Indian Journal of Science and Technology, Vol 8(25), DOI: 10.17485/ijst/2015/v8i25/80241, October 2015
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Integrated User Authentication Method using BAC(Brokerage Authentication Center) in Multi-clouds

Jeong-Hee Choi[1], Sang-Ho Lee[1*] and Mi-Kyoung Kim[2]

[1]Department of Software, Chungbuk National University, Korea; heebest75@gmail.com
[2]IT Convergence Technology Research Lab, ETRI, Korea; yoohee@etri.re.kr

Abstract

The problems (lack of storage, system trouble, problem of power supply etc.) in a single cloud with increase the use of clouds are being raised. For these reasons, multi-clouds are needed. In multi-clouds, Cloud providers as different forms of systems with different authentication systems demand an authentication to a user. So users using multi-clouds have different authentication ID and Password for each cloud because they have to authenticate differently for each cloud. As a result, users are inconvenient to manage multiple of authentication IDs and Passwords and users' authentication information exposed whenever they access to multi-clouds is in a poor security. Integrated user authentication is needed to ensure an authentication of user's safety and convenience in multi-clouds. So this paper suggests the user focused integrated authentication methods in multi-clouds environment using an authentication ticket issued from Brokerage Authentication Center (BAC).

Keywords: Authentication, Cloud-of-Clouds, Intercloud, Multi-Clouds, Security

1. Introduction

Various forms of cloud services are appearing with developing of a cloud computing technology. Accordingly, users using a cloud want to receive services from various forms of clouds instead of receiving a service from a single cloud. That is, a cloud environment is being changed a single cloud computing environment into multi-clouds computing environment[1,2]. But there are many problems about standardization and security on heterogeneous cloud systems, different authentication methods for each cloud service enterprise and different API in using multi-clouds[3,4]. Users have to authenticate themselves for each cloud because each cloud service enterprise demands a different user authentication and they have to manage their own authentication individually given from many clouds. Because of these inconveniences, users tend to use same ID and Password to many cloud authentications[4,5]. Many papers discuss a danger of security about it[6,9].

So this paper suggests a user integrated authentication method that can access to many clouds by using an integrated authentication ticket from Brokerage Authentication to the users using multi-clouds. This paper is organized as follows. Section 2 will describe about clouds and authentication and section3 will arrange the security and the requirements in a cloud computing. Section 4 will describe about a user's authentication using BAC suggested in this paper and section 5 will analyze on a suggested user's authentication method. Finally, section6 will describe a conclusion.

2. Related Work

2.1 Cloud Computing

NIST defined cloud computing as "a model for enabling ubiquitous, convenient, in-demand network access to a shared pool of configurable computing resources

*Author for correspondence

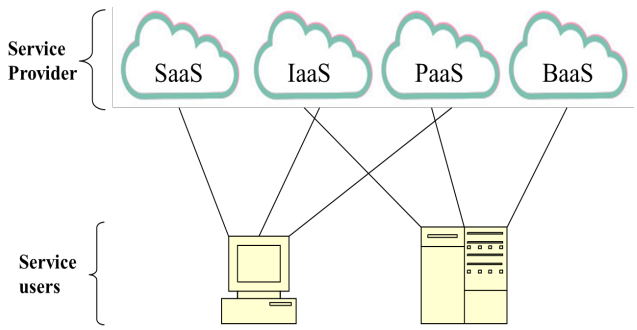

Figure 1. Multi-clouds.

(eg. Networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction[7]. Users using a single cloud can be damaged by data loss, integrity, security and so on because of an interruption of power supply, lack of storage, system overhead or other reasons[8,11,12]. Therefore demand of using multi-clouds is increasing. Multi-clouds are called cloud-of- clouds or intercloud which are introduced in the paper by Vukolic[10,13]. Multi-clouds mean not using a single cloud service but using several forms of cloud services. We can solve the problems when using a single cloud by using multi-clouds and the problems of dependency and reliability from using a single cloud.

2.2 Authentication

2.2.1 The Problem of Authentication in Multi-Clouds Environment

Multiplicity clouds are used to handle a service interruption status (power interruption, component failure, lack of storage and so on) in a using cloud. At this time, an authentication is required with different authentication system for each cloud because an authentication policy is different for each cloud. Hence, users have to be authenticated through several authentication procedures and they cannot help but using different ID and Password for each cloud. Generally users using many internet servers have used same ID and Password in many servers because they had difficulty managing many IDs and Passwords[ ]. So users will use same ID and Password in many clouds if different authentication is required in many clouds. This type of user authentication has inconvenience to users and a secure fragility. So this paper suggests an integrated user authentication method available in multi-clouds.

3. Security Risks and Requirements in cloud computing

This section will discuss the security risks and requirements in cloud computing. First, we research on authentication factors required generally in a cloud computing environment and many situations which are threatened to the security are described.

3.1 Security Requirements

3.1.1 Service Availability

Whenever a rightful user wants to use the service, they must be able to use it at any time. With increase of using a cloud, many users want to receive the service that they want to use by accessing a cloud anywhere and at any time. So there should not be happened the occasions that users can't be received the service because of a lack of storage in a cloud system or component failure. That is, users must be able to be received the service they want from the cloud through a rightful authentication procedure at any time.

3.1.2 Data Integrity

Only a rightful user must be able to access the data through the authorized mechanism because many users can use many clouds in a multi-clouds environment at the same time. And users' data, ID and Password using a cloud must be protected even though an interruption is occurred by malicious users or faults in the system.

3.1.3 Confidentiality

When users try to access to a cloud by using a ticket through a rightful procedure, confidentiality of user authentication ticket has to be ensured. Confidentiality in a transfer process of the ticket and from a malicious insider or an attacker has to be ensured as well.

3.1.4 User Authentication

Users in the multi-clouds environment are provided the service by accessing several clouds so as to receive different services. At this time, users will have inconvenience if they use different ID and Password in order to access a cloud suggesting different services. Also users will be inconvenient and confused if there are different authentication systems for each cloud. So an integrated

authentication system is needed between multi-clouds and users.

3.2 Security Risk

3.2.1 Snooping

This is to abuse by doing unauthorized access about users' data stored in a cloud or by taking users' authentication information.

3.2.2 Man-in-the-Middle Attack

It requires an attacker to have the ability to both monitor and alter or inject messages into a communication channel. Many users access to many clouds frequently when they use multi-clouds. This is that a malicious user overhears the authentication information of a normal user and the malicious user communicates between the user and a cloud when the user accesses to the cloud.

3.2.3 Modification

A malicious user can change user's information and data saved in a cloud by intercepting the user's authentication information.

3.2.4 Impersonation

The cloud can be accessed by many users. So if an unauthorized user masquerades as an authorized user, it's possible to attack as snooping. Eventually they take a rightful user's authentication information and can do unauthorized access to the user's data.

3.2.5 Replaying

A malicious user tries to access to a server or a system by using the authentication information again after he gets a copy of a rightful user's authentication message. Users can access to the cloud frequently, read and write their own data and use an application frequently in a cloud computing environment. If the copy of the user's authentication is reused by using replaying attack in this situation, there will be a serious treatment to security.

3.2.6 Password-guessing

This is an attack by guessing a user's password by an attacker. Users can use same ID and Password for every cloud, when ID and Password are maintained and used for every each cloud provider in multi-clouds. So if there is a success of password guessing in a cloud, there will be security problems in the other clouds.

3.3 Perfect Forward Secrecy

PFS is that when a key is opened, encoded messages are decoded by the key. So the key for password decode must not be used to induce a new key.

4. Proposal Method

A user authentication method in multi-clouds suggested in this paper uses Brokerage Authentication Center (BAC). A user who will be provided a service from the cloud is issued an authentication ticket which authenticates a rightful user from Brokerage Authentication Center (BAC). And they receive the service by accessing the cloud with the ticket. Also they can receive the service by accessing many clouds with one user authentication ticket despite multi-clouds environment. Before we discuss the suggesting method, we are assumption that many clouds are service providers authenticated from same BAC and they have API interface which can be integrated one another. And users are issued the authentication ticket from the same BAC.

4.1 Authentication Phase for Ticketing

A user authentication is achieved from BAC (Brokerage Authentication Center) before users request a service to a cloud. In this stage, users are issued a user authentication ticket to authenticate a rightful user. Figure 2 shows that how the user is got an authentication ticket.

- At first, user1 encodes his ID and Password by a public key of BAC, and requires an issue of an authentication ticket to BAC.

$$E_{puk_b}\left[ID_{u1}, PASS_{u1}\right]$$

- BAC issues the ticket which accepts the authentication with arbitrary precision N to user1. At this time, BAC stores $\left\{h\left(ID_{u1}, PASS_{u1}\right), h_b{}^n\left(ID_{u1} \,||\, PASS_{u1}\right)\right\}$ to a safe storage device.

$$E_{puk_{u1}}\left[ID_{u1}, ID_b, h_b{}^n\left(ID_{u1} \,||\, PASS_{u1}\right), N\right]$$

- User1 makes a key value of a user authentication ticket by decreasing 1 from N value from BAC. And he/she gives information as being received the ticket normally to BAC. At this time, user1 stores the value of $\left\{ID_b, h_{u1}{}^{n-1}\left(ID_{u1} \,||\, PASS_{u1}\right)\right\}$ to a safe storage device.

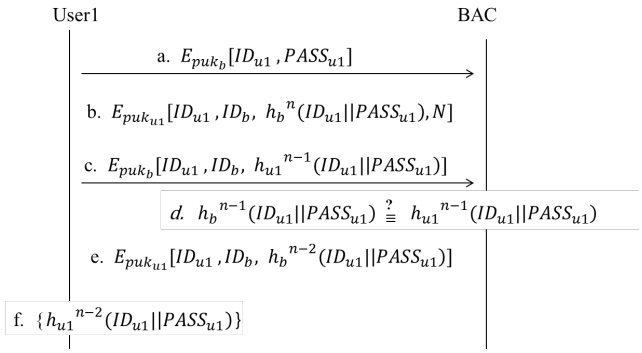

Figure 2. Authentication phase for ticketing.

$$E_{puk_b}\left[ID_{u1}, ID_b, h_{u1}^{\,n-1}\left(ID_{u1}||PASS_{u1}\right)\right]$$

- BAC receives the message from user1, and compares the value from user1 to N value from $h_b^{\,n}\left(ID_{u1}||PASS_{u1}\right)$ stored in it after decreasing 1.

$$h_b^{\,n-1}\left(ID_{u1}||PASS_{u1}\right) \stackrel{?}{\equiv} h_{u1}^{\,n-1}\left(ID_{u1}||PASS_{u1}\right)$$

- If the value is same in d-phase, BAC informs that the issue of ticket is normally achieved and user1 can use the ticket. Also BAC stores as the new value $\left\{h\left(ID_{u1}||PASS_{u1}\right), h_{u1}^{\,n-2}\left(ID_{u1}||PASS_{u1}\right)\right\}$ by decreasing 1 from user1's N value in its storage.

$$E_{puk_{u1}}\left[ID_{u1}, ID_b, h_b^{\,n-2}\left(ID_{u1}||PASS_{u1}\right)\right]$$

- If user1 receives the message that can use the ticket for authentication from BAC, he changes n-1 value stored in his own system into n-2 and starts to use it.

$$\left\{h_{u1}^{\,n-2}\left(ID_{u1}||PASS_{u1}\right)\right\}.$$

4.2 Service Request to Single Cloud

Figure 3. shows that user1 who is received a user authentication ticket for using a cloud1 accesses to single cloud and is received the service. User1 starts to require a service to a cloud1 by using the authentication ticket which is stored in his storage device from BAC.

- User1 requires the service to cloud with the ticket authenticated from BAC. At this time, user1 encodes the user authentication ticket which is stored in his storage device as a cloud1's public key and send a request message to cloud1.

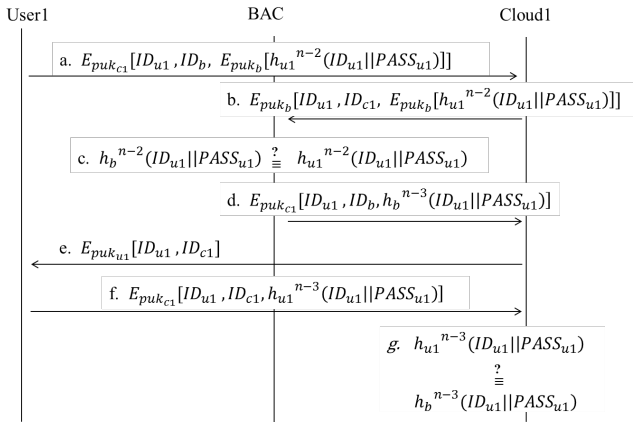

Figure 3. Service request to single cloud.

$$E_{puk_{c1}}\left[ID_{u1}, ID_b\ E_{puk_b}\left[h_{u1}^{\,n-2}\left(ID_{u1}||PASS_{u1}\right)\right]\right]$$

- After the cloud1 decodes the request message received from user1 with a secret key, the cloud1 checks ID_b and requests whether user1 is a rightful user or not to BAC which is an authentication institution.

$$E_{puk_b}\left[ID_{u1}, ID_{c1}, \mathrm{E}_{puk_b}\left[h_{u1}^{\,n-2}\left(ID_{u1}||PASS_{u1}\right)\right]\right]$$

- After BAC decodes the message from the cloud1 with its secret key, compares user1's ticket value stored in its system to the value of $h_{u1}^{\,n-2}\left(ID_{u1}||PASS_{u1}\right)$ from the cloud1 and checks if the value is same.

$$h_b^{\,n-2}\left(ID_{u1}||PASS_{u1}\right) \stackrel{?}{\equiv} h_{u1}^{\,n-2}\left(ID_{u1}||PASS_{u1}\right)$$

- If the value is same, after making the new key value $h_b^{\,n-3}\left(ID_{u1}||PASS_{u1}\right)$ by decreasing 1 from user1's N, it has to be sended to cloud1 as an authentication message on a user1. And BAC makes the new key value with $\left\{h\left(ID_{u1}||PASS_{u1}\right), h_b^{\,n-3}\left(ID_{u1}||PASS_{u1}\right)\right\}$ by decreasing 1 from user1's N value and stores it.

$$E_{puk_{c1}}\left[ID_{u1}, ID_b, h_b^{\,n-3}\left(ID_{u1}||PASS_{u1}\right)\right]$$

- Cloud1 received the message that user1 is a rightful authentication from BAC sends a message informing the start in using service to user1.

$$E_{puk_{u1}}\left[ID_{u1}, ID_{c1}\right]$$

- User1 who is received the message from cloud1 sends a respond message. He sends the ticket value by decreasing 1 from N which is from the ticket value $h_{u1}^{n-2}(ID_{u1}||PASS_{u1})$ stored in his own storage to cloud1. And he stores the new key value $\{h(ID_{u1}||PASS_{u1}), h_{u1}^{n-3}(ID_{u1}||PASS_{u1})\}$ in his storage.

$$E_{puk_{c1}}\left[ID_{u1}, ID_{c1}, h_{u1}^{n-3}(ID_{u1}||PASS_{u1})\right]$$

- Cloud 1 starts the cloud service to user1 if that ticket value is same after comparing the ticket value $h_b^{n-3}(ID_{u1}||PASS_{u1})$ from BAC to the ticket value $h_{u1}^{n-3}(ID_{u1}||PASS_{u1})$ from user1.

$$h_{u1}^{n-3}(ID_{u1}||PASS_{u1}) \stackrel{?}{\equiv} h_d^{n-3}(ID_{u1}||PASS_{u1})$$

4.3 Service Request to Multi-Clouds

Figure 4. shows that user1 logs on a cloud1 and is received the service. And it shows that user1 intends to request the service to cloud2 as well. In this stage, user1 doesn't be received different authentication from cloud1 to authenticate cloud2 and he requests the authentication to cloud2 by using an authentication ticket from BAC.

- User1 accesses to cloud2 with the ticket from BAC in order to use the service. And user1 generates the new key value $h_{u1}^{n-4}(ID_{u1}||PASS_{u1})$ by decreasing 1 from the ticket's N valued and sends to cloud2 by encoding as the public key of cloud2. And he stores the new key value $\{h_{u1}^{n-4}(ID_{u1}||PASS_{u1})\}$.

$$E_{puk_{c2}}\left[ID_{u1}, ID_b, E_{puk_b}\left[h_{u1}^{n-4}(ID_{u1}||PASS_{u1})\right]\right]$$

- Cloud2 decodes an authentication ticket with its private key in order to request the service of user1 and requests whether user1 is a rightful or not to BAC.

$$E_{puk_b}\left[ID_{u1}, ID_{c2}, E_{puk_b}\left[h_{u1}^{n-4}(ID_{u1}||PASS_{u1})\right]\right]$$

- BAC received message from cloud2 and decodes with BAC's secret key. Then, BAC finds out the user1's key value. After that, it decreases 1 from stored user1's key value $\{h_b^{n-3}(ID_{u1}||PASS_{u1})\}$ and compares the user1's key value from cloud2 to it.

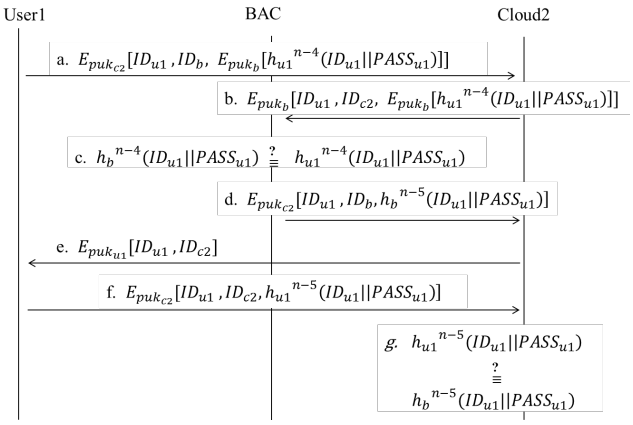

Figure 4. Service request to multi-clouds.

$$h_b^{n-4}(ID_{u1}||PASS_{u1}) \equiv h_{u1}^{n-4}(ID_{u1}||PASS_{u1}) \stackrel{?}{}$$

- If the key value is same in the c-phase, BAC sends the message that user1 is rightful to cloud2 with the key value $\{h_b^{n-5}(ID_{u1}||PASS_{u1})\}$ and stores $\{h(ID_{u1}||PASS_{u1}), h_b^{n-5}(ID_{u1}||PASS_{u1})\}$ that is decreased 1 from N value in its storage.

$$E_{puk_{c2}}\left[ID_{u1}, ID_b, h_b^{n-5}(ID_{u1}||PASS_{u1})\right]$$

- Cloud2 sends a message to inform that the service is started to user1 after user1 is regarded as a rightful user from BAC.

$$E_{puk_{u1}}\left[ID_{u1}, ID_{c2}\right]$$

- After User1 received the message to start the service from cloud2 sends the respond message, user1 sends the ticket value from $h_{u1}^{n-5}(ID_{u1}||PASS_{u1})$ by decreasing 1 from N of the ticket value h^n-4(IDu || PASSu) stored in his storage. And user1 stores the new key value $\{h(ID_{u1}||PASS_{u1}), h_{u1}^{n-5}(ID_{u1}||PASS_{u1})\}$ into his storage.

$$E_{puk_{c2}}\left[ID_{u1}, ID_{c2}, h_{u1}^{n-5}(ID_{u1}||PASS_{u1})\right]$$

- Cloud2 checks the ticket value $h_b^{n-5}(ID_{u1}||PASS_{u1})$ from BAC in d-phase and the user1's ticket value

$h_{u1}{}^{n-5}(ID_{u1} || PASS_{u1})$ in f-phase, if they are same, it starts the cloud service to user1.

$$h_{u1}{}^{n-5}(ID_{u1} || PASS_{u1}) \stackrel{?}{\equiv} h_b{}^{n-5}(ID_{u1} || PASS_{u1})$$

5. Evaluation

Cloud system requests identify whether a user1 requiring a service is rightful or not to BAC and if BAC judges that the user1 is rightful, BAC sends the key value made by decreasing 1 from N value to the cloud system. Cloud system sends the message permitting the user's access for fair service to the user1. And the user1 decreases 1 from N value and accesses in order to start the service by making a new key value. A cloud service provider compares the key value from BAC with the key value from a user. If that key values are same, user starts the service. The stability is ensured about attacks in this course. Also, efficiency of computing speed can be improved because one-way function is used when the user's authentication is generated.

5.1 Impersonation Attack

Suppose an attacker logs on as a normal user. The key value $h_{u1}{}^{n-3}(ID_{u1} || PASS_{u1}) \stackrel{?}{\equiv} h_b{}^{n-3}(ID_{u1} || PASS_{u1})$, $h_b{}^{n-4}(ID_{u1} || PASS_{u1}) \stackrel{?}{\equiv} h_{u1}{}^{n-4}(ID_{u1} || PASS_{u1})$, $h_{u1}{}^{n-5}(ID_{u1} || PASS_{u1}) \stackrel{?}{\equiv} h_b{}^{n-5}(ID_{u1} || PASS_{u1})$ authorized an authentication from BAC and the key value from the attacker can't be matched because the attacker doesn't know the N-value in a g-phase from Figure 3 and Figure 4. Even though the attacker logs on as a normal user, the success of impersonation attack is impossible.

5.2 Man-in-the-middle Attack

Even though an attacker intercepts a sender's message, attacker can't decode because the message is send by encoding as a sender's public key. To find N-value is impossible because the key value of a user's authentication ticket is calculated as hash function that is one-way function if it is decoded. The attacker doesn't know the correct N value when an attack is achieved because N value is decreased every 1 from N-value whenever the authentication is required. So the attack is impossible.

5.3 Replaying Attack

Replaying attack is to store the message of the authentication ticket by tapping and to use repeatedly the same message. It's impossible for an attacker to reuse the tapped authentication message because the new key value is made as N-m in order to make the new key value for an authentication even though messages are tapped.

5.4 Password Guessing Attack

Because they don't log on by using their ID and Password and they use the authentication ticket from a middle authentication server, in any case, user's ID and Password are not stored and used when users want to receive the service from the cloud. So it's impossible to analogize a user's password.

5.5 Insider Attack

Malicious insiders can't attack because user's ID and Password are not stored in the cloud system in any case. Also the attack can't be achieved by an inside malicious user because the ticket used in the user authentication phase and service request phase is used by decreasing every 1 from N value from BAC and users whenever the authentication and services are required.

5.6 Forward Secrecy

Even though it's possible to find out the user's key value of the authentication ticket, the attacker can't know the n-value because the key value of the ticket is changed all the time when a user of the cloud and a middle authentication center ask for the authentication and respond about the authentication. Thus, forward secrecy attack is impossible.

6. Conclusion

Users who intend to use the cloud are increased with development of IT technology and they want to use more different kinds of clouds as increasing uses of the cloud. So cloud ecosystem is changing because multi-clouds computing environment is appeared. However, the different authentication procedure for each cloud is achieved in user's environment which intends to use different cloud services. Thus, users have different authentication certificates, ID or Password for different cloud systems. This paper suggested an integrated authentication

method for users who can use in multi-clouds to handle user's inconvenience and security risks. I hope that the convenience of user center will be increased with suggested authentication method by using BAC. We also hope that security and stability will be improved by using BAC(brokerage authentication center) instead of the authentication method that is vulnerable to the security because of using same ID and Password when a user accesses to many servers.

7. References

1. AlZain MA, Pardede E, Soh B, Thom JA. cloud computing security: From single to multi-clouds. 45th Hawaii International Conference on System Sciences. 2012; 5490 –9.
2. Bernstein D, Ludvigson E, Sankar K, Diamond S, Morrow M. blueprint_for_the_intercloud_-protocols for cloud computing to cloud computing interoperability. Internet and Web Applications and Services. 2009; 328 –36.
3. Bessani A, Correia M, Quaresma B, Andre F, Sousa P. EDPSKY- dependable and secure storage in a cloud-of-clouds. European Conference on Computer Systems. 2011; 31–46.
4. Wi Y, Kwak J. OpenID Based User Authentication Scheme for Multi-clouds Environment. The journal of Digital Policy & Management. 2013; 11(7):215–23.
5. Tsai J-L. Effieient multi-server authentication scheme based on one-way hash function without verification table. Computers and Security. 2008; 27(3-4):115–21.
6. Das ML, Saxena A, Gulati VP. A Dynamic ID-based Remote User Authentication Scheme. IEEE Transactions on Consumer Electronics. 2004; (50)2:629–31.
7. Mell P, Grance T. The NIST Definition of Cloud Computing. NIST SP 800-145. 2011.
8. Singh Y, Kandah F, Zhang W. A Secured Cost-effective Multi-Cloud Storage in Cloud Computing. IEEE INFOCOM 2011 Workshop on Cloud Computing. 2011.
9. Takabi H, James Joshi JBD, Ahn G-J. Security and Privacy Challenges in Cloud Computing Environments. The IEEE Computer and Reliability Societies. 2010; 649–24.
10. Vokolic M. The Byzantine empire in the intercloud. ACM SIGACT News. 2010; 41(3):105–11.
11. Swarup B, Chukkala Varaha S, Pothabathula S. Design and Implementation of a Secure multi-cloud Data Storage Using Encryption. IJARCET. 2014; 3(5):1595–9.
12. Soo JH. Efficient and secure group key generation protocol for small and medium business. Journal of Convergence Society for SMB. 2014; 4(1):19–24.
13. Han K-H. Biometric Certificate on Secure Group Communication. Journal of Convergence Society for SMB. 2014; 4(1):25–30.