Attribute Based Access to Versatile Media in Cloud-Supported Substance distribution impart Systems

[1]K.Satya Narayana & [2]S.Naga Lakshmi

[1]M.Tech Research Scholar, Department of CSE, Priyadarshini Institute of Technology & Science, Chintalapudi, India
[2]Assistant Professor, Department of CSE
Priyadarshini Institute of Technology & Science, Chintalapudi, India

Abstract-

Cloud computing is the significant computing paradigm which allows the users to store their data into cloud this paper presents an Attribute-Based access to the media in the cloud where it uses cipher-text policy Attribute-Based Encryption (CP-ABE) technique to create an access control structure. By using the algorithms in the access policy the attributes are used to generate a public key in order to encrypt the data and a secret key consisting of user attributes to decrypt the data and is used as an access policy in order to restrict the access of the user. . By using ABE technique the encrypted data is trustworthy even on the untrusted server. This requires flexible and accessible cryptographic key management to support difficult access policies. The policy is to assign a key to each user attribute and encrypts the data based on the appropriately distributed keys to corresponding user.

Keywords-

Cloud computing; CP-ABE; Access Control; scalable media content

1. INTRODUCTION

Cloud computing offers the abstract view to the users and developers. It hides much of the Implementation details. It is mainly used in content sharing networks. Examples for these networks are social networking where they are dynamic in terms of storage requirement. However due to the weak security issues the use of cloud is not very fast in content sharing networks.

A promising approach to access control in content sharing services is to empower users to enforce access controls on their data directly, rather than through a central administrator. However, this requires flexible and scalable cryptographic key management to support complex access control policies.

Access policy is a mechanism that provides security facilitates the data to user in a controlled manner. The traditional mechanism is that the data is encrypted with the user's public keys. The data owners encrypt the data using this user's public key and then upload the file to the cloud. The user whenever wanted to download the file should decrypt the file with his generated secret key. By doing this there are a few problems like the owner has to get the public key of the user and the same data is encrypted with different public keys this results in storage overhead. For example cryptic text c=E(E(m,sk1),sk2) here encrypting multiple times with the key

pair(sk1,sk2) here one user has an attribute key sk1 and another user has an attribute key sk2 this may collude to decrypt the data. Hence for a particular shared data among the multiple users we need to encrypt the data with every user's Public key in order to provide security hence an ordinary encryption is unsatisfactory. Instead if the cipher text consists of the set of attributes then by using the key and access policy we can decrypt the data i.e. the key works only when the attributes in the cipher text satisfies the access policy.

For this purpose, we introduce a novel Multi-message Ciphertext Policy Attribute-Based Encryption (MCP-ABE) technique. MCP-ABE encrypts multiple messages within one ciphertext so as to enforce flexible attribute-based access control on scalable media. Specifically, the scheme constructs a key graph which matches users' access privileges, encrypts media units with the corresponding keys, and then encrypts the key graph with MCP-ABE; only those data consumers with the required user attributes can decrypt the encryption of the key (sub)graph and then decrypt the encrypted media nits. The experiments demonstrate that the present scheme is applicable on Smartphone, especially when a cloud platform is available.

2. RELATED WORK

The relationship between the user identification and resource in content sharing applications is dynamic. There are two forms of access management strategies they're user attribute access management structure and Media Structure minded Access management structure

A) User attributes access management structure

Easier is a design that supports fine-grained access control policies and dynamic cluster membership by victimization CP-ABE theme. a lot of works are projected to style versatile ABE schemes There are two methods to comprehend the fine-grained access management supported ABE they are KP-ABE and CP-ABE. In KP-ABE the cipher text consist of some descriptive attributes which are labeled by the sender and the trusted authority issues a user's private key and the access policy is involved in the private key which specifies the decryption of the cipher text with the key. Here the disadvantage of this encryption is that the access policy is constructed into user's personal key. So data owner does not have the option on who can decrypt the data except encrypting the data with the set of attributes. Hence it is not suitable for certain applications as the information owner must trust the authority who gives the user's key. The KPABE is secure beneath the final cluster model because it is monotonic access structure and additionally it cannot categorical the attributes to reject the parties with whom the knowledge owner didn't got to share the knowledge from membership. To overcome this weakness cipher text policy attribute based encryption has been created that is proved to be secured below the quality model. In CP-ABE the access policy is made within the encrypted data and also the attributes is with the user's private key. The attribute based encryption will be divided into monotonic or no monotonic based on the sort of the access structure and based on the access policy the schemes will be classified as key policy or cipher text policy. The ideal attribute based encryption must support data privacy, scalability, fine grained access control, user accountability,

user revocation and collusion resistant. But the provided access policies are not appropriate for the scalable media content.

B) Media structured access control

For a video the secure scalable streaming is the progressive encryption technique. This should be integrated with error correction technique since it may result in decryption failure due to the packet loss. An access control scheme is designed by wu et al which is highly secured and efficient and predominantly the scheme is flexible as its "*encrypt once, decrypt many ways"* is compatible with the features of jpeg 2000.Zhu et al. [19] proposed an access management schemes

for streams determined by the MPEG-4 Fine granularity scalability (FGS) normal thus on allow one encrypted stream to support each forms of scalabilities simultaneously. The organization of the media data will be ruined by the media structured access control in request to ensure the data so that the client will unscramble the separate figure content with the important keys. These plans are constrained to productive key generations furthermore ordinarily expect the presence of an online key spreading center; and they don't manage access policies, e.g., how to give user attributes to access rights.

All the above media structure based access control schemes exploit the format of media data to generate protected objects so that users with the necessary keys can decrypt the corresponding ciphertext. These schemes are limited to efficient key generations and normally assume the existence of an online key distribution center; and they don't deal with access policies, e.g., how to assign user attributes to access privileges.

3. PRELIMINARIES

3.1. Bilinear Map

Let G0 and G1 be two multiplicative cyclic groups of prime order p. Let g be a generator of G0 and e be a bilinear map,
e : G0 × G0 → G1. Then e has the following properties:

Bilinearity: for all u,v € G0 and a, b € Zp, we have

$e(u^a, v^b) = e(u, v)^{ab}$

Non-degeneracy: e(g, g) ≠ 1

3.2. CP-ABE

For **CP-ABE** scheme, a message is encrypted under an access policy and a secret key of a user is associated with a set of attributes. A user could decrypt the message only if his attributes satisfy the access policy. The CP-ABE scheme includes five algorithms that can be illustrated as follows:

Setup: The setup algorithm chooses a bilinear group G0 of prime order p with generator g, and two random exponents α, β € Zp. Public key PK and master key MK are returned as:

$\{PK = G0, g, h = g^{\beta}, f = g^{1/\beta}, e(g, g)^{\alpha}\}$ $\{MK = \beta; g^{\alpha}\}$.

Encryption: Given an input message M, this algorithm encrypts M under the access tree T. Firstly, a polynomial px is chosen for each tree node x in a manner as: (1) Set the degree dx of the polynomial px to be dx = kx - 1, where kx is the threshold value of node x; (2) For the root node R, choose a random s € Zp and set pR(0) = s, and randomly choose other points of polynomial pR; (3) For any other node x, set px(0) = pparent(x)(index(x)), and randomly choose other points of px. Let L be the set of leaf nodes in T , the ciphertext is given as

$CT = (T, \acute{C}=Me(g, g)^{\alpha s}, C = h^{s}$

Key Generation: Taking a set of attributes S as input, the key generation algorithm outputs a secret key that identifies with the set. First, it selects a random r € Zp and a random ri € Zp for every attribute in S. The key is computed as

$SK =(D = g^{(\alpha+r)/\beta}$

Decryption: The decryption algorithm takes as input a message encrypted under an access policy, a secret key SK of a user, and the public key PK. It first calculates e(g, g)rpx(0) for each leaf node x. Then it recursively computes the corresponding values for non-leaf nodes In a bottom-up manner using polynomial interpolation technique. If the attributes completely satisfy the access policy of the tree, it could compute the value for the root node as
$A = DecryptNode(CT, SK, r) = e(g, g)^{rp}{}_{R(0)} = e(g, g)^{rs}$.

3.3 Access Tree

T - access tree
Nj - j th hub of the access trees
A - attributes of the data user or clients
AA - Attribute authority
ai - ith attribute of the user
L - leaf hubs of the access tree
S - set of attributes of the leaf nodes belonging to a particular non-leaf node
An access tree is a graph representation of the access policy. Such a tree includes non-leaf nodes and leaf nodes. Each leaf node is associated with a user attribute (e.g., age, gender, profession), while each non-leaf node has child nodes which may be leaf nodes, other non-leaf nodes or both.

A Boolean function which is derived from the access policy is related with every non-leaf node and the Boolean function of the non-leaf node is represented as nj/n where n is nothing but the child nodes belonging to Nj and its Boolean value is calculated to be true if and only if it has atleast n_j child nodes. For suppose the Boolean function for the node N2 is 2/3 or equally a1a2+a2a3+a1a3 is true if a1 and a2 belongs to S and we can say nj is evaluated to be true. T(A) is evaluated to be true if the Attributes of the user satisfies the access policy and this can be explained as following. For any leaf node Nj which is accompanying with the attributes belonging to A its Boolean value is evaluated to be true and for every non-leaf node the Boolean value is the significance of its Boolean function and the T(A)= true if and only if the root nodes Boolean value is true.

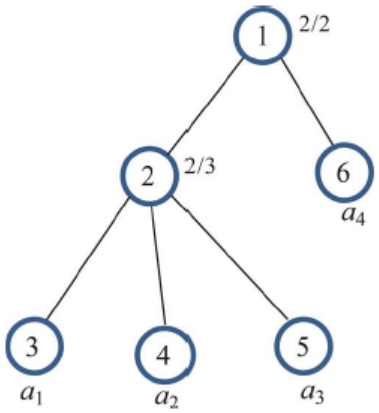

Fig.1. Access tree—the graph representation of access policy.

4. MEDIA SHARING IN CLOUD ENVIRONMENT

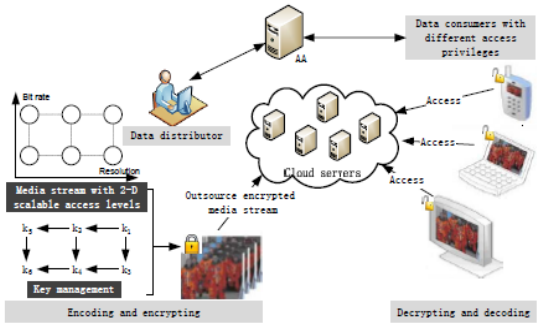

Fig.2. System Architecture

A. System Architecture

As shown in Fig.2, the scalable media content sharing in cloud environment involves the following parties: the data distributor, the attribute authority (AA), the cloud servers, and the data consumers.

On the data distributor side, a media stream is encoded into a manner that is scalable in two dimensions. For example, Fig.2 shows a media stream that is encoded by SVC, and is scalable in terms of bit rate and resolution. The media data, including the base layer and enhancement layers, are then encrypted with specific access keys. Note that if a user could acquire the access key at a specific level, he can also acquire all of the access keys for every lower level. For instance, a user with key k2 is capable to derive k4; k5; k6 and encode all the media not exceeding the middle resolution level and the middle bit-rate level. The generation of the access keys will be discussed in details in section 4.2. The encrypted media stream is then output to the cloud server ready for sharing.

A data consumer downloads the media of his interest from the cloud server, and decrypts the content by recovering the access keys with the assistance of the cloud server. Intuitively, we would like to enable the data consumers having a larger set of desired attributes to obtain the media data with a higher quality, and limit those who have a smaller set of attributes to access a lower quality media data. For example, a consumer with the attributes of "Tara's classmate, Tara's friend, same Location with Tara" would be expected to have higher access privilege in terms of Tara's media data than the consumer only having the attribute "same location with Tara".

In this data owner-consumer model, the backend servers provide the fundamental platform for storage, networking, etc; the foreground servers provide the interface for media generation, transmission, and computational assistance to users; while AA issues personal secret keys so that access control can be enforced flexibly based on user attributes and media scalability.

B. Data Structure of Media

We could exchange the formats of all files in media sharing applications. Particularly, some media file formats such as Text, PDF, JPEG 2000, Microsoft Word, SVC files and presentations, their content can be segmented into logical units. We refer to such media content as *scalable* media content. For example, assuming that an SVC video file includes one base layer and two enhancement layers, we may assign three access privileges to users. If the unit for base layer is assigned to a consumer, she obtains a video experience of basic quality, but if the base layer and the two enhancement layers are available, the video shown to her will be of full fidelity.

5. ACCESS CONTROL ON SCALABLE MEDIA

A. Scalable Media Encoding

Given a media stream for sharing, the data distributor first encodes the media in a manner that is scalable in two dimensions. Based on the encoded media stream, we will have a desired access key management structure. For example, if the media stream is encoded into six layers, the corresponding key management structure will be like the one in Fig.3(a), which is composed of six access keys.

B. Access Key Generation

Let us treat the key management structure as a directed graph G. Then the edge edge(ki; kj) indicates that the generation of kj is based on ki. It is also guaranteed that kj can be obtained by a user if any ki(edge(ki, kj) € G) could be obtained by the user. Unlike the traditional scalable media access control schemes, where kj is the combination of the hashing components of ki and is generated from ki using hashing, here we generate kj based on the desired attributes and the value for the key node that locates in the access tree and indicates ki.

Specifically, we adapt CP-ABE to build a scalable access policy for generating the access keys, in the following fashion.

(1) First, we convert the key management structure (a graph) into a binary tree, by keeping all nodes and edges in the graph and locating high level access keys at low layers of the access tree. Fig. 3 (a-b) show an example of such conversion;

(2) Next, we select a set of attributes for each of the edge, indicating a computable way from the higher level access key to the lower level access key when the attributes are satisfied, and construct a referencing tree, as shown in Fig. 3(c);

(3) Then, we construct a scalable access tree T based on the referencing tree. Nodes in T include leaf nodes, non-leaf nodes, and key nodes Vj that are related to access keys kj .

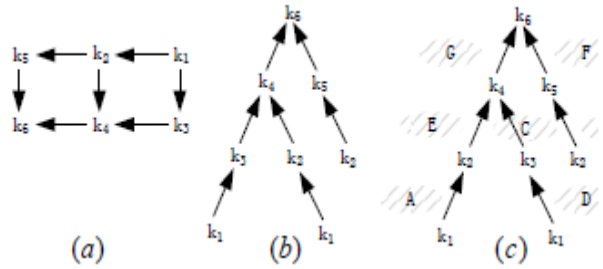

Fig.3. The access key structure

From bottom to up, we iteratively construct subtrees for the key nodes, following the structure of the referencing tree. For instance, in Fig. 3 that is generated from the referencing tree shown in Fig. 2(c), we treat V1 as a leaf node, use this and respectively choose attributes from attribute set A and D to construct a subtree for V2 and V3. We then iteratively construct the subtrees for V4, V5, V6. Apart from the attribute sets in the referencing tree, another attribute attriCommon, which is the common attribute owned by all authorized consumers but not owned by the cloud server, is added as the child of the root node in T . Then a polynomial px is chosen for each node x in T in the same way as CP-ABE. Since there may be two ways to recover an access key from the higher level access keys, some key node may appear twice in T , either as a leaf node or as a non-leaf node. To keep the value of an access

key consistent, we choose the same polynomial for the two repeated key nodes.

(4) Finally, we select a random value m 2 Zp and generate the key seed e(g; g)m, and then encrypt it under the access policy. Let L be the set of leaf node, the ciphertext will be

$$CT = (\mathcal{T}, \tilde{C} = (e(g,g)^m)e(g,g)^{\alpha s}, C = h^s, \\ \forall i \in L : E_i = g^{p_i(0)}, E_i' = H(att(i))^{p_i(0)})$$

After the construction of T, we can generate the access keys by calculating kj = (Kj ⁄ e(g, g)^m) = e(g, g)$^r_{Kj}$, where
Kj = e(g, g)$^{rpv_j(0)}$ is the value for the key node Vj , and
R_{kj} = (r . pv_j (0) – m) € Z_p.

C. Media File Creation

After the access keys are generated, each layer is then encrypted by the corresponding access key. Standard encryption algorithms such as AES could be adopted. The encrypted media layers, the syntax of the access tree, and the values for the key nodes, i.e. Kj , are then sent by the data distributor to the cloud server. The cloud server will store this information and control the access on the encrypted media data.

6. DISCUSSIONS

A. Cloud-Assisted Decryption

The data user executes this algorithm by taking the cipher text, secret key and the set of attributes in order to decrypt the cipher text as per the access policy.

a) Set the Boolean value of N_j is TRUE if N_j is a
Leaf node and the attribute a_i € A ∩ S.

V_j = $DeNode_1$(CT, SK, j, a_j)
= e(g^r, $g_i^{f(0)}$)e($H_1(a_i)^r_j$,$g_i^{f(0)}$
=e(g^r, $g_i^{f(0)}$) = e(g,g)$^{rf(o)}_j$

The Boolean value of N_j is not true if the last three equations are not due to the bilinear property.

b) If N_j has child nodes then let S_j be the random set of the n_j sized set of the child nodes *z.* For suppose if the node has x child nodes then it call the DeNode(CT,SK,Z) and stores output as f_x, $f_x \neq \nabla$. If no such set exist then it is not satisfying the access policy otherwise it fulfills the access

policy setting N_j value to TRUE.

f x =*DeNode*(CT,SK,Z)

The final result is:

f_x = e (g, g)$^{rf(0)}_j$.

B. Provable Security

In CP-ABE the respective user's private key is used to decrypt the data where the access policy is built into the encrypted data. In CP-ABE the encrypted data can select who can decrypt it whereas this remained as the disadvantage in KP-ABE. The attributes in the user's private key plays a vital role as these are responsible to fulfill the access policy built into the encrypted data. The CP-ABE access control supports in the real time environment. The concept of CP-ABE is also used in MCB-ABE. In MCB-ABE the CP-ABE is used to encrypt the multiple messages with the same public key and the access policy is built into the encrypted data. When the user attributes satisfy the access policy then the corresponding message will be decrypted the remaining message will be in encrypted form because the multiple messages are encrypted together with the same public key.

C. Flexible Access Control

The proposed scheme allows a data distributor to define fine grained access policy for the media data by integrating attributes with access key structure. This is very much desired for sharing of social media and cloud media in which multiple levels of access privileges are needed. The media data for sharing is encrypted under symmetric encryption algorithm. CP-ABE is applied to compute the key of the symmetric encryption algorithm. The confidentiality is achieved based on the security of symmetric encryption algorithm and the security of CP-ABE. Besides, although the cloud servers have all of the Ki, it cannot decrypt the key seed if it does not own the common attribute. Therefore, the cloud server shall not be able to recover the access keys, and hence could not obtain the media content, either.

D. Fine-Grained Access Control

By changing the polynomials for the tree nodes, Ki can be changed, and hence the access keys can also be changed. In order to recover the new access keys to decrypt the new media data, the revoked consumer will need to acquire K'_j, which is equal to $e(g, g)r.p'_{Vj}(0)$. However, the data consumer only has $Kj = e(g, g)^{r.pV_j(0)}$. Without knowing the value of $^{p'}V_j(0)$ and pVj (0), which are randomly chosen and kept private by the data distributor, there is no way for the consumer to compute K0j from Kj . Therefore, we can claim that the revoked data consumer cannot acquire the subsequent access keys, given that he has all old access keys. As a result, forward secrecy is satisfied in the proposed scheme. When a new data consumer joins in the system, he will be assigned with a secret key related to his attributes. If the attributes satisfy part of the scalable access policy, he can obtain the current K_j' and the key seed, then recover k'_j . In order to recover the previous old access keys, the consumer has to compute Kj according to K'_j, which is infeasible according to the discussion in forward secrecy. This means that the proposed scheme also satisfies backward secrecy.

7. EXPERIMENTS

To demonstrate the multi-level access control, we constructed a presentation with one text file and a fireman sequence (30 frames, encoded into one base layer and two enhancement Layers, with a total 103,871 bytes), and generated the encrypted presentation and its enabling block with the method laborated in Section. A Smartphone is used to carry out all the Decryption operations by exploiting the JAVA bilinear map Software package [31] after obtaining the encrypted presentation and its enabling block.

We observe that MCP-ABE has smaller overhead than CP-ABE. Statiscally, the overhead of MCP-ABE is 5437 ± 2627 bytes, while the overhead of CP-ABE is 9588 ± 4452 bytes.

Fig.4. Garbled text file shown to unauthorized User 1 or User 2.

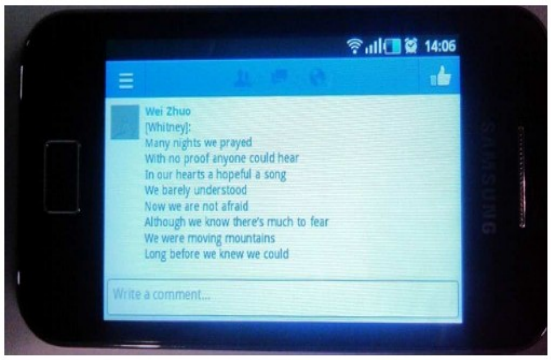

Fig.5. Clear text file shown to User 3.

Fig.6. Base layer shown to User 4 and User 5.

Fig.7. Full video shown to User 6.

User 1 nor User 2 is authorized to access anything; hence, their hand phones output garbled data shown in Fig. 4. User 3 is able to view the text file shown in Fig. 5, and User 4 obtains the base layer shown in Fig. 6 in addition to the text file in Fig. 9. User 6 gets the full content, i.e., the text file shown in Fig. 9 and the full video shown in Fig. 7.In other words, MCP-ABE realizes the multi-message encryption for meeting the multi-privilege access control requirement.

8. CONCLUSION

CP-ABE primarily based access management permits a data owner to enforce access management supported attributes of data customers while not explicitly naming the particular information customers. However, CP-ABE supports just one privilege level and therefore isn't suitable for access management to ascendable media. In this paper we presented a basic development of the CP-ABE and how the access structure is built in the CP-ABE. Cloud computing is the highly adaptive technology and mobile devices are becoming widespread the above presented CPABE access control helps to free from the computational demanding operations on the cloud server. The experimental results show that the CP-ABE is flexible, scalable, user, accountability, collision, resistant, user revocation. With the assistance of the cloud the acceleration of the decryption increased but it is still slow in some lowend devices because a Integrated exponentiation operation is required. We have confirmed the effectiveness of the proposed scheme through both numerical analysis and mobile terminal implementation with typical laptop and smart phones.

REFERENCES

[1] E. Messmer, “Are security issues delaying adoption of cloud computing?,” *Network World*, Apr. 2009 [Online].

Available:http://www.networkworld.com/news/2009/042709-burning-security-cloud-computing.html

[2] E. Messmer, "Security of virtualization, cloud computing divides IT and security pros," *Networkworld.com*, Feb. 2010 [Online].Available:http://www.networkworld.com/news/2010/022210-virtualization-cloud-security-debate.html

[3] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attributebased encryption," in *Proc. IEEE Symp. Security Privacy*, 2007, pp. 321–334.

[4] *National Inst. Standards and Technol., Secure Hash Standard (SHS)*, FIPS Publication 180-1, 1995.

[5] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," NIST Special Publication 800-145, 2011.

[6] M. D. Soete, "Attribute certificate," in *Encyclopedia of Cryptography and Security*, H. C. A. Van Tilborg and S. Jajodia, Eds., 2nd ed. Berlin, Germany: Springer, 2011, p. 51.

[7] B. Carbunar, Y. Yu, W. Shi, M. Pearce, and V. Vasudevan, "Query privace in wireless sensor networks," *ACM Trans. Sensor Networks*, vol. 6, no. 2, 2010, Art. ID 14.

[8] X. Zhang, M. Nakae, M. J. Covington, and R. Sandhu, "Toward a usage-based security framework for collaborative computing systems," *ACM Trans. Inf. Syst. Security*, vol. 11, no. 1, pp. 1–36, 2008.

[9] S. Jahid, P.Mittal, and N. Borisov, "EASiER: Encryption-based access control in social networks with efficient revocation," in *Proc. ACM Symp. Inf. Computer Commun. Security*, Mar. 2011, pp. 411–415.

[10] S.Yu, C. Wang,K. Ren, andW. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *Proc. IEEE Int. Conf. Comput. Commun.*, 2010, pp. 1–9.

[11] V. Goyal, O. Pandey, A. Sahai, and B.Waters, "Attribute based encryption for fine-grained access conrol of encrypted data," in *Proc. ACM Conf. Comput. Commun. Security*, 2006, pp. 89–98.

[12] M. Pirretti, P. Traynor, P. McDaniel, and B. Waters, "Secure attribute based systems," in *Proc. ACMConf. Comput. Commun. Security*, 2006, pp. 99–112.

[13] J. A. Akinyele, C. U. Lehmanny, M. D. Green, M. W. Pagano, Z. N. J. Petersonz, and A. D. Rubin, "Self-protecting electronic medical records using attribute-based encryption," in *Proc. ACM Conf. Comput. Commun. Security: Workshop on Security and Privacy in Smartphones and Mobile Devices*, Oct. 2011, pp. 75–86.

[14] R. Baden, A. Bender, N. Spring, B. Bhattacharjee, and D. Starin, "Persona: An online social network with user defined privacy," in *Proc.ACM SIGCOMM Conf. Data Commun.*, 2009, pp. 135–146.

[15] S. J. Wee and J. G. Apostolopoulos, "Secure scalable streaming enabling transcoding without decryption," in *Proc. IEEE Int. Conf.Image*, 2001, pp. 437–440.

[16] V. Gergely and G. Feher, "Enhancing progressive encryption for scalable video streams," in *Proc. EUNICE,*
Open European Summer School and IFIP TC6.6 Workshop on The Internet of the Future, 2009, vol. 5733, Lecture Notes in Computer Science, pp. 51–58.

[17] Y. Wu, D. Ma, and R. H. Deng, "Flexible access control to JPEG2000 image code-streams," *IEEE Trans. Multimedia*, vol. 9, no. 6, pp. 1314–1324, Oct. 2007.

[18] ISO/IEC 14496-2, Coding of Audio-Visual Objects-Part 2: Visual.

[19] B. B. Zhu, C. Yuan, Y. Wang, and S. Li, "Scalable protection for MPEG-4 fine granularity scalability," *IEEE Trans. Multimedia*, vol. 7, no. 2, pp. 222–233, Apr. 2005.

[20] T. W. H. Schwarz and D. Marpe, "Overview of the scalable video coding extension of the h.264/avc standard," *IEEE Trans. Circuits Syst.Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.

[21] S.-W. Park and S.-U. Shin, "Efficient selective encryption scheme for the H.264/scalable video coding (SVC)," in *Proc. Int. Conf. Networked Computing Adv. Inf. Manag.*, 2008, vol. 1, pp. 371–376.

[22] S. Lian, "Secure service convergence based on scalablemedia coding," *Telecommun. Syst.*, vol. 45, no. 1, pp. 21–35, 2010.

[23] C. Li,X. Zhou, andY. Zhong, "NAL level encryption for scalable video coding," in *Proc. Pacific-Rim Conf. Multimedia*, 2008, vol. 5353, Lecture Notes in Computer Sci., pp. 496–505.

[24] G. B. Algin and E. T. Tunali, "Scalable video encryption of H.264 SVC codec," *J. Vis. Commun. Image Representation*, vol. 22, no. 4, pp. 353–364, 2011.

[25] Y. G. Won, T. M. Bae, and Y. M. Ro, "Scalable protection and access control in full scalable video coding," in *Proc. Int. Workshop Digital Watermarking*, 2006, pp. 407–421.

[26] H. Hellwagner, R. Kuschnig, T. Stutz, and A. Uhl, "Efficient in-network adaptation of encrypted H.264/SVC content," *Image Commun.*,vol. 24, no. 9, pp. 740–758, 2009.

[27] M. Green, S. Hohenberger, and B.Waters, "Outsourcing the decryption of ABE ciphertexts," Usenix Security, 2011.

[28] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute-based encryption with non-monotonic access structures," in *Proc. ACM Conf. Comput. Commun. Security*, 2007, pp. 195–203.

[29] J. Bethencourt, A. Sahai, and B. Waters, Ciphertext-Policy Attribute-Based Encryption Mar. 24, 2011