Cloud Access Secure with Identity Based Cryptography

Houria Hamadi(✉), Abdelkrim Khireddine, and Abdelkamel Tari

University of A/Mira, Bejaia, Algeria
{hamadi.houria,abdelkrim.khired,tarikamal59}@gmail.com

Abstract. With the fast evolution of networks and Internet, appeared the concept of the Cloud, the requirements in terms of safety become more and more essential. This requires the introduction of advanced authentication methods, access control and identity management, while respecting the constraints of the services offered by this evolution, such as data exchange capacity and resources of terminals. The aim of our work is to propose cloud security architecture to be able to establish secure sessions between the nodes of a cluster in the cloud, and allow to different users a secure access to their data.

Keywords: Virtual networks · Cloud · Security · Authentication · Access management · Identity management · Identity-based cryptography

1 Introduction

In the classic computer systems, companies, operators, service providers put up computer parks to host their data and activate their services. But today, Cloud Computing represents an economic and practical alternative for these entities. Just like the electric power one century ago, the computing power and storage of information would be proposed for consumption by specialized companies. Therefore, companies would no need more appropriate servers, but would entrust this resources and services to Cloud providers using a set of physical resources (servers) organized within a cluster to provide, in the request, resources and services to users. The Cloud became today an effective solution to optimize the use of the physical resources and reduce their costs.

Cloud computing is a technology with vast impact on computer systems. The costs can be significantly reduced through the purchase at the request of CPU time, memory and storage, offering a great flexibility. However, the opening of systems and sharing of associated resources create many problems of security, which remains one of the major barriers for the adoption of these technologies [1]. In addition, cloud providers secure their systems only against external enemies using firewalls and secure connections. Ignoring the internal opponents, who represent a big threat.

S. Boumerdassi et al. (Eds.): MSPN 2015, LNCS 9395, pp. 109–118, 2015.
DOI: 10.1007/978-3-319-25744-0_10

Identities constitute a key element of safety. This information must be correct and available to all elements of cloud witch needs to validate access. Access control bases on the identity information to allow and constrain the access to cloud functioning and the underlying infrastructure. According to reports published by groups of protection of personal information supervisory in the United States, more than 148 incidents of identity theft, affecting about 94 millions identities, were held in 2005 in the United States alone (Mark, 2006). Identity theft is one of the most serious threats to online services security. Therefore, it is irresistible that SaaS providers have the means to authenticate the identity of each user trying to access to the system.

In order to improve the security of cloud infrastructure, security of identities and limiting access to authorized users, and by basing on the work already done and re-use of already existing solutions for other type of networks which we will adopt for virtual networks and reduce operating costs. Our purpose in this article is to define a system of access control to filter users wishing to connect, and preserve access to critical user's data while protecting their identities using a system of protection based on the identity-based cryptography and smart cards.

This paper is organized as follows: introduction, the second and third sections present an overview of existing works and definitions of various basic elements which will be used in this approche. Section 4 approaches the model of the solution and the proposed architecture. Section 5 details some security evaluations of our solution. Finally, Sect. 6 concludes and proposes perspectives of our work.

2 Basic Notions

2.1 Cloud Computing

Cloud computing is a model that allows quick access and request for a pool of shared computing resources (servers, storage, applications, bandwidth, etc.) without strong interaction with service provider. This definition of the NIST (National Institute of Standards and Technology) is widely taken to define the Cloud. Several major players like Microsoft and Google already propose different Cloud solutions. Companies hope to have more efficiency and reduce costs if they can access to an online service, and not manage their ICT (Information and Communication Technologies). Cloud brings, beyond the added value technology such as scalability, performance, etc., cost reduction and flexibility (to develop efficiency). In the following we are going to describe the cloud architecture that will be used in our work:

- **Cloud Service Provider:** the provider of cloud services, he provides to users and customers virtual infrastructure so that his customers can access to their storage spaces, and services requested and allocated.
- **Client:** is a user of the services offered by a cloud provider, he has access to the resources allocated, and has the rights and permissions on all of its data and services. A customer can be a single individual or a company. Customers can communicate between them and share data.

- **Other Users:** who are not necessarily customers in a cloud provider, but can access to the data stored in the cloud by a client, and having a number of authorizations defined by owners of the data (Fig. 1).

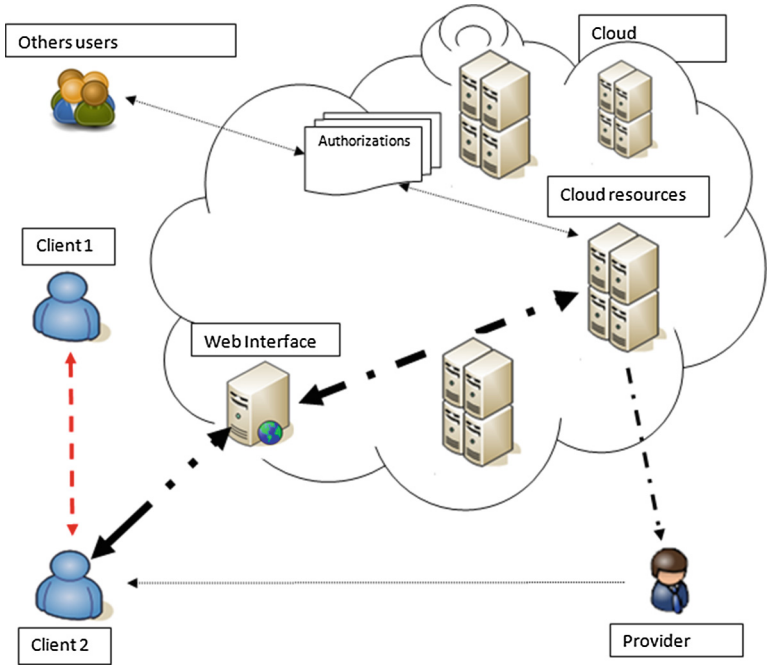

Fig. 1. Cloud architecture.

2.2 Identity-Based Cryptography

The concept of "identity-based cryptography" was introduced by Adi Shamir in 1984 [2]. Contrary to the conventional schemes, schemes based identity (IBC), offer the possibility of choosing freely the public key. Electronic certificates, known for their complexity, are replaced with easily remembered information, such as email address, IP address, such as public key. As far as the identity-based schemes do not require certification mechanism, they allow to simplify considerably the implementation of secure communication systems.

By their construction, plans based on identity (IBC) manage to solve the authentication problems, management and revocation public keys, however, they require the presence of an ultra-powerful authority PKG (Private key Generator), who provides to each user the corresponding private key to its public key. The confidence granted to this authority must be without defect, because it is

inherently capable of regenerating the private key of any user, and therefore able to perform unauthorized signatures or decryption.

During the last decade, IBC has been improved by the use of Elliptic Curve Cryptography [3]. Consequently, the new identity based encryption and signature scheme appeared. To be able to take a client's private key, the PKG must first define a set of identifiers based on public characteristics. PKG product groups G_1, G_2 up to G_T and the coupling function ê of $G_1 \times G_2$ in G_T. G_1 and G_2 are additive subgroups of the group of points of an elliptic curve.

However, G_T is a multiplicative subgroup of a finite domain. G_1, G_2 and G_T have the same ordre q. Moreover, G_1, G_2 and G_T are generated by P, Q and the generator $g = \hat{e}\ (P, Q)$. The bilinear function ê is fired from the Weil [4].

After the specification of the groups, the PKG defines a set of hash functions in accordance with the IBC and the signature scheme used [5]. As such, the PKG defines a hash function $Hash_{pub}()$ to transform the client's identity (ID) into a public key as follows: $Pub_{ID} = Hash_{pub}(ID)$.

Generally, the public key of a customer is calculated as a hash of one of its identities which is a point of an elliptic curve [6] or a positive integer [7]. The PKG generates the private key of an entity using a local secret $S_{PKG} \in Z^*$ and a private key generation function $Priv_{Gen}()$. Note that the private key is calculated as follows: $Priv_{ID} = Priv_{Gen}(S_{PKG}, Pub_{ID})$.

3 Related Works

In the last years, cloud security and virtual environments became a topic attracting who research. Several studies have addressed the various problems of security and so proposing different solutions. To develop and improve the standard authentication with password [8] various works were realized to be in order to solve new security problems in a cloud environment, the use of identity-based cryptography (IBC) stays a very effective way to make sure of the identity of the user and be able to manage the identities of a number of users who increases more and more. With the development of internet technology to the cloud, access and identity management became crucial for the safety, and she allows limiting access to data and applications to the only authorized users.

Since their introduction, schemes based identity, have been the subject of intensive researches, yet it will have been necessary to wait 2001, and the works of Cocks [9], Boneh and Franklin [6], to have crypto systems that reply to conditions of safety and efficiency. The identity-based cryptography, is a new occurrence and an interesting domain for the security of virtual networks. IBC has was adapted for grids networks, this idea was explored by Lim and Robshaw in 2004 [10]. In their proposal, each virtual organization has its own PKG and all users share the same public characteristics certified by the network certification authority.

In 2005, Lim and Robshaw [11] explain a new concept of dynamic key infrastructure for the grid in order to simplify the management of keys already made in [10]. That is to say, every user takes care of the construction of its public characteristics and its distribution to other entities. It is in 2005, that Lim and

Paterson [12] suggest using IBC to secure grid environments. They describe several scenarios where IBC simplifies many cases, as the elimination of the use of the certificate, and the savings of bandwidth while using the approach proposed by Boneh and Franklin [10].

H. Li, Y. Dai, L. Tian, and H. Yang, [13] propose to use IBC as an alternative to SSL (Secure Sockets Layer)authentication protocol in the cloud, however, these solutions still suffer from necessary fiduciary hierarchy to assure a safe working system. [14] Presented a new security infrastructure by using the IBC for applications oriented Cloud services. In their proposal, the service URL is used to generate the public key.

4 Model and Architecture

The choice of IBC is motivated by several reasons. At first, IBC will allow us to take advantage of its simple keys management mechanism, which does not require the deployment of a public key infrastructure (PKI). Secondly, IBC allows generating public keys without needing for a prior calculation of corresponding private keys. That is, contrary to traditional patterns public key generation, IBC does not require to generate the private key before the public key. Indeed, customers need only first, generate public key-based characteristics that constitute their identities to define their public keys lists. Finally, for each new discussion, a different public key will be used to encrypt messages so the client generates and uses a new private key to decrypt, that will prevent us the use of the same key to encrypt or decrypt the discussions.

The objective of this work is to design and develop security architecture for the cloud. The proposed architecture will allow the management of identities, access and the security of communications between nodes in a cluster in Cloud. Our idea consist in naming every customer as a PKG that generates its own private keys, from a secret S_C, recovered after authentication with the PKG, and one of its public key Pub_C. These private keys used to authenticate, encrypt and decrypt messages exchanged between the client and the various nodes of the cluster. To define our architecture, we have to identify the different actors. In our context, the actors are the basic entities that our solution will count (provider, customer,...). We define the various entities as follows:

The provider, customer and other customers are already defined in the previous section: Basics notions.

- **PKG:** is an authority that authenticates and generates secret keys for all entities of the cluster. Each delivered key must have issued a validity date in order to manage and eliminate nodes which do not belong any more to the cluster. Every node that does not re-authenticate in a specific date its secret key will be invalid, and it cannot generate private keys. It also has a secret key and a public key list, that broadcast regularly in the network for new clients to have a way to communicate it.
- **Index Server:** Server in the cloud where are stored indexes that define the location of each data stored in a storage server.

- **Indexes Database:** contains all indexes of stored data. A batch of data stored or identifier may be designated by multiple indexes.
- **Storage Server:** it is a storage space where the data are stored. Each space memory where data is stored and saved under a code that we are going to name Index. Each lot of data saved by a client must have an identifier (keyword) that will be indicated when searching.

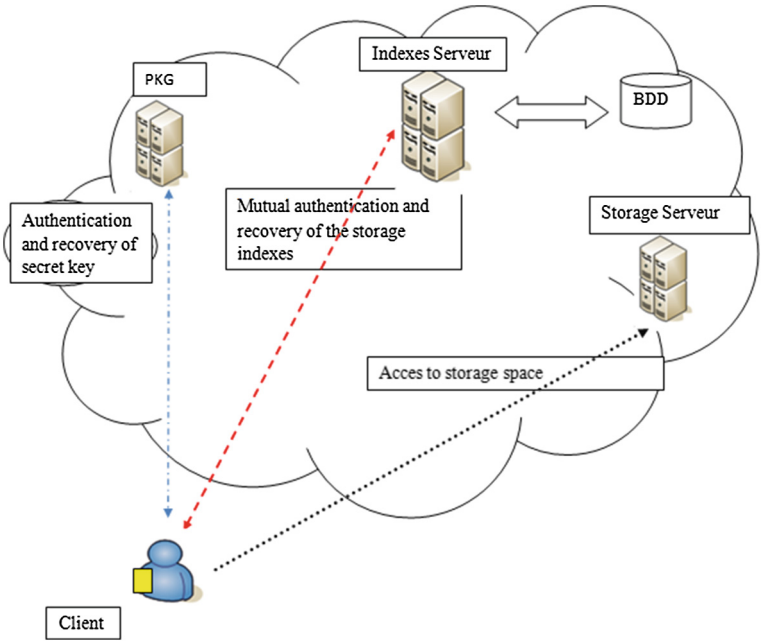

Fig. 2. Proposed architecture.

4.1 Prerequisites

In this section we will define the prerequisites used to design our model. First, each client generates its own list of public keys pub_C, that will be integrated in a certificate, signed by a certification authority, to validate their identity. This list will be used and distributed so this client is known within its cluster (Fig. 2).

We suppose as well, that the customer have already a registration and authentication with the provider, who granted him different access to its allocated services. Each customer holds a smart card which will allow him to protect its list of public keys as well as its identifiers of data stored in the storage servers. The choice of smart card came to avoid to the customer the memorization and loss of its important informations.

Note also, that the channel established between the client and the PKG is secured. This channel supports the mutual authentication, integrity and confidentiality of data exchanged. TLS (Transport Layer Security) [15], is chosen as it is among the best protocols which ensure secure transmission of messages.

Once the client is connected to a web space to reach his cluster, the first thing to do is to distribute its public key list on the network so that other nodes can connect with him. The client begins by mutual authentication with the PKG, always based IBC. Once authenticated, the customer request for secret key, PKG generates the secret key with a validity date. What will allow PKG to require the authentication of customer every time to remove inactive clients and release the cluster. PKG sends the secret key and the algorithm for generating private keys. The client's secret key will be stored in the smart card (Fig. 3).

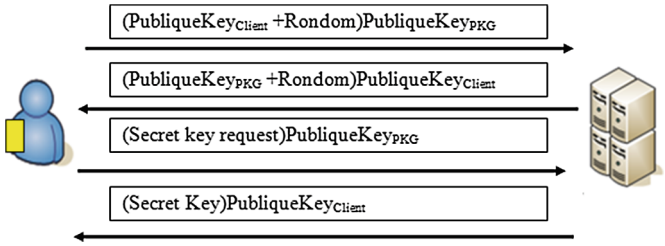

Fig. 3. Recovery of the secret key.

Once he had his secret key, the customer can create himself its private keys with its public keys. Which will allow it to communicate with the server indexes to have access to its storage space. After a mutual authentication with the server indexes, the client sends a request with the list of data identifiers which it wants to access. The server indexes side verifies the existence of identifiers, sent by the client, in its database. If identifiers exist well, he gets back the list of appropriate indexes, that will allow the customer to have direct access to the storage server. Otherwise if the client has no access to the data it asked, the server indexes sends unauthorized access.

Our solution provides another advantage to the customer, which is the encryption of their data before saving. When the customer decides to take out of his storage space, the server gives him an opportunity to encrypt data with a key of his choice, thus, it will be the only person who holds the key for encryption and decryption. And as the client already uses identity-based cryptography then the choice of encryption its data will be based on IBC (Fig. 4).

5 Security Evaluation

In this section, we present an informal security analysis of our proposal. In addition, we express the possible refinements to limit other threats.

- **Data Privacy:** In our approach, the customer is responsible for encrypting data before storing them in the cloud. As the client acts as a PK authority, he

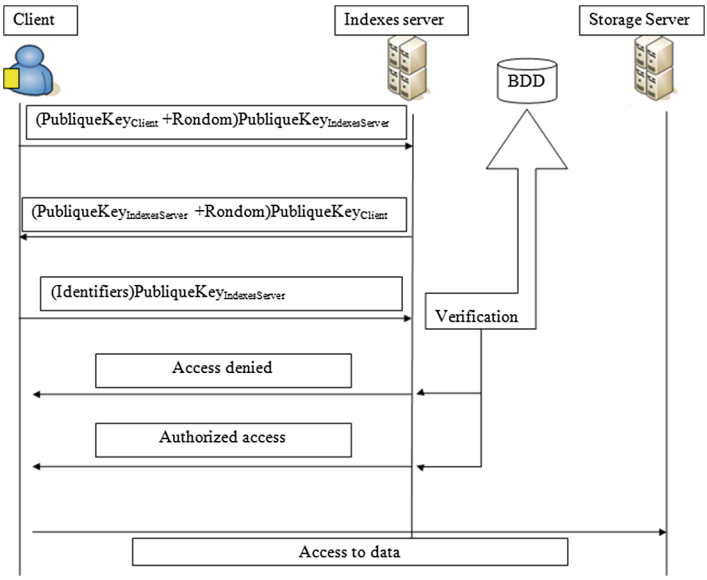

Fig. 4. Schema of access to storage servers.

is able to manage its secrets and generate private keys. So, it is the only entity that knows the secret S_C, necessary to generate any decryption key. Thus, it is impossible for the provider or an attacker to recover the decryption key to encrypt data.

- **Control Data Access:** through our security model, data access is limited to authorized clients. IBC allows customers to protect their identity, preserve access to its data and even access to messages exchanged between the different nodes of the cluster to which it belongs. Even if an attacker has a list of client's public keys and tries to steal his identity, he cannot decrypt the messages which intended for him because he has not the secret key that will allow him to generate private keys. Secondly, although the supplier or a malicious user can obtain the access to data, we always guarantee the data privacy, as they can only have access to encrypted data. And they have no private key to decrypt the data.
- **Key Management:** cryptography based identity suffers attacks against the deposit of encryption keys, that is the PKG. However, our solution declines this problem, because every customers acts as a PKG for his own data and encryption and decryption keys. In addition, the secret key is saved in smart card, which makes it the mission impossible for an attacker to generate private keys, or even find them.

Finally, we are anxious to explain that our solution, limits some attacks and provides maximum protection of the private life and the information of the users of the Cloud.

6 Conclusion and Perspectives

The increasing need to secure access to sensitive data and to avoid the theft of identities in the cloud became a challenge for researchers in the field of security. In this article, we presented architecture of authentication for the cloud, with identity-based cryptography (IBC). Our solution is based on a particular method of IBC. First, each client is named as a PKG and takes care of the creation of his list of public keys, the generation of his private keys if necessary, and the secret key backup. The smart card comes to facilitate the job for the customer to properly protect its secret key and its various meta data. Our method also offers the user a chance to encrypt critical data as well as its meta data to avoid those who offers more privacy to its information.

We are anxious to make different tests security while simulating some attacks which aim authentication, client access safety and capacity of the authentication server to support different requests of authentications, and a comparative study between the different methods of encryption in performance time and CPU usage. Finally we presume that access security and data storage on the cloud is always full of challenges and of paramount importance and several research problems remain to be identified.

References

1. Cloud Security Alliance. Top Threats To Cloud Computing. Technical report, March 2010. http://www.Cloudsecurityalliance.org/topthreats.html
2. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (1985)
3. Hankerson, D., Menezes, A.J., Vanstone, S.: Guide to Elliptic Curve Cryptography. Springer-Verlag New York, Inc., Secaucus (2003)
4. Blake, I., Seroussi, G., Smart, N., Cassels, J.W.S.: Advances in Elliptic Curve Cryptography (London Mathematical Society Lecture Note Series). Cambridge University Press, New York (2005)
5. Ratna, D., Rana, B., Palash, S.: Pairing-based cryptographic protocols: A survey (2004). http://eprint.iacr.org/
6. Boneh, D., Franklin, M.: Identity-based encryption from the weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001). http://dl.acm.org/citation.cfm?id=646766.704155
7. Sakai, R., Kasahara, M.: Id based cryptosystems with pairing on elliptic curve, Cryptology ePrint Archive, Report 2003/054 (2003). http://eprint.iacr.org/
8. Secure Password by Using Two Factor Authentication in Cloud Computing, Ali A. Yassin, Hai Jin, Ayad Ibrahim, Weizhong Qiang, and Deqing Zou, Services Computing Technology and System, Lab Cluster and Grid Computing, Lab School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, 430074, China
9. Cocks, C.: An identity based encryption scheme based on quadratic residues. In: Honary, B. (ed.) Cryptography and Coding 2001. LNCS, vol. 2260, pp. 360–363. Springer, Heidelberg (2001)

10. Lim, H.W., Robshaw, M.J.B.: On identity-based cryptography and grid computing. Lecture Notes in Computer Science, pp. 474–477 (2004). http://www.springerlink.com/content/ylj95gfgjxlb2131
11. Lim, H.W., Robshaw, M.: A dynamic key infrastructure for GRID. In: Sloot, P.M.A., Hoekstra, A.G., Priol, T., Reinefeld, A., Bubak, M. (eds.) EGC 2005. LNCS, vol. 3470, pp. 255–264. Springer, Heidelberg (2005)
12. Lim, H.W., Paterson, K.G.: Identity-based cryptography for grid security. Int. J. Inf. Secur. **10**(1), 15–32 (2011). http://dx.doi.org/10.1007/s10207-010-0116-z
13. Li, H., Dai, Y., Tian, L., Yang, H.: Identity-based authentication for cloud computing. In: Jaatun, M.G., Zhao, G., Rong, C. (eds.) Cloud Computing. LNCS, vol. 5931, pp. 157–166. Springer, Heidelberg (2009)
14. Schridde, C., Dörnemann, T., Juhnke, E., Smith, M., Freisleben, B.: An identity-based security infrastructure for cloud environments. In: Proceedings of IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS2010) (2010)
15. Dierks, T., Rescorla, E.: RFC 5246 - The Transport Layer Security (TLS) Protocol Version 1.2, Technical report, August 2008. http://tools.ietf.org/html/rfc5246