Impact Factor Value: 4.029 ISSN: 2349-7084
International Journal of Computer Engineering In Research Trends
Volume 4, Issue 6, June-2017, pp. 269-276 www.ijcert.org

CloudSim Framework for Federation of identity management in Cloud Computing

Rohit Shere*[1], Sonika Shrivastava[2], R.K. Pateriya[3]

[1] *M.Tech Scholar, Department of Computer Science & Engineering, MANIT, Bhopal, 462003, India*
[2]*Ph.D. Scholar, Department of Computer Science & Engineering, MANIT, Bhopal, 462003, India*
[3]*Associate Professor, Department of Computer Science & Engineering, MANIT, Bhopal, 462003, India*
{rohitshere07[1], ms271104[2], pateriyark[3]}@gmail.com

Abstract: - Cloud computing is built on several components for managing and making provision of abundant resources to business, on demand. Identity management is the essential element in cloud computing, and it is an inevitable standard security module that keeps away unauthorized users with unintentional interference to the system. The majority of work is being done to enhance this identity management component to overcome current limitations in authentication mechanisms. Federation among different clouds can be helpful in minimizing overhead and cost in overall identity management. Many cloud service providers are present in the industry with their independent identity management, but very few of them supports the federation among themselves to tackle the whole business collapse situation due to any disaster caused by nature. The Federation among these vendors can bring healthy competition in business markets that will lead to boost the confidence of cloud user in cloud computing. In this paper, our research work addresses a framework for researchers in identity management in cloud computing. The framework takes minimal effort and time for creating and simulating test environment for the generalized cloud environment.

Keywords: - FID, ECC algorithm, Hashing technique, CloudSim, Sign up Authentication.

1. Introduction

With the technological advancement in Cloud computing in last decade, this field is providing the basis for designing of services that improve the quality of digital applications that are used by the user in the day to day life. Cloud computing provides a platform for the development and deployment of the services. The services deployed on the cloud are mostly guaranteed for maximum availability and are offered to users on demand. These services are metered which means the user is agreed to pay-per-use. The characteristics of multi-tenancy are the notion of reusability which means different instances are created out of same entity. Cloud offers enormous scalability, self-operability, on-demand access in a cost effective manner. This recent technology has become prevalent due to a wide variety of offerings to users for their daily computational needs. For a comparative understanding of this technology, we can quote similarity between cloud service provider (CSP) and Internet service provider (ISP). The interest of the users will be kept intact in cloud services if the confidence of using these services remains

© 2017, IJCERT All Rights Reserved

unaffected by the failure due to disaster. The need for quick on-demand collaboration among cloud vendors plays a major role in this situation. In the internet technology, some innovative development in Federation for identity management and virtualization in distributed computing and accessing of the high-speed network with low cost attract the focus of users toward this technology.

This paper focuses on designing and contribution of a framework for research in identity federation technique. The implementation details are discussed in the third section of this paper, whereas subsequent section presents experimental setup and result in discussion with current limitations.

1.1 Standalone Authentication

Passwords are currently the most widely used authentication mechanism. A typical standalone authentication system implemented in a personal computer. Systems implementing standalone authentication is not networked which keeps them away from exploitation by network attacks. However, in cloud computing scenario, the users have to be connected to SP over a untrusted communicational channel of the internet which brings a challenge of proving the identity of the remote user. This has to be handled by service providers by ensuring correct authentication mechanism that identifies intended user and keeps the system safe from unauthorized entities.

1.2 Third Party Authentication (TPA)

It is carried out through trusted third party or Identity-service Provider (IdP). It plays a role for authenticating the user on behalf of the Service Provider (SP). It manages all the authorization, providing user attributes to mutually trusted SPs through agreed upon protocols. All the housekeeping activity for the user profile is carried out by IdP, and it is responsible for secure service provisioning to the user as well. In figure 2 below, the IdP is mutually trusted by Cloud Service Provider (CSP). Any user requesting the services has redirected to IdP which authorizes the user and issues a security token which user can keep it for using services from CSP in near future before the session ends at the service end. Upon receiving a token, the user presents it to SP. The service provider, in turn, validates the token from IdP. This takes significant time for overall authentication process which increases response time. Depending on the IdP's response for validation SP can offer or deny the service request.

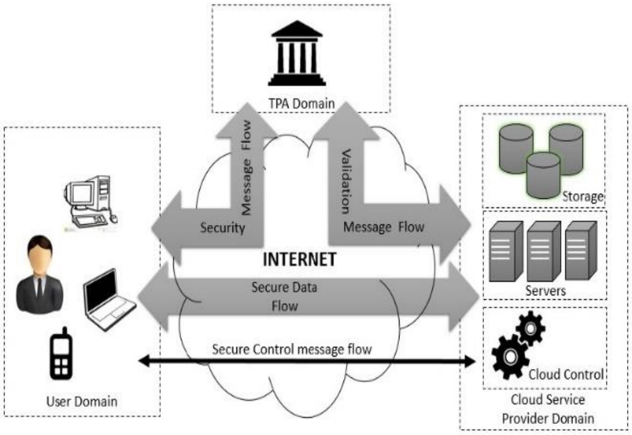


Figure 2: Architecture for Third party IdP service

2. The Security Assertion Markup Language (SAML)

It is widely accepted and preferred to be implemented by many organizations instead of OAuth. Since its latest second version release in 2005, many companies adopted it for Identity management systems and also contributed to it. In web application development, the authentication state information in the cookies from one domain is never available to another domain. SAML comes handy to achieve this, and it enables SSO and automatic user provisioning with little changes in assertions and response handling. SAML assertions and protocol messages are encoded in XML which facilitates use other protocols to communicate either regarding an assertion itself or the subject of an assertion. These messages can be embedded in other structures for transport such as HTTP POST requests or XML- encoded SOAP messages.

An assertion is a package of information that supplies one or more statements made by a SAML authority. SAML defines three different kinds of assertion statement that can be created by a SAML authority:

- **Authentication:** User generates this SAML assertion.
- **Attribute**: Specific feature associated with the object
- **Authorization Decision:** An Authority like IdP provides the decision related to validation of authentication token

Depending on the requirements, the system designer can choose from signed or unsigned SAML responses with encrypted signed/unsigned assertion to be implemented. Advantages in encryption of assertion remover communication vulnerability of being eavesdropped. Signing assertions guarantee no intermediate modification to the source generated responses.

2.1 Advantages of using SAML:

Following are the key advantages,

a. **Platform neutrality –**
   It abstracts security framework from platform architectures
b. **Loose coupling of directories –**
   User information across directories need not be maintained and synchronized.
c. **Improved online experience –**
   A user once authenticated to IdP, no additional authentication at the service provider is required for service access.
d. **Reduced Administration cost –**
   Active session of user at IdP can be reused by user which minimizes overall computational effort for SP
e. **Risk Transference –**
   SP can trust IdP for user profile management which pushes IdP beyond its verification and validation responsibilities.

With these advantages, for our framework implementation, we opted for SAML as per OASIS standard. An unsigned SAML Response with encrypted signed assertions was communicated between IdP and CSPs for secure hassle free interaction.

3. Problem Identification

In the multi-cloud environment where data is divided into segments, and these segments are distributed among the available clouds, integrity verification, in this case, is a significant issue. Since every segment resides on the different cloud, so chances of anomaly are raised. Studying these anomalies in a real cloud environment is challenging task. Thus a generalized framework for creation, simulation of federated of non-federated identity management on cloud and proof of concept for any advancement in FIM system that can be achieved with minimal time and effort.

The main three principles of security which are confidentiality, integrity, and availability need to be addressed by any component that serves as a security agent of the system which can be achieved by a robust encryption, signature algorithms, and active redundancy respectively. In CloudSim toolkit there is no support for federated identity handling thus in this work, we aim to provide a framework by implementing lightweight and robust encryption algorithm for confidentiality in SSO authentication in the federation environment using the CloudSim toolkit. We discuss the design and implementation of this mechanism, providing the services ensuring confidentiality, integrity, and non-repudiation of the identity information.

4. Related Research And Literature Review

Benjamin Ertl[12] proposed an approach, which is based on the standardized System for Cross-domain Identity Management (SCIM) protocol. They have added the support for account linking and pre-service verification. The author tried to put the concept of this system into the context of currently existing federated infrastructures and demonstrated it within a federated e-infrastructure. In their paper,

© 2017, IJCERT All Rights Reserved

an algorithm was proposed which utilize different user, and group claim results in the mappings outlined. Here one user identity uses, and there are no groups provided. Only verify the user against the local user management service and add a verified user to the default group.

Yong Yu[14] embedded key-homomorphism cryptographic primitive to newly constructed identity-based (ID-based) RDIC protocol to reduce the system complexity. It also minimizes the cost of establishing and managing the public key authentication framework in PKI-based RDIC schemes as shown in Figure 4. Author implements ID-based RDIC and its security model with security against both malicious cloud server and third party verifier using zero knowledge privacy. The proposed ID-based RDIC protocol is data leak proof as metadata of the stored data cannot be revealed to the verifier during the RDIC process.

Jaweher Zouari[13] proposed a single sign-on concept based on Identity as a Service framework in which Automated Identity Finder (AIDF) system associates one or more service provider with the suitable identity provider after user consent. To maintain the same Identity context some additional functionalities that claim a transform between different standards and semantic mapping among different attributes have been proposed. Identity providers register to the AIDF system by indicating the identity types, standards their system supports, and the level of assurance they grant to asserted identities.

5. Proposed Work

For best possible federated identity management framework in CloudSim, the proposed a framework where a researcher can configured system for simulation scenario which supports for multiple IdPs providing authentication support to multiple CSPs. In this framework, a CSP can be configured to support multiple IdPs for the same user. Thus the user can choose to provide security token from any of the suitable IDPs where he has registered an account. It enables the user to have multiple identities. Cloud user can set a default IdP for regular authentication. Once the user receives successful login response, the user can set another active authentication with other IdP at signed in the cloud to avoid interruption due to the failure of its previous IdP. This feature supports the availability characteristic of the cloud as it offers smooth and hassles free experience of cloud services. At each simulated cloud end, a temporary user profile is created for future surveillance purpose which can be used to track cloud user activities. This profile is stored in the database and it keeps track of session ID, time of login, a list of services access, etc.

To regulate access of cloud user within a cloud, RBAC is implemented at a novice level with locally stored XML access policy files. Depending on the configuration of the user of this framework the behaviour of cloud user can be controlled for that specific simulated cloud. SHA-2 with a key length of 256 bits which is not breakable for time bound the brute force attack, is used for signing the SAML responses to maintain the integrity of the authentication communications. Thus the proposed work aims to achieve confidentiality, integrity, and availability for this framework. In a subsequent portion of this paper, a brief discussion is done about the details about the details of the framework for its authentication mechanisms along with encryption algorithm details.

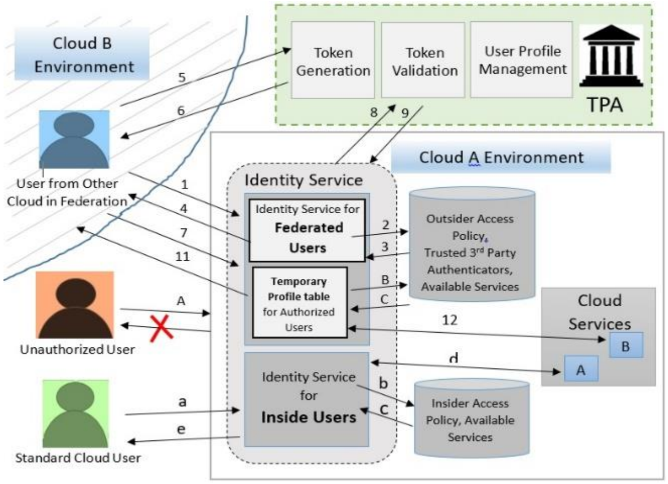

Cloud B Environment
Token Generation
Token Validation
User Profile Management
TPA
5
6
8
9
User from Other Cloud in Federation
Cloud A Environment
Identity Service
1
4
7
11
Identity Service for Federated Users
2
3
Outsider Access Policy, Trusted 3rd Party Authenticators, Available Services
Temporary Profile table for Authorized Users
B
C
A
12
Unauthorized User
Cloud Services
B
A
d
Identity Service for Inside Users
b
c
Insider Access Policy, Available Services
a
e
Standard Cloud User

Figure 5: Federation single sign-on model architecture

5.1 Proposed Framework

The figure 5 above, gives the architectural structure followed in the proposed framework. It shows the authentication sequence between TPA, Cloud Users and CSP.

5.1.1 Normal User Access:

a. User from Cloud A requests normal access for Service B running in Cloud A
b. As the application is from inside user, Identity Service carries out the standard authentication process. It queries the credentials to the standard user database.
c. The fetched credentials are matched for authentication
d. If match is found, Resource access confirmation message is sent to the standard user
e. A standard user can now access the registered services from cloud A.

5.1.2 Unauthorized User Access [Denial case]

A. An unauthorized random outside user requests normal/federated access for Service B running in Cloud A
B. As the request is from the user, Identity Service carries out the standard authentication process. It queries the credentials to the standard user database.
C. As NO Match should observe, it should send authorization failure message to the requestor.
D. The unauthorized user gets delivered Authentication failure message on the screen.

5.1.3 Federated User Access:

1. User from Cloud B (in collaboration/federation) requests federated access for Service A running in Cloud A
2. Federated Identity Service (FIS) looks for an available trusted 3rd party authentication service provider in the database.
3. Database service replies to the Identity service with the available trusted 3rd party list
4. FIS forwards the list to the client.
5. The client gets redirected to one of the 3rd party authenticators within the list. It Authenticates to it.
6. Upon successful authentication, Client receives Authentication ticket & it gets stored locally for future use
7. The client provides this ticket to Federated Identify service of cloud A.
8. FIS sends the ticket to 3rd Party authentication service for ticket validation.
9. 3rd party sends approval/rejection of the ticket to FIS.
10. Upon approval by 3rd party authentication service, FIS creates a temporary user profile for federated client for monitoring purpose
11. FIS sends a successful authentication message to the federated client.

Federated cloud client can now access only the registered service that is mentioned in the policy for the federated user, running in Cloud A.

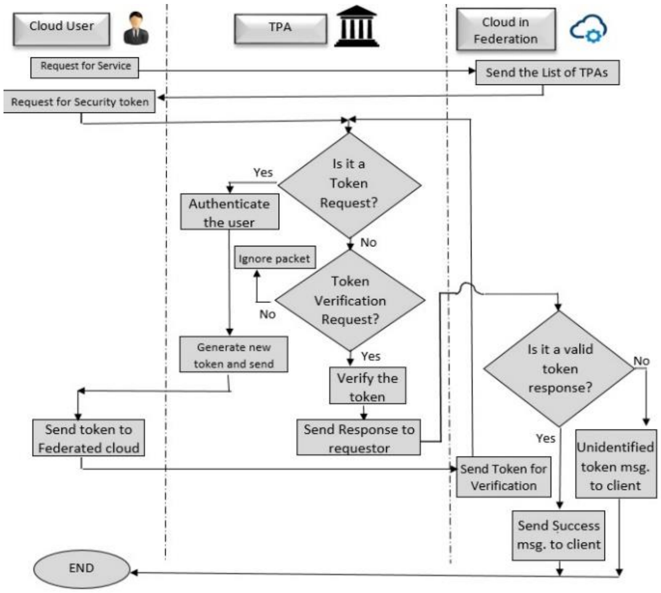

Cloud User
TPA
Cloud in Federation
Request for Service
Send the List of TPAs
Request for Security token
Is it a Token Request?
Yes
Authenticate the user
No
Ignore packet
Token Verification Request?
No
Yes
Generate new token and send
Verify the token
Is it a valid token response?
No
Send token to Federated cloud
Send Response to requestor
Yes
Unidentified token msg. to client
Send Token for Verification
Send Success msg. to client
END

Figure 6: Flowchart for implemented federated authentication

5.2 ECC Algorithm for encryption of SAML assertions

Elliptical curve cryptography (ECC) is a public key encryption technique based on elliptic curve theory that can be used to create faster, smaller, and more efficient cryptographic keys. ECC generates keys through the properties of the

© 2017, IJCERT All Rights Reserved

elliptic curve equation instead of the traditional method of generation as the product of very large prime numbers. The technology can be used in conjunction with most public key encryption methods, such as RSA.

5.3 Hyper-Elliptic Curves Cryptography

The security of Hyperelliptic Curve Cryptosystem depends on the discrete logarithm problem[2]. This problem helps to avoid the eavesdropper from breaking of keys even both Q and P values are known publicly. Different types of the curve have to be understood for public key (Q) and group point (P).

HEC curve of genus $g >= 1$ over finite filed F is set of solution $(x, y) \in F * F$ to the equation

E: $y^2 + h(x)\, y = f(x)$ (1)

Where h(x) is a polynomial of degree g and h(x) ϵ $F(x)$, $f(x)$ is a monic polynomial of degree 2g+1 and $h(x) \in F(x)$.The curve E is said to be non-singular curve, if there are no pair$(x, y) \in F * F$. The polynomial $f(x)$ and $h(x)$ are chosen such that it has to satisfy the following equations

$2y + h(x) = 0$ (2)

$h'(x).y - f'(x) = 0$ (3)

5.3.1 Types of genus curve

Genus curve decides the processing time of the Elliptic Curve Cryptosystem such as key generation, encryption, and decryption process. The value of g decided the polynomial of curve EC like $g = 2, 3, 4$. Polynomial chosen for genus 2, 3, 4, 5 and 6 over prime field[3] F_P given below

Genus $g = 2$

$Y^2 = x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$ (4)

Genus $g = 3$

$Y^2 = x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$ (5)

Genus $g = 4$

$Y^2 = x^9 + a_8x^8 + a_7x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$ (6)

Genus $g = 5$

$Y^2 = x^9 + a_9x^9 + a_8x^8 + a_7x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$ (7)

Genus $g = 6$

$Y^2 = x^9 + a_9x^9 + a_9x^9 + a_8x^8 + a_7x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$ (8)

HECC consists of three processes that are a key generation, encryption and decryption. These processes involved in divisor generation by choosing proper polynomial of genus curve 2, 3, 4, 5 and 6 as shown in Equation (4), (5), (6), (7) and (8). ElGamal method is used to design Hyper elliptic Curve Cryptosystem process, known as the HECElGamal algorithm is discussed in next section.

5.4 Key generation

Input: The public parameters are hyperelliptic curve C, prime p and divisor D.

Output: The public key P_A and private key a_A.

a. ***Private Key:*** $K_A \in R_N$; Random prime K_A is chosen in order of N.

b. ***Public key:*** $P_A \leftarrow K_A$.D; PA is a pair of polynomial [(u_x, v_y)] and D is Divisor

c. ***Key pair:*** [(K_A, P_A)]

5.5 Encryption stage

The plaintext' is converted into ASCII value and these values are represented as Sequence of points (u_x,v_y). The encoded message is referred asE_m. The following step encrypts message m into E_m which is then sent to user B.

Agreed key: $Q_A \leftarrow K_A P_B$: P_B is represented as receiver's public key.

Cipher text: $C_m \leftarrow [\, Q_A\, ,\, E_m + P_A\,]$; C_m is represented as [(u_x,v_y)].

5.6 Decryption stage

To decrypt the cipher text C_m, user B extracts the first coordinate 'Q_A' from the cipher text then multiply with its private key (a_B) and subtract the result from the second coordinate. This can be written as follows:

E_m+ k P_B− a_B (Q_A) = E

= E_m+ k P_B− k (a_BD)

= m + k P_B− a_B (k D)

= E_m+ k P_B− k P_B= E_m

© 2017, IJCERT All Rights Reserved

We have chosen ECC for SAML signature encryption, as the only possible Side Channel Attack, strict safety policy adherence can suppress backdoor attacks. The quantum computing attacks are insignificant in the current situation.

6. Experimental Setup

To perform simulation, A CloudSim API, with Apache server using Java language is used. The aim of CloudSim is to offer a comprehensive simulation basis that enables researchers to model, simulate and to perform an experiment on evolving Cloud computing infrastructures. To experience web authentication scenarios, we have used the Apache framework, Java language and JSP framework to design and render web pages. The general details of the development and deployment platform used in this work are briefed in the section below.

The simulation experiment was performed on a 64 bit Windows 10 platform machine with Intel® Core™i7- 3770 assisted with 4GB of DDR3 SD RAM. For implementation purpose we have used CloudSim-3.0.3, NetBeans IDE version 8.0, MySQL Server Version 5.0 (GPL) for Windows 10 and Java version 1.8.0_65.
Java Implementation for SSO was done with the help of open source SAML, an OASIS standard [11] which to carry out Authorization module for the framework. It also eliminated Replay attacks by storing the ID of the SAML messages already processed, to avoid processing them twice. For storing user credentials, user attributes, CSP details and IdP parameters, MySQL database was used. During execution, the RBAC policies for user were fetched from corresponding project files into MySQL database for faster retrieval operations. The user active session details were also stored in the separate database. The mappings of cloud user, CSP, and IdP, formed a table in the database. This framework takes care confidentiality and integrity of communication by mentioning the proposed HECC algorithm in SAML assertions mark-ups attributes.

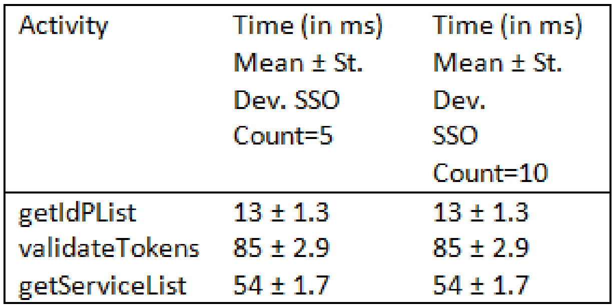
7. Result Analysis

The execution performed by considering different scenarios which include a different number of service access requests and different numbers of clouds in Federation. The timings were observed for lookup, token validation time and response time for the access request. There could be various computational parameters that could have been specified, but the most prominent parameters are discussed below as part of measuring performance

Table 1: Time statistics for Federated Identity service

Sub-Component	Time (in ms) Mean ± Std. Dev.
Temp. User Profile Creation	346 ± 25.7
Temp. User Profile Updating	72 ± 7.9
Temp. User Profile Deletion	23 ± 4.6

Table 2: Time statistics for Authentication related parameters for access to multiple services in SSO



Activity	Time (in ms) Mean ± St. Dev. SSO Count=5	Time (in ms) Mean ± St. Dev. SSO Count=10
getIdPList	13 ± 1.3	13 ± 1.3
validateTokens	85 ± 2.9	85 ± 2.9
getServiceList	54 ± 1.7	54 ± 1.7

The user profile creation takes higher time than user profile management time. Once the user gets logged on for the first time, his profile will be inactive session till the time log out activity gets called by the user. The observed statistics show that the proposed algorithms for the framework are suitable for federated identity management along with and single sign-on services. By applying different encryption algorithms or innovation in encryption technique to communication during authentication sequencing, makes this framework a basis for researchers to test their approach against latest attacks possible. The Single Sign-On approach has reduced the overhead both users and developers of facing difficulties in the management of multiple credentials that require accessing a variety of services from different CSPs in the collaboration.

8. Conclusion

The current framework implementation is supported for SAML-Browser based SSO. With SAML implementation for SSO defined by the OASIS Security Services, CSP in this CloudSim framework gets added functionality of deep linking, automatic renewal of sessions. This approach has limitations for command line users of cloud. These users need to invoke browser for authentication before line clients can control cloud resources or services. As a part of future work, enhancement in user profile needs to be done for the command line oriented users. This can be a motivation for researchers in this field to find alternative techniques which enhance this framework by contributing to it. The proposed work was carried out with open source technologies like Java SE, MySQL, and Apache Framework so as to keep the interest of open source contributors. The results of experimentation were promising for the federated cloud environment. The proposed algorithms were found to be efficient and reliable.

References

1. Hong Liu, Student Member, IEEE, Huansheng Ning, Senior Member, IEEE, Qingxu Xiong, Member,IEEE, and Laurence T. Yang, Member, IEEE2014, "Shared Authority Based Privacy-preserving Authentication Protocol in Cloud Computing."
2. Adleman, L., "A subexponential algorithm for the discrete logarithm problem with application to cryptography", Proc. 20th IEEE Found. Comp. Sci. Symp., 1979, 55-60.
3. Ganeshan R. et.al, "Performance analysis of Hyper-Elliptic Curve Cryptosystems over Finite Fields F_p for Genus 2 & 4", IJCSNS Vol. 8 No. 12, Dec 2008
4. "The Notorious Nine - Cloud Computing Top Threats in 2013," https://downloads.cloudsecurityalliance.org/initiatives/top_threats"Panda: Public Auditing for Shared Data with Efficient User Revocation in the Cloud," IEEE Transactions on Services Computing, VOL. X, NO. X, XXXX 2014, accepted.
5. Jin Li, Yan Kit Li, Xiaofeng Chen, Patrick P. C. Lee, Wenjing Lou, "A Hybrid Cloud Approach for Secure Authorized Deduplication", IEEE Transactions on Parallel and Distributed Systems, 2014.
6. Lluis Pamies-Juarez, Pedro Garcia-Lopez, Marc Sanchez-Artigas, Blas Herrera, "Towards the Design of Optimal Data Redundancy Schemes for Heterogeneous Cloud Storage Infrastructures" ," Computer Networks, Vol.55, 1100-1113, 2011.
7. Deyan Chen, Hong Zhao, "Data Security and Privacy Protection Issues in Cloud Computing", International Conference on Computer Science and Electronics Engineering 2012.
8. K. Ren, C. Wang, and Q. Wang, "Security Challenges for the Public Cloud," IEEE Internet Computing, vol. 16, no. 1, pp. 69–73,2012.
9. Qian Wang, Cong Wang, Kui Ren, Wenjing Lou, Jin Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing", Proc. IEEE transactions on parallel and distributed systems, vol. 22, no. 5, may 2011 .
10. OASIS: Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V2.0. OASIS Standard (2005)
11. Benjain Ertl, "Identity Harmonization for Federated HPC, Grid and Cloud Services", IEEE, pp. 621-627, 2016.
12. Jaweher Zouari, "An Identity as a service framework for the cloud", IEEE, pp. 1-5, 2016.
13. Yong Yu, "Identity based Remote Data Integrity hacking with perfect data privacy preserving for cloud storage", IEEE, pp. 1-11, 2016.

© 2017, IJCERT All Rights Reserved