**Organization of the protection mechanisms in the cloud storage**

Tatiana Galibus

Viktor Krasnoproshin

Belarusian State University  
220030 Minsk, 4, Nezavisimosti   
Minsk – Belarus.  
+375 017 226-59-40

{tan2tan@gmail.com, [krasnoproshin}@bsu.by](mailto:krasnoproshin%7d@bsu.by)

Thiago P.B. Vieira

Rafael Timóteo de Sousa Júnior

João Paulo C. L. Costa

University of Brasilia  
UnB - FT – ENE – CP: 4386  
70910-900, Brasília - DF - Brazil  
+ 55 61 3107-5580

joaopaulo.dacosta@ene.unb.br, desousa@unb.br

Edison Pignaton de Freitas

Federal University of Rio Grande do Sul  
UFRGS – INF – CP: 15064  
91501-970 Porto Alegre - RS - Brazil   
+55 51 3308 6168

{jcsanjos, edison.pignaton, geyer}@inf.ufrgs.br

Anton Zaleski

H.E.R.M. Vissia

Byelex Multimedia Products BV Argon 1, 4751 XC Oud Gastel, The Netherlands

{ a.zalesky@by.byelex.com, [h.vissia@byelex.com](mailto:h.vissia@byelex.com)}

ABSTRACT: Providing effective security mechanisms is an important requirement to any secure cloud solution. In this sense, we propose to use the hybrid approach to cloud security that includes both warning and detection mechanisms in order to minimize the possibility of successful attack. According to our proposal, the customized attribute-based encryption is a comprehensive access control solution for cloud storage including user accountability and key revocation. We apply the state-of-the-art signal processing techniques in order to detect the malicious activities and MITC-attacks in the cloud environment.

**Introduction**

Recently, many organizations and enterprises shift to a cloud infrastructure, contributing to a global transition to a distributed system paradigm. The cloud services make the data accessible for multiple users and the concept of access control in the cloud data storage should be carefully considered both by providers and end-users/organizations. There are numerous attacks on the cloud systems and the one most significant in the protected cloud environment is called “Man in the Cloud” (MITC). In the typical scenario of such attack, the hacker steals the user credentials (a token or password), and uses this data in order to substitute or to steal the protected data.

There are two basic approaches to address this attack: the cryptographic/key-based mechanisms that work as a precaution against it by means encryption and secret sharing, and the data collection/traffic analysis mechanisms that allow to detect the attack and to prevent it from being successful as fast as possible.

The main difficulty of relying solely on the cryptographic methods is that the attacks are often based on the reverse or social engineering and, therefore, the majority of the attack scenarios cannot be handled completely.

We propose a novel approach to the cloud security based on the hybrid protection system. First we apply the proactive attribute-based-encryption in order to protect the access to the protected cloud and we adopt signal processing techniques similarly to the honeypot detection system in order to provide an immediate alarm system for the rapid identification of the MITC-attacks. The system monitors the activity of the mobile device users and warns the domain administrator in the case of suspicious actions of the user/device so that he can takes the appropriate decision to restrict the access of the dangerous/untrusted device or remove it from the domain.

**The components of the cloud storage security**

The proposed infrastructure for the protected cloud includes the following components illustrated in Figure 1.

Figure 1: Components of the proposed security infrastructure and their interactions

1) Encryption server. In Figure 1, the encryption server manages all AC and encryption operations and grants the user access to the data storage. This server can store the encryption keys and/or connect to a separate Key Storage.

2) File storage. The file storage is secure in the sense that some of the files specified by the domain administrator are store, encrypted and have restricted access as show in Figure 1. Due to the fact that the file storage data is partially stored in the cloud i.e. externally it is recommended to encrypt this external part of file storage completely.

3) Client UI. The client can connect to the encryption server and ask for the permission to access the file storage in order to view/edit/upload specific files or folders.

4) Key storage. The key storage is accessible only for the domain administrator.

5) Attack detection system. The attack detection system monitors the mobile device activity and detects the threats and suspicious actions.

In order to increase the speed of data encryption on-the-fly, a hybrid encryption system is set up which is combined of both symmetric and attribute-based (asymmetric) encryption. More importantly, the basic ABE [8] , [9] approach is modified in order to increase the speed of encryption and implemented the configuration parameters in order to set up the user key expiry dates and more sophisticated attribute sets corresponding both to the file shares and to the user groups.

The basic functionality of the security components are briefly described in the following:

1. File storage: The bulk data in the protected file storage is encrypted with the appropriate block cypher (AES, Blowfish, IDEA, Serpent). The key to the encrypted data is stored in the key storage and has expiry period in order to increase the protection.

2. Key storage: The symmetric keys for the data in the file storage are kept in the separate storage. The protection of the key storage is implemented via some strong authentication method i.e. two-factor authentication. Additionally, administrator can set up the key expiry period, use the separate keys for the separate files and/or apply the secret sharing mechanism in order to store the keys for the most important sensitive data.

3. Encryption server: The most important cryptographic services are run on the Encryption server. This server generates the user keys and connects to the client UI, i.e. a separate user of the system and decides whether the access to the specific dataset should be granted to this user. In addition, the server runs the key renewal routines, stores the user public keys and attributes besides the auditing data.

4. Client UI: The client UI connects to the encryption server and checks the expiry period of the user keys (in the case it has been configured) and permits the device user to view/edit/upload the data. Client UI stores the user keys for the ABE encryption and the unique symmetric session keys which serve for restricting the access to the downloaded files. The symmetric keys are encrypted with the ABE keys. The client allows the whole system to work in the heterogeneous environment as it supports different platforms and operating systems.

5. Attack detection system. The client requests are processed via the detection system, which collects the request time/frequency/amount data and analyzes it in order to detect the malicious user activity.

The described modular infrastructure allows setting up different components separately and configuring the security system according to specific needs of the enterprise/organization. The main purpose of the proposed security infrastructure is the setup of the efficient access control and attack detection in the cloud-based protected environment.

**The access control system**

The important component of the proposed security infrastructure is the setup of the access control in the cloud storage. Access control mechanisms serve to run the following tasks:

• Authentication of users in the cloud system: The initial and the simplest authentication is performed by the user password and email id. For the highly sensitive data it is necessary to implement more sophisticated two-factor authentication when apart from the password and access to e-mail, the possession of a specific device is verified too. This can be used in the government services or services with the highly sensitive data.

• Provision of access control functions and protection of data from the unauthorized access: The AC services are run by the encryption server. The server generates and distributes the user keys and keeps the group attributes along with the file sharing ids. Access control allows to securely distribute and show to the user (or accept from user) only the data he is permitted to view/edit. In order to achieve this protection, a special version of ABE is used with the implementation of both possible policies: key policy – KP, and cyphertext policy – CP, in order to support the attributes of the groups of users as well as the attributes of the file shares. This algorithm is developed specifically for the access structure of the proposed cloud architecture.

• Protection of of the user data privacy: Once the user wishes to access a separate file downloaded on user device the client uses his/her ABE key after performing the authentication in order to decrypt the symmetric session key and open the file.

The access control in the protected cloud storage is based on the selective ABE encryption. The ABE encryption allows to set up user attributes, corresponding to the set of the access identificators of the user groups Group1, Group2, ..., Group*n*:

...

The hash-value of the open text is specified by M. Additionally, there are user attributes and the set of attributes of the encrypted text . If at least one attribute in the set is equal to the attribute in the set , the corresponding user U can decrypt the text M.

The key generation and encryption is implemented on the server side, while the the decryption is performed on the client side. The additional parameters of the ABE encryption allow to implement the key revocation and renewal.

**The client workflow**

The additional alarm protection system based on the signal processing techniques is correlated with the client workflow. Below we describe the typical activities of the client. The most important feature of the client-side encryption is the uniqueness and unchangeability of the session key to access the separate shares (files) which allows to access the share in the off-line mode but, of course, can compromise security. To address this problem, we provide session key encryption by means of modified ABE with the key expiry period. Thus, we implement a hybrid encryption system:

* File encryption: A unique symmetric key is generated for each file. Then, the file is encrypted with AES-128 using this unique key. Encrypted file is sent to the client (Iphone, Ipad or Android device). The AES key is encrypted with modified ABE with respect to the file sharing participants list and sent to the client to be stored locally.
* File decryption: The client checks if the private key of a user is still valid. If so, the file key is decrypted with his appropriate ABE key. If the file key is decrypted successfully then the file is decrypted with the AES-128. ABE serves for the access control polytics to be preserved and does not allow the access to the files for the unauthorized users.

The typical actions of the user are as follows:

1. The user starts-up the cloud storage client application, enters the PIN code and opens the domain. Meanwhile, the login procedure receives the actual user ABE private key.

2. User selects the files for synchronization with the server. This means, that encrypted files are saved to the device storage. A user can see them in his file browser, but these files are encrypted and cannot be used directly.

3. User clicks via the context menu/button “Decrypt”. The file will be decrypted, and directly opened with the application that is associated with the appropriate extension. In the background the application checks the private key validity, decrypts the file and saves the non-encrypted file in the file system.

4. User modifies the decrypted files and saves them locally. When the file is modified it will be uploaded to the server and will be encrypted again. It is the responsibility of the user to save the file back to the globally controlled storage and not to save it anywhere else. When the user saves the file, it is sent to the encryption server and encrypted.

5. User synchronizes with the global storage in order to save the modifications if allowed.

**The attack scenario**

Regarding the private keys of the ABE encryption, they are protected by the following mechanisms: uniqueness, revocation, centralized generation. Therefore, the hacker cannot do much even if he steals the whole set of the keys. He needs to have the access to the device itself, and is able to have this acces only within the key expiry period. He cannot substitute these keys as it is performed in the classical MITS attack.

The most insecure scenario for the proposed system is when the hacker steals the permanent user credentials, namely, password and login. He can do it by means of social engineering. In order to protect the user from this threat we propose to use such cautionary methods as hashing/secret sharing or PBE/ PAKE protocols.

Also, to increase the security we propose to install the attack detection system, which allows to quicly discover the malicious activity based on the user actions analysis. This system functionality is based on the typical behaviour of the hacker after stealing the credentials:

1. Downloading user files randomly
2. Modifying and uploading files randomly or massively
3. Repeatedly downloading the files that are already on the device

All these types of activities can be detected with the signal processing techniques. We propose to analyze the following info:

1. Number of files uploaded/downloaded
2. Time delay between downloading the files (if it is too short – this is, obviously, a bot, not a real user)
3. Number of files re-downloaded/uploaded – normally the user keeps the files on his device and doesn’t need to refresh it for some time.

We propose to extract the info from the log in the following form:

1. Apply feature selection techniques to identify valuable features;
2. Using Moder Order Selection schemes for abnormalities detection;

Enriching the attack detection through techniques for obtaining detailed information about attacks.**The log analysis algorithm**

Most of the attacks incurs into significant variation on the standard behavior of information systems or adotps well-known signatures that can be easily detected by monitoring system. Intrusion detection and intrusion prevention systems are security systems used respectively to detect (passively) and prevent (proactively) threats to computer systems and computer networks. Such systems can work in the following fashions: signature-based, anomaly-based or hybrid [3, 8].

Signal processing techniques have been successfully applied to network anomaly detection [2,7] and have been a research problem in order to achieve improvements on detection accuracy and computational cost.

In the context of anomaly-based schemes, the proposed log analysis algorithm applies signal processing techniques, such as Principal Component Analyis and Model Order Selection schemes, for automatic identification of attacks or malicious behaviors. Additionally, other techniques can be used to obtain detailed information about the malicious behavior, making possible to identify patterns and obtain the necessary information for performing reactive and proactive actions against possible threats.

Therefore, the desired information is extracted from the collected log, in order to obtain useful features that shall be modeled as matrices that represents a signal superposition containing noise, legitimate and malicious traffic.

From the extracted features shall be performed the behavioral evaluation for identification of abnormalities over time, such as outstanding abnormalities or less expressive variations on the observed behavior. For this analysis we adopts the eigenvalue analysis based on covariance and correlation, for highlight behavior changing that shall be used as input for attack detection through Model Order Selection schemes.

The selected Model Order Selection scheme detects the attack occurrences, that can be enriched by tecniques to extract detailed information of the detected attack. For detailed information extraction and attack identification we apply eigen analysis and similarity analysis for obtaining detailed information about accurate time and attacker identification.

**Conclusion**

In this paper, we propose a novel set of access control and attack detection mechanisms to be used for the organization/enterprise cloud storage protection. This set of mechanisms is rather flexible and allows solving various complex security problems including the attack on the user login/password and unauthorized access to the data. Currently, the system is implemented as the commercial software and being tested.