Identity Management Frameworks for Cloud

Roshni Bhandari
Department of Computer Science and Engineering
Parul Institute of Technology, Vadodara

UpendraBhoi
Department of Computer Science and Engineering
Parul Institute of Technology, Vadodara

Dhiren Patel
Department of Computer Engineering
National Institute of Technology, Surat, India

ABSTRACT

Cloud computing is a new trend of computing paradigm that provides a set of scalable resources on demand. However, it also being a target of cyber attacks and creates risk for data privacy and protection. An Identity Management System (IDM) supports the management of multiple digital identities for authentication and authorization. This paper reviews various identity management frameworks that help making Cloud environment more secure.

General Terms

Identity management, Service provider, Identity provider, Cloud computing.

Keywords

Single-Sign-On (SSO); Identity and Access Management, Personal identifiable information (PII); identity federation management; provisioning; de-provisioning

1. INTRODUCTION

1.1 Cloud Computing

In 2011, NIST [1] defined Cloud computing as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., Networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. Cloud model consists of five essential characteristics that are on-demand self-service, broad network access, resource pooling, rapid elasticity, measured service; three service models that are software as a service (SaaS), platform as a service (PaaS), infrastructure as a service (IaaS); and three deployment models that are private Cloud, public Cloud, and hybrid Cloud(see Figure 1).

1.2 Identity Management System

An identity is a set of unique characteristics of anuser: an individual, a subject, or an object. An identity used for identification purposes is called an identifier [3].Service providers provide the authentication to the user identifiers. Identifiers provide digital identity to an SP about the user's identity, which helps the SP to decide whether to allow the user to use a service or not. Users may have multiple digital identities. An Identity Management System (IDM) supports the management of multiple digital identities. It also decides how to disclose personally identifiable information (PII) of entities to obtain a specific service. IDM performs the following tasks [4]:

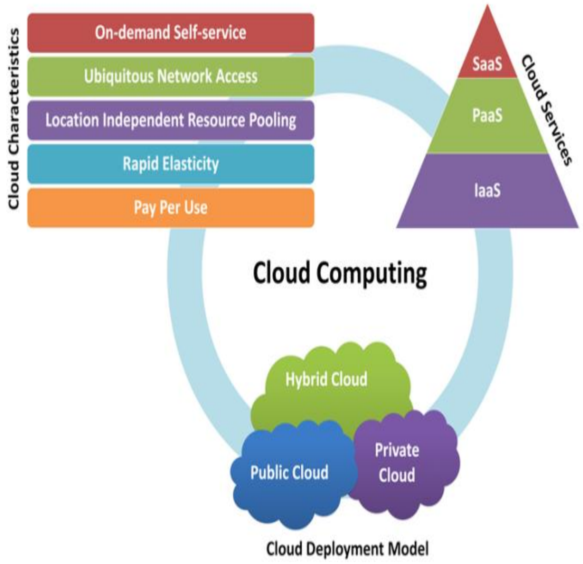

Fig 1: Cloud Computing [2]

Set up identities: Comparing personally identifiable information with auser.

Describe identities: Delegate attributes identifying auser.

Record the use of identity data: Storethe identity activity in a system.

Destroy an identity: After the completion of the work personally identifiable informationof the user become unusable.

A set of parties uses IDM and collaborate to identify an entity. These parties are [3]:

Identity provider (IdP). It issues digital identities. For example, governments issues identities of social security number or driving license, master card provider issues identities enabling payment.

Service provider (SP). It provides services to entities on the bases of valid authentication. For example Net banking or online transaction.

User. Users about whom claims are made. A claim could be, for example, a user name, or an email etc.

Identity verifier. It receives requests from the Service Provider for verifying a claim of users. It verifies whether the claims are correct or not.

An IDM uses one of the following three categories of identifiers:

Identifiers that both auser and Service Provider know.

Identifiers that an entity knows and Service Providers can verify via the Identity Provider.Identifiers that an entity is (unique markers, for example retina).The 7 Laws of identity which are developed by Kim Cameron given below [5]:

User Control and Consent: With the user's consent, the technical identity systems must only reveal information identifying a user. The System must protect the user against fraud and also verifying the identity of the users.

Minimal Disclosure for a Constrained use: The solution which discloses the minimum amount of identifying information and it is the long-term solution.

Justifiable Parties: It involves the user and the service provider. The justification applies both to the userand service provider.

Directed Identity: It involves only directional and unidirectional. Omni-directional identifiers use by public entities and "unidirectional" identifiers use by private entities. For example,when a computer user enters aclass room equipped with the projector it is an omni-directional identity beacon could be utilized to decide whether he wants to interact with it. If he does, a Uni-directional identity relation could be established between the computer and the projector.

Pluralism of Operators and Technologies: In identity system the inter-working of multiple identity technologies run by multiple identity providers.

Human Integration: The identity metasystem must define the human user to be a component of the distributed system integrated through unambiguous human-machine communication mechanisms offering protection against identity attacks.

Consistent Experience Across Contexts: It means that user request for services to the service provider then the service provider match the digital identities and provide his services on the basis of digital identities.

Functional area of identity management[6]:

User provisioning: Creating and maintaining user identities for access to services.Modeling and mapping: Using a management model to efficiently map users to resources.

Delegated administration: Delivering a means to distributed administrators for defining a hierarchy of roles to manage to access services.

Self-registration or self-service: Delivering delegated identity and editing down to individual users.

Workflow: Managing identity approval processes.

Auditing, logging, and reporting: Managing the necessary use of tools to history of useable life-cycle management steps, and reporting that information as well as reporting it against actual access-control lists on managed platforms.

Password management: Providing an administrative interface specifically for password policies and synchronization.

Integration: Using a “toolkit” such as a meta directory service to link multiple identity sources together for easier updating.

Various Identity Management Framework which manages the electronic identities. These Frameworks include SAML, Liberty Alliance, Windows Card space, PRIME, OpenID, OAuth, OneLogin, Windows identity Foundation, etc..

The rest of the paper is organized as follows: Section 2 derives Importance of Identity management in the Cloud. Section 3 describes the Identity management Frameworks. Section 4 summarizes and compares Identity management frameworks in theCloud. Section 5 concludes this survey and the Section 6 covers the references.

2. IMPORTANCE OF IDENTITY MANAGEMENT IN CLOUD

In Cloud computing, entities may have multiple accounts associated with different Service Providers. Likewise, entities may use multiple services offered by the same Service Provider (e.g., Gmail and Google Maps are offered by Google). A Cloud user has to provide his personally identifiable information (PII), which identifies him. Sharing PIIs of the same entity across services along with associated attributes can lead to the mapping of PIIs to the entity [7]. The main issue is how to secure PII from being used by unauthorized or untrusted parties in order to prevent serious crimes against privacy, such as identity theft [8]. The owner of data, including PII, is responsible for his privacy in Cloud computing. The owner needs technical controls supporting this challenging task.In a Public Cloud [9], to improve security use identity management features such as a single-sign-on. This would limit user provisioning and avoid multiple access system that work of monitoring of the activities of the end user.In a Private Cloud [9], which is managed by the organization, thus it is protected by a firewall. These types of Clouds integrate with client’s identity management systems for SSO, user authentication, authorization, audit, provisioning, role management and compliance.Identity Management Maturity Model for three types of services i.e. Software as a service (SaaS), Identity as a service (IDaaS) and Infrastructure as a Service (IaaS) is providing the security in it. Identity Management reduces the cost and the complexity of internal infrastructure to external SaaS and Cloud applications.

Identity management Lifecycle (see Figure 2) is useful to manage user identities and their credentials. Cloud provisioning means on-demand Provisioning. User Provisioning includes Active Directory and Light Weight Directory Access Protocol.

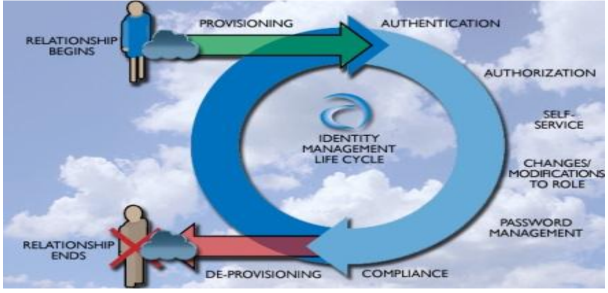

Fig 2: Identity management Lifecycle [10]

Single-Sign-On: It is useful to manage personal information of user in online business. User can Only one time login and use multiple application. One log-on provides access to all resources of the network, LAN or WAN. It can be illustrated in two different scopes. client/server relationship and e-commerce domain.

Client/server relationship: Single sign-on is a user authentication process that permits a user to enter one name and password in order to access multiple applications.

E-commerce: The single sign-on is designed to centralize consumer financial information on one server- not only for the consumer's convenience but also to offer increased security by limiting the number of times the consumer enters credit card numbers or other sensitive information used in billing [21].

Types of Single-Sign-On are given below [21]:

Password Synchronization

The password synchronization is the process of changing each password for different applications of the same value, so that the user always enters the same password. Once the user installs password synchronization software, users will enter the same password when they login to any of the synchronized systems, such as to their network, finance system, e-mail, calendar.

Legacy SSO (Employee/Enterprise SSO)

There are two types in which Script based and Application wizard based involve. In Script based Write a script that would take the target application credentials and launch the application and in Application Wizard based Runs a service on the client that continually monitors the workstation for login dialog boxes.

Web Access Management (WAM)

It is a form of identity management that controls access to web resources, providing authentication management, policy-based authorizations audit and reporting services. It is a browser based application.

Cross Domain SSO

In Cross Domain SSO Multiple states that manage user credentials. A user authenticated in one states gets signed-on to an application using another states typically within the same enterprise.

Federated SSO

Example of federated SSO is Liberty Alliance, OASIS, IBM/Microsoft. It is used to Establishment of trusted partnerships, New revenue opportunities and New, efficient, and production models. In One Login Only one password can make very secure. It can encrypt the sensitive data and send it by the SSL save channel, Reduced operational cost, Reduced time to access data It Improved user experience no password lists to carry.

3. IDENTITY MANAGEMENT FRAMEWORKS

3.1 SAML

Security Assertion Markup Language is an XML-based Open standard data format for exchanging authentication and authorization data between two parties an identity Provider and Service Provider over an internet[11].The Consortium for defining SAML standard and security is OASIS(Organization for the Advancement of Structured Information Standards)[12].There are three SAML versions: SAML 1.0, SAML 1.1 and the new major version of SAML is 2.0 became an official OASIS standard in March 2005.

The Component of SAML are assertions, Protocols, Bindings and Profiles[12].

Assertions: The transaction from the identity Provider to the Service Provider is called a SAML assertion.

Protocols: It is used to communicate assertions between the service provider and identity Provider.

Bindings: It is used to Map the SAML Protocol on to lower level network communication Protocols which are used to transport the SAML assertion between the identity Provider and Service Provider.

Profiles: It is the highest level of SAML Component that is the use cases between identity Provider and Service Provider that indicates how assertion,Protocols and Bindings will work together to Provide single-sign-on.

The web browser Single-Sign-on profile may be initiated by the Identity Provider or the Service Provider. If Initiated by the Identity Provider, the assertion is either signed, encrypted, or both. The Figure 3 shows that the Identity Provider Initiated SAML, assertion Flowchart. In this Browser send the request to an identity provider for access resource. Identity provider redirects with AuthRequest to Browser. Identity provider getsAuthRequest from Browser. The identity provider sends the challenge for credentials or proof to Browser like username or password. Browser login with username and password. Identity provider response with signal in HTML form to Browser. Browser POST SAML response to the Service Provider. Service Provider checks authentication and authorization from SAML Assertion. Then service provider supply resources to the browser. In the case of service provider initiated SAML Assertion flowchart, the service provider redirected the user back to the identity provider's federation web page with a SAML request.

SAML is Single-Sign-On Process so that only one time login with username and password and use multiple applications and services at a time. So it reduces the time taken by users to log into multiple applications and platforms. SAML also improves the effectiveness of all Networks. It also reduces the Administrative Overhead. SAML does not require user information to be maintained and synchronized between databases. The limitations of SAML are single point of failure. It added the cost and also the necessary information disclosure between the trusting site and SSO authority.

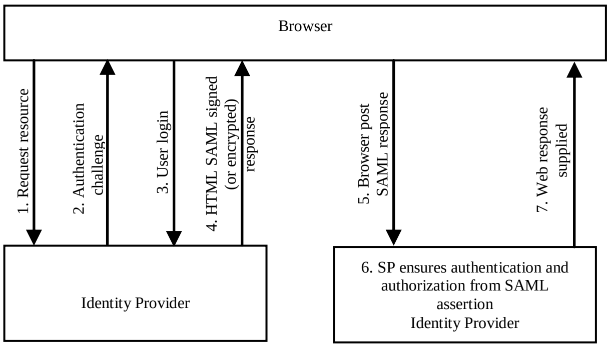
Browser
1. Request resource
2. Authentication challenge
3. User login
4. HTML SAML signed (or encrypted) response
5. Browser post SAML response
7. Web response supplied
Identity Provider
6. SP ensures authentication and authorization from SAML assertion
Identity Provider

Fig 3: Identity Provider Initiated SAML Assertion Flowchart [12]

3.2 Liberty Alliance

Liberty Alliance defines sets of protocols that collectively offer solutions for identity federation management, cross-domain authentication and session management [13]. Liberty Alliance Circle includes User, Identity Provider and Service Provider. It is the single-sign-on in which no need to authenticate again. Steps for single-sign-on being as given in Figure 4.

1) User → SP: Request for services.

2) SP → User: Choose the IDP,where the User federated his identity.

3) SP → User: SP acknowledges the IDP chosen and redirects the User to IDP sites.

4) User → IDP: Authenticate the User against the IDP and also contain the Information about the SP.

5) IDP → User: IDP generates SAML Token.

6) IDP → User: Send Response with the Token inside the message.

7) User → SP: Send the Token.

8) SP → IDP: SP Communicate with IDP for Verification of User Authentication.

9) IDP → SP: IDP response with assertion to the SP.

10) SP → User: SP Provide the services to the user on the basis of Assertion.

11) SP → User: Response with the service.

12) SP ↔User:SP Provide the service to the User.

While Authentication proves the identity, there is another concept which is called Authorization. The purpose of authorization is to provide access rights to the users with certain levels. After completing the Authentication, Authorization and Single Sign-On mechanism, Liberty Alliance specifications also support the Single Sign-Out mechanism.

The user sends request for single Logout to identity Provider. Identity Provider Forward this request to Service Provider. The service Provider does Process for Log out. Service Provider sends response to the Identity Provider. Identity Provider forward Single Log out confirmed to the user.The Liberty Alliance is Single-Sign-On and it authenticate and Authorize user Profiles. It also Provide the Scalability [14].

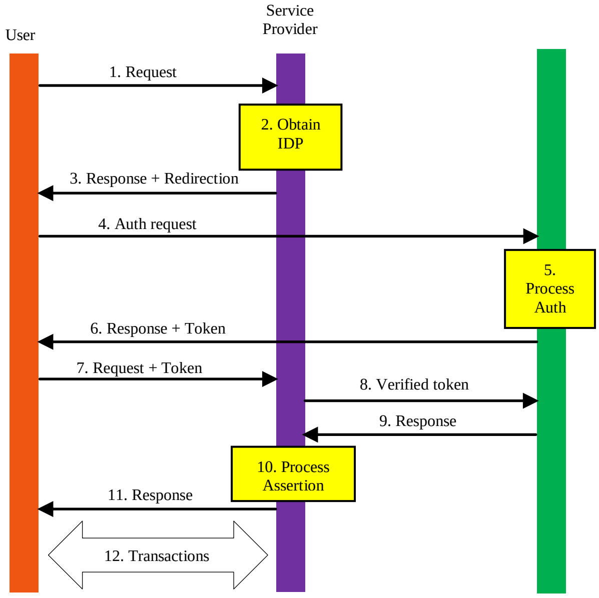

Fig 4: Single-Sign-On [14]

3.3 Windows CardSpace

Windows CardSpace is an Identity-metasystem which provides a way, for managing multiple digital identities of a user [15]. It is based on the Concept of an Information Card (InfoCard).It is claims based access platform/ architecture, developed for windows XP. It uses a plug-in for Internet explorer 7 browser. Microsoft CardSpace is built on WS-Federation protocol which consists of the following standards providing a basic model for federation between Identity Providers and Relying Parties: WS-Trust, WS-Security Policy, WS-Security. In this identity system three parties are involved: Identity provider: It issues digital identities (as trusted third-party). For example, a credit card provider might issue digital identities (security tokens) enabling payment. Even individuals can be Idp if they use self-issued identities like the signing of websites, using username and password. Relying Parties: It requires identities to provide a service to a user for example, a web site. Service requestor: they are individuals and other entities about whom claims are made.

The CardSpace identity metasystem makes use of XML based protocols, including the Web Services protocols and SOAP. The message flows of the CardSpace framework are as follows. Here (see Figure 5) CardSpace-enabled User agent (CEUA), Relying Party (RP) and Identity Provider (IDP) are involved[15].

1) CEUA → RP: HTTPGETLoginHTML Page Request.
2) RP → CEUA: HTML Login Page + InfoCard Tags (XHTML or HTML object tags).
3) CEUA ↔ RP: CEUA retrieves security policy via WS Security Policy.
4) CEUA ↔ User: User picks an InfoCard.
5) CEUA ↔ IdP (Identity Provider): User Authentication.
6) CEUA ↔ IdP: CEUA retrieves security token via WS Metadata Exchange and WS-Trust.
7) CEUA → RP: CEUA presents the security token via WS-Trust.
8) RP → CEUA: Welcome, you are now logged in.

The messages in steps 3, 5, 6, and 7 must be carried over and SSL/TLS channel to preserve their confidentiality.

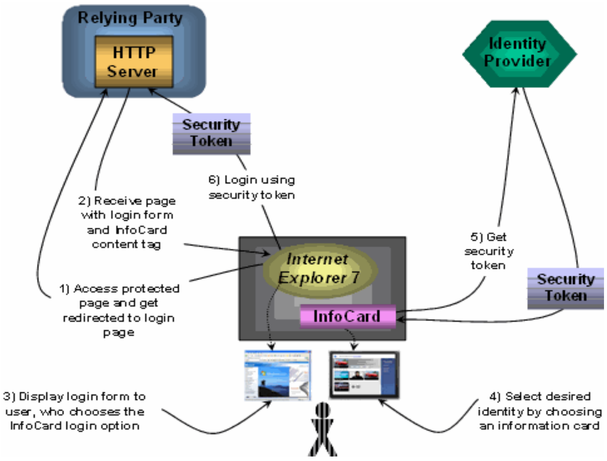
Relying Party
HTTP Server
Security Token
Identity Provider
6) Login using security token
2) Receive page with login form and InfoCard content tag
Internet Explorer 7
InfoCard
5) Get security token
Security Token
1) Access protected page and get redirected to login page
3) Display login form to user, who chooses the InfoCard login option
4) Select desired identity by choosing an information card

Fig 5: CardSpace Model of Identity Management [15]

They are more flexible than simple user names and passwords. It employs strong cryptography, which makes their use more secure than passwords. It can potentially present any type of identity claim that makes sense to all of the interacting parties and which users are willing to release. The CardSpace framework is criticized due to its reliance on the user's judgment of the trustworthiness of an RP. Most do not pay attention when asked to approve a digital certificate of an RP, either because they do not understand the importance of the approval decision or because they know that they must approve the certificate in order to get access to a particular website. RPs without any certificates at all can be used in the CardSpace framework. In a case where a single IdP and multiple RPs are involved in a single working session the security identity metasystem within the session will rely on a single layer of authentication that is the authentication of the user to the IdP. If a working session is hijacked or the password is cracked the security of the entire system is compromised. To defeat the security imitation mentioned above use the Zero-Knowledge Proofing, Selective Disclosure, Anonymous Credential [15].

3.4 PRIME

Privacy and Identity Management for Europe (PRIME) Project Produced Privacy architecture, and a Prototype and various application Scenarios[16]. Three parties are involved in PRIME: User, Service Provider and Certification Authority. User requests for services or resources to service provider and Service Provider provide the services as per user demand. Certification Authority is a special type of service provider is certifying authority that issues certificates that is digitally-signed statement. The PRIME involves four cryptographic tools namely secure communication, anonymous communication, pseudonyms, credentials and proofs of ownership of credentials. The execution of transaction is given in Figure 6.

PRIME is an User-controlled privacy-enhancing means that each individual user is put into control w.r.t. His/her PII as much as possible. It is Comprehensive means bringing diverse research areas (cryptography, system architecture, policies) and prototypes together e.g. Designing and evaluating early prototypes, learning some lessons how to integrate their achievements, and closing the remaining gaps. It is a large scale means that system architecture, security and privacy mechanisms, prototypes, terminology, and tutorials are developed, presented to the public and evaluated. The

Limitations of PRIME is the product is not standardized and it is only possible unless it is interoperable with existing systems. It has its middleware which should be implemented on senders and receiver side console, which is an extra overhead [16].

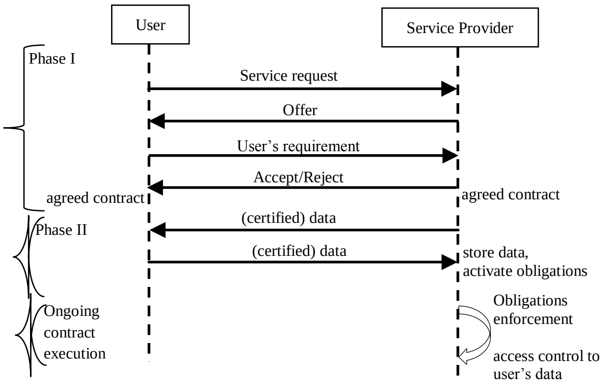

Fig 6: Execution of a transaction[16]

3.5 OpenID

OpenID is a Safe, Faster, and Easier way to log in to websites. OpenID is a decentralized model for identity management, which allows service providers to delegate the authentication of users to identity providers. In this model, the identity of a user is represented by a URL, called an OpenID identifier. Hence, users don't need to create a separate account for each site; instead, they just have to use their OpenID identifier, and the authentication procedure will be conducted through the user's identity provider [17]. OpenID is supported in many programming languages like Java, PHP, Perl C/C++, C##, python.

The Flow of OpenID as shown in Figure 7and given below [17]:

The user requests access to a service or resource at the Service Provider site.The Service Provider requires the authentication of the user and asks for his OpenID identifier. The user provides an OpenID identifier. OpenID 2.0 allows the user to simply provide the identifier of his identity provider, enhancing this way his privacy by reducing the chances of being traced through his identifier.The SP performs a discovery process using the supplied identifier to locate the IdP of the user.The SP and the IdP perform an association process, that is, they generate a shared secret through a DiffieHellman key exchange.The SP constructs an authentication request and redirects the user to the IdP site through an HTTP redirection.The user gets authenticated by the IdP, for example, by providing his credentials.

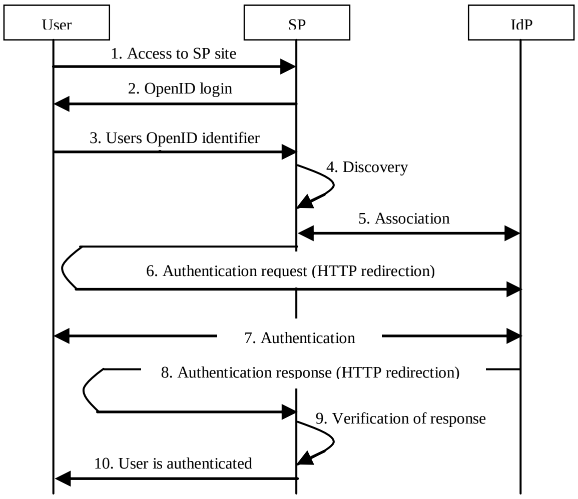
User
SP
IdP
1. Access to SP site
2. OpenID login
3. Users OpenID identifier
4. Discovery
5. Association
6. Authentication request (HTTP redirection)
7. Authentication
8. Authentication response (HTTP redirection)
9. Verification of response
10. User is authenticated

Fig 7: OpenID Authentication protocol [17]

The IdP constructs an authentication response, which contains an assertion about the result of the authentication. The IdP signs the request. The user is then redirected back to the SP site in order to continue with the authentication process.The SP verifies the authentication response and reads the attribute values included within.The user gets authenticated at the SP site and is able to access to the requested service. It provides the control of sharing information and faster & easier registration login. The limitations of OpenId are Phishing Attacks.

3.6 OAuth

In OAuth is an Open Authentication method where user can share his stored resources to one site with another site without having to hand out his/her username and password[18]. It is flexible and designed to work with mobile devices and desktop applications. Example of OAuth is given as below. User as BOB, Service Provider as Fuji (a photo sharing site), 2 photos BOB uploaded are protected resources. BOB wants to share his vacation journey photos with his friends, BOB using Faji, a photo sharing site, for sharing his journey photos, he signs with faji.com with his account and share his photos which he marks private. BOB wants to share his photos with his grandma but his grandma has no internet connection so BOB plans to order prints and have them mail to grandma. Being a responsible person BOB use, Beppa, environmentally photo printing service. Here Beppa is the consumer and must use OAuth to gain access to the photos in order to print them. In Beppa, printing service, 3 locations or supports or options are there: flicker, Fotolog and Fuji. When Beppa added support for Fuji photo import, Beppa obtain a consumer key and consumer secrete from Fuji to be used with Faji'sOAuth-enabled API. Beppa requests from Faji a request token at this point, the request token is not User specific and can be used by Beppa to gain User Approved from BOB to Access his Private photos. When Beppa receive the request token, it redirects BOB to the Fuji OAuth User authorization URL with the request token and asks Fuji to redirect BOB back once approval has been granted to the URL. After successfully logging into Fuji, BOB is asked to grant access to the pipe. Fuji informs BOB and the type of access being granted. BOB can approve or deny access. BOB waits for Beppa to present him with his photos fetched from his Fuji account. While BOB waits, Beppa uses the authorized request token and exchanges it for an access token. Here Request tokens are only good for obtaining user's approval while access tokens are used to access protected resources, BOB's photos. In this case Beppa exchanges the request token for an access token in the first request and in the second request gets the photos. BOB show that Beppa capture his photos without using his username and password. OAuth use Digital Signature, Hash Algorithm, Shared Secret, Nonce, TimeStamp. If the OAuth standard is extended with support for info cards or other functionality in the future, it might easily be supported in your application. It is Easier to manage or maintain configure them for example extranet login models with mixed authentication like SAML. It is Less data to store on servers.

3.7 OneLogin

OneLogin is the Single-Sign-On and Identity management for Cloud based Applications. It is nice web applications for Saas. It is used to improve the security of Saas Apps, having a Centralized Password, user can have a more secure password throughout his network of apps because the user does not have to remember them all. OneLogin works by installing a browser extension which effectively Pastes the credentials into the apps and logs user in. It supports the major Browser like Firefox, Safari, Chrome and also supports the major OS's- windows, Mac, Linux. The Active Directory &LDAP, are available. It also provide the De-Provisioning. OneLogin's Cloud identity platform comes complete with secure single sign-on for web, mobile and iPad, federated search, user provisioning, deep directory integration with real-time user sync, out-of-band multi-factor authentication, VPN integration and compliance reporting. OneLogin's catalog contains thousands of pre-integrated applications, including Google Apps, Innotas, LotusLive, NetSuite, Microsoft Office 365, Oracle CRM On-Demand, Salesforce.com, Success Factors, WebEx, Workday, Yammer, Service Now [19].The Advantages of OneLogin are [19]:Beautiful UX (User experience), Simple to set up, easy to use, Cross Platform and Cross Browser Support, Autologin from email links- nice, 2 factor authentication available, Directory Integration (AD, LDAP), Easy to add new apps (including custom apps). The limitations of OneLoginare [19]:Doesn't do role based security, only application level, De-provisioning doesn't delete or lock the applications, only prevents access to onelogin.

3.8 Windows Identity Foundation

Windows identity Foundation is a Microsoft software framework for building identity aware applications. It is a set of .NET Framework classes. It is a framework for implementing claims-based identity in applications. The web services that use Windows Identity Foundation, the .NET frame work version 3.5 SP1 [20]. The Characteristics of Windows Identity Foundation are as below [20].It allows developers to build claims-aware applications by providing a application programming interfaces (APIs). It provides templates which building claims-aware applications.It provides utilities that create a trust relationship between a Relying Party application and a Security Token Service (STS).It also provides a ASP.NET controls that help developers create web pages in claims-aware applications.It includes a utility that helps developers translate between claims and NT tokens. It includes functionality that allows identities to be maintained across multiple service boundaries. It provides tools to help developers build custom security token services using ASP.NET.The Claim-based identity involves Claim, Security Token, Security Token Service and Relying Party. Claim is identity information such as a name, email address, age. In Security Token the user delivers a set of claims along with his request. In a Web service, these claims are carried in the security header of the SOAP envelope. In a browser-based Web application, the claims arrive via an HTTP POST from the user's browser, and may later be cached in a cookie if a session is desired. They must be serialized somehow, and this is where security tokens come in. It is a serialized set of claims that is digitally signed by the issuing authority. In security token service it is the plumbing that builds, signs, and issues security tokens according to the interoperability protocols. In Relying Party When you build an application that relies on claims.

4. REVIEW OF IDENTITY MANAGEMENT FRAMEWORK IN CLOUD

A comparison of different identity frameworks is given in Table 1. It also suggests that in some of the identity frameworks, the registration and identity provider initiation are not required. Although most of the frameworks support single-sign-on, earlier identity frameworks were adopted by e-mail providers and corporate organization use and government; currently they are extensively used in social networking sites and mobile apps.

5. CONCLUSION

In this paper, we have discussed the concept of Cloud Computing, Identity Management and reviewed different Identity Management Frameworks. An Identity Management Framework helps align Identity Management initiatives with the organization's business goals and security strategy. It also focuses on issues related to Authentication, Authorization, Integrity, Non-repudiation, Data Confidentiality, Provisioning and De-provisioning.Based on the requirements of an application and regulatory compliances; suitable framework can be chosen.

Table 1. The Comparison of Different identity Frameworks

Identity Framework	**SAML**	**Liberty Alliance**	**Windows Cardspace**	**PRIME**	**OPENID**	**OAUTH**	**OneLogin**	**Windows Identity Foundation**
Started in	2001	2001	2003	2004	2005	2006	2009	2011
Current Version	SAML 2.0	ID-WSF 2.0, Liberty	.NET framework 3.5	PRIME	OPENID 2.0	OAUTH 2.0	OneLogin	WIF 4.5
Registration required?	NO	Yes	Manifested through the installation of managed cards into the selection	Restricted to registered user	NO	Explicit identity services pre-register for a consumer key & secret	Yes	Yes
Protocols	SAM,	LDAP,XML	XML based	Cryptographi	XRDS,	JSON,HTTP	RDF,X.509	WS-Trust, WS-

Used	XML, SOAP, HTTP			c protocols	HTTP			Security, WS-Federation
Relying party/service provider initiated	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Identity provider initiated	Yes	Yes	NO	Yes	NO	Expected for OAUTH V2.0	Yes	Yes
Main Purpose	Single-sign-on for enterprise users	Create an open network identity infrastructure	Single-sign-on for websites	For Data Minimization	Single-sign-on for Consumers	API authentication between applications	Single-Sign-On for Companies to secure access web application	Tempering/Disclosure of Credential or other Sensitive data
Currently used in	Google Apps	Healthcare, govt, Financial services	Windows Vista	Android Apps	Google, Yahoo, Facebook	Twitter	Mobile Apps	Yahoo, Google

6. REFERENCES

[1] Mell, P., and Grance, T. 2011. The nist definition of Cloud computing (draft), NIST. [Online]. avilable: http://csrc.nist.gov/publications/drafts/800-145/Draft-SP-800-145_Cloud-definition.pdf.

[2] Bhavesh, B., Patel, A., Patel, D. R. and Patel, H. 2012. Incorporating honeypot for intrusion detection in Cloud infrastructure. In Proceedings of the 6th International Conference on Trust Management.

[3] Angin, P., Bhargava, B., Ranchal, R., Singh, N., Linderman, M., Othmane, L. Ben and Lilien, L. 2010. An entity-centric approach for privacy and identity management in Cloud computing. In Proceedings of the 29th IEEE Symposiumon. IEEE in Reliable Distributed System.

[4] Wikipedia. 2010. Identity management systems.

[Online].avilable:http://en.wikipedia.org/wiki/Identity_management_systems.

[5] The Laws of identity. [Online]. avilable: http://msdn.microsoft.com/en-us/library/ms996456.aspx.

[6] king, C. and Perkins, C. 2013.The role of identity management in information security: part 1-The planning view. [Online]. avilable: www.znet.com.

[7] A. Gopalkrishnn, "Cloud Computing Identity Management", Journal of SETLabs Briefings, 2009, Vol. 7, No. 7, pp. 45-54.

[8] Gellman, R. 2009. Privacy in the Clouds: risks to Privacy and Confidentiality from loud, World Privacy Forum.

[9] Telecom Review. 2011. Importance of identity management in Cloud. [Online]. Available: http://telecomreviewna.com/index.php?option=com_content&view=article&id=92:importance-of-identity-management-in-Cloud-computing&catid=37:november-december-2011-issue&Itemid=73.

[10] Identity and Access Management: IAM Architecture and Practice. [Online]. Available: http://mscerts.net/image/201011/IAM%20Architecture%20and%20Practice_2.jpg.

[11] K. D. Lewis, and J. E. Lewis, "Web single sign-on authentication using SAML", International Journal of Computer Science Issues, 2009, Vol. 2, pp. 41-48.

[12] J. Somorovsky, A. Mayer, A. Worth, J. Schwenk, M. Kampmann, and M. Jensen, "On breaking SAML: Be whoever you want to be," In WOOT, 2012.

[13] Cantor, S., Kemp, J., Champagne, D., Aarts, R., and Kavsan, B. Liberty ID-FF Bindings and Profiles Specification, Version: 1.2-errata-v2. 0. [Online]. Available: www.projectliberty.org.

[14] Dwiputera, A. F., and Ruppa, I. S. 2012. Single sign-on architecture in public networks (Liberty Alliance). In Proceedings of the INFOTECH seminar on advanced communication Services (ACS).

[15] Bhargava, B., Singh, N., and Sinclair,Asher. 2011. Privacy in Cloud Computing Through Identity Management. Technical Report. Computer Science, Purdue University.

[16] Camenisch, J., Shelat, A., Sommer, D., Fischer-HÄubner, S., Hansen, M., Krasemann, H., Lacoste, G., Leenes, R., and Tseng, J. 2005. Privacy and identity management for everyone. In Proceedings of the Workshop on Digital Identity Management.

[17] Nunez, D., Agudo, I., and Lopez, J. 2012. Integrating openid with proxy re-encryption to enhance privacy in Cloud-based identity services. In Proceedings of theIEEE 4th International Conference on Cloud Computing Technologyand Science (CloudCom).

[18] Hueniverse Beginner's Guide to OAuth. [Online]. Available: http://hueniverse.com/2007/10/beginners-guide-to-oauth-part-ii-protocol-workflow

[19] OneLogin. [Online]. Available: http://www.justinpirie.com/2010/03/onelogin-saas-app-review-1-the-good-bad-and-ugly.

[20] Windows Identity Foundation. [Online]. Available: http://en.wikipedia.org/wiki/Windows_Identity_Foundation.

[21] Single-Sign-On. [Online]. Available: http://en.wikipedia.org/wiki/Single_sign-on.