A Federated Identity And Access Management For Cloud Computing Model Based On Single Sign On

Alyaa M. Ghazi, Mahmood K. Ibrahem

Abstract— Federation Identity is an expansion of identity and access management to different security fields. The objective of this paper is to give the participation of identities, so a client can be validated a single time and afterward get to resources and applications over multiple domains which called Single Sign On (SSO) that give three main services (authentication, data integrity and confidentiality) by suggesting a secure model for registration and login using RSA & SHA512 for encryption. Single-Sign-On (SSO) suffer from security problems such as user credential leakage and single sign out which solved by proposed a new system to overcome the weakness of SSO system. The proposed system implemented for three different applications and shows successful results.

Index Terms— Identity Federations Management; Authentication; Security; Single Sign On, Signature; Decryption.

1 INTRODUCTION

Federated Identity Management (FIDM) is an idea that permits cooperative on identity operation, policies and technologies across company sides. It is considered a promising concept to simplify trust resources participation between cooperative partners in Information Technology (IT) environments, and show that the individuals move between companies borders [1].

So, providing a successful service in a new way to the user is cloud computing which based on service provided. Cloud computing can be categories to three type of services; Infrastructure as a Service, (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) [2].

- IaaS manage the facility of computing or storing or some other equipment. One of the cloud providers that offering IaaS is amazon, EC2 and S3 are one of these administrations.

- SaaS manage the utilization of any assistance or application using the cloud. One of the models that give mutual effort on various applications is Google Apps, like project or event management etc. via the web.

- PaaS gives stages in regards to the operation system and framework programming that utilized to develop custom application by the customers. Customers rearrange and build up the applications on a practical stage. An example of PaaS is Microsoft Azure [3,4].

2 LITERATURE REVIEW

A brief literature survey is presented for existing works related to integrated federated identity management related to this paper, it can be summarized as follows:

PRIME (Privacy and Identity Management for Europe) (2004–2008) [5] This project was focused about the diffusion of individual information and with secure communication to monitor people information. PRIMELIFE (2008–2011) [6] implemented the understanding of identity management issues for practical life which including privacy on the web and applications to expand friendly devices for identity management. Integral Federated Identity Management for Cloud Computing (2012) [7] shows a new model and platform for federated identity management that mapping from Software as a Service (high level identities) to Infrastructure as a Service (low level identities). ABC4 Trust (2013 – 2015) [8] The project proposed unknown qualifications and the utilization of clingy policy. The access control schema utilizes identification proof cards, which require equipment to access the clients. GE´ANT (2015 – 2016) [9] help to provide network, technology and development network in the

research and education. PRISMACLOUD (Privacy and Security Maintaining Services in the Cloud) (2015 – 2018) [10]; The goal is to create cryptographic equipment that save the privacy and protection of client information during life in the cloud. CREDENTIAL (Secure Cloud Identity Wallet) (2015 – 2018) [11] The version of CREDENTIAL is to create arrangements with cryptograph to ensure client data. CREDENTIAL venture gets financing from the European Union's 2020 Research and Innovation program Horizon. Cloud Bursting Galaxy (Federated Identity and Access Management) (January 2020) [12], Utilized web protocol best-practice so the client data never been sent or store. Executed the methodology in the Galaxy stage, which is utilized over the world for enormous scope biomedical analyses.

3 FEDERATED IDENTITY AND ACCESS MANAGEMENT ARCHITECTURE

Federated identity management mentioned to the technologies, agreements and principle that enable the versatility of attributes, identities and privileges over various enterprises, supporting a huge number even millions of clients and various applications. At the point when various organizations execute able to exchange and make use of information federated identity schemes, a representative in one community can utilize a single sign-on to get to access over the federation with confidence connections related with the identity. For instance, a worker may sign onto his/her corporate intranet and be validated to implement approved capacities and access approved administrations on that intranet. The representative can access their medical advantages from an outside health-care insurance supplier without having to reauthenticate. Past SSO, federated identity management gives different abilities. One is a normalized method for speaking to attributes. Progressively, digital identity consolidate attribute other than just an identifiers and verification data, (for example, passwords), Instances of attribute incorporate account number, physical area, document ownership and organizational jobs. A client may have various identifiers; for instance, every identifier might be related with a unique job with the access permissions [13].

Fig. 1. Explain the entities and information's flows in the identity management architecture. A principal is an identity processer. Commonly, this is a human client that looks for access to services and assets on the system. Client devices, operator procedures, and server frameworks may also work as administrators. Principals verify themselves to a character supplier. The personality provides partners verification data with a head, just as properties and at least one identifier. Progressively, digital identities combine properties other than just an identifier and validation data, (for example, passwords). An attribute administration deals with the maintenance and performance of such attribute. For instance, a client needs to give a delivery heading each time a request is set at another internet merchant, and this data should be modified when the client leave. Identity management enable the client to give this data once, so it is kept up in a solitary spot and discharged to information customers as per approval and security arrangements [14].

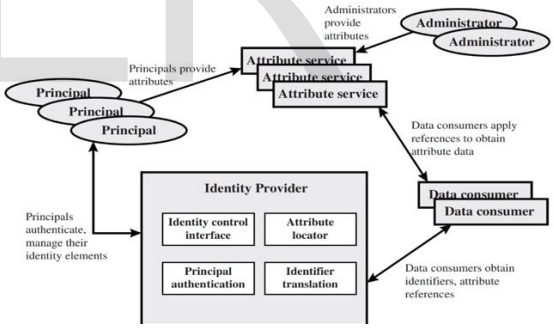

Fig. 1. General Architecture of Identity Management [13]

4 SINGLE SIGN ON ENABLING TECHNOLOGIES AND PROTOCOLS [15]:

The end-users can use a many of protocols to implement the SSO, for example Kerberos and Security Assertion Markup Language (SAML), etc. A Few of them are listed:

1. Kerberos: For the communities that need to authenticate the users by using the single sign on model to different applications over multiple technologies the Kerberos is a good

schema but the system applications should support Kerberos model for this work. Kerberos is a protocol of network authentication which works depending on the ticket (security token) to permit the nodes to communicate over a non-secure network to confirm their identity to each other in a safely method. The clients require the ticket from Ticket Granting Service (TGS) which is a service working on the server on their network, this what the customer need to do to obtained on secured service. After the client obtaining the ticket, he/she request a Service Ticket (ST) from another service running on the same network called an Authentication Service (AS). Then use the Service Ticket by the client to authenticate the required service. Both of Ticket Granted Service (TGS) and the Authentication Service (AS) running inside an closed service called the Key Distribution Center (KDC).

2. Lightweight Directory Access Protocol (LDAP): LDAP used for inquiries the directory servers which working on gathering information about an organization for example users' names, user's address's, telephone numbers and certification. Active Directory of Microsoft's version of LDAP used to give the authority true SSO using Kerberos but for Window's environment only. Major LDAP is more functional than structure authentication into each application for application authentication.
3. RADIUS Protocol: is a service which enable the users to request a remote authentication. The protocol is used for authenticate the users from all over the country for example the people that connect by Virtual Private Network (VPN). RADIUS server may usually invisibly run on UNIX or windows tool when provided with the user certification, able to support multiple authentication technique such as PAP, PPP or Unix login. Also is a connectionless client/server depending on User Datagram Protocol (UDP).
4. Agent Scripts: Can use the programming code (Script) that work on a central authentication authority for synchronize the user's credential over the systems when security rules are revised or passwords are changed. Extensible Markup Language (XML) codes and Structured Query Language (SQL) codes can be used for encryption the data in databases.
5. Cookies: The parts of programming that are downloaded onto the client machine are called cookies [16]. Cookies are token depending on SSO schema for Hyper Text Transfer Protocol (HTTP) service that are used to authenticate sessions during a certain period of time. After the cookie has been expired, the user should re-authenticate itself, which mean he/she should enter his/her credential to access the organization.
   - When a user visits a web page, his/her name can be stored in a cookie.
   - Next time the user visits the page, the cookie "remembers" his/her name.
6. Digital Certificates and Public Key Infrastructure (PKI): A Public Key Infrastructure (PKI) is a model which used for maintaining and saving encryption keys. This system used the public key cryptography for client authentication. The system depends on the function of Certification Authority (CA) for the management and issuance of the digital certificates and as a consequence the users' digital identity. Firstly, the user has to distinguish herself/himself to an authentication administrator which issues a public key certificate to the authenticated client. The user creates a token and includes its public key (digital certificate) in it and signs it with the user private key whenever the authenticated client needs to access a protected resource in next authentication request. The aim server communicates with the certification authority to check the identity of the user requesting on the reception request. There is a trust relationship as the latter's certificate between the primary and secondary certification authority (CA) that being issued by the previous one. For that any secondary certification authority can accept

the certificate that issued by the primary certification authority [17].

7. Web Security Service: Among different commercial entities, WSS supports cross domain and cross platform communication.

5 E-GOVERNMENT SECURITY CHALLENGES

All relationships are depend on trust, regardless of whether business relationships, private relationships and also government relationships with residents who are its customers. The most issue that related to e-government is trust issues. Trust in e-government is an assessment of "Whether political organizations work as per regularizing desires held by people in general" [18].

Resident's trust of e-government service is a complex idea that underlies a lot of connections. Resident's trust of government that offering necessary services over internet, which assume important roles in inescapable appropriation of e-government activities. Without clients' trust in government portals, information, processes and different parts of government offices the vision of completely electronic service conveyance will stay a difficult objective. This may result a few people don't confide in E-government and wouldn't utilize the open web services, despite the fact that the utilization of e-government is highly active [19].

5.1 Confidentiality:

The confidentiality service protects information from unauthorized access. At the point, when information leave one parts of a system, for example, a customer's PC in a system, it adventures out into a non-confiding in condition. Thus, the recipient of that information may not completely trust an outsider source data like a cryptanalysis or a man in the middle has eavesdropped on the information. This service utilizes algorithms encryption to guarantee that nothing of the sort occurred while the information was in the wild [20].

In E-government framework, significant data ought to be encrypted to keep unapproved clients from utilizing the original information. Can be used symmetric and asymmetric encryptions relying upon the idea of the framework. In the case of symmetric encryption, Advanced Encryption Standard (AES) and Data Encryption Standard (DES) are the most generally symmetric encryption methods.

In asymmetric encryption, the Rivest-Shamir-Adleman (RSA) public key cipher algorithm is the most widely utilized [21]. In our thesis will describe RSA algorithm.

-Rivest-Shamir-Adleman (RSA):

RSA is a cryptographic algorithm, meets the requirement of public-key systems. The RSA schema is a block cipher which the plaintext and ciphertext are integers between 0 and n-1 for some n (where n is number of bits).Size of n is 1024 bits, or 309 decimal digits [21]. Plaintext will be encrypted in blocks. So, each block will having binary values less than n. The size of block must be less than or equal to log2(n); the size of block is i bits, where $2i < n \leq 2i+1$.

Encryption and decryption process are as following structure:

for some plaintext M and ciphertext C:

$C = M^e \bmod n$ ----------(5.1)

$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$ ----.(5.2)

The sender and receiver both of them must know the value of n. The sender knows the value of e, and the receiver only knows the value of d. This is a public-key encryption algorithm with a public key of PU = {e, n} and a private key PR = {d, n}.

To ensure the requirements for public-key encryption of this algorithm is done, it must be done as follow [22]:

A. It is possible to find values of e, d, n such that Med mod n = M for all
M < n.

B. It is relatively easy to calculate mod Me mod n and Cd for all values
of M < n.

C. It is infeasible to determine d given e and n.

Fig. 2. & 3. illustrate Encryption, Decryption, Key generation and Process in RSA.

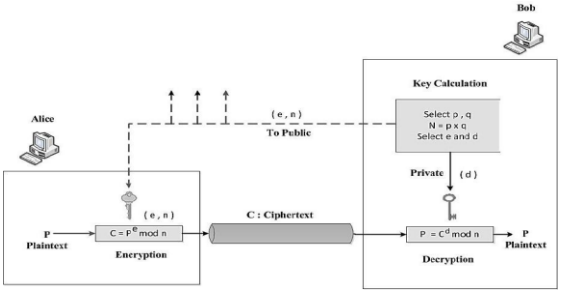

Fig. 2. Encryption, Decryption and Key generation in RSA [21]

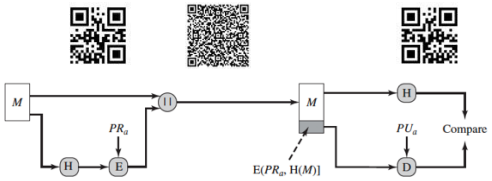

Fig. 3. RSA approach [21]

5.2 Data Integrity:

The integrity service protects data from buoyant threats, such as those that may edit it. Just like information confidentiality, in transition between the sending and receiving is oversensitive to a lots of threats from eavesdroppers ,hackers and cryptanalysts, whose goal is to intercept the data and changed it based on their intention. This service, through encryption and hashing algorithms, ensures that the integrity of the transient information is right [14]. Message authentication and cryptographic hash functions are commonly used to maintain the integrity of data.

- Cryptographic Hash Function

Hash function is one-way encryption function, accepts a variable length block of data as input and produces a fixed-size hash value. The main concept of a hash function is data integrity. When use a hash function to provide message validation, is often referred to as a message digest. A hash code does not use a key but only a function of the input message. The hash code is a function of all the bits of the message that provides an error-detection capability: A change on any bit or bits in the message results in a change on the hash code. A hash value (h) is generated by a function H of the form:

h = H(M) ……..…….(5.3)

Where M is a variable-length message and H(M) is the fixed-length hash value. The hash value is appended to the message at the source at a time when the message is assumed or known to be corrected. The receiver authenticates that message by re-computing the hash value.

6 PROPOSED FEDERATED IDENTITY MANAGEMENT ARCHITECTURE

In this proposed system, it is imperative to know which SaaS client is executing which activities on the infrastructure level. This not just empowers a fair accounting framework; it likewise allows an exact recognizable identification of clients for fine grained examining and access control in IaaS.

In view of these requirements, proposed another architecture for federated identity management the executives that integrates the SaaS and IaaS layers. Some parts are worked on the PaaS layer to hide the execution details of identity interpretation from the IaaS contractual. Likewise structured a way to deal with permit the SaaS client to utilize the SaaS application.

Fig. 4. shows a model architecture including all the elements described previously. The IaaS contractual acquires the contracted assets, as virtual machine occurrences, from two unique suppliers (IaaS Providers One and Two). The IaaS contractor then deploys his/her SaaS application over the contracted resources. It is important to install authenticate relationships between the provider of identity which responsible for the SaaS client' authentication.

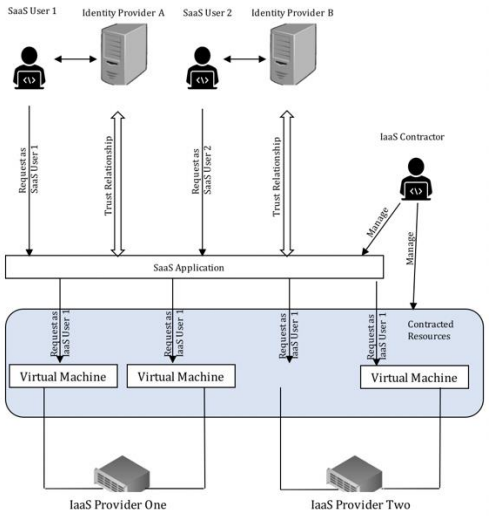

Fig. 4. Architecture of Federated Identity Management

6.1 The System Model:

The proposal system has been designed to achieve system secure, flexible and defender against hacker; the design of the proposed system consists of four main entities defined as follows.

1. Identity Providers (IdP): Ensure from the authorization and authentication of clients, as well as managing and partition credential information with various trusted SPs.
2. Service Providers (SP): Provide particular services to the clients who are authenticated by trusted IdPs.
3. Database (DB): Store data and provide facilities of searching specific record in given data.
4. Clients: User who are interested to get services from multiple SPs.

The overview of system as show in fig. 5. gives a general view of the system.

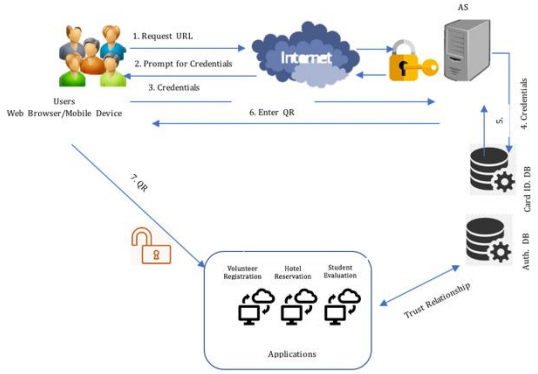

Fig. 5. System layout of the Proposed System

6.2 Algorithm of the Proposed System

Fig. 6. Illustrate the flowchart of the proposed model. When the user tries to access the applications is promoted with login page, if the user doesn't has an account, must registered. So, the user must entered his/her information and make a submit. Then, the server will check the ID if match with the database, then send the QR to the user via email. Login page will appear and the user can enter the username and password that entered in registration stage, if match then the page of camera will display to the user to enter the QR that received on the email. If successful, the homepage will appear, if not, error page will display.

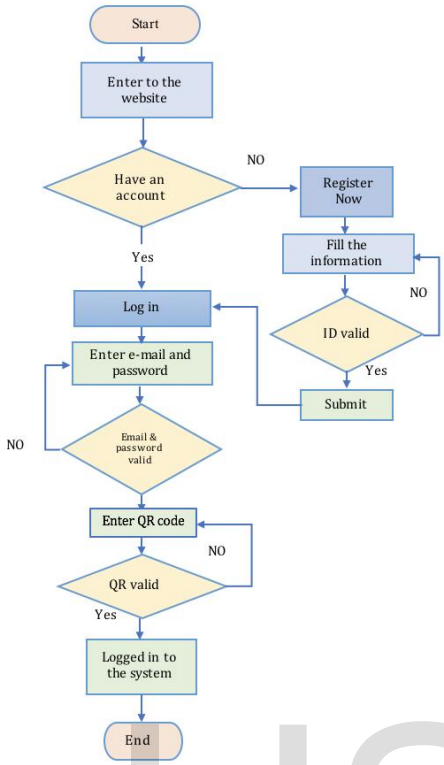

Fig. 6. Flowchart of the Proposed Model

6.3 Proposed System Protocol

The improved protocol for the proposed system is listed below:
Where:
A: Client, B: Application, ID: Identification No., AS: Authentication Server, ||: Concatenation, S: Signature, M: Plaintext block and C: ciphertext block.
Generate Public (PU) and Private keys (PR) of the server.
AS → All Sites: Send public key to all registered sites.

- **Registration Stage:**

-Create Account (All clients should create accounts)
Step-1:
Enter ID, Username, Email, Password.
Then:
A → B : M
B → AS : M

-Verification Stage:

Step-1:
Application send the information of registration (ID) to the main server to check if the client is verified.
If Succeed
Then, the main server will generate a QR depending on the ID and generate Private Key using RSA algorithm. Then ehash of the message using SHA512.
H = SHA512(M) ---------- (6.1)
and encrypt this message by private key of the server. The encryption results is the signature of the message
S = E(Private Key, H) ---------- (6.2)
AS : H(M)
AS : S=E(PRS , H(M))
AS → A : S=E(PRS , H(M))
And the bellow message will appear:
The QR has been sent to the (*Email*).
Step-2:
Otherwise, If the user enter ID already registered.
The message bellow will appear:
Username already registered please check your email.

- **Login Stage:**

Step-1:
User enter the email and password that created in the registration stage.
A → B : S=E(PRS , H(M))
Step-2:
Verify email and password.
If Succeed, login page with camera will appear to the client to insert the QR.
Step-3:
Otherwise: the message below will appear:
Your Login Name or Password is invalid.

- **QR Stage:**

Step-1:
After login stage, the user should enter the QR that received on his/her email and the bellow message will appear:
Welcome (Username) please provide your QR
Step-2:
Verify QR by decrypt the message using public key and compare the result.
B : M=D(PUB, E(PRS , H(M))).
if succeed then a homepage of the site will appear.
Step-3:

Otherwise: the message below will appear:
Your QR ID is incorrect.

- **Communication Stage:**

After the user is authenticated by the server, the information will be stored in COOKIE (which is a small file which are stored on a user's computer. Its designed to hold a modest amount of data specific to a particular client and website, and can be accessed either by the web server or the client computer) to implement the Single Sign On algorithm. So, the server will send notification to other sites to inform them that this user is authenticated and can access without login as shown in the following code.

- **Logout Stage:**

In this stage, worked on enhancement the SSO workflow. When the client signed out from on application, all other application will be also signed out once the page reloads. Also, the webpage will sign-out automatically once 3600 sec. the user didn't use the application.

6.4 Enhancement of The Federated Identity Management System

The proposed system is an enhanced version of the SSO work. Through studying single sign on, shown that there is some weakness suffer from. Some enhancements have been made to overcome the drawback and vulnerabilities of theses weakness and to increase its performance and strengthen its security and authentication. Enhancements can be described as follow:

1. In the first step, user should be register before access the application, the user cannot enter the username and password to the server without registration. The registration information will protect the system from any eavesdropping attack conducted to obtain any information to penetrate the system. Strong authentication protocol is used to control access to all sites.
2. The second improvement in the proposed system states, while it provides single sign on it does not provide single sign off, in the proposed model this problem is solved depending on cookies. So, when the user signs out in one of web sites, a notification will be sent to other websites tell them to logout. So, the other web sites will also signed out (just refresh the webpage) automatically.
3. The third enhancement in the existing models, the webpage will stay opened even the user doesn't used it. So, this disadvantage solved by using time limit of 3600 sec. and the sites will be signed out automatically if the user doesn't use the web site during this time.
4. The fourth improvement in the login stage, where the user/client must use the username and password, which were previously sent to the server during the registration stage and then if valid, enter the QR which is signed by the server. This procedure has been added in order to protect the system from unauthorized users to enter the system.
5. The proposed protocol suggests send the signed QR to the user via email (which the user inserted during registration stage after verify from the authority of the user). The generation of QR depending on the national ID which is uniquely to each user and when the user lose it, he/she can click on the button of (LOSE) and the server will stored the lost ID in revocation list and, user can create new one to prevent any eavesdropping.

6.5 System Implementation

This section presents the implementation of the proposed architectures over three web sites: Site '01' (Hotel Reservation), site '02' (Student Evaluation), and site '03' (Volunteer Registration) as shown in Fig. 7. When the user try to access the website first will display the login page, he/she must enter his/her own ID of Unified Card or National Identification Number (NID) (Which is unique to every user), the name of the user, and his/her own email, then the request will delivered to the authentication server trying to authenticate the user. So, the server will compare the entries, if authentication succeeds, then the server will send

the QR to the user via the entered email, if not, error page will appear. After that, the client will use the QR that received on the email to access the applications and home page will appear. The user can browse the other websites without enter the QR again.

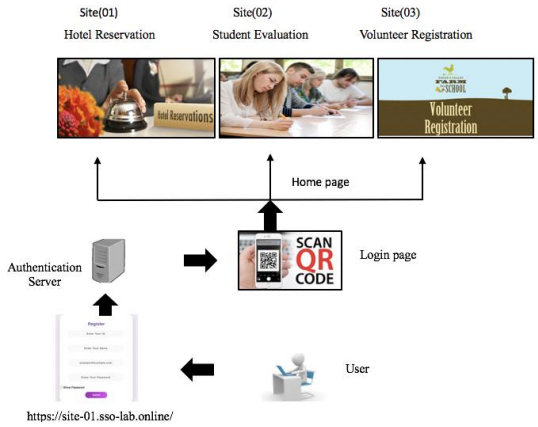

Fig. 7. Actual sites of the proposed Cloud Computing Applications

7 RESULTS

-A process of an authentication consists of two steps:

• Identification step: Presenting an identifier to the authentication system (clients should be assigned carefully avoids errors, because authenticated entity are the basis for security service).

A service provider can receive trustworthy information from the authentication server. Indeed, by sharing ID and Name with the identity provider, the service provider can read a part of a token it receives and compare it to the database.

• Verification step: Presenting or generating authentication information that corroborates the binding between the entity and the identifier.

- First verification, If the result matches the database then, the service provider knows that the information does come from the other entity possessing the key which is the identity provider.
  Fig. 8 show the registration page that will show when the user try to create an account through the site (01) https://site-01.sso-lab.online/ in Google Chrome.
- Second verification, the customer should entered the username and password that made in the registration stage as shown in Fig. 9. So, when the result match the database, the server will generate QR depending on the ID, and send it to the client to use it to access the application.

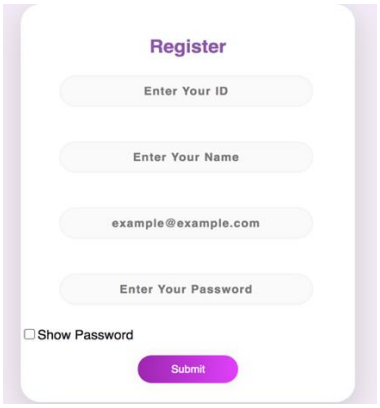

Fig. 8. Screen Shoot of Registration Step

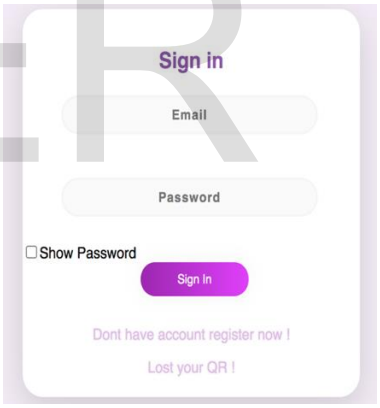

Fig. 9. Screen Shoot of Login Step-1

- Third verification when the client entered the QR as shown in fig. 10.. The user must first login on the application and stored it in the cookies file to succeed the single sign-on and can access the other application without entered the credential again. Then, the application will verify if the QR is correct which is depending on the ID to avoid any attempt to hackers to use the same QR. Otherwise the user would be prompted with an error message. After

successfully login, the home page of the site will appear as shown below in Fig. 11.

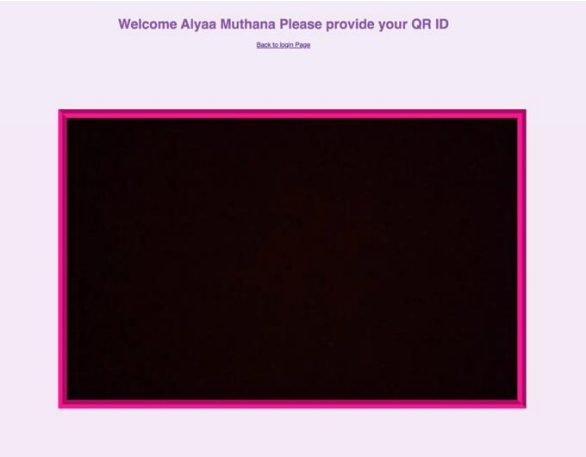

Fig. 10. Screen Shoot of Login Step-2

Fig. 11. Home Page of Site(01)

The user can browse to the site(02) Student Evaluation or site (03) Volunteers Registration without require to scan the QR again because the user already authenticate in identity provider and stored his/her identity in the database as an authenticate user.

Also one of disadvantage of SSO, while it provides single sign on it does not provide single sign off, in the proposed model this problem is solved depending on cookies. So, when you sign out in one of web sites, the other web sites will also signed out (just refresh the webpage) automatically. The second problem, in the existing models, the webpage will stay opened even the user doesn't used it. So, this disadvantage solved by using time limit of (3600 sec), and the sites will be signed out automatically if the user doesn't use the web site in this time.

-Database MySQL Server

In the database server three tables are created, the first one is filled by the admin with ID and name of the users as shown below fig.12 . The second table is a user table to record user's information such as user name, password, email, Fig. 13.

The third table, if the user identity has been stolen, will become a victim of identity theft, the user should take immediate action. A fraudster could have several pieces of your personally identifiable information (PII). This is unique, sensitive information like your national number that can enable someone to commit more than one type of fraud in your name. It's important to remember that even if the user have been diligent in trying to keep your personal information private, cybercrimes like the Equifax data breach show your information can still be vulnerable various ways.

So the user can click on icon (LOSE) in login page and enter the username and password and the server will take up the old QR from the database and put it in the revocation list and generate new QR to the user and send it via email, Fig. 14.

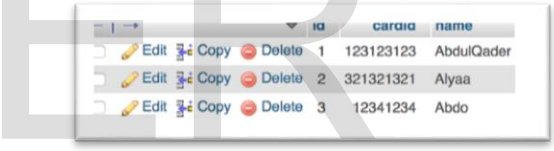

Fig. 12. Card ID Table

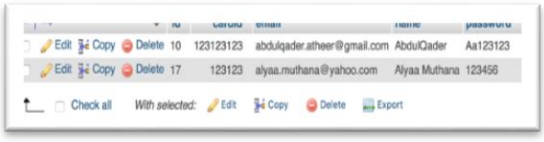

Fig. 13 User Database Table

Fig. 14. Revocation List Table

8 DISCUSSION

This technology represents the future of E-Government & E-Application in general. Due to the increase in applications, this problem has confused

the business and weaknesses in security, so this model presents a solution towards future of this applications. FIDM is not a security model, rather than is a management model, it requires a registration for all the potential users on the server, which depends on the national ID, so whenever the ID is available the application becomes smoother and easier. FIDM based single sign on is implemented using cPanel, which is a simple and easy program and compatible to all web browsers, google chrome, safari, etc. It's enabled the administrator to use the ready commands without use the complexity of configuration. The implementation provides a control for all users to all application by just one time to log in. The previous works implemented FIDM architecture based on just SSO, while the contribution of this paper is to design and implement FIDM architecture based on Single Sign on & Single Sign Off. In addition, add time expire to each application which put the application in log off if the user didn't use the application during this time, in contrast the previous works are just to single sign on.

9 CONCLUSIONS

The work in the proposed system focuses on the designs and architectures of the FIDM which represents a solution towards the future of E-Government & E-Application in general that enable the user to access the applications with only once sign-in with centralized database to facilities their work and reduce the time spending in log on procedure. The following conclusions summarize the results that records during proposal work time:

- The proposed system manages the problem of multi accounts that lead to weakness in the security and dada protection and to simplify the username and password management and faster access to applications by single account, which is applied in three different websites belong to different organization.
- The proposed system was demonstrated the work of applications that give desired security with less complexity.
- With SSO, the user can to utilize different services in the same domain or trust circle with one set of sign-in credentials.
- The chance of shutting down all applications with a single log out which named Single Sign-off also solved.
- The second problem that been solved which the client session normally remain open after the client has finished his/her utilization which lead to session hijacking, its solved by put a time expire.
- Besides, the application administrator also acquire features like ability to apply security polices for single clients on the infrastructure level.

10 REFERENCES

[1] Keltoum, Bendiab, and Boucherkha Samia, "A dynamic federated identity management approach for cloud-based environments." Proceedings of the Second International Conference on Internet of things, Data and Cloud Computing, (pp. 1-5), 2017.

[2] Gholami, Ali, "Security and privacy of sensitive data in cloud computing", Diss. KTH Royal Institute of Technology, a survey of recent developments. arXiv preprint arXiv:1601.01498, 2016.

[3] Kumar, P. Ravi, P. Herbert Raj, and P. Jelciana, "Exploring data security issues and solutions in cloud computing." Procedia Computer Science 125-691-697, 2018.

[4] Shaikh, Rizwana & Mukundan, Sasikumar, "Identity Management in Cloud Computing" International Journal of Computer Applications. 63. 17-19. 10.5120/10509-5467, 2015.

[5] Loruenser, Thomas, et al., "PRISMACLOUD tools: a cryptographic toolbox for increasing security in cloud services." 11th International Conference on Availability, Reliability and Security (ARES), Salzburg, pp. 733-741, IEEE, 2016.

[6] Sen, Arun Kumar, and Pradeep Kumar Tiwari. "Security Issues and Solutions in Cloud Computing." IOSR Journal of computer Engineering 19.2, 67-72, 2017.

[7] Stihler, Maicon, et al. "Integral federated identity management for cloud computing." 5th International Conference on New Technologies, Mobility and Security (NTMS), 2157-4960,IEEE, 2012.

[8] Bendiab, Gueltoum, et al. "FCMDT: A novel fuzzy cognitive maps dynamic trust model for cloud federated identity management." computers & security 86, 270-290, 2019.

[9] Avery, Atiya, and Dream Gomez. "Ethical Considerations of Online Identities." Proceedings of the 13th Pre-ICIS Workshop on Information Security and Privacy. Vol. 1. 2018.

[10] Kebande, Victor R., et al. "CVSS Metric-Based Analysis, Classification and Assessment of Computer Network Threats and Vulnerabilities." International Conference on Advances in Big Data, Computing and Data

Communication Systems (icABCD), Durban, pp. 1-10, IEEE, 2018.

[11] Nugroho, Eddy Prasetyo, and Rizki Cahyana. "A development of cloud-based PHP learning system.", 3rd International Conference on Science in Information Technology (ICSITech), Bandung, pp. 674-680, IEEE, 2017.

[12] Vahid Jalili, Enis Afgan, James Taylor, Jeremy Goecks, "Cloud bursting galaxy: federated identity and access management", Bioinformatics, Volume 36, Issue 1, 1 January 2020.

[13] William Stallings, "Cryptography and Network Security; Principals and Practice", 5th Ed, 2009.

[14] Shere, Rohit, Sonika Srivastava, and R. K. Pateriya. "A review of federated identity management of OpenStack cloud." International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE), Bhopal, pp. 516-520, IEEE, 2017.

[15] Bazaz, Tayibia, and Aqeel Khalique, "A review on single sign on enabling technologies and protocols." International Journal of Computer Applications 151.11,18-25, 2016.

[16] Pattan, Neha Gangadhar. "Setting cookies across applications." U.S. Patent No. 9,288,118. 15 Mar. 2016.

[17] Bazaz, Tayibia, and Aqeel Khalique, "A review on single sign on enabling technologies and protocols." International Journal of Computer Applications 151.11,18-25, 2016.

[18] Pradhan, Pratima, and Subarna Shakya. "Big Data Challenges for e-Government Services in Nepal." Journal of the Institute of Engineering 14.1: 216-222, 2018.

[19] Jameel, Arif, et al. "Assessing the Moderating Effect of Corruption on the E-Government and Trust Relationship: An Evidence of an Emerging Economy." ,MSPI Journals, Sustainability 11.23, 2019.

[20] Kebande, Victor R., et al. "CVSS Metric-Based Analysis, Classification and Assessment of Computer Network Threats and Vulnerabilities." International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD), Durban, pp. 1-10, IEEE, 2018.

[21] A. Abusukhon, Z. Mohammad and A. Al-Thaher, "Efficient and Secure Key Exchange Protocol Based on Elliptic Curve and Security Models", IEEE JordanInternational Joint Conference on Electrical Engineering and Information Technology (JEEIT), Amman, Jordan, pp. 73-78, 2019.

[22] Jana, Bappaditya, and Jayanta Poray. "A performance analysis on elliptic curve cryptography in network security." International Conference on Computer, Electrical & Communication Engineering (ICCECE), Kolkata, pp. 1-7, IEEE, 2016.

Author's Details:

Alyaa Muthana Ghazi and Mahmood Khalel Ibrahem.
Collage of Information Engineering / Department of Information & Communication Engineering.
Al-Nahrain University/ Baghhdad-Iraq.
alyaamuthana@yahoo.com
mahmoodkhalel@coie-nahrain.edu.iq