RESEARCH ARTICLE OPEN ACCESS

Expressive, Efficient and Revocable Data Access Control for Multi-Authority Cloud Storage

Chetan Bulla
Associate Professor
Akshata R. Patil, Priyanka B. Guttedar and Reshma G. Giddenavar
Students, Department of Computer Science and Engineering
KLE's KLE College of Enggineering. & Tech, Chikodi
Belagavi – India

ABSTRACT

Data access control is an effective way to ensure the data security in the cloud. Due to data outsourcing and un-trusted cloud servers, the data access control becomes a challenging issue in cloud storage systems. Cipher text-Policy Attribute-based Encryption (CP-ABE) is regarded as one of the most suitable technologies for data access control in cloud storage, because it gives data owners more direct control on access policies. However, it is difficult to directly apply existing CP-ABE schemes to data access control for cloud storage systems because of the attribute revocation problem. In this paper, we design an expressive, efficient and revocable data access control scheme for multi-authority cloud storage systems, where there are multiple authorities co-exist and each authority is able to issue attributes independently. Specifically, we propose a revocable multi-authority CP-ABE scheme, and apply it as the underlying techniques to design the data access control scheme. Our attribute revocation method can efficiently achieve both forward security and backward security. The analysis and simulation results show that our proposed data access control scheme is secure in the random oracle model and is more efficient than previous works.

Keywords:- Access control, multi-authority, CP-ABE, attribute revocation, cloud storage

I. INTRODUCTION

CLOUD storage is an important service of cloud computing, which offers services for data owners to host their data in the cloud. This new paradigm of data hosting and data access services introduces a great challenge to data access control. Because the cloud server cannot be fully trusted by data owners, they can no longer rely on servers to do access control. Cipher text-Policy Attribute-based Encryption (CP-ABE), is regarded as one of the most suitable technologies for data access control in cloud storage systems, because it gives the data owner more direct control on access policies. In CP-ABE scheme, there is an authority that is responsible for attribute management and key distribution. The authority can be the registration office in a university, the human resource department in a company, etc. The data owner defines the access policies and encrypts data according to the policies. Each user will be issued a secret key reflecting its attributes. A user can decrypt the data only when its attributes satisfy the access policies.

There are two types of CP-ABE systems: single-authority CP-ABE [2], [3], [4], [5] where all attributes are managed by a single authority, and multi-authority CP-ABE [6], [7], [8] where attributes are from different domains and managed by different authorities. Multi-authority CP-ABE is more appropriate for data access control of cloud storage systems, as users may hold attributes issued by multiple authorities and data owners may also share the data using

access policy defined over attributes from different authorities. For example, in an E-health system, data owners may share the data using the access policy ''Doctor AND Researcher'', where the attribute ''Doctor'' is issued by a medical organization and the attribute ''Researcher'' is issued by the administrators of a clinical trial. However, it is difficult to directly apply these multi-authority CP-ABE schemes to multi-authority cloud storage systems because of the attribute revocation problem. In multi-authority cloud storage systems, users' attributes can be changed dynamically. A user may be entitled some new attributes or revoked some current attributes. And his permission of data access should be changed accordingly. However, existing attribute revocation methods either rely on a trusted server or lack of efficiency, they are not suitable for dealing with the attribute revocation problem in

data access control in multi-authority cloud storage systems.

In this paper, we first propose a revocable multiauthority CP-ABE scheme, where an efficient and secure revocation method is proposed to solve the attribute revocation problem in the system. As described our attribute revocation method is efficient in the sense that
it incurs less communication cost and computation cost, and is secure in the sense that it can achieve both backward security (The revoked user cannot decrypt any new ciphertext that requires the revoked attribute to decrypt) and forward security (The newly joined user can also decrypt the previously published ciphertexts1, if it has sufficient attributes). Our scheme does not require the server to be fully trusted, because the key update is enforced by each attribute authority not the server. Even if the server is not semi-trusted in some scenarios, our scheme can still guarantee the backward security. Then, we apply our proposed revocable multi-authority CP-ABE scheme as the underlying techniques to construct the expressive and secure data access control scheme for multi-authority cloud storage systems.

II. EXISTING SYSTEM

This new paradigm of data hosting and data access services introduces a great challenge to data access control. Because the cloud server cannot be fully trusted by data owners, they can no longer rely on servers to do access control. Cipher text-Policy Attribute-based Encryption (CP-ABE) is regarded as one of the most suitable technologies for data access control in cloud storage systems, because it gives the data owner more direct control on access policies. In CP-ABE scheme, there is an authority that is responsible for attribute management and key distribution.

In a multi-authority cloud storage system, attributes of users can be changed dynamically. A user may be join some new attributes or revoked some current attributes.

In 2010, S. Yu, C. Wang, K. Ren, and W. Lou, worked on **"Attribute Based Data Sharing with Attribute Revocation".** This paper use semi-trustable on-line proxy servers. This server enables the authority to revoke user attributes with minimal effort. This scheme was uniquely integrating the technique of proxy re-encryption with CP-ABE, and also enables the authority to delegate most of laborious tasks to proxy servers. The advantages of this scheme is More Secure against chosen cipher text attacks. Provide importance to attribute revocation which is difficult for CP-ABE schemes.

Drawback:

The storage overhead could be high if proxy servers keep all the proxy re-key.

In 2011, S J. Hur and D.K. Noh, worked on **"Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems"**. This paper proposes an access control mechanism based on cipher text-policy attribute-based encryption to enforce access control policies with efficient attribute and user revocation method. The fine-grained access control can be achieved by dual encryption scheme. This dual encryption mechanism takes advantage of the attribute-based encryption and selective group key distribution in each attribute group. The advantage of this scheme is securely managing the outsourced data. This scheme achieve efficient and secure in the data outsourcing systems.

Drawback:

Huge issue in Enforcement of authorization policies and the support of policy updates

In 2011, S. Jahid, P. Mittal, and N. Borisov, worked on **"Easier: Encryption-Based Access Control in Social Networks with Efficient Revocation".** The proposed Easier architecture that supports two approaches are fine-grained access control policies and dynamic group membership. Both scheme achieved by using attribute-based encryption, however, is that it is possible to remove access from a user without issuing new keys to other users or re-encrypting existing cipher texts. We achieve this by creating a proxy that participates in the decryption process and enforces revocation constraints. The advantage of this scheme is the Easier architecture and construction provides performance evaluation, and prototype application of our approach on Face book.

Drawback:

Does not Achieve Stronger Security Guarantees

III. PROPOSED SYSTEM

In this paper, we first propose a revocable multi-authority CP-ABE scheme, where an efficient and secure revocation method is proposed to solve the attribute revocation problem in the system. Our attribute revocation method is efficient in the sense that it incurs less communication cost and computation cost, and is secure in the sense that it can achieve both backward security (The revoked user cannot decrypt any new cipher text that requires the revoked attribute to decrypt)and forward security (The newly joined user can also decrypt the previously published ciphertexts1, if it has sufficient attributes). Our scheme does not require the server to be fully trusted, because the key update is enforced by each attribute authority not the server. Even if the server is not semi trusted in some scenarios, our scheme can still guarantee the backward security. Then, we apply our proposed revocable multi-authority CP-ABE scheme as the underlying techniques to construct the expressive and secure data access control scheme for multi-authority cloud storage systems.

Objectives:

- he main objective of the project is to design an Expressive, Efficient, and Revocable data access control scheme for multi-authority cloud storage systems, where there are multiple authorities co-exist and each authority is able to issue attribute independently.
- e modify the framework of the scheme and make it more practical to cloud storage systems, in which data owners are not involved in the key generation.
- We greatly improve the efficiency of the attribute revocation method.
- We also highly improve the expressiveness of our access control scheme, where we remove the limitation that each attribute can only appear at most once in a cipher text.

IV. ALGORITHMS

A. Key Policy Attribute-Based Encryption (KP-ABE)

KP-ABE [15] is a public key cryptography primitive for one-to-many communications. In KP-ABE, data are associated with attributes for each of which a public key component is defined. The encryptor associates the set of attributes to the message by encrypting it with the corresponding public key components. Each user is assigned an access structure which is usually defined as an access tree over data attributes, i.e., interior nodes of the access tree are threshold gates and leaf nodes are associated with attributes. User secret key is defined to reflect the access structure so that the user is able to decrypt a ciphertext if and only if the data attributes satisfy his access structure. A KP-ABE scheme is composed of four algorithms which can be defined as follows:

The setup algorithm takes no input other than the implicit security parameter. It outputs the public parameters PK and a master key MK.

The setup algorithm chooses a group G of prime order p and a generator g .

Step 1: A trusted authority generates a tuple $G=[p,G,G_1,g \in G,e] \leftarrow Gen(1^k)$.

Step 2: For each attribute a_i where $1 \le i \le n$, the authority generates random value $\{a_{i,t} \in Z^*_p\}$ $1 \le t \le n_i$ and computes $\{T_{i,t} = g^{a_{i,t}}\}$ $1 \le t \le n_i$

Step 3: Compute $Y= e(g,g)^{\alpha}$ where $\alpha \in Z^*_p$

Step 4: The public key PK consists of $[Y,p,G,G_1,e,\{\{T_{i,t}\}1 \le t \le n_i\}1 \le i \le n]$

The master key MK is $[\alpha, \{\{a_{i,t} \in Z^*_p\}1 \le t \le n_i\} 1 \le i \le n]$

B. Proxy Re-Encryption (PRE)

Proxy Re-Encryption (PRE) is a cryptographic primitive in which a semi-trusted proxy is able to convert a cipher text encrypted under Alice's public key into another cipher text that can be opened by Bob's private key without seeing the underlying plaintext. More formally, a PRE scheme allows the proxy, given the proxy re encryption key $rk_{a\leftrightarrow b}$, to translate cipher texts under public key pk_a into cipher texts under public key pk_b and vice versa.

IV. SYSTEM ARCHITECTURE

A revocable multi-authority CP-ABE scheme, to solve the attribute revocation problem in the system. This method is an efficient and secure revocation method. The attribute revocation method can efficiently achieve both forward security and backward security. In backward security scheme the revoked user cannot decrypt any new Cipher text that requires the revoked attribute to decrypt. In Forward security the newly joined user can also decrypt the previously published cipher texts, if it has sufficient attributes. Moreover, while updating the cipher texts, all the users need to hold only the latest secret key, rather than to keep records on all the previous secret keys.

We consider a data access control system in multi-authority cloud storage, as described in Figure1. There are five types of entities in the system: a certificate authority (CA), attribute authorities (AAs), data owners (owners), the cloud server (server) and data consumers (users).

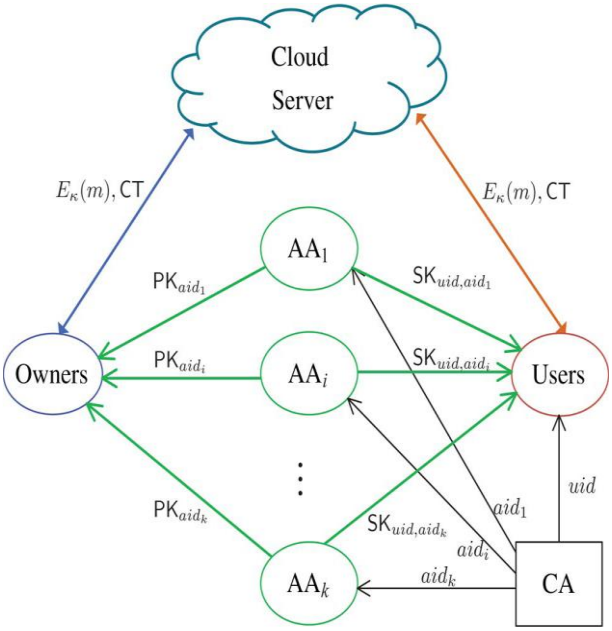

Figure1: System model of dataccess contro in multi-authority cloud storage.

Certificate Authority:

The CA is a global trusted certificate authority in the system. It sets up the system and accepts the registration of all the users and AAs in the system. For each legal user in the system, the CA assigns a global unique user identity to it and also generates a global public key for this user. However, the CA is not involved in any attribute management and the creation of secret keys that are associated with attributes. For example, the CA can be the Social Security Administration, an independent agency of the United States government. Each user will be issued a Social Security Number (SSN) as its global identity.

Attribute Authorities:

Every AA is an independent attribute authority that is responsible for entitling and revoking user's attributes according to their role or identity in its domain. In our scheme, every attribute is associated with a single AA, but each AA can manage an arbitrary number of attributes. Every AA has full control over the structure and semantics of its attributes. Each AA is responsible for generating a public attribute key for each attribute it manages and a secret key for each user reflecting his/her attributes.

Data Consumers:

Each user has a global identity in the system. A user may be entitled a set of attributes

which may come from multiple attribute authorities. The user will receive a secret key associated with its attributes entitled by the corresponding attribute authorities.

Data Owners:

Each owner first divides the data into several components according to the logic granularities and encrypts each data component with different content keys by using symmetric encryption techniques. Then, the owner defines the access policies over attributes from multiple attribute authorities and encrypts the content keys under the policies.

Cloud Server:

Then, the owner sends the encrypted data to the cloud server together with the cipher texts. They do not rely on the server to do data access control. But, the access control happens inside the cryptography. That is only when the user's attributes satisfy the access policy defined in the cipher text; the user is able to decrypt the cipher text. Thus, users with different attributes can decrypt different number of content keys and thus obtain different granularities of information from the same data.

V. OUR DATA ACCESS CONTROL SCHEME

In this section, we first give an overview of the challenges and techniques. Then, we propose the detailed construction of our access control scheme which consists of five phases: System Initialization, Key Generation, Data Encryption, Data Decryption and Attribute Revocation.

To design the data access control scheme for multi-authority cloud storage systems, the main challenging issue is to construct the underlying Revocable Multi-authority CP-ABE protocol. In Chase proposed a multi-authority CP-ABE protocol, however, it cannot be directly applied as the underlying techniques because of two main reasons:

1) Security Issue: Chase's multi-authority CP-ABE protocol allows the central authority to decrypt all the cipher texts, since it holds the master key of the system.
2) Revocation Issue: Chase's protocol does not support attribute revocation.

We propose a new revocable multi-authority CP-ABE protocol based on the single-authority CP-ABE proposed by Lewko and Waters . That is we extend it to multi-authority scenario and make it revocable. We apply the techniques in Chase's multi-authority CP-ABE protocol to tie together the secret keys generated by different authorities for the same user and prevent the collusion attack. Specifically, we separate the functionality of the authority into a global certificate authority (CA) and multiple attribute authorities (AAs). The CA sets up the system and accepts the registration of users and AAs in the system. It assigns a global user identity uid to each user and a global authority identity aid to each attribute authority in the system. Because the uid is globally unique in the system, secret keys issued by different AAs for the same uid can be tied together for decryption. Also, because each AA is associated with an aid, every attribute is distinguishable even though some AAs may issue the same attribute.

To deal with the security issue, instead of using the system unique public key (generated by the unique master key) to encrypt data, our scheme requires all attribute authorities to generate their own public keys and uses them to encrypt data together with the global public parameters. This prevent the certificate authority in our scheme from decrypting the cipher texts.

To solve the attribute revocation problem, we assign a version number for each attribute. When an attribute revocation happens, only those components associated with the revoked attribute in secret keys and cipher texts need to be updated. When an attribute of a user is revoked from its corresponding AA, the AA generates a new version key for this revoked attribute and generates an update key. With the update key, all the users, except the revoked user, who hold the revoked attributes can update its secret key (Backward Security). By using the update key, the components associated with the revoked attribute in the cipher text can also be updated to the current version. To improve the efficiency, we delegate the work load of cipher text update to the server by using the proxy reencryption method, such that the newly joined user is also able to decrypt the previously published data, which are encrypted with the previous public keys, if they have sufficient attributes (Forward Security). Moreover, by updating the cipher texts, all the users need to hold only the latest secret key, rather than to keep records on all the previous secret keys.

A cipher text policy attribute based encryption scheme consists of five fundamental algorithms: Setup, Key Generation, Encryption, Decryption and Attribute revocation.

Setup: The setup algorithm takes no input other than the implicit security parameter. It outputs the public parameters PK and a master key MK.

The setup algorithm chooses a group G of prime order p and a generator g .

Step 1: A trusted authority generates a tuple $G=[p,G,G_1,g \in G,e] \leftarrow Gen(1^k)$.

Step 2: For each attribute a_i where $1 \le i \le n$, the authority generates random value $\{a_{i,t} \in Z^*_p\}\ 1 \le t \le n_i$ and computes $\{T_{i,t} = g^{a_{i,t}}\}\ 1 \le t \le n_i$

Step 3: Compute $Y= e(g,g)^{\alpha}$ where $\alpha \in Z^*_p$

Step 4: The public key PK consists of $[Y,p,G,G_1,e,\{\{T_{i,t}\}1 \le t \le n_i\}1 \le i \le n]$

The master key MK is $[\alpha, \{\{a_{i,t} \in Z^*_p\}1 \le t \le n_i\}\ 1 \le i \le n]$

Key Generation (MK,S): The Key Generation algorithm takes master key MK and the attribute list of the user as input and do the following.

Let $L=[L_1,L_2.....L_n]=\{v_{1,t1}, v_{2,t2},,,,v_{n,tn}\}$ be the attribute list for the user who obtain the corresponding secret key.

Step 1: The trusted authority picks up random values $\lambda_i \in Z^*_p$ for $1 \le i \le n$ & $r \in Z^*_p$ and computes $D_0=g^{\alpha-r}$.

Step 2: For $1 \le i \le n$ the authority also computes $D_{i,1}, D_{i,2}=[g^{r+\lambda_i a_{i,t}}, g^{\lambda_i}]$ where

$L_i = v_{i,ti}$ The secret key is $[D_0\ D_{i,1}, D_{i,2}]$.

Encrypt (PK,A, M): The encryption algorithm takes as input the public parameters PK, a message M, and an access structure A over the universe of attributes. The algorithm will encrypt M and produce a ciphertext CT such that only a user that possesses a set of attributes that satisfies the access structure will be able to decrypt the message. Assume that the ciphertext implicitly contains A.

Step 1: Select $s \in Z^*_p$ and compute $C_0=g^s$ and $C^{\sim}=M.Y^s=M.e(g,g)^{\alpha s}$

Step 2: Set the root node of W to be s, mark all child nodes as un-assigned, and mark the root node assigned.

Recursively , for each un-assigned non leaf node, do the following

a) If the symbol is ^ and its child nodes are unassigned, we assign a random value $s_i, 1 \le s_i \le p-1$ and to the last child node assign the value.
$S_t = s - \sum s_i \bmod p$. Mark this node assigned.

b) If the symbol is V, set the values of each node to be s. Mark this node assigned.
Each leaf attribute a_i ,can take any possible multi values, the value of the share s_i is distributed to those value and compute.

c) Each leaf attribute a_i can take any possible multi values,the value of the share s_i is distributed to those values and compute.

$[C_{i,t,1}, C_{i,t,2}]= [g^{si}, T_{i,t}^{si}]$.The cipher text CT is

$[C,C_0,\{\{C_{i,t,1}, C_{i,t,2}\}1 \le t \le n_i\}1 \le i \le n\}]$

Decrypt(PK,CT,SK): The decryption algorithm takes as input the public parameters PK, a ciphertext CT, which contains an access policy A, and a private key SK, which is a private key for a set S of attributes. If the set S of attributes satisfies the access structure A then the algorithm will decrypt the ciphertext and return a message M.

The recipient tries to decrypt CT, without knowing the access policy W by using his SK_L associated with the attribute list L as follows

$$M = \frac{C^{\sim} \prod_{i=1}^{n} e(C_{i,t,2}, D_{i,2})}{e(C_0, D_0) \prod_{i=1}^{n} e(C_{i,t,1}, D_{i,1})}$$

Attribute revocation: A multi-authority Ciphertext-Policy Attribute-Based Encryption system with identity-based user revocation is comprised of the following algorithms:

Global Setup(λ)$\rightarrow$GP The global setup algorithm takes in the security parameter λ and outputs global parameters GP for the system.

Central Authority setup(GP)→(SK^*,PK^*) The central authority (CA) runs this algorithm with GP as input to produce its own secret key and public key pair,SK^*,PK*.

Identity KeyGen(GP,RL,GID,SK^*)→K^*_{GID} The central authority (CA) runs this algorithm upon a user request for identity secret key.It checks whether the request is valid and if yes(i.e.GID € RL),generates K^*_{GID} using the global parameters and the secret key of the CA.

Authority Setup(GP)→(PK,SK) Each attribute authority runs the authority setup algorithm with GP as input to produce its own secret key and public key pair,SK,PK.

KeyGen(GP,SK,GID,i)→$K_{i,GID}$ The attribute key generation algorithm takes in an identity GID, the global parameters, an attribute i belonging to some autjority, and the secret SK for this authority.It produces a key $k_{i,GID}$ for this attribute, identity pair.

Encrypt(GP,CT,(A,ρ),{PK},PK^*,RL)→CT The encryption algorithm takes in a message *M*, an access matrix(A,ρ),the set of public keys for relevant authorities ,the public key of the central authority, the revoked user list and the global parameters.It outputs a ciphertext CT.

Decrypt(GP,CT,(A,ρ),{$K_{i,GID}$},K^*_{GID},RL)→*M* The decryption algorithm takes in the global parameters, the revoked user list, the ciphertext, identity key and a collection of keys corresponding to attribute, identity pairs all with the same fixed identity GID,It outputs either the message *M* ,when the collection of attributes I satisfies the access matrix corresponding to the ciphertext. otherwise, decryption fails.

VI. CONCLUSION

In this paper, we proposed a revocable multi-authority CPABE scheme that can support efficient attribute revocation. Then, we constructed an effective data access control scheme for multi-authority cloud storage systems. We also proved that our scheme was provable secure in the random oracle model. The revocable multi-authority CPABE is a promising technique, which can be applied in any remote storage systems and online social networks etc.

FUTURE SCOPE

We identify the following future research directions for access control models in cloud computing environments:

1. Develop attribute-driven role-based access control models such that the user role and role-permission assignments be separately constructed using policies applied on the attributes of users, roles, the objects and the environment; and the attribute-based user-role and role-permission assignment rules be applied in real-time to enforce access control decisions.
2. Develop a location-aware role-based control model incorporated to the Policy Enforcement Point of a cloud (thereby, preventing the disclosure of user's identity, role, or location directly to a remote server in the cloud that may not be fully trusted), and enable/activate the role only when the user is located within the logical positions (computed from real positions by specific mapping functions) that lie within the spatial boundary of a role.

REFERENCES

[1] Chase and S.S.M. Chow, ''Improving Privacy and Security in Multi-Authority Attribute-Based Encryption,'' in Proc. 16th ACM Conf. Computer and Comm. Security (CCS'09), 2009, pp. 121-130.

[2] A.B. Lewko and B. Waters, ''Decentralizing Attribute-Based Encryption,'' in Proc. Advances in Cryptology-EUROCRYPT'11, 2011, pp. 568-588.

[3] S. Yu, C. Wang, K. Ren, and W. Lou, ''Attribute Based Data Sharing with Attribute Revocation,'' in Proc. 5th ACM Symp. Information, Computer and Comm. Security (ASIACCS'10), 2010, pp. 261-270.

[4] M. Li, S. Yu, Y. Zheng, K. Ren, andW. Lou, ‘‘Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute-Based Encryption,’’ IEEE Trans. Parallel Distributed Systems, vol. 24, no. 1, pp. 131-143, Jan. 2013.

[5] J. Hur and D.K. Noh, ‘‘Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems,’’ IEEE Trans. Parallel Distributed Systems, vol. 22, no. 7, pp. 1214-1221, July 2011.

[6] S. Jahid, P. Mittal, and N. Borisov, ‘‘Easier: Encryption-Based Access Control in Social Networks with Efficient Revocation,’’ in Proc. 6th ACM Symp. Information, Computer and Comm. Security (ASIACCS’11), 2011, pp. 411-415.

[7] S. Ruj, A. Nayak, and I. Stojmenovic, ‘‘DACC: Distributed Access Control in Clouds,’’ in Proc. 10th IEEE Int’l Conf. TrustCom, 2011, pp. 91-98.

[8] K. Yang and X. Jia, ‘‘Attribute-Based Access Control for Multi-Authority Systems in Cloud Storage,’’ in Proc. 32th IEEE Int’l Conf. Distributed Computing Systems (ICDCS’12), 2012, pp. 1-10.

[9] Boneh and M.K. Franklin, ‘‘Identity-Based Encryption from the Weil Pairing,’’ in Proc. 21st Ann. Int’l Cryptology Conf.: Advances in Cryptology - CRYPTO’01, 2001, pp. 213-229.