ISSN:2231-0711

Secure E-Government Framework: Design and Implementation

Mahmood Khalel Ibrahim, Mohammed A. Jabbar Hamid

Information and Communication Engineering Department,
College of Information Engineering,
Al-Nahrain University,
Baghdad, Iraq

***Abstract*— this paper suggests a security framework for e-government systems. The proposed system is a dynamic web application, represents a secure e-government model, provides the three main security services (authentication, confidentiality and integrity) and could be applied as a security reference model for e-government applications. National ID and password authentication scheme has been used in authenticating system's users. Advanced Encryption Standard (AES) have been used in encrypting citizen information and keyed massage authentication code algorithm (HMAC-MD5) has been used to guarantee the integrity of message contents and to authenticate the source of message. (RSA) algorithm has been used in (Digital Envelope) scheme for managing and distributing AES's key. RSA's private key, in turn, is encrypted by Triple Data Encryption Standard (3DES) using admin provided key .Three informational services were presented as e-services represented by (civil information system, passport issuance system and driving license issuance system). To prevent the backend sub-systems from attack, front-end sub-system (portal) was developed as a middleware system.**

***Keywords*—e-government, security framework, integrity, authentication, confidentiality, Portal, web application.**

I. INTRODUCTION

E-government is narrowly defined as the use of information technology, especially the Internet, to deliver government services and information to citizens, businesses, and other government bodies. E-government is the most important accomplishment of Internet. There is, however, one main downside that is occurring within each process of e-government, namely security. E-government processes and services must be secure as technology will allow and the users must follow the policies set to make their own actions safe. One of the primary challenges and obstacles to successful deployment and operation of e-government is security [1].

From a technical point of view, there are three types of threats that are found in both e-government and e-commerce, unauthorized access which affects confidentiality, unauthorized change or modification to the information which affects the integrity of information, and the threat that affects the availability of both information and services. There is also a fourth type, namely the threats against accountability such as denying involvement in a transaction [2].

Citizens feel vulnerable when using e-government systems. They want to have security solutions which provide subjective trust. Citizens have no opportunity to verify provided security. They have to accept a certain security uncertainty. The difference with e-commerce is that citizens using e-government will not accept the same uncertainties as they might when using e-Commerce systems. They need more trust when they use e-government systems [1].

II. E-GOVERNMENT SECURITY CHALLENGES

The most key issue about e-government is its trust problems. Citizen trust of e-government service is a complex concept that underlies a lot of relationships. Citizen's confidence of government providing imperative online services plays important roles in the widespread adoption of e-government initiatives. Without customers' trust in the government portals, processes, information, and other aspects of government, the vision of fully electronic service delivery will remain a challenging target. This may result some people do not trust e-government and wouldn't like to use the public service on the web portals, even though the application of e-government is highly efficient [3].

With development of e-government and increase of the public's reliance on information, security problem of e-government has become more and more prominent; more and more security bugs have been discovered in e-government system, attacks targeting at e-government system are emerging constantly.

E-government system is mainly faced with the following security challenges [4]:

A. *Confidentiality*: It is referring that information is subjected to attack of sniffer or amend in the process of transmission or storage. In E-government system, important information should be encrypted to prevent unauthorized users from using the original data.

B. *Data Integrity*: refers to maintaining and assuring that data received are exactly as sent by (i.e., contain no modification, insertion, deletion, or replay) and that the purported identity of the sender is valid. Message authentication codes are used for maintaining the integrity of data.

C. *Identity authentication*: In order to prevent illegal intervention, system should verify the legitimacy of the user's identity. This process is called "Entity Authentication". This can be done with one of three kinds of witnesses: something know, something possessed, or something inherent. Password, challenge-response, zero-knowledge, and biometrics are the most common authentication methods [5].

III. E-GOVERNMENT ARCHITECTURE FRAMEWORK

One of two types of architecture can be adopted by e-government systems: Centralized and De-Centralized. Centralized architecture standardizes the IT services across the government departments. It provides the services that span across various departments with a centralized resource allocation mechanism. Centralized model helps greater integration across various departments and the E-Governance Portal. De-centralized architecture is preferred when government agencies conduct separate implementations of different business processes across individual departments [6].

IV. WEB BASED APPLICATION VULNERABILITIES

With the increase of online commerce, Web applications have become fertile ground for attackers attempting to penetrate systems and misuse private data. This leads to a concern among consumers that their data and systems can be at risk due to Web vulnerabilities. The three most common Vulnerabilities of web applications are: SQL injection, Denial of Service and Cross-site Scripting [7].

A. SQL injection attack

SQL injection is a technique often used to attack data driven applications. This is done by including portions of SQL statements in an entry field in an attempt to get the website to pass a newly formed rogue SQL command to the database [8]. An even better approach is to use a parameterized command or a stored procedure that performs its own escaping and is impervious to SQL injection attacks.

B. Denial of Service attack

A denial of service (DoS) is an attempt to prevent legitimate users of a service from using that service. An attacker is able to recruit a number of hosts throughout the Internet to simultaneously or in a coordinated fashion launch an attack upon the target. One common method of attack involves saturating the target machine with external communications requests, such that it cannot respond to legitimate traffic, or responds so slowly as to be rendered essentially unavailable. Such attacks usually lead to a server overload [9]. There are three lines of defense against DoS attacks [9]: Attack prevention and preemption (before the attack) , Attack detection and filtering (during the attack) and Attack source trace back and identification (during and after the attack).

C. Cross-site Scripting attack

Cross Site Scripting (CSS) is one of the most common application level attacks that hackers use to sneak into web applications today. Cross site scripting is an attack on the privacy of clients of a particular web site which can lead to a total breach of security when customer details are stolen or manipulated. The goal of the CSS attack is to steal the client cookies, or any other sensitive information, which can identify the client with the web site. With the token of the legitimate user at hand, the attacker can proceed to act as the user in his/her interaction with the site – specifically, impersonate the user [10].

To prevent CSS attack, many web applications rely on the security session for authentication between individual HTTP requests.

V. E-GOVERNMENT PORTAL

The diversity of e-government applications and functions can be presented to users through various types of websites and portals organized within a whole-of-government architecture and applying the principles of no-wrong-door and content easily discovered on basis of user requirements, rather than government structures and integrated channel management. That portals, sometimes called “gateways”, aggregate and organize content and services, often with links to websites of individual ministries or programs. The goal of a portal is to efficiently guide users to the information and services they seek. National web portals can represent the face of a country to the world and the face of government to the citizenry [3].

The benefits of a Portal are immense. Portals act as a one stop resource for information. Having a secure e-government portal will reduce the costs for the government in delivering timely information to its citizens [4]. E-Governance Portal is built based on Service Oriented Architecture (SOA). Service Oriented Architecture (SOA) may be defined as a group of services that communicate with each other through data-passing or two or more services coordinating some activity [6].

VI. IDENTITY MANAGEMENT IN E-GOVERNMENT SYSTEMS

E-governments over the world need to develop digital identity management systems to provide services such as user identification, authentication, and authorization in an e-government environment. A digital identity management system is based on a schema for representing digital identities (a database subset, for example, that includes name, last name, date of birth, photo, certificate, serial number, etc.) and authentication mechanisms and protocols that entities use to demonstrate they are the owners of a given digital identity. Accordingly, the purpose of a digital identity is to tie a particular transaction or a set of data in an information system to an identifiable individual. With the help of a digital identity, a user can be identified and authorized to use a given resource or service [11]. Identities and identity management are of primary importance for governments as they encompass the identification of citizens and their interactions with public services and government. Trusted, secure and accountable identity management solutions are key e-government enablers [11]. Federated identity management plays a key role in this phase. Federated identity management (FIM) refers to an infrastructure that consists of technologies, standards and use-cases which serve to enable the portability of identity information across otherwise autonomous security domains. FIM enables users of one domain to securely access data or systems of another domain seamlessly, and without the need for completely redundant user administration. Use of FIM can increase security, lower risks, and reduce cost by eliminating the need to deploy multiple identity management systems. It can also drastically improve the end-user experience by eliminating the need for account registration through automatic "federated provisioning", i.e. authenticate the user once and use the same identity information across multiple systems (also referred to as single sign-on) [11].

VII. SYSTEM OVERVIEW

Figure (1) shows the proposed system layout. The proposed system is web-based application, prototyped on the internet and represent proposed security model for an e-government systems. The designed system consists of four application servers: Portal server, Civil Information server, Passport server and Driving License server. Portal server is the front-end system where the clients of the system are interacting with it and represent the gateway for the proposed e-government application. The three rest servers (Civil Information, Passport, and Driving License server) are representing the back-end system. Any user over the internet can reach the e-government portal and browsing its website.

The proposed system suppose that each citizen has unique identifier called National ID (NID) and assume generating NID for any citizen that does not possessed one. The portal provides services that target the citizen. To get one of these services , the citizen must be log in by typing his/her Full Name (first name ,father name , grand-father name) and his/her NID at login wizard. After that, the citizen can fill the information required in the order of that service and a message box will present telling the citizen about the date of order completion. Then, that order is hashed, encrypted and sends to the server of that service. Upon arrival , the order is decrypted , and the integrity of the data is checked , then the order is inserted in the service's database in order to processed it.

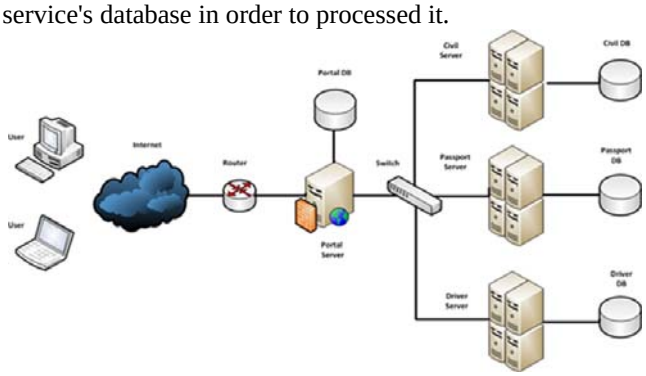

Figure (1) system layout

VIII. SYSTEM ARCHITECTURE

4-layer architecture was used in the proposed system. It consists of Four Layers: Presentation layer, business logic layer, data access layer and data store layer. Figure (2) shows the system architecture.

A. ***Presentation layer:***

The first layer is the presentation layer. This layer manages the input/output data and their display. The users are allowed to access the system by using any existing web browser software. This layer receives the information from the business logic layer and displays them in HTML pages form.

B. ***Business logic layer:***

This layer contains the business logic and set of operational rules particular to the application and talks to the data access layer to fetch data on which it has to apply rules , save updated data after applying rules to it and to perform operations and validate data. This layer consist of control module and six functional modules. The control module controls the flow of functions execution and

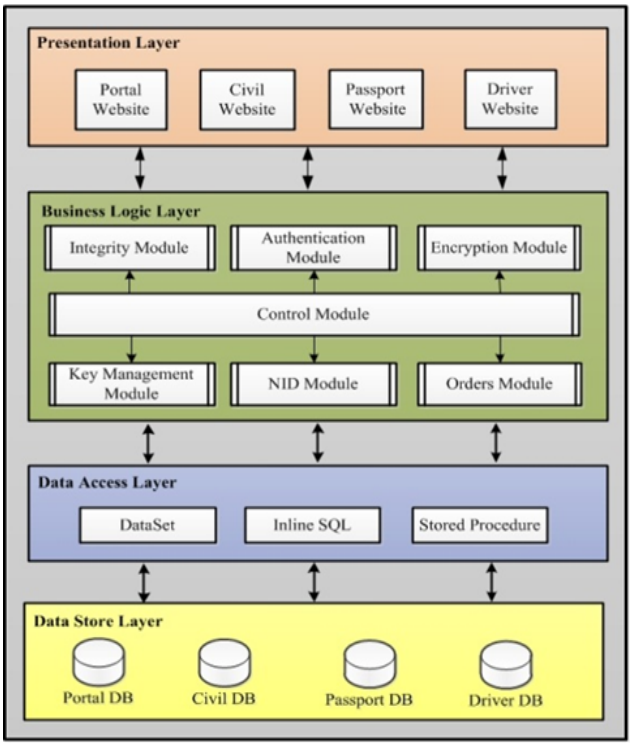

Figure (2) Proposed System Architecture

transferring required information between them and data access layer. The functional modules are:

a) Authentication module

This module responsible for authenticating the clients of the system before granting them an access to the services provided by the portal. Hashed password based authentication was used as authentication method that most commonly used within e-government portals worldwide and does not require additional hardware capabilities like biometric devices that commonly used within windows based-applications. There are three types of the clients, ordinary users , citizens and administrators. Administrators, in turn, are classified into portal admin and service admin. Each of them has its own credential. NID that generated by NID module is used in authentication the citizens. After the client of the system entered his credential , whether citizen credential or admin credential , he open a security session that allowed him to travel between portal's pages and get one of the three services.

b) Key management module

This module is responsible for generating, managing and distributing the symmetric and asymmetric keys. Random key generator was used to generate one-time 256-bit session key (AES's key) for each encryption process. In order to keep this session key secret, each service –via this module- generate one RSA's key pair (public and privet). RSA's private key, in turn, is encrypted by (3DES) using admin provided key (Passphrase). After generating service's key pair, the service requires the admin entering his passphrase that used in encrypting the private key using (3DES).Only the public key being known by the portal and the privet

keep secret at the service itself in encrypted format. After encrypting citizen's information by AES's key, this key will encrypt by the service's public key at the portal. When arrived to the service, the system requires service's admin entering his passphrase in order to decrypt the privet key of that service that , later, will used in decrypting AES's key.

c) Encryption module
This module is responsible for encrypting and decrypting the information that filled by the citizen in the order of the service. Symmetric cryptography was used instead of public-key cryptography which seems more transparent than public-key scheme. Rijndal algorithm has been used for this purpose as an official standard. One-time key (session key) that generated by Key management module was used for this purpose.

d) Integrity module
This module is responsible for guaranty the integrity of the information when sent from the portal server until they reached to the server of the service. HMAC-MD5 algorithm was used as one of the message authentication techniques. Before encrypting the information of the order, the message digest is computed and attached at the beginning of the message. The idea behind that is for easily separating the message digest from the original message, where HMAC-MD5 output is fixed 256-bit block of data, and the original message has a variable length, depending on the nature of information inserted by the citizen.

e) NID module
This module is responsible for generating a new NID for the citizen. It benefits from the constant personal information of the citizen like his/her first name, father's name, grand-father's name and mother's name. The 16-digits NID is a mixed from these information and random key, generated by random key generator. The essential purpose of the NID is to use it together with the citizen's full name in authenticating him/her at the portal.

f) Orders module
This module is responsible for control the number of orders inserted by the citizen to prevent flooding the server of the service by duplicated orders. When a citizen fills an order of the service for a first time, a message box will appear telling him about the date of order completion after 30 days. If the same citizen try to insert an order of the same service in less than month (30 days) , a message box will display telling the citizen that last order have been inserted before less than month. This prevents an attacker from flooding the server by duplicated orders.

C. *Data access layer*

Data access layer is the actual code that use in our applications to connect to a database. It used to encapsulate data access methods like Create, Read, Update and Delete operations and other methods accessing data from a data store layer. This layer involves: Inline SQL , Datasets and Stored procedures.

D. *Data store layer*

This layer represents the database itself. There are four databases, Portal DB, Civil DB and two identical databases: Passport DB and Driver DB. Portal DB include Credential table which used to store citizen's credential (full name & NID). The rest of DBs (Civil DB , Passport DB and Driver DB) are contain two tables : Order table which used to store encrypted order information , and Master table which used to store clear order information after decrypting them and checking their integrity.

IX. SYSTEM IMPLEMENTATION

A. *Generating Service's Key Pair*

For each service at the proposed system, there is one key pair (public and private) for that service. The public key is used by the portal system in encrypting citizen's orders before sending them to the service and the private is used by the administrator of that service in decrypting that encrypted citizen's orders. The generation process is done by the administrator at the service domain. Figure (3) shows "Key Pair Generation" page.

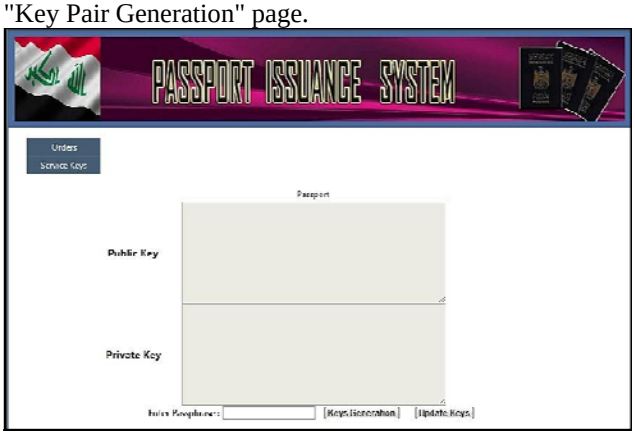

Figure (3) "Key Pair Generation" page

In order to avoid storing the private key in clear text, admin provided key has been used as a passphrase for encrypting /decrypting the private key. (3DES) algorithm has been used for that purpose. After the admin enter his passphrase at "Key Pair Generation" page , the hash value of that passphrase will computes and the result used as an input key for encryption/decryption process. MD5 algorithm has been used for computing the hash value. Figure (4) and figure (5) shows entering the admin passphrase and key pair generation, respectively.

Figure (4) Admin's passphrase insertion

Figure (5) Key Pair Generation

The public key is send to the portal database for storing it in "Service Key" table. An encrypted private key will store in "Service Key Pair" table, at service's database, together with the public key, associated with it. These keys can be updated by the admin. When the admin clicked "Update Keys" button, a new key pair will generates and the public key at the portal database will updated in transparent way.

B. ***Generating NID for the citizen***

Because of the nature of the e-government portals worldwide, any user across the internet can reach the Portal and navigate it. He has a permission to see the three pages within the portal only: AboutIraq.aspx (named "About Iraq"), E-GovProject.aspx (named "E-Gov Project") and ServicesDiscription.aspx (named "Portal Services"). He cannot access the serviceAcess.aspx page because he has not been entered his credential (which is the Full name and NID). If this citizen does not have NID, Civil Information System will assume generating a one for him.

In "NIG Generation" page, the admin must be fill the information related to the citizen. After inserting them in the specific textboxes, these information – later- will convert to capital letters in order to convert the first letter from each of these textboxes to ASCII code which, in turn, enter in NID generation process. After generating a new NID, it will hashed by (SHA-512) algorithm and stored in NID table. The hash value of each NID that generated recently will send to Portal website in order to update the Credential table within Portal DB. Figure (6) and figure (7) shows entering citizen's information and generating the NID for him.

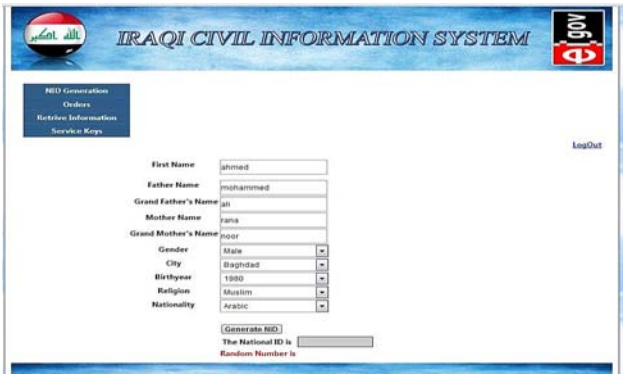

Figure (6) Entering citizen's personal information

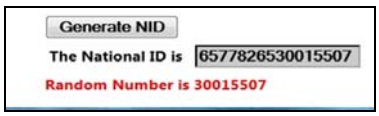

Figure (7) The new NID and its random number

C. ***Adding the new NID to Portal Database***

Each citizen that Civil system have been generated NID for him could not access the Portal until that NID's hash will added to the Portal database. The admin of the Portal system will assume this task. When opening the Portal website , "About Iraq" page will present as a default page

After entering his valid credential at login wizard and click on "Login" button, the Portal will redirect the admin to "Orders" page that contain a list of recent orders. Figure (8) shows these orders.

ID	FullName
135	JASIM AHMED ALI
136	MUSA ALI AHMED
138	AHMED MOHAMMED ALI

Figure (8)"Orders" page

This page contains the full names and hash values of the NIDs' of the citizens that recently generated NIDs for them. Each full name is represented by a link button. When click on one of these the link buttons, the website will redirect the admin to "Credential Check" page. Figure (9) shows this page.

Figure (9) "Credential Check" page

As notice in this page, it contains the new credential (full name and NID) and two buttons: active "Credential Check" button and inactive "Insert" button. When click "Credential Check" button, the system will check if this credential is exist in Credential table or not. If the answer is yes, a message box will display, telling the admin the credential is exist. In this case, "Insert" button will still inactive to prevent the admin from storing duplicated credentials at the table (Credential table). If the answer is no, a message box will display, telling the admin the credential is not exist. In this case, inactive "Insert" button will convert to active button in order to click by the admin. When click on it, this credential will add to the credential table and website will redirect the admin to "Delete Order" page. "Delete Order" page contains the credential that has been stored and "Delete" button.

When click on this button, this credential will deleted from the list of credentials that exist at "Orders" page. This prevents the accumulation of added credentials. Now, the citizen associated with the added credential can access the portal services.

D. ***Order submission by the citizen***

As mentioned previously, any user can navigate the portal website expect accessing it's services until enter a valid credential. When the citizen enters his full name and NID, the portal will compute the hash value of the entered NID and compare the computed hash with the NID's hash that stored at the Credential table. If the match, the portal will redirect the user (the citizen) to "Service Access" page, named "Portal services". Figure (10) shows this page.

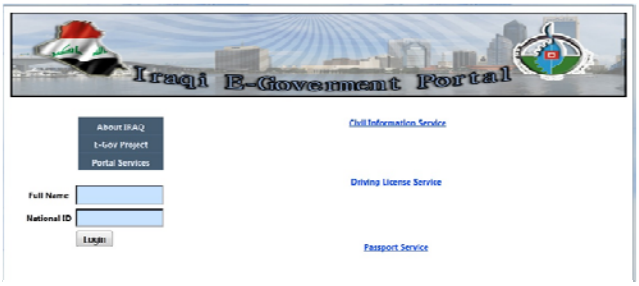

Figure (10)"Service Access" page

This page contains three hyperlinks that led to the three services (Civil, Passport and Driver License service). Each hyperlink that clicked will redirect the citizen to the order requesting page that associated with the service that citizen request it. If the citizen clicked on the hyperlink of the Passport service, portal will redirect him to "InsertEncryption" page which is an order requesting page of the Passport service. Once "InsertEncryption" page loaded, the portal will check if the current citizen has been sent an order at the current service. This checking is transparent to the citizen and done depend on the citizen's NID. If the answer is "no", a label message will display to the citizen , telling him this is his first visit. Figure (11) shows the citizen first visit to this page (Insert Encryption page).

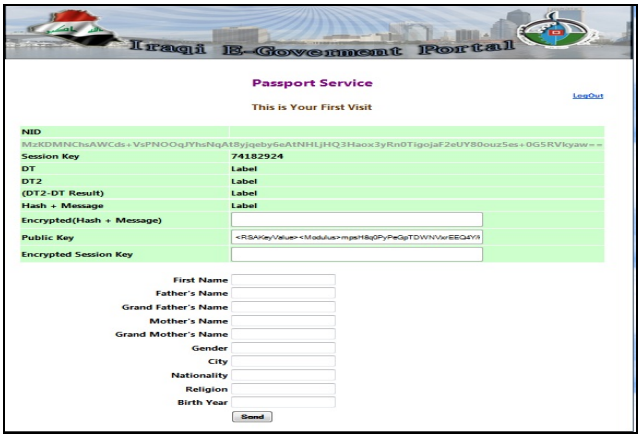

Figure (11) the citizen first visit to "InsertEncryption" page

Depend on displaying this label message , the portal will added 30 days to the current server date and stores this date as a last order date (DT2) for that citizen. After filling the required information by the citizen – in the textboxes -and clicks on "Send" button , a message box will display asking the citizen to contact the office (Passport office) after month (30 days) from the server date. If the answer is "yes", the portal will compare the date of the last order (DT2) with the current date of the server (DT) in terms of the number of days. This number of date will stored in "X" variable. After the citizen fills the required information, he clicks on the "Send" button. If "X" value is less than 30, a message box will display telling the citizen that his last order have been inserted before "X" days.

Order sending involves a series of hidden steps. These steps include:

a) Once "InsertEncryption" page loaded , a random key generator will used to generate a 256-bit one-time session key as symmetric encryption key. This key will stored at "Session_Key" field at Order table within Passport database.
b) The data filled in these textboxes will concatenated and stored at the variable "message".
c) The value of "message" will used as an input to HMAC-MD5 algorithm to compute the hash code of the message.
d) A same 256-bit key that used in encryption/decryption process has been used as an input to HMAC-MD5 algorithm.
e) The result hash stores at "Recived_Hash" field in the Order table within Passport database.
f) After that, the hash code will appended by the message to create "hash+message".
g) Hash plus message , that result in step "F" and the key generated in step "A" will used as an input to the Encryption algorithm .
h) The symmetric key will encrypted by the public key of that service.
i) The encrypted (hash+msg) and the encrypted AES's key will stored in Order table within Passport database.

Figure (12) shows these steps.

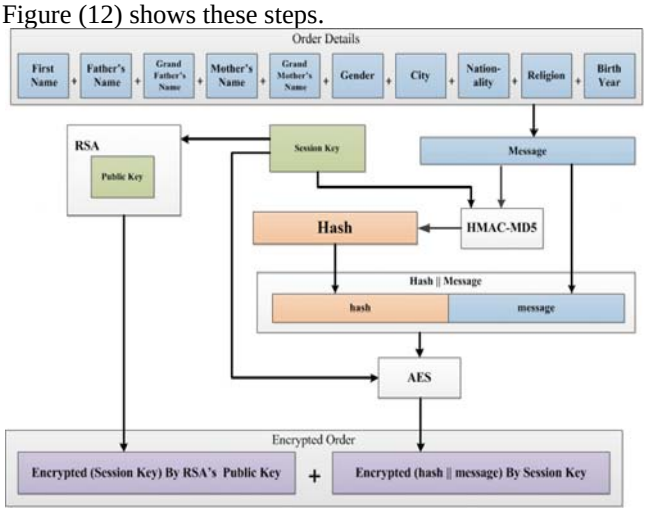

Figure (12) Order Sending

E. ***Order receiving by the Passport admin***

Any order of particular service that filled by the citizen at the portal will sends to the order table of that service. For each order that inserted at the order table recently, there is a link button in the order list at the "Orders" page. Figure (13) shows the order list.

Figure (13) Passport order list

When the admin click on this button, the website will redirect him to "DecryptionInsert" page. This page contains the textboxes, labels and buttons in green table that used in decrypting AES's key, decrypting message, integrity checking and inserting the order information at the Master table of that service. Figure (14) shows this page.

As shown in the figure, there are five buttons, "Decrypt Private Key", "Decrypt Session Key" and "Decryption" active buttons, and two inactive buttons: "Integrity Checking" button and "Insert" button. Once "Decryption Insert" page loaded, the website will retrieve the current order information depend on its ID. A set of labels and text boxes have been used to display these information in the green table. Figure (15) shows this table.

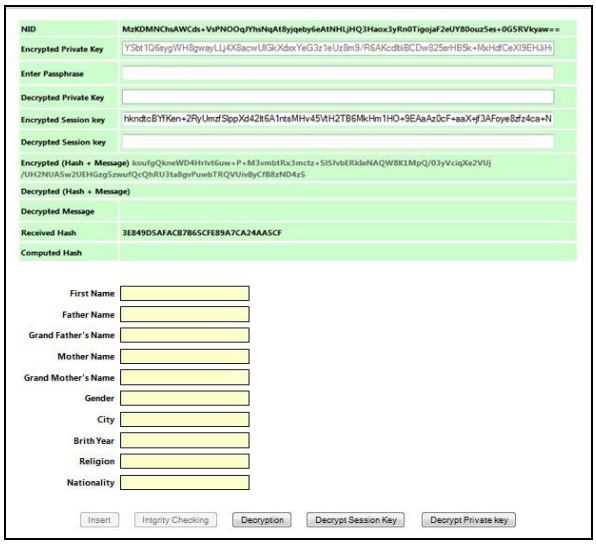

Figure (14) Passport "DecryptionInsert" page

Figure (15) "Decryption Insert" page labels

For decrypting the information of the current order, the admin must enter the same passphrase that has been entered it when generating the keys, in order to decrypt the private key of the service. Figure (16) shows this step.

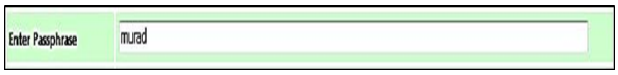

Figure (16) Admin's passphrase entering

When the admin clicks "Decrypt Private Key" button, the encrypted private at the text box will be decrypted Then, when the admin clicked "Decrypt Session Key" button, an encrypted session key will decrypted. When "Decryption" button clicked, the encrypted order details will decrypt and fill an associated text boxes. Figure (17) shows the decryption process.

After the success of the decryption process , "Integrity Checking" button will be active. Once "Decryption" button clicked , the hash value of the received order will sprat from the message and display at "Received Hash" label. When clicks "Integrity Checking" button, the system will compute the hash value of the decrypted message and compare it with the received hash. "Computed Hash" label will display the computed hash. Figure (18) shows "Computed Hash" label and "Received Hash" label.

Figure (17) Decryption process

Figure (18) "Computed Hash" label and "Received Hash" label

If they match, a message box will appear telling the admin that received message is integrated. Then, "Insert" button will be active.

When clicks "Insert" button, the integrated information of the current order will stored at the Master table in clear format as shown in figure (19).

Figure (19) Stored order in master table

For illustrate the Order receiving steps, figure (20) shows these steps.

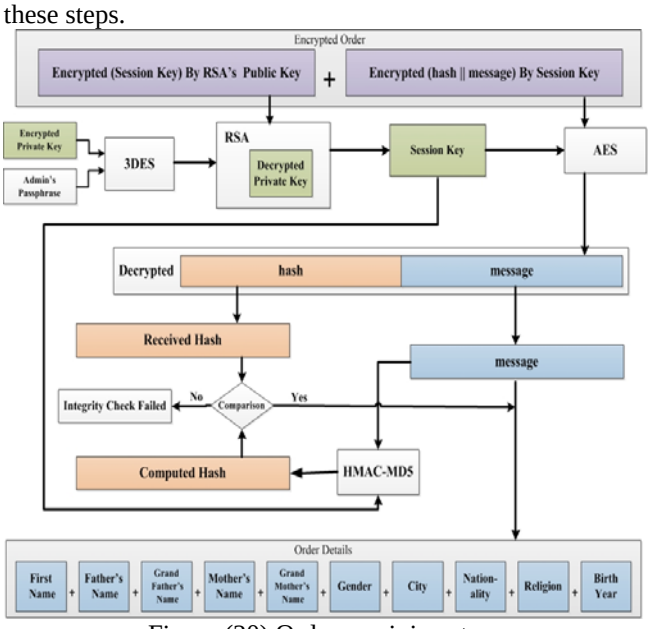

Figure (20) Order receiving steps

X. CONCLUSIONS

Through the design and implementation of the system, some conclusions are drawn, these are:

a) The proposed system maintains the interoperability between the services across suggesting a federated identification system, represented by a unified authentication data.
b) Authentication scheme have been used by the proposed system makes it accessible from desktop computers, mobile devices such as laptops, palmtops and even cellular phones (cross-platform). Thus, maintains the availability demand.
c) The system's architecture makes it flexible for future changes. Any changes in one layer do not affect the other layers.
d) The proposed security model used a standard cryptographic suit which is used by standard security protocols such as SMIME, SSL and IPsec.
e) Procedures followed by the proposed system makes it safe against the web application vulnerabilities like SQL injection attack, denial of service attack and cross-site scripting attack.
f) The security policy adopted by the proposed system encourages the citizens for adopting the public services that e-government systems provide.

REFERENCES

[1] M. Hwang, et al., “Challenges in e-Government and Security of Information.” in Information & Security, vol.15, no.1, 2004.
[2] Maria Wimmer ; Bianca von Bredow ,"A Holistic Approach for Providing Security Solutions in e-Government" ,35th Hawaii International Conference on System Sciences, 2002.
[3] "e-government primer", infoDev /World Bank, Washington , 2009.
[4] Anil Saldhana ," secure e-government portals", W3C Workshop on e-Government and the Web, National Academy of Sciences, Washington, 2007.
[5] John Kelsey, Stefan Lucks, Bruce Schneier, Mike Stay, David Wagner, and Doug Whiting, "Improved Cryptanalysis of Rijndael", 2000.
[6] Gopala Krishna Behara, Vishnu Vardhan Varre and Madhusudhana Rao ,"Service Oriented Architecture for E-Governance" , BPTrends, 2009.
[7] Joanne Kuzma & Sue Barnes, "Web Vulnerability Study of European Insurance Sites", 2010.
[8] Matthew M.Donald, Adam Freeman, and Mario Szpuszta, " Pro ASP.NET 4 in C# 2010" ,Fourth Edition, Apress, 2010.
[9] William Stallings," Cryptography and network security: security principles and practices ", Fifth Edition, Prentice Hall, 2010.
[10] Bryan Sullivan & Vincent Liu, "web application security: a binger's guide", McGraw-Hill, 2010.
[11] Dr. Ali M. Al-Khouri," PKI in Government Digital Identity Management Systems ", European Journal of ePractice ,No.16, February2012.