG will not generate key for this IP address. Therefore, to make a DoS possible, the adversary should have the power to receive message for significantly many IP addresses. This is more likely to be happened that the adversary mimicking IP addresses within a same domain. Then it is the responsibility to find the adversary.

Basically, DoS problem is a hot topic in Intrusion Detection System (IDS). Since the application is restricted to key generation, we address this problem by introducing domain key at the domain side. At the PKG side, we can setup an IDS to filter possible attacks.

5.3: REVOCATION OF SYSTEM CRACKER

DRM is founded on compliant devices. If hacker does reverse engineering to the system and retrieves the key from the device, he is able to redistribute the content. A compromised device should be black listed from the license server. In Device-based DRM, every device has been registered in the license server database. If a device appears to be malicious, the license server will revoke this device and black list it from further license update and content purchase.

In IP-based DRM, there are two options in revoking an IP address: black listing an IP address in 1) PKG, or 2) license server. We prefer using the later method. It is because PKG is not only used in IP-based DRM, it may further used in secure email service, or instant message service. Misconduct in DRM should be punished in DRM only. Another reason is the hacker would request the license server to change his IP address after he found himself being blacklisted. In this case, revoking devices in license server has a better management.

How to trace a malicious user is a hot topic in cryptography and DRM. Usually we refer this to *Traitor Tracing*. We refer interested reader to [22].

6: COMPARISON

In this section, we compare the architectures of ID-based DRM, Device-based DRM, IP-based DRM, and H-DRM in several different aspects. The comparison is somehow subjective owing to the reason that architectures are difficult to be compared. We illustrate the comparisons in Table 1.

TABLE I - COMPARISON BETWEEN DIFFERENT ARCHITECTURE

	ID-based	Device -based	IP-based
Portability	Good	Poor	Very Good
Privacy	Good	Poor	Very Good
Security	Good	Very Good	Good
Flexibility	Medium	Poor	Very Good
Extra Device	Yes	No	No
Hardware Cloning	Partially Secure	Insecure	Secure
Home Network	Not suitable	suitable	Very Suitable
Geographical License	Not suitable	Not suitable	Very Suitable
Fairness	Medium	Poor	Very Good

The above table is self-describable except the item fairness. Fairness means how the user right is achieved in the system. For the same architecture, we can have different choices of implementations. Some implementations compromise the security level to allow more user rights. So, we compare different architectures with their implementations with the same level of security assumptions. For example, we may assume the compliant devices can be compromised, or the legitimate users may try redistributing the content and the licenses, etc. As far as we know, Device-based systems generally impulse more constraints on the usage of content. While IP-based is ready to implement many user friendly functions like geographic license, share among friends gathering. We comment IP-based DRM is fairer than the other implementations. For the performance and cost effectiveness are heavily depends on implementation; thus are not discussed here. But we believe effective IP-based implementations do not introduce too many computation or transmission overheads. Regularly updating IP private key may bring some overhead, but efficient key update in broadcast encryption can lower the complexity.

7: CONCLUSION

In this paper, we proposed the IP-based DRM which is a new DRM architecture authenticating user by IP address. IP-based DRM system has a better privacy and is more users friendly. Some special functions like home network and geographical license can be easily implemented using IP-based DRM architecture. We suggest that IP-based DRM system can be implemented

using IBE encryption which can preserve user privacy efficiently. One may also consider combining Device-based DRM as a hybrid mode of DRM system to handle mobile devices and connection limited devices. We also suggest a few security threats that may be faced in IP-based DRM and provide some possible solutions.

ACKNOWLEDGEMENT

This work was supported by Industrial Technology Research Institute (V0-95003).

REFERENCES

[1] Apple iTunes, Available: http://www.apple.com.tw/itunes/

[2] T. Aura, "RFC3972: Cryptographically Generated Address (CGA)," Available: http://rfc3972.x42.com/-

[3] C. Conrado, F. Kamperman, C.J. Schrijen, and W. Jonker, "Privacy in an Identity-based DRM System," in IEEE Proceedings of the 14th International Workshop on Database and Expert Systems Applications (DEXA'03), Prague, pp. 389-395, September 2003.

[4] DomainKeys, Yahoo Anti-Spam Resource Center. Available: http://antispam.yahoo.com/domainkeys

[5] R. Dutta, R. Barua, and Palash Sarkar, "Pairing-Based Cryptographic Protocols: A Survey," Cryptology ePrint Archive, Report 2004/064, 2004.

[6] A. Eskicioglu and E. Delp, "An overview of multimedia content protection in consumer electronic devices," *Signal Processing: Image Communication*, 16(5), pp. 681-699, 2001

[7] J. Feigenbaum, M. Freedman, T. Sander, A. Shostack, "Privacy Engineering for Digital Rights Management Systems," *Proceedings of the 2001 ACM Workshop on Security and Privacy in Digital Rights Management*, LNCS, Vol. 2320, Springer, pp. 76-105, 2002.

[8] JSR-000139 Connected Limited Device Configuration 1.1 (Final Release), Java, Sun. Available: http://jcp.org/aboutJava/communityprocess/final/jsr139/index.html

[9] F. Kamperman, S. van den Heuvel, and M. Verberkt, "Digital Rights Management in Home Networks, " In *Proc. IBC 2001,* pp. 70-77, 2001.

[10] M. Kim and K. Kim, "A new identification scheme based on the Gap Diffie-Hellman problem," *SCIS* 2002.

[11] M. Kim and K. Kim, "A new identification scheme based on the Bilinear Diffie-Hellman problem," *ACISP* 2002.

[12] W. Ku and C.H. Chi, "Survey on the Technological Aspects of Digital Rights Management," In *Proceeding of the 7th Information Security Conference*, LNCS 3225, pp. 391–403, 2004

[13] Microsoft DRM, Available: http://www.microsoft.com/windows/windowsmedia/drm/default.aspx

[14] No-IP.com. Available: www.no-ip.com

[15] OMA DRM. Available: http://www.openmobilealliance.org/

[16] B. Popescu, B. Crispo, A. Tanenbaum and F. Kamperman, "A DRM security architecture for home network," *ACM DRM 04*, pp. 1-10, 2004

[17] RealNetworks DRM solution. Available: http://www.realnetworks.com/products/drm/index.html

[18] M. Ripley, C. Traw, S. Balogh, and M.Reed, "Content protection in the digital home," *Intel Technology Journal*, 6(9), pp 49-56, 2002

[19] A. Shamir, "Identity-based cryptosystems and signature schemes," In *Proceedings of CRYPTO 84 on Advances in cryptology*, pages 47-53, Springer-Verlag New York, Inc., 1985

[20] H.M. Sun, C.F. Hung, and B.H. Ku, "An Improved Identity-Based DRM System," Information Security Conference 2005, June 9-10, 2005

[21] H.M. Sun, K.C. Yang, and K.H. Wang, "Transferable DRM System for Two New Business Models," Information Security Conference 2005, June 9-10, 2005.

[22] V.D. To, R. Safavi-Naini, and F. Zhang, "New Traitor Tracing Schemes Using Bilinear Map," *ACM DRM 2003,* pp. 67-76, 2003

Appendix

Proof of the *PKG* registration protocol. In this section, we try to prove that in the *PKG* registration phase is secure in the sense – a successful key registration will guarantee that only the user will know the private key. Therefore, if the license key is encrypted with the IBE using IP as the identity, no adversary should be able to learn the license key. We formulate the security notion as below.

The adversary are able to execute the commands **send**, **test**, and **validate**. The command **send(*U, m, A*)** allow the adversary to send a message to entity *U* with message *m* from an IP address *A*. The output of this command is the message that the entity *U* would generate upon receipt of message *m* from an IP address *A*. The command **test(*A*)** can only be issued to an entity who processes IP address *A* and accepts the received private key *R*. The output of this command depends on the result of a private random toss, either hash value of the private key *R* or the a random key of the same size. The command **validate(*R, A*)** is available for the adversary and a real world user. The output of this command is a Boolean value; it is true if and only if *R* is the private key of the IP address *A*.

Security: We define the event S if adversary is able to guess the output of **test(A)** is H(*R*), the hash value of a private key respect to *A* or a random key. We say the protocol is secure for any adversary and for some fixed *c*, there always has a *k* such that if $\Pr(S) < 1/2 + 1/c^k$.

Assumption 1: Without the oracle **validate**, adversary should have no knowledge that any pair *R* and *A* is a valid public key and private key pair. Formally, given a random pair *A* and *R*, which has a probability of 1/2 being a pair of valid public key and private key pair, anyone can validates the value *A* and *R* by the oracle **validate(*R, A*)**. For any adversary, if *A* and *R* are not queried before, the probability of adversary correctly validate the pair is bounded by $1/2 + 1/c^k$ for some fixed *c* and some *k*.

This assumption implies that no one is able to validate H(*R*) and *A* that *R* and *A* are indeed a public key and private key pair. In our security notion, we cannot afford giving the private key *R* directly in the **test** command. It

is because the adversary can validate the private key R by the command **validate**.

Assumption 2: Given a public key A, only PKG can generate the corresponding R. Formally, if the private key space is Z_R, any adversary function Q takes input A and outputs a possible key R'. Pr(**validate**(R', A) = true | $Q(A) = R$') < $1 / |Z_R| + 1/c^k$ for some fixed c and some k.

Theorem: Our *PKG* registration protocol is secure

Proof: A user processes the IP address A will accept the message R only if (A, R) passes the validate function. We define the event P_1 be the event that a random R' passes the validate function. Assume the public key space is Z_A, where $|Z_R| \geq |Z_A|$. This happens with probability at most $1/|Z_A|$. Let P_2 be the event that a random message r'(R') which decrypted with r would output R. Let the length of the ciphertext $r(R)$ be t_1. P_2 is bounded by probability $1/2^{t_1}$.

Now, we simulate the protocol with an adversary. We override the both symmetric and asymmetric encryption and decryption by encryption list and decryption list. Instead of honestly running the protocol, we choose a random number for the first and third flows of the protocol. The value r will also be randomly drawn. The fourth round message will be also replaced with random number. These all random number will be stored in a database. If the adversary attempts to access the decrypt oracle to try to decrypt the first or third round messages, he will success if he can correctly guess the *PKG* private key with probability $1/|Z_R|$. If the adversary tries to decrypt the symmetric encryption in the fourth round of message, he will success if he can correctly guess the value r with $1/2^{t_1}$. Now, all the message is irrelevant to the value R, except the IP address A.

If the adversary issues the test command, we flip a coin b. If b is head, we return the hash of the decryption of fourth round message; otherwise, we return a random string. We notice that if the adversary did not query the decryption of the fourth round message with key r, neither the encryption from some message with key r that output the fourth round message, the hash of the decryption of fourth round message is also a random number. Unless the adversary has queried the value R to the hash function, otherwise he cannot distinguish the hashed value from a random string; it is because all the message is irrelevant to the value R. Therefore, by assumption 1 and 2, the probability of him to guess the coin correctly is bounded by $1/2 + 1 / |Z_R| + 1/c^k$, which is negligibly larger than 1/2.

Let Q_s be the total number that adversary queries on symmetric encryption/decryption. Let Q_a be the total number that adversary queries on asymmetric encryption/decryption. The transformation above is distinguishable for the adversary bounded by the following terms.

$$Q_s / 2^{t_1} + Q_a / |Z_R| + 1/2^{t_1} + 1/|Z_A|$$

Where the above term will diminish exponentially as the growth the security parameter. Therefore, our protocol is secure.