ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 5, Issue 5, May 2016

An approach towards digital rights management system using blind decryption algorithm

Miss. Ankita A. Deshmukh, Prof. V.B.Gadicha

Abstract- **Cloud computing is an emerging technology where a huge amount of copyrighted data is stored from different vendors and organizations. The digital content can be easily copied and distributed over the internet. Most of the current digital rights management systems rely on trusted third parties to provide privacy to the entities involved. However, a trusted third party can become malicious and break the privacy protection of the entities in the system. The current system focusing on these problems and using an enhanced scheme enabled with Digital rights management to prevent illegal distribution of the content. It uses simple primitives such as blind decryption and one way hash chain to avoid the trusted third party assumption. The proposed scheme also preserves the privacy of the users without relying on any third party for mutual trust.**

***Indexterms*-cloudcomputing,binddecryptiom,digitalrightsmanagement,privacy**

I INTRODUCTION

Digital Right Management (DRM) is a system for protecting the copyrights of data circulated via the Internet or other digital media by enabling secure distribution and/or disabling illegal distribution of the data. For achieving security and for managing the digital rights some researchers have used a TTP but still it is not much secure because a TTP can also be malicious. This emerges the need to preserves the digital rights of the data with secure and efficient schemes without relying on a TTP for future Cloud Systems.[1]In DRM environment, only legitimate users are allow to access and use the content technology .The digital right management not only protect the piracy but it also protects the interest of copy-rights related parties. In essence, the intellectual property right is exercised by permitting an authorized use of the copyrighted work. The innovation of internet and digital contents has provided a platform as market place to trade various copyrighted works. Illegal copies bring monetary damages to the copyrighted holders, thus the copyright holders started to use the digital rights management as the defense to the piracy or illegal use to the data.[2]

The growth of Internet has made it easy for replicating, and distributing digital contents without any loss of quality to the contents. This has resulted in widespread illegal copyright violations of digital contents. Hence, digital rights management (DRM) technologies have been developed to protect the intellectual property rights of the entities involved. Although, advances in DRM technologies have controlled the copyright violations of digital contents, it has resulted in the violations of privacy of the entities involved [7], [9]. In DRM systems, the distributors and users must be accountable for any misuse of their purchased contents/licenses. To achieve the accountability of users, content providers perform usage tracking and monitoring via license acquisition transactions and user authentication mechanisms. However, accountability affects the user's privacy as it reveals the link between users and their usage patterns. User data gathered in this process can be later used to generate detailed profiles of the users and their activities. The resulting profiles of the entities can be misused by the content providers.Privacy preserving mechanisms for DRM have been proposed by several authors [12], [13]. However, simultaneous consideration of accountability and privacy has not been addressed well yet. Some schemes that take care of the accountability and privacy need the user to trust a third party. Whereas, other schemes which use complex cryptographic mechanisms to avoid trusted third parties fail to satisfy many of the desirable properties of DRM.Trusted third parties (TTP) are undesirable in DRM because users can never be assured that their privacy will be secured by the these entities.

II LITERATURE SURVEY

In Privacy in an Identity Based DRM System paper, author C.Conrado propose a DRM system using Identity Based Encryption System (IBES) that overcomes the deficiencies and vulnerabilities of the existing DRM systems. The proposed DRM system is an approach towards an open framework, wherein, the content processing application, can be independent of the DRM provider, and the security is controlled with the help of the smart cards.[13]

In identity Based Encryption system, Identity based Encryption System (IBES) is a public key cryptosystem designed mainly to remove the redundant complexity involved in the Public key Infrastructure's certification. and certificate verification process. In this system, a recipients well known unique identity ID, like email address, mobile phone number, IP address, URL, etc is used as the public key for the encryption. The system architecture ensures that only the owner of this particular unique Identity has the private key for this ID, and hence none others can decrypt it. This is ensured by a Private Key Generator (PKG), the trust component of the system. It depicts the system architecture and the various steps for secure message transfer. The concept of IBES was proposed by Shamir way back in 1984. But the first successful and computationally feasible system was published in 2001 by Dan Boneh and M Franklin [9]. Their system makes use of a concept called Weil Pairings, a bilinear function. But the computationally feasible system for mobile phones was first demonstrated in 2006, by J. S. Hwu, R. J. Chen, and Y.B. Lin with their Fast computation method for Weil Pairings [10]. With this advancement, IBES services can now be ported onto handheld devices also, which are prime content usage devices today. The ID-

based scheme consists of four algorithms which includes: Setup, Extraction, Encryption, and Decryption. Setup is run by the PKG to generate a master key and the system parameters. This is done on input of a security parameter kID, which specifies the bit length of the group order and is regarded as the key size of the ID-based scheme. The Extraction algorithm is carried out by the PKG to generate a private key corresponding to the identity of a user. As with regular public key cryptography, the Encryption algorithm takes a message and a public key as inputs to produce a cipher text. Similarly, the Decryption algorithm is executed by the owner of the corresponding private key to decrypt the cipher text.[6].

In LMSAT paper, the author jiangzhang propose a license management scheme named LMSAT (License Management Scheme with Anonymous Trust) which provides a more powerful and flexible license acquisition and usage tracking scheme to allow the user access the contents anytime, anywhere, and on any compliant devices anonymously. The user buys a token with a secret Anonymity ID (Anonymity ID) and corresponding password on it anonymously from the provider through a mechanism such as the pre-payment scheme in advance, and then the user can request a license bound to the Anonymity ID. The Anonymity ID is a string of random binary digit which represents an anonymous account. When the user requests the license of the digital contents, he can input the Anonymity ID and corresponding password according to the requirements of DRM system, and then the DRM system will charge the anonymous account for the corresponding contents.[16]

In "A Ticket based digital rights management model" paper the term Digital Rights Management (DRM) generally refers to a set of policies and techniques that guide the proper use of digital content. Due to the nature of digital content, it is easily modified and distributed. DRM is already progressive in many industries and has received a fair amount of attention in recent years. To prevent multimedia content from being usurped, there is much research which focuses on how to build a reliable DRM system. In this paper, Ming-Kung Sun proposes a DRM model to provide improvements for several existing research issues. We use a "ticket" to achieve anonymous consumption and design a protocol to protect against malicious servers. A solution is proposed which addresses these issues and brings more functionality to users.[12]

A. Anonymous consumption,In this scheme, Ming-Kung provides a method to achieve anonymity. After the process of our protocol, the user gets a ticket. The user can use this ticket to obtain content license. Based on Maitland et al.'s restrictive partially blind signature [8], we provide the user with anonymous consumption which is impossible when using e-currency. Even though the server has all the information in the proposed protocol, it is not able to discover who the original buyer of the ticket is. This also means the user can get a license without revealing any personal information.

B. Protection against malicious servers

To the best of our knowledge, current DRM systems focus on preventing the illegitimate use of their content. However, there exists another issue which is also important to real life commercial DRM systems. In the case where the various servers are controlled by different entities, it is possible the content server or the license server may be malicious. In traditional DRM schemes, any malicious server has access to the entire content key and this could lead to abuse of the commercial model. To avoid this problem, we provide a content key management protocol to protect against malicious servers and prevent them from getting a complete content key. Malicious server has access to the entire content key and this could lead to abuse of the commercial model. To avoid this problem, we provide a content key management protocol to protect against malicious servers and prevent them from getting a complete content key.[13]

A User needs to perform the authentication with the Owner as a qualified user with its real identity credentials at the registration stage. The Owner will perform blind decryption for the registered users only. This scheme satisfies the non-anonymous authentication/registration property. This scheme provides privacy protection to the Users. A User after getting the Anonymous Token Set interacts only with the Anonymous Tokens. Though a User has been authenticated by its real identity in the non anonymous authentication process, the Content Provider or the Owner cannot link the real identity with the anonymous identities of the User in other transactions such as license acquisition, tracking and revocation but due to identity the privacy gets damaged so that there is need to maintain privacy by hiding identity. So in the current research work we maintain privacy by using blind decryption. Users are accountable for the contents/licenses they had purchased

III. SYSTEM ARCHITECTURE

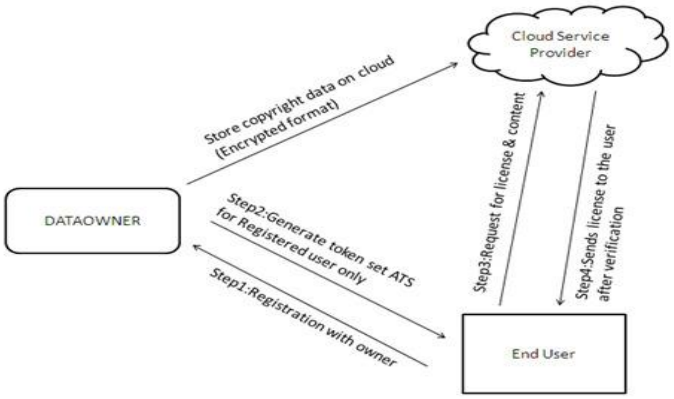

Fig (a) system architecture

A. Anonymous Token Generation

The Owner generates a collection of Anonymous Token Sets for the Users. A user can get a token set from the Owner anonymously and use it for content purchase. Each token set consists of anonymous tokens. The value is used to generate all anonymous tokens sets during the time period. Let be another secret value with the Owner. Now the Owner computes belonging to using a hash function. The Owner updates the value of after the time and generates new sets of anonymous Token Set with the new All the expired tokens

will be deleted. Each Anonymous Token Set is generated for transactions. Finally, each encrypted with a key using the symmetric-key encryption algorithm.

B. Content and License Creation With Access Control

Our privacy scheme can be used in any DRM systems. In particular, it can be used in the conventional two-party DRM systems [12], multiparty multilevel architectures [1] and multiple authorized domain architectures [2], [4] which involves redistribution licenses and usage licenses for transactions. To control the access of a content by unqualified or illegal distributors or end-users trusted access control can be enforced using the concept of attribute keys as given below. The Owner creates two attribute keys: for the distribution of the content and for the usage of the content. The Owner specifies the required attributes for each content. Only the distributor or end-user which has the specified attribute keys can access or redistribute the content. Each user gets their respective attribute keys according to the user information at Step 2 of the registration protocol. Suppose that the specified attribute to use the content is as follows: user must be older than 18 years, citizen of USA, and living in the New York City. Then the attribute keys for the usage of the content can be the 3 tuple,Suppose that the specified attributes for distribution of the content is as follows: distributor should have distributorship for the city of New York. Then the attribute key for the distribution of the content can be, . The structure of the attribute key varies with the contents. After specifying the respective attribute keys for the content , the Owner generates content encryption key using the user attribute key set of the content and a content usage key as , where is a hash function. The Owner encrypts the content with the key. The usage key will be inserted in the usage license and the usage license for the content is created as where is a token used in the content and license purchase, are the requested rights by the user or rights predefined by the Owner, a unique ID of the content . Therefore, only qualified and authentic end-users can get the correct. Similarly, attribute based redistribution key and redistribution license can be created for distribution of contents only by qualified distributors. A content package is composed of two parts: the content header and the encrypted content. The header part stores the content information such as content type, content resolution, required attribute for eligible end users and distributors and other content related information. The Owner stores the content packages in its content server. If a user qualifies for the attributes listed in the header part, the user downloads the encrypted content package from the content server.

C. Registration and Acquisition of Anonymous Token

Before communicating with the system for content purchase, each User needs to be registered with the Owner. Who requires anonymity, he/she first obtains an Anonymous Token Set Package from the Owner prior to the registration process. can get an Anonymous Token Set Package only if he/she has first made the payments for the service using an anonymous payment scheme [1], [9]. After making the payment, is provided with a payment receipt with no identity information but with a time stamp signed by the Owner. The presents the receipt to the Owner to get an Anonymous Token Set Package. To use the Anonymous Token Set Package, needs the decryption of the key. at a later point (the Owner will not know with which User he/she is interacting) of time requests the decryption of using the following blind decryption protocol [14].

1) chooses a random secret blinding factor such that then computes and sends to the Owner together with its PKI certificate, identity information and decryption request encrypted with owner's public key.

2) Owner decrypts and verifies the PKI certificate and the identity information of . Owner then computes and sends to . Owner saves the PKI certificate and the identity information of in its database.

3) Computes and obtains the decryption key.

After obtaining the decryption key , uses it to decrypt to get the Anonymous Token Set uses each token for each transaction with the Content Providers. A Content Provider only verifies the signature on the encrypted ID of the token . Thus the Content Provider will not get access to the real ID of the token . This is to avoid any misuse of the token ID by the Content Provider. Thus, will not be required to decrypt . The protocol for the decryption of is required to be performed by only at the first contact of to the system. The protocol is executed again only if the anonymous tokens of are expired. To prevent the stealing/loosing of the anonymous tokens of the User, the tokens in the Anonymous Token Set need to be securely stored at the User side. This is done by binding the tokens with the DRM agent at the User side using the seal storage function of the Trusted Platform Module (TPM) of the client device. The TPM contains a set of registers, called Platform Configuration Registers (PCR) containing measurement digests. The TPM protects all the tokens by encrypting them using a non migratable Storage Root Key bound to it. The tokens are bounded to the current platform configuration (as defined by the PCRs) via the following TPM operation: where is the encryption of with the key. The TPM verifies the integrity of the list of PCR values, and then compares them against the current values of those PCRs. If they match, the TPM decrypts and outputs the resulting. If any of the checks fail, the TPM returns an error.[2]

D. Anonymous License Acquisition Process

We now describe how a license can be acquired anonymously. A User first downloads the content from the content server of the Content Provider. It then obtains the license using an anonymous token. Before buying a license for the content , presents a token to the Content Provider for authentication. The anonymous license acquisition protocol for the content is given below.

1) Generates a secret key and sends the message after encrypting with the public key of the Content Provider.

2) Content Provider decrypts the message and Checks the expiry time , Verifies in the token using the public key of the Owner; Checks whether is a revoked/used token.

3) If everything is in order, the Content Provider sends the license to after encrypting with the secret key. makes the payment for the license in this step using an anonymous payment scheme [10]. Content Provider stores the token in its database.

4) Decrypts the encrypted license using the key and obtains the license.

VI. PROPOSED METHODOLOGY

Blind decryption introduced by Sakurai and Yamane [1] is similar to the Chaum's blind signature scheme. Suppose

that, Alice has a message encrypted with Bob's public key. The blind decryption allows Alice to get the message decrypted by Bob without Bob learning the message and Alice not learning the private key of the Bob. It is implemented using RSA cryptosystem. A blind decryption scheme is a public-key encryption (PKE) scheme that admits an efficient protocol for obliviously decrypting cipher texts. In this protocol a User who possesses a cipher text interacts with a Decryptor who holds the necessary secret key. At the conclusion of the protocol, the User obtains the plaintext while the Decryptor learns nothing about what it decrypted. Furthermore, the User should gain no information about any other cipher text. To formalize the latter guarantee, we will restict our investigation to secure blind decryption schemes that retain their security under aaptive chosen cipher text attack. Blind decryption has many applications to privacy-preserving protocols and systems [1].

Advanced Encryption Standard (AES)-Like DES, AES is a symmetric block cipher. This means that it uses the same key for both encryption and decryption. However, AES is quite different from DES in a number of ways. The algorithm Rijndael allows for a variety of block and key sizes and not just the 64 and 56 bits of DES' block and key size. The block and key can in fact be chosen independently from 128, 160, 192, 224, 256 bits and need not be the same. However, the AES standard states that the algorithm can only accept a block size of 128 bits and a choice of three keys - 128, 192, 256 bits. Depending on which version is used, the name of the standard is modified to AES-128, AES-192 or AES- 256 respectively. As well as these differences AES differs from DES in that it is not a feistel structure. Recall that in a feistel structure, half of the data block is used to modify the other half of the data block and then the halves are swapped. In this case the entire data block is processed in parallel during each round using substitutions and permutations. A number of AES parameters depend on the key length. For example, if the key size used is 128 then the number of rounds is 10 whereas it is 12 and 14 for 192 and 256 bits respectively. At present the most common key size likely to be used is the 128 bit key. This description of the AES algorithm therefore describes this particular implementation [8].

Rijndael was designed to have the following characteristics
• Resistance against all known attacks.
• Speed and code compactness on a wide range of platforms.
• Design Simplicity.
Advantages of AES:AES is more secure (it is less susceptible to cryptanalysis than 3DES).AES supports larger key sizes than 3DES's 112 or 168 bytes.AES is faster in both hardware and software.AES's 128-bit block size makes it less open to attacks via the birthday problem than 3DES with its 64-bit block size.AES is required by the latest U.S. and international standards.

Step by step explanation of the current scheme is given as below.

- STEP I: The owner will upload its copyrighted content on the cloud
- STEP II: The user who wants to access the Cloud content will now register with data owner.
- STEP III: The data owner will now generate Anonymous Token Set for the registered user.
- STEP IV: The end user now request for license to access the encrypted content.
- STEP V: The DRM agent on the cloud will now send the License for accessing the content to the user.
- STEP VI: If user request with the same token again then CSP revoked the user and sends token to the data owner
- STEP VII: Data owner will now update the list of revocated tokens in CSP

Detailed explanation of the current scheme with each phase is given below:

Token Generation

The data owner will generate a set of anonymous token sets $\{ATS_1, ...ATS_n\}$ for only the users who are registered with data owner. Each ATS consist of n number of tokens ATS= $\{T_1 \dots \dots T_n\}$. A user U_i will request for $ATS_i$to the data owner and used each token in the setATS_i for each transaction with Cloud servise Provider (CSP).[10]

Let $TID_{(i,j)}$ denotes the ID of the token $T_{(i,j)}$ and $TID^{enc}_{(i,j)}= \varepsilon_{pub}(TID_{(i,j)},K_{pb})$ denotes the encryption of $TID_{(i,j)}$ with the public key $K_{pb}$of the owner.

Let T_{exp} denotes the expiry time of all the tokens and $TID^{sgn}_{(i,j)}= S_{ign}(TID^{enc}_{(i,j)}) \|T_{exp},K_{pr})$ Denotes the Digital signature of the concatenation of $TID^{enc}_{(i,j)}$ with T_{exp}using the private key $K_{pr}$of the Owner.

Now, each token $TID_{(i,j)}$ in the token set ATS_i is given by $TID_{(i,j)}= \{TID^{enc}_{(i,j)} , TID^{sgn}_{(i,j)}, T_{exp}\}$.

Content Encryption:The data owner defines at his attribute for each content separately. Only the user who has the required usage attribute keys can access the content. The user gets the usage attribute key during the registration process from the data owner on the basis of the details given by the user during registration.[11]

Registration with the Owner: In order to purchase the digital content the user needs to be registered with the owner. The steps to register with the owner are as follows :

- The user selects the desired digital content from the list of the digital content displayed on the website.
- Once the digital content is selected the user needs to fill the registration form providing appropriate details.
- After filling the registration form, once the use makes the payment the users system details(Hard disk number and Mac address) are acquired.
- The acquired Hard disk and Mac address details are encrypted and stored in the database.[2]

Key Generation: The acquired system details are used to generate Users. A User after getting the Anonymous Token Set interacts only with the Anonymous Tokens. Though a User has been authenticated by its real identity in the non anonymous authentication process, the Content Provider or the Owner cannot link the real identity with the anonymous identities of the User in other transactions such as license acquisition, tracking and revocation. Users are accountable for the contents/licenses they had purchased. Usages are tracked by the Content Provider using the Anonymous Token of a User rather than the real identity of the User.[6] If a misuse of a license by a User is found, the Anonymous Token Set of the User is retrieved and revoked by the Owner. Further, the Content Providers are accountable for malicious users and the contents sold by them. For each Anonymous Token Set Package generation, the major computations need to be performed at the Owner side are: hashing operations, public-key encryptions, digital signature generations, symmetric-key encryptions where is the number of sub tokens in an Anonymous Token Set Package. The Owner needs to store the Anonymous Token Set Package of all generated token sets. Each one-time registration involves 3 rounds of communication between the Owner and a User. The major computation at the User side is one RSA encryption and the major computation at the Owner side is one RSA decryption. The User needs to store the Anonymous Token Set Package of length l. The Owner needs to store only his RSA private key.[15] The anonymous license acquisition process involves 2 rounds of communication between the User and the Content Provider. The major computation at the User side is one public-key encryption and the major computation at the Content Provider side are one public-key decryption, one signature verification, one symmetric key encryption and checking of the token in the Revocation List. The User needs to store the license. The Content Provider needs to store all the unexpired and un-revoked tokens sent by the users for tracking purpose and to prevent the reuse of the tokens. Revocation of a user involves one round of communication between the Owner and the Content Provider. The major computations at the Owner side are one public-key decryption, hash operations and public-key encryptions. The Owner and the Content Provider need to store the encrypted token Ids in the Anonymous Token Set for each unexpired and revoked token.

V RESULT ANALYSIS

A User needs to perform the authentication with the Owner as a qualified user with its real identity credentials at the registration stage. The Owner will perform blind decryption for the registered users only. This the scheme satisfies the non-anonymous authentication/registration property. The proposed scheme provides privacy protection to the Users. A User after getting the Anonymous Token Set interacts only with the Anonymous Tokens. Though a User has been authenticated by its real identity in the nonanonymous authentication process, the Content Provider or the Owner cannot link the real identity with the anonymous identities of the User in other transactions such as license acquisition, tracking and revocation. Users are accountable for the contents/licenses they had purchased. Usages are tracked by the Content Provider using the Anonymous Token of a User rather than the real identity of the User. If a misuse of a license by a User is found, the Anonymous Token Set of the User is retrieved and revoked by the Owner. Further, the Content Providers are accountable for malicious users and the contents sold by them. To block the Users who are no longer eligible tomake content transactions with the Content Providers the revocation of those Users has to be performed. In the proposed scheme, revocation of that User . A trusted third party (TTP) is an entity that facilitates the interactions between two parties who both trust the third party. In real life a TTP can become untrusted or malicious. In the proposed system the anonymity of the Users are preserved without the need to trust on any third parties.
The comparison of the proposed scheme with various other schemes is given in the Table.

Features	**Proposed Mechanism**
Non Anonymous User Authentication	Y
Content Accountability	Y
No Reliance on TTP	Y
Prevent Hacking of Key	Y
Privacy of User's system details	Y
Revocation of malicious User	Y
High security of Digital Content	Y

VI. CONCLUSION

The current work presented a novel privacy enabled digital rights management mechanism without the trusted third party assumption using simple primitives. The proposed scheme satisfies the conflicting requirements of accountability and privacy in digital content distribution.

Further, the proposed scheme supports access control without degrading user's privacy as well as allows revocation of even malicious users without violating their privacy. The implementation, analysis and comparison, demonstrate that the proposed scheme is efficient, satisfies the good design properties and out performs the related works.

REFERENCES

[1]Cong Wang; Chow, S.S.M.; Qian Wang; Kui Ren; Wenjing Lou, "Privacy-Preserving Public Auditing for Secure Cloud Storage," Computers, IEEE Transactions Feb. 2013.
[2] Lei Lei Win, Tony Thomas, and Sabu Emmanuel, "Privacy Enabled Digital Rights Management without Trusted Third Party Assumption", IEEE Transaction on Multimedia. 2012.
[3] Petrlic, R., "Privacy-Preserving Digital Rights Management in a Trusted Cloud Environment," Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on , Jun1 2012.
[4] Barsoum, A.; Hasan, A., "Enabling Dynamic Data and Indirect Mutual Trust for Cloud Computing Storage Systems," Parallel and Distributed Systems, IEEE Transactions, 2012.
[5] Khaled M. Khan and Qutaibah Malluhi,"Establishing Trust In Cloud Computing",IEEE Computer Society, September/October 2010
[6] L. L.Win, T. Thomas, and S. Emmanuel, "A privacy preserving content distribution mechanism without trusted third parties," in Proc. IEEE Int. Conf. Multimedia, Barcelona, Spain, 2011,
[7] Qian Wang; Cong Wang; Kui Ren; Wenjing Lou; Jin Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing," Parallel and Distributed Systems, IEEE Transactions, May 2011.
[8] Chih-Ta Yen; Hrong-Twu Liaw; Nai-Wei Lo; Ting-Chun Liu; Stu, J., "Transparent Digital Rights Management System with Superdistribution," Broadband, Wireless Computing, Communication and Applications (BWCCA), 2010 International Conference on, Nov. 2010
[9] A. O. Durahim and E. Savas, "A-MAKE: An efficient, anonymous and accountable authentication framework for WMNs," in Proc. ICIMP, 2010
[10] E. McCallister, T. Grance, and K. Scarfone, "Guide to Protecting the Confidentiality of Personally Identifiable Information (PII)," National Institute of Standards and Technology (NIST), Special Publication 800-122, Apr. 2010.
[11] Beum-Su Park,shirlyl Lee,Hoon-Jae Lee, On A Digital-Right-Management System using One-Time-Password,2010
[12] M. K. Sun, C. S. Laih, H. Y. Yen, and J. R. Kuo, A Ticket Based Digital Rights Management Model In Proc. CCNC 2009
[13]C. Conrado, F. Kamperman, C. J. Schrijen, and W. Jonker, "Privacy in anIdentity-based DRM System," Proceeding of the 14th IEEE Int. Workshop on Database and Expert Systems Applications, 2003.
[14]Bok-Nyong Park, Jae-Won Kim and Wonjun Lee, "PrecePt: A privacy-enhancing license management protocol for digital rights management," Proceedings of IEEE AINA'04, 2004, pp.574-579.
[15]H.M. Sun, C.F. Hung, and C.M Chen, "An improved Digital Rights Management System Based on Smart Cards," 2007 Inaugural IEEE International Conference on Digital Ecosystems and Technologies (IEEE DEST 2007).
[16]Jiang Zhang, Bin Li, Li Zhao, Shi-Qiang Yang" license management scheme with anonymous trust for digital rights management" Department of Computer Science , Tsinghua University, Beijing China zhang-jiang03, libin98, zhaoli

Ankita A.Deshmukh received the B.E. degrees in Computer Science & Engineering from Sipna College of Engineering & Technology in 2013. Now pursuing ME (CSE) from P.R.Pote (Patil) College of Engineering &Management, Amravati.

Prof. V.B.Gadicha received the M.E. degrees in computer science and engineering ,Amravati**.**