Design and Implementation of SET-Enabled E-Commerce System

Asst. Prof. Dr. Sufyan T. Faraj
College of Computers
Al-Anbar University, Al-Anbar, Iraq

Asst. Lect. Media Abdul Razak Ali
Computer & Software Eng. Dept., College of Eng.
Al-Mustansiriya University, Baghdad, Iraq

Abstract

This study presents the design and implementation of a business to consumer e-commerce system that provides the basic e-commerce security requirements including confidentiality, integrity, non-repudiation, replay protection and the most important entity authentication. The above security features are obtained by adopting the Secure Electronic Transaction (SET) Protocol as being the emerging e-payment security standard. The security of the protocol is further enhanced using some of the most powerful cryptographic algorithms such as the Advanced Encryption Standard (AES) algorithm, the RSA-OAEP encoding, SHA-1 hashing and HMAC authentication functions.

The system is based on the modular Three Tier client\server architecture and guarantees portability across any hardware and software platform. This feature is basically provided by the cross-platform capability feature of the Java language. Indeed, Java Servlet technology gives the system the very important multithreading feature. MySQL database along with the HTML language were also used for system implementation. The system had been successfully installed and tested. Experimentally, our system was found to be of a high security level, flexible, portable, robust, and a relatively good performance.

الخلاصـــــة

يقدم العمل نظام تجاري ألكتروني يوفر المتطلبات الرئيسية لأمنية التجارة الألكترونية المتضمنة الخصوصية، عدم التحوير، عدم التنصت، حماية إعادة الأستعمال والمتطلب الأكثر أهمية وهو الموثوقية أو التحقق من الهوية وهذا يتم بإستخدام النظام القياسي (Secure Electronic Transaction). تم بناء نظام تجاري ألكتروني (B2C) كامل متكون من تاجر، مستهلك، سلطة توثيق إساسية (CA) وبوابة دفع (Payment Gateway). جميع التراسلات بين عناصر المنظومة تكون محمية بواسطة ببروتوكولات التراسلات الأمنية السرية (SET)، ابتداءاً بإثبات الهوية والشراء والتوثيق والدفع. تم تحسين أمنية بروتوكول الـ(SET) بإستخدام الخوارزمية القياسية الجديدة AES Algorithm . إستخدمت أيضاً خوارزميات أخرى المتضمنة نظام الترميز RSAOAEP ودوال الموثوقية SHA-1 و الـ HMAC لتوفير السرية المطلوبة.

ان النظام مبني على أساس معمارية النظام المركب من المستفيد والخادم (Three Tier Client/Server Architecture) ويضمن صفة النقلية (Portability) على أي قاعدة تصميمية من خلال خاصية قابلية الأستقلال عن الآلة (Cross Platform Capability) التي تتمتع بها لغته التطبيقية Java بنوعيها التطبيقات وتقنية Servlet الجديدة التي توفر أيضاً صفة أكثر من خدمة (Multithreading). ان قاعدة البيانات (MySQL) ولغة المستندات (HTML) هي أيضاً مستخدمة في البحث الحالي.

1. Introduction

According to the International Electronic Consortium (IEC), e-commerce can be broadly defined as " a model of selling and buying in which buyers are able to participate in all phases of purchase decisions, while stepping through these processes electronically rather than physically, including enabling a customer to access product information, select items to purchase, pay for items securely only with simple point and click interactions" [1]. As the scale of e-commerce transaction has grown, it has become very attractive to criminals, and the volume of fraudulent e-commerce transactions is growing rapidly. Therefore, there has been an increasing amount of attention given to the security of e-commerce being the most important feature facing its growth.

The best way to characterize e-commerce security requirements is by examining the overall process, beginning with the consumer and ending with the commerce server. Considering each link in the "commerce chain", web commerce security can be defined as a" set of procedures, practices and technologies for assuring the reliable and predictable operation of the web server, web browser and other data that is in communicate with the web server and the surrounding Internet infrastructure" [2,3]. Consequently web security has three primary facts:

i- Securing the web server and the data that is on it.

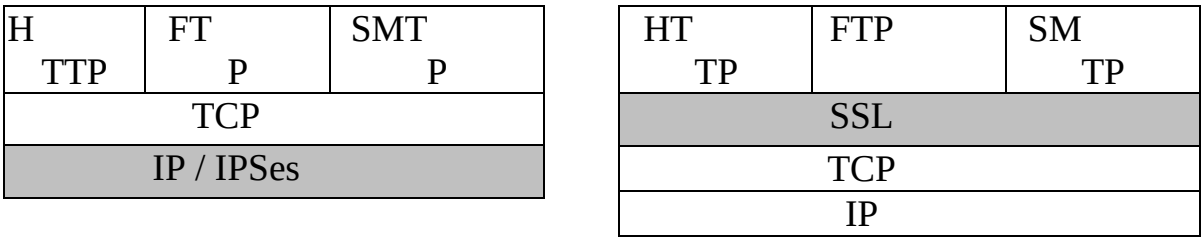
ii- Securing the information that travels between the web server and the user.

iii- Securing the end user computer.

Neither the user nor the web server has any control over the path their messages will take, nor can they control who examines the message content along the route, but at least the user and web server security are often thought to be their own responsibility. Accordingly this study concentrates on web transaction security as the communication channels are the major assets to be protected. A number of approaches for providing web security are possible. These approaches are similar in the services they provide and to some extent in the mechanisms that they use, but they differ with respect to their scope of applicability and their relative location within the TCP/IP protocol stack [4,6]. At the network level, there is the IP Security (IPSec) protocol which provides a security option for the current IP protocol. It supports authentication and/or encryption services at the network level. The advantage of using IPSec is that it is transparent to the end users and application, and offers a general-purpose solution mostly based on IP packet filtering. However, the approach is not suitable with general e-commerce applications where the client request can be from any where. In spite of this, IPSec is typically used to build and maintain a virtual private network offering secure connection between a company and its business partners, branch offices and employees over the Internet [5,8].

At the transport level, it is possible to use the Secure Socket Layer (SSL) protocol. The protocol was launched in 1994 by Netscape to make use of the TCP/IP protocol to provide a reliable end-to-end secure communication between the web server and the web browser. Security services include server authentication, data confidentiality, optional client

authentication and data integrity [6]. **Figure (1)** illustrates the usage of above mentioned protocols.



(a) Network level security (b) Transport level security

Figure (1) TCP\IP layers security

Although SSL does eliminate some security risks such as eavesdropping and unauthorized modification, it does not fulfill the basic e-commerce security requirements and this fact is not due to a weakness in the protocol but due to being a secure web session protocol, not an entire e-commerce security specific protocol and hence there remains a number of risks and threats which can lead to a card fraud.

Thus, in conclusion, application level security services are more suitable and have much more benefits for our concern because these are embedded within a particular application, having the big advantage of being tailored to the specific needs of that application. In the context of e-commerce these are called the e- payment protocols. The most important such protocol is the Secure Electronic Transaction (SET) protocol [2,7].

After this introductory section, the rest of this paper is organized as follows: Section 2 gives the required background on SET. In Section 3, the general architecture of the proposed SET-enabled system is presented. Then, sections 4, 5, 6, and 7 illustrate each of the participating entities in the proposed architecture in a more detail. Next, the main system operation flow is described in Sections 8. System performance is considered in Section 9. Finally, some important conclusions are listed in Section 10.

2. SET Background

SET is an open encryption and security specification designed in 1998 to protect credit card transaction on the Internet. The protocol emerged from a call for security standard by Master and Visa card, and a wide range of companies were involved in developing the protocol including IBM, Microsoft, Netscape, RSA, Terisa and Verisigne. In essence, SET provides a secure communication channels among all parties involved in transaction, provides trust by the use of X.509v3 digital certificate and ensures privacy because the information is only available to parties in a transaction when and where necessary [6,9].

2-1 SET Network Architecture

Figure (2) gives a schematic representation of the network architecture for the SET protocol; it has basically the following components [3,9]:

a) Cardholder; an authorized holder of a payment card issued by an issuer.

b) Merchant; organization that has goods/ services offered through a web site.

c) Issuer; this is a financial institution which provides the payment cards.

d) Acquirer, this is the financial institution that establishes an account with the merchant and processes the payment card authorizations and capture.

e) Payment gateway; this is a function operated by the acquirer to interface between the SET protocol and the existing bankcard payment networks for authorization operations.

f) Certification authority; this is a trusted entity to issue X.509v3 public key certificate to the cardholders, merchants and payment gateways.

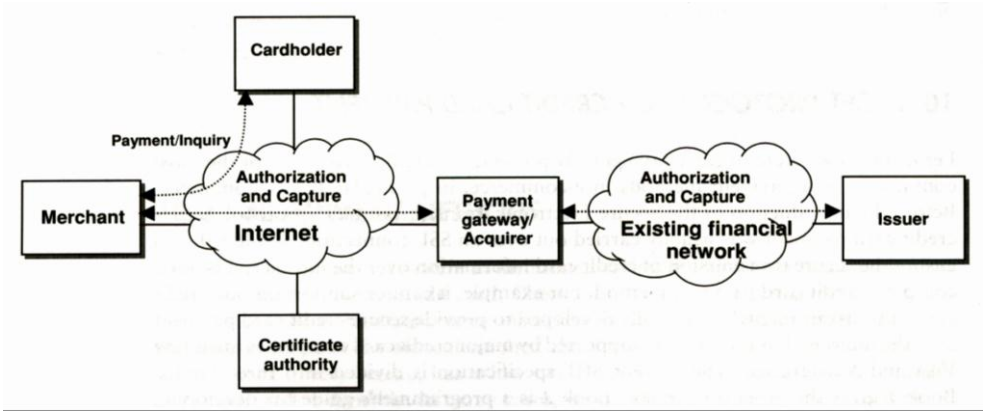


Figure (2) SET protocol network architecture

Some important SET key features are:

1. ***Data Confidentiality:*** The Usage of Digital Envelope; SET obtains the benefits of both public and secret key cryptography schemes by employing the Digital Envelope cryptographic technique. The RSA-OAEP (Optimal Asymmetric Encryption Padding) 1024-bits and 128 bit AES algorithm is used.
2. ***Data Integrity:*** Usage of Dual Signature; SET introduces a new application of digital signature namely the concept of Dual Signature (DS). This approach prevents the SET merchant from obtaining the account details of their consumers. DS provides a linkage between two messages needed to be linked securely in order to be sent to two parties, each has to read only one of the pair. SHA-1 hashing and HMAC algorithm is used [9,10]. SET implements the DS technique when the customer sends the Order Information (OI), to the merchant and the Payment Instruction (PI) to the payment gateway tunneled through the merchant. The merchant should not know the account details or the bank need to know the order details, but the messages needed to be linked together so that the merchant cannot attach another (OI) and claims to be the customer order. DS provides a proof that this payment is intended for that order and not for another, providing protection from any

disclosure or fraud from the merchant side. The customer software takes the hash of the PI and the hash of the OI. These two hashes are then concatenated together and the hash of the result is then taken. Finally, the client software encrypts the final hash with the client private signature key, developing the DS. The DS, along with the digest of the PI (PIMD) is sent to the merchant [6,7].

3. ***Entity Authentication:*** The Usage of Digital Certificates; SET relies upon a hierarchical (tree structure) arrangement of nine components for the management of digital certificates as shown in **Fig.(3)**.

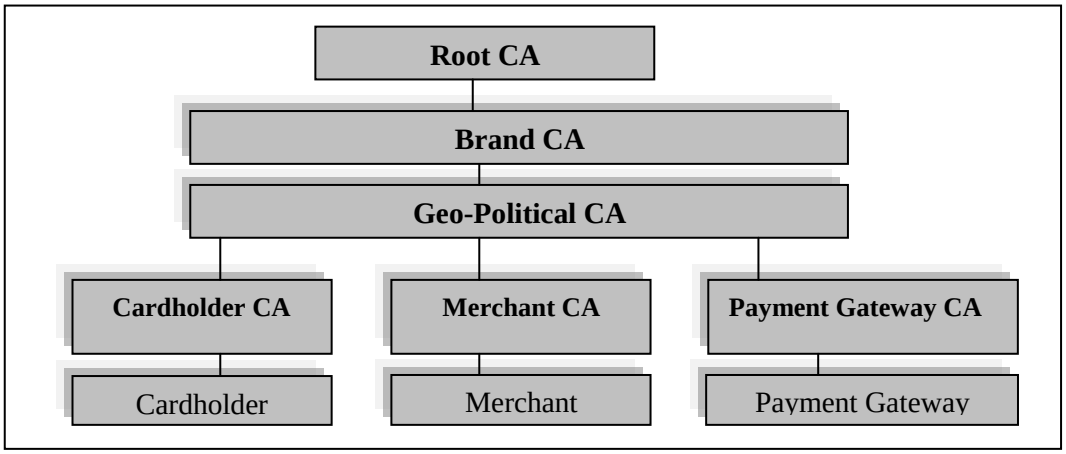


Figure (3) The SET certificate architecture

2-2 SET Protocols

SET is not itself a payment protocol, rather it is a set of security protocols and formats that enables users to employ the existing credit card payment infrastructure on an open network such as the Internet in a secure fashion. Since all the provided protocols cannot be covered in this study, only the main protocols were implemented, these considered as the main and the most important protocols for general e-commerce systems [11]:

1. ***The Cardholder Registration Protocol:*** In order to purchase using the SET protocol, cardholders must register with a Certificate Authority (CA) to obtain SET digital certificate so that they can send SET messages to merchants and payment gateways. Cardholders certificates function as an electronic representation of the payment card and since they are digitally signed by a financial institution, they cannot be altered by a third party and can only be generated by a financial institution. The SET cardholder certificate does not contain the account number and expiration date. Instead the account information and a secret value known only to the cardholder's software are encoded using a HMAC by SHA-1 hashing algorithm [3,10].
2. ***The Merchant Registration Protocol:*** The merchants have to register with a CA before they can receive SET payment instructions from cardholders or process SET transactions through a payment gateway. Merchant certificates, on the other hand, are digitally signed by

the merchant's financial institution, and hence cannot be altered by a third party and can only be generated by a financial institution.

3. ***The Purchase Protocol:*** The Purchase protocol is the most central data structure in SET. It is used to pass the data required to authorize a payment card payment from the cardholder to the payment gateway, which will use the data to initiate a payment card transaction through the traditional payment card financial network. The data is encrypted by the cardholder and sent via the merchant, such that it is hidden completely from the merchant.
4. ***The Authorization Protocol:*** During the processing of an order from a cardholder, the merchant authorizes the transaction with the payment gateway. The payment authorization ensures that the transaction is approved by the issuer. This authorization guarantees that the merchant will receive the payment and can, therefore, provide the services to the customer.
5. ***The Capture Protocol:*** After completing the processing of an order from a cardholder, the merchant will request payment. Amounts captured must be previously authorized using the authorization protocol [9,12].

3. The Proposed Web System Architecture

The proposed design follows the three-tier architecture model, as shown in **Fig.(4)**. This model breaks the application into three distinct layers or tiers: user presentation, business logic, and data services tier; each with its own goals and design constraints:

1. The user presentation tier; it is the client side layer that hosts the front-end tool with which the end user interacts at a client PC. Basically it contains a Graphical User Interface (GUI) and some additional application specific business rules.
2. Business logic tier; this tier performs the business operations that the application designed to automate, through interacting with the data service tier, its resident on a computer called the management or the application server.
3. Data service tier; this tier supports the business logic tier in fulfilling the user requests through providing the data and information necessary to compete a transaction.

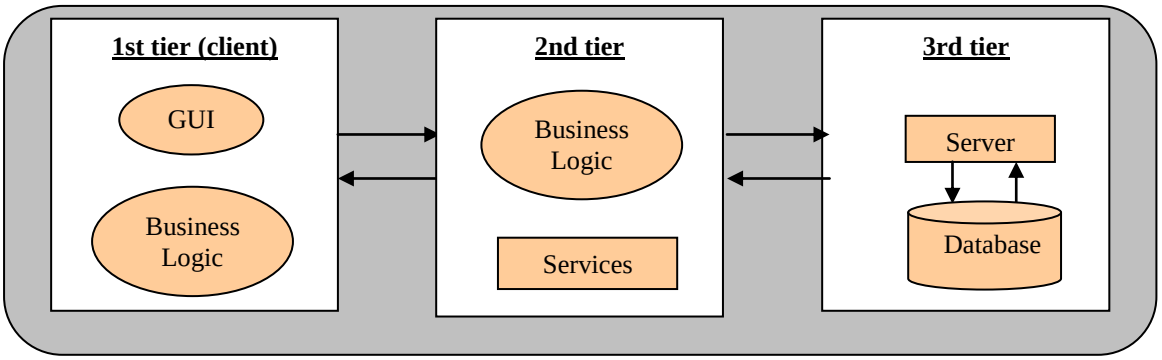


Figure (4) The three tier model

Based on the three-tier model, and according to the shopping and SET payment working procedure, the general architecture of the proposed SET-enabled e-commerce system (see **Fig.(5)**), was developed to be consisting of the followings:

1. Consumer Application Part; including the web browser and the digital wallet software.
2. Online Shopping Service Part; including the web server and the database server.
3. SET Payment Service Software.
4. Payment Gateway Service Software.
5. Certificate Authority Service Software.

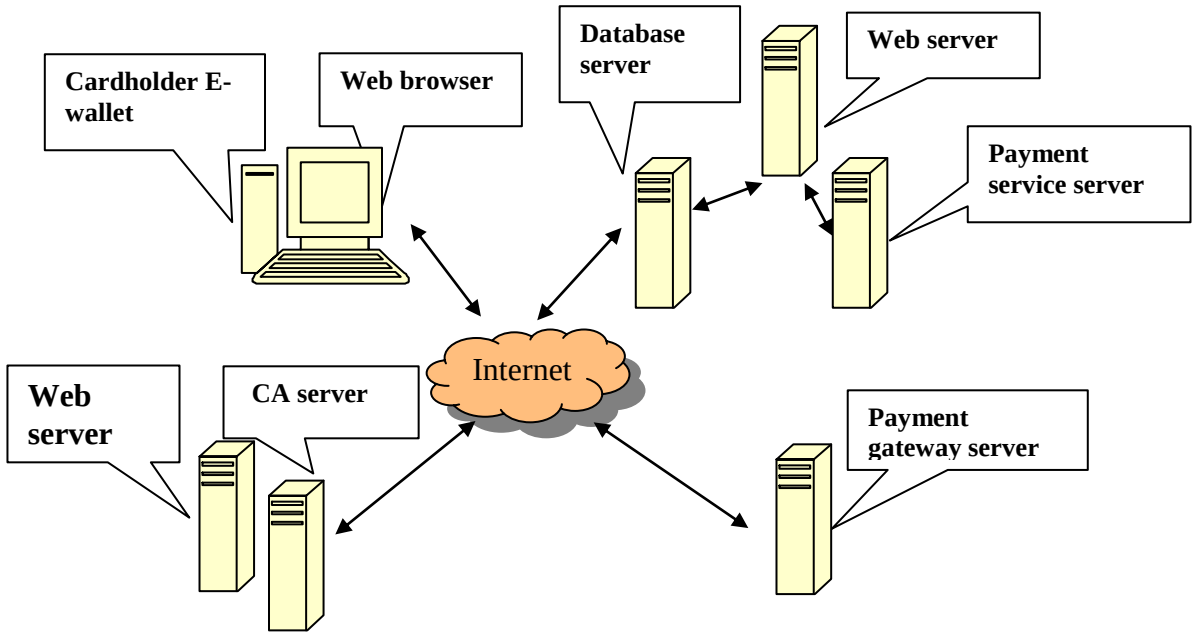


Figure (5) System web architecture

The following four sections (sections 4-7) contain a more detailed description of the design and software implementation for each of the participants involved in the e-commerce process. However, before that it is important to mention that there are some software classes that are implemented by all system participants. These include:

1. SET Settings: This class is used to hold the SET protocol general components, the reversed ports for the participating parts, and the basic message header to indicate the type of the message; so that the recipient can invoke the appropriate processing.

2. The Security Package: This component provides all required cryptographic operations in the SET protocol, including the following: 1024-bit RSA encryption/decryption, RSA with SHA-1 signature and signature verification, 160-bit SHA-1 hashing function, 128-bit Rijndael encryption\decryption, symmetric and asymmetric key generation, and SET's version of OAEP to support its "extra encryption" and "extra decryption" operators that combine RSA encryption and Bellare-Rogaway Optimal Asymmetric Encryption Padding

3. The Certification Component: This component is responsible for managing the digital certificates for the SET entity; it basically acts as a key store which contains the participating entity certificates to be provided during the processing.

4. Server Side Architecture and Development Tools

The SET merchant software consists of the server side applications that implement the business logic of the web system. The modular approach for the design calls for the separation of work on dedicated servers each has its own functionality. Distributing the work in this way assures the highest availability of resources and meets the scalability needs. Accordingly, the merchant environment consists of three essential servers: the merchant web server, merchant database server, and merchant payment server. As shown in **Fig.(6)**, these three servers together form a logical entity which can be called the merchant.

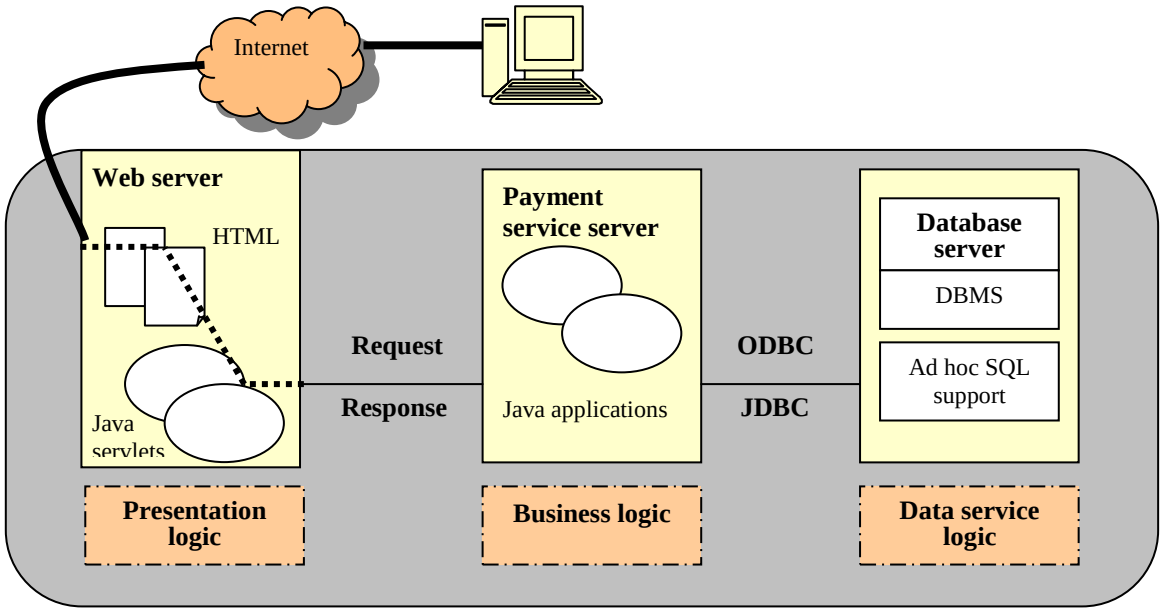


Figure (6) The Merchant Software

4-1 Merchant Web Server

Java web server was used in this work as the merchant web server, due to the valuable features it has and especially being Servlets enabled. These features include cross-platform, simplicity, local and remote administration, session support, flexible security model, being standard based technologies, and extensibility. The web server is built as a multithreaded Servlets container responsible for implementing the merchant required tasks, managing the web site, establishing the connection with the payment service server and the database server, and handling the new client's tasks. There are essentially three Servlets chained together to perform the above tasks. The Servlets are built using the Java Servlet programming language, SQL database instructions, and the HTML language. These Servlets are:

1. ***Home Page:*** The homepage Servlet is responsible for managing the web site products and displaying them to the cardholder when connected, logging on the database server. Upon cardholder selection for items, the home page passes these choices to the Order form Servlet once clicking on the "Order Form" button.
2. ***Order Form:*** As the name implies, this Servlet is responsible for managing the cardholder requested order by checking with the merchant database and constructing the order form to the user to confirm it and redirect the order description and purchasing amount to the Start SET Servlet
3. ***Start SET Servlet:*** This Servlet is responsible for initializing the SET protocol through making the necessary connection and address allocation between the communicated parties and displaying the appropriate web pages when needed. **Figure (7)** illustrates the merchant web server structure.

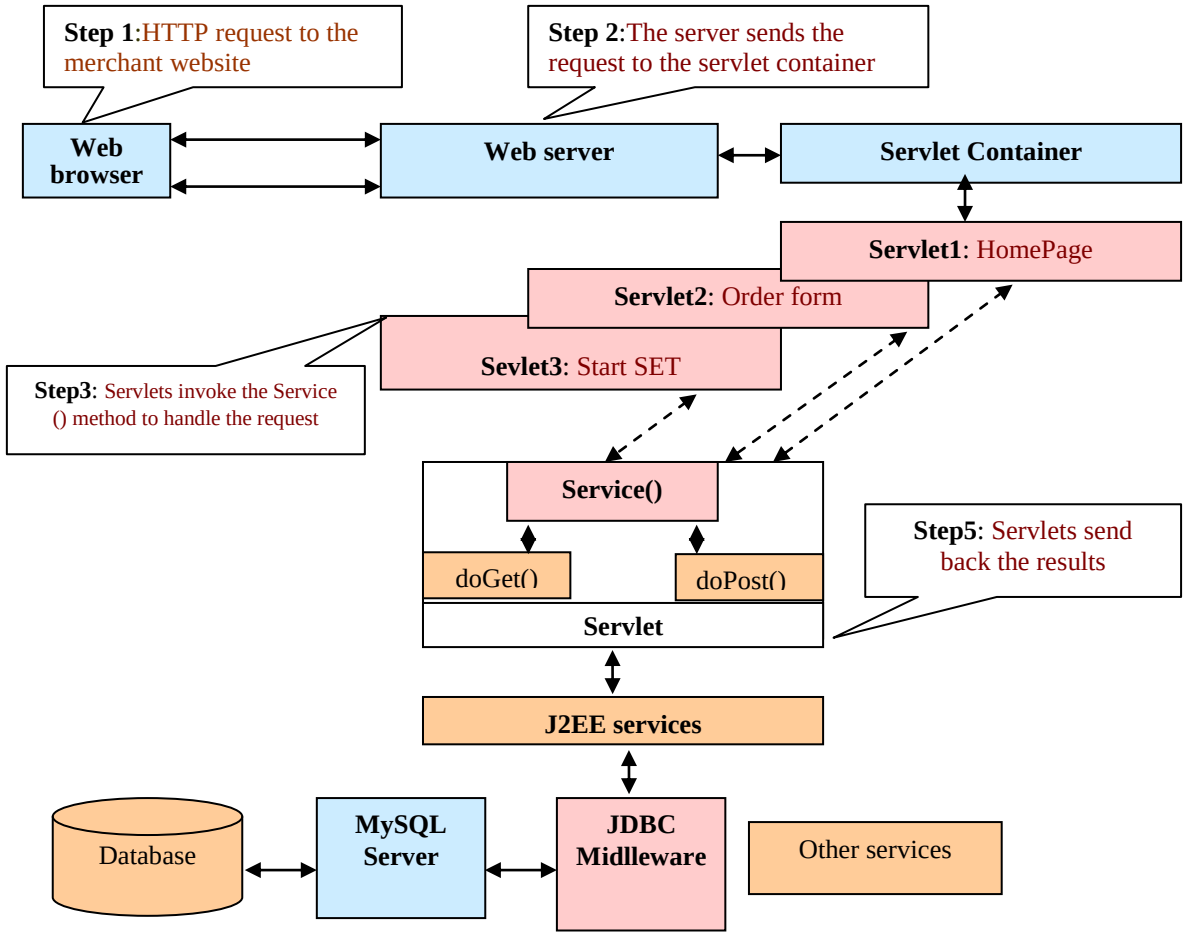


Figure (7) The merchant web server architecture

4-2 The Merchant Payment Service Server (PSS)

The PSS is where SET-related work is performed. This is the computer that loads the required SET software, placing these components into separate server, permits their sharing usage by more than one merchant server software on the network. The PSS can be configured being the server side analogous of the cardholder e-wallet handling the required cryptography, managing the merchant digital certificates and the communication with the Payment Gateway Service Server (PGWS) to log the transaction authorization and settlement operations. The designed PSS consists of the following components (see **Fig.(8)**):

1. ***The PSS\Web Listener:*** This Payment listener is the entry point to the PSS for the merchant's web server. The component keeps listening on a dedicated IP port collecting requests from the merchant web server and forwards them accordingly to the Payment Intializer for the appropriate processing.
2. ***The Payment Intializer:*** This component is responsible for handling the new client required management processing.
3. ***The PSS\Wallet Listener:*** This component is responsible for listening to the wallet dedicated port waiting for the corresponding SET messages to arrive, to forward them to the PSS application for processing.
4. ***The PSS Application:*** This software is the core component of the PSS. It is responsible for performing the entire required SET message processing at the merchant level, by providing the following encryption and message handling: PSS purchasing, PSS authorization and capturing, and PSS registration applications.

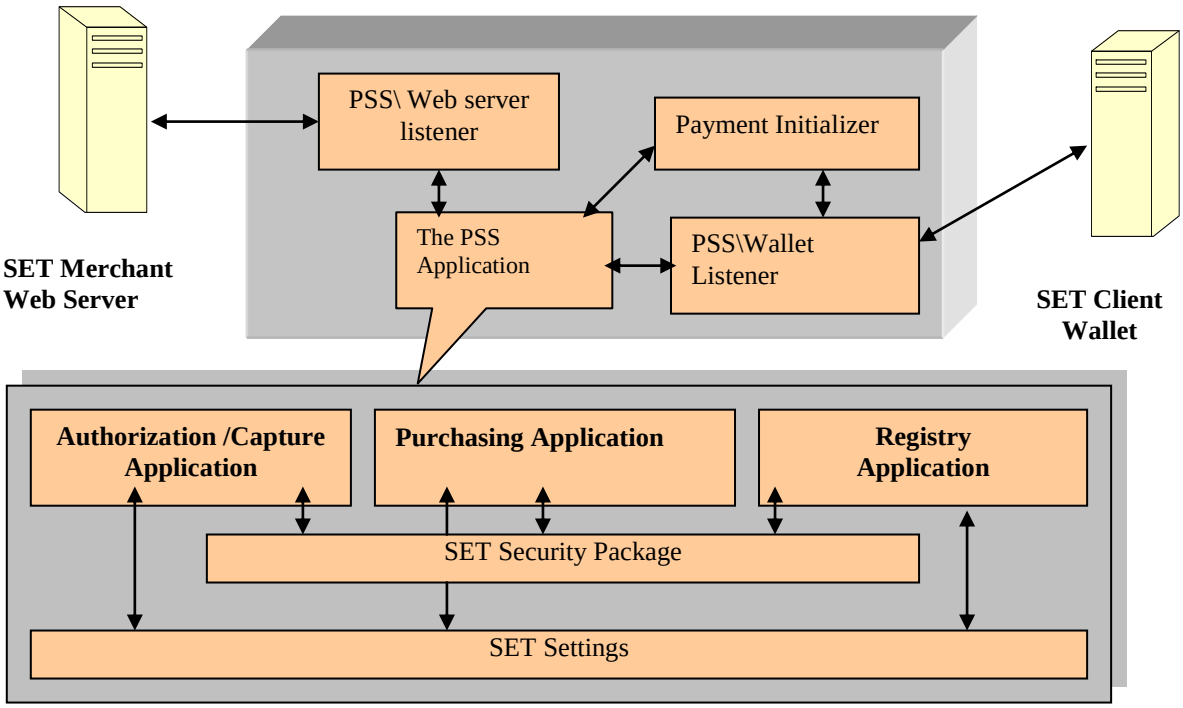


Figure (8) The Merchant Payment Service Server (PSS)

5. The Payment Gateway Service Server (PGWS) Structure

The PGWS software is the application that provides the acquirer payment gateway function of SET protocol. It functions as an intelligent router for incoming and outgoing messages to and from the Internet. It interfaces between the legacy credit card processing systems and the SET protocol. The designed PGWS (see **Fig.(9)**) has three main components as follows:

1. ***The Payment Gateway Listener:*** This is the entry point to the PGWS for the merchants. It listens on a dedicated IP port and collects requests from the merchant PSS and presents them to the payment gateway application to process them.
2. ***The Payment Gateway Application:*** This is the software that performs all the required SET processing operations through providing the encryption services and message handling services to perform the following:
   - Processing the payment instruction from the cardholder.
   - Validating the consistency between the merchant authorization requests and the cardholder payment instruction data.
   - Providing the legacy system with the appropriate data fields from the received SET messages.
   - Responding to the merchants with the appropriate authorization response.
   - Performing all the required operation during the registration phases.

Essentially it consists of the two main components: the PGWS authorization components and the PGWS registration components.

3. ***The Payment Gateway Legacy Application:*** The role of this component is to serve the request sent by the payment gateway application, to process them by interacting with the connected financial institution.

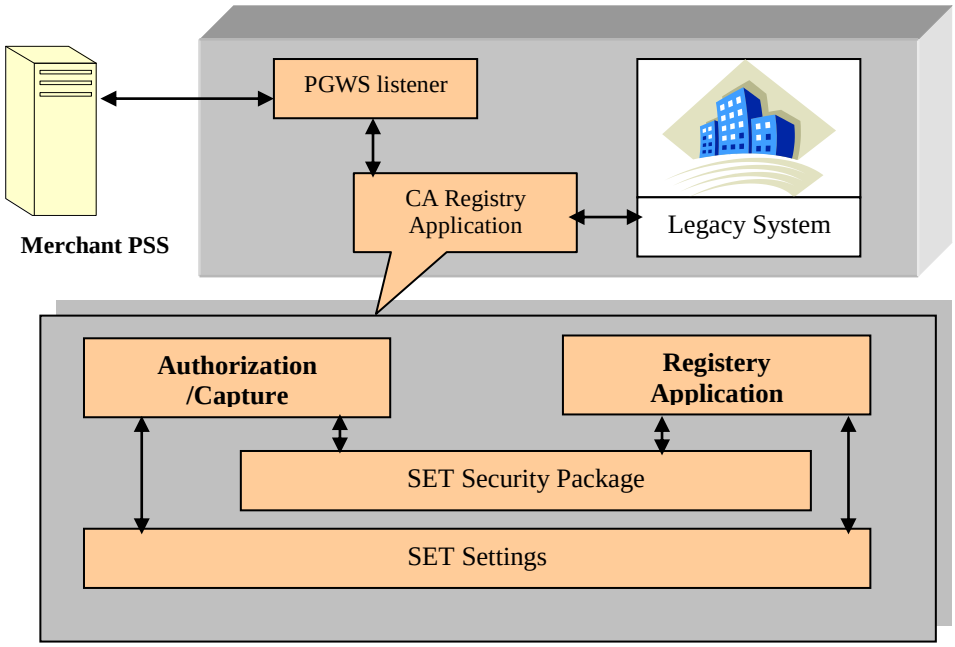


Figure (9) The Payment Gateway Service Server (PGWS)

6. The Certificate Authority Service Software

The SET protocol relies on the certificates to validate all the parties in a transaction. The software that provides the registration function is the certificate server software; it provides a certificate management infrastructure for cardholders, merchants, and acquirers to facilitate secure payment over the Internet using the SET protocols. The designed Certificate Authority (CA) server (see **Fig.(10)**) consists of the following components:

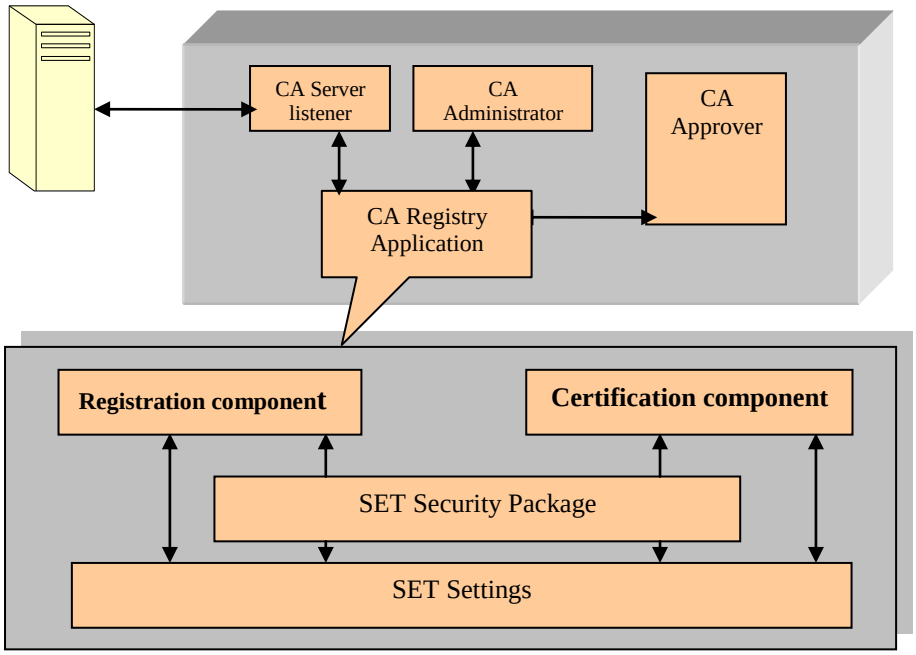


Figure (10) The Certificate Authority Service

1. The CA Web Server: Similar to the merchant web server, this CA component receives client requests for the registration purpose. Its operation is also built using the Servlet technique. Essentially, there are three web servlets, the *CAHome Page* which represents an introduction web site to the CA, the *CAPolicy* which contains the CA registration policy, and finally the *CAReg* Servlet which passes the requester the IP address of the CA server to perform the SET registration protocol.

2. The CA Certification Server: This CA component consists of the following:

- The CA server listener: This component is responsible for listening to the dedicated SET port, collecting requests from the SET participants requesting to be certified, and forwarding these requests to the CA registry application to process it.
- The CA Administrator: This is the application and associated interface responsible for the operational management of the server, handling the brand registration forms, establishing the root brand policies, etc.
- The CA Approver: This component is the application that is responsible for validating the user authentication information. The registry application forwards this component

such authentication information to validate it logging on the corresponding finical institutions. The component operation is also just assumed here.

- The CA Registry Application: This is the software that handles and processes all the SET registration messages, including the registration component and the certification component.

7. Consumer Application Part

In order for the SET protocol to be successful, it is imperative that the cardholder is protected from the underlying complexity of the protocol, and to solve this problem, the client side in the cardholder is developed as two parts: the *Client Web Browser* as a universal client to visit and shop, and the *Cardholder E-Wallet Application*. The E-wallet is a fundamental requirement. This component carries out the cardholder part of the SET protocol in a salient manner. It embodies the SET required operations and provides a means to store and manage the certificates to digitally sign the required messages, along with the security aspects consumer demands to keep private data. It also communicates with SET complaint merchant servers.

The security of E-wallet is very essential, it must be safe as least as safe as a real wallet. Accordingly the designed wallet is secured using two pairs of UserID and password working as follows: When the cardholder first accesses his/her E-wallet he will be asked for a userID and a password to open it, permitting anyone to access the stored data in it. After the cardholder purchase and wants to buy using the E-wallet, another pair of userID\password will be requested. This pair permits the E-wallet to carry out the SET protocol and purchase using the supplied credit card information unless it is correct. The designed E-wallet (see **Fig.(11)**) consists of the following components:

1. ***The Wallet\Web Server Listener:*** This component of the wallet is responsible for listening to the merchant web server waiting for a wakeup wallet to launch and activate the cardholder wallet, starting the SET protocol.
2. ***The Wallet* Administrator*:*** This is the software and associated interface responsible for operational management of the wallet.
3. ***The Wallet/ PSS Connector:*** This component is responsible for establishing the required connection with the PSS during the processing and terminating when the purchasing operation is complete.
4. ***The Wallet Application:*** This component supports processes and manages all SET messages at the cardholder level through handling the cardholder transactions providing cryptographic functions and communicates with the merchant PSS. It basically consists of two components: the wallet purchasing application and the wallet registry application.

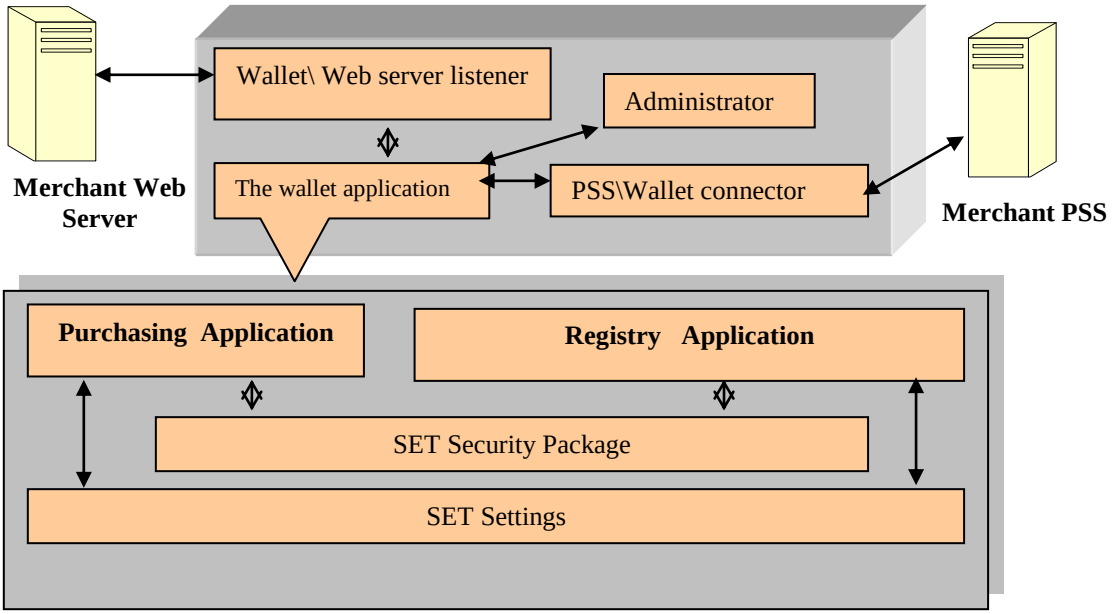


Figure (11) The Cardholder E- wallet

8. System Operation Flow

In this section, the system operation is emphasized by examining its main flow steps. In the following subsections the most important operation phases are considered in a more detail. These are the registration, authentication, and purchasing phases.

8-1 The Registration Scenario

To illustrate the relationship between the CA server components, the registration flow is examined, as follows:

1. The cardholder using his browser accesses the CA web server home page.
2. After reading the policy and agreeing on it by pressing the accept button, the *CAReg* Servlet, sends a launch message to the requester which is here the E-wallet. This message contains the CA sever address and the dedicated port for this new client. According to the received message, the cardholder E-wallet launches, asking the user to provide his username and password. As described in the e-wallet design, this password is not used in any network communication. For all online transactions, the system uses digital certificate to authenticate the user; this password simply acts as a key to lookup and run the wallet. After providing the correct ID and password, the user is notified that he must certify his credit data before making a purchase using its SET wallet. The wallet then sends *SET Regform* request to return the appropriate registration form.
3. On the CA server, the listener is permanently waiting for request message on the dedicated port to forward it to the registry application to be handled. The listener maintains the connection for the duration of the registry operation.

4. The registry application receives the request and forwards the initiation information to the CA administrator.
5. The CA administrator upon receiving the brand information retunes the appropriate registration information to the registry application.
6. The registry application properly formats the registration information in SET message, and forwards it to the CA listener.
7. The CA listener forwards the message to the end entity application through the established connection.
8. The end entity application returns the filled in registration form with some required information in certificate request message, the message confirms to the SET specification.
9. The listener again forwards the received data to the correct destination, the registry application.
10. The registry application performs the required processing and validates its data through the CA approver.
11. The CA approver verifies the registration form information logging on the corresponding legacy system and returns the appropriate approval which contains the required certificate (see **Fig.(12)**) or denial of the request.
12. The registry application upon the approval results constructs an appropriate certificate response and forwards it to the CA listener.
13. The CA listener through the established connection returns the response to the end entity application and terminates the connection.

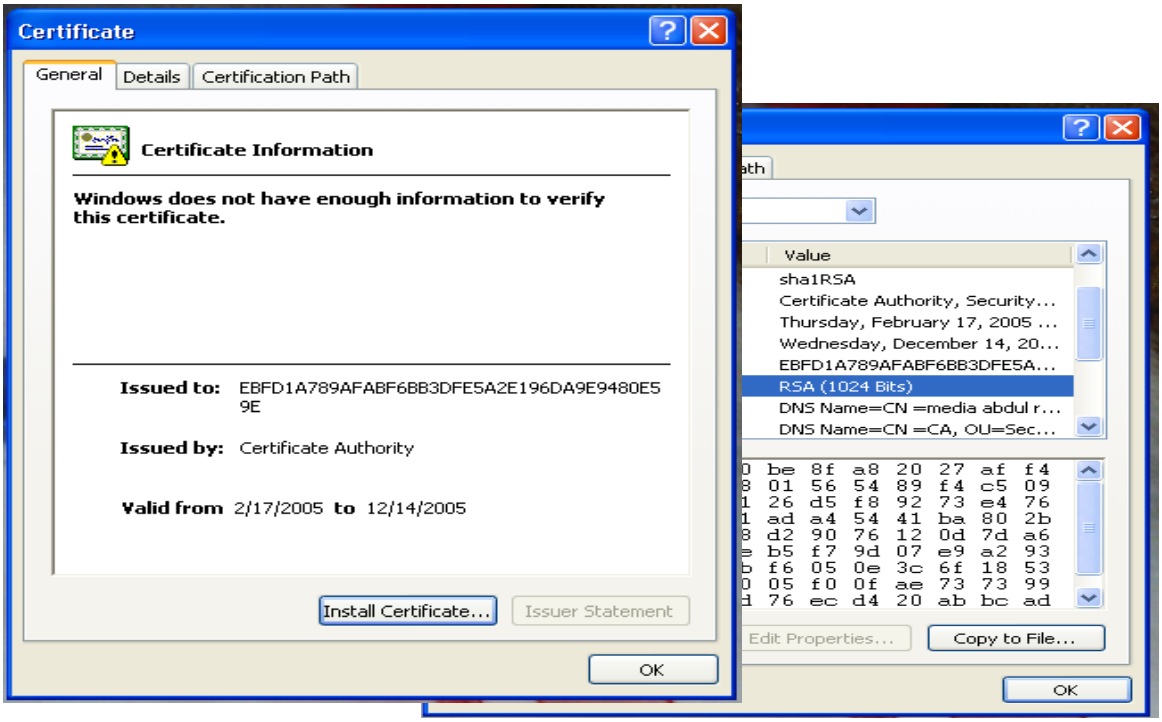


Figure (12) The provided cardholder certificate

8-2 The Authentication Scenario

To illustrate the relationship between the different components implied in the payment gateway system, the payment authorization flow is examined between the merchant PSS and the PGWS, as follows:

1. The merchant PSS sends a payment authorization request to the PGWS, the request is sent over the TCP\IP and the destination port is the one defined in the SET setting record. The request data confirms to the SET specification.
2. On the acquirer system, the payment gateway listener is permanently waiting for requests on a dedicated IP port. The listener is responsible for extracting the core data from the incoming request message to submit it to the payment gateway application accordingly. The payment gateway listener maintains the connection with the merchant PSS for the duration of the process.
3. The payment gateway application processes the request, decrypts the SET message and validates what needed to be validated, logging on the payment gateway legacy application to return the true response.
4. The payment legacy application performs any processing required to determine the answer to be sent back to the merchant PSS. This process includes any connection to the legacy host system and the result is returned back to the payment gateway application.
5. The payment gateway application does any encryption and signing required using the gateway public key certificate to return a properly formatted SET response which is sent back to the payment gateway listener.
6. Gateway listener sends the response message to the merchant PSS with which the connection is still established. The connection is then closed.

8-3 The Purchasing Scenario

The main steps of this phase are as follows:

1. The consumer views with his browser an online merchant web page.
2. After completing browsing, purchasing and confirming the order, the web server, forwards the new client order information to its PSS through the established connection.
3. On the PSS, the PSS\Web server listener permantly keeps listening to the web server dedicated port waiting for the Payment Initialization message to arrive, and forwarding it to the Payment Initializer to process it.
4. The Payment Intializer component is responsible for two things. The first is reading the order description sent from the web server and forwarding it to the PSS purchasing application to be used in the processing required. The second is to construct the wakeup wallet message containing the IP address of the PSS and the dedicated port number for this new client and forwarding this message to the PSS\Web listener and finally constructing the accordingly PSS\Wallet listener.
5. The PSS\Web listener forwards the wakeup wallet to the merchant web server through the established connection.

6. The start SET servlet on the web server when receives the wakeup message, appends it to the client order information and forwards it to the cardholder wallet application.
7. The wallet\web server listener on the wallet software keeps listening to the dedicated SET port number, waiting for the wakeup wallet to lunch and activate the SET protocol, once receiving it, the listener forwards it to the administrator to process it.
8. The wallet administrator when receiving the wakeup message it performs the following tasks; achieving the URL of the PSS (IP address and dedicated port for this client) to forward it to the wallet\ PSS connector and secondly to achieve the order description of the client purchasing operation and performs the following:
   - Formats the received order description and purchasing amount and presents them, using the client interface to the cardholder to verify it.
   - Presents the cardholder with his certified credit data to choose the card to buy with.
   - Detects the client identity to carry on purchasing with the supplied credit card data. The wallet will never purchase with the provided credit information unless the correct UserID and Password are provided, as shown in **Fig.(13)**.

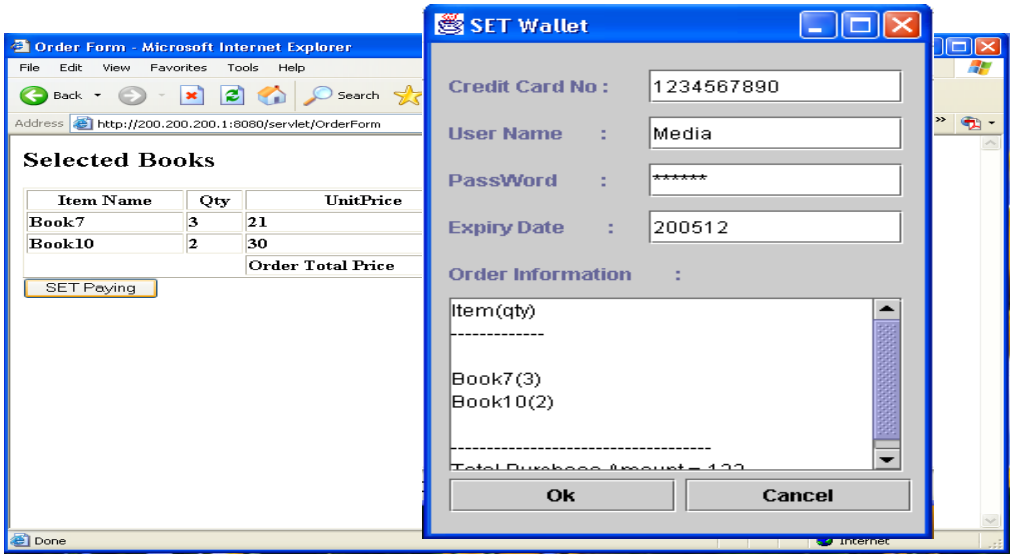


Figure (13) The verification user interface

If the order information and the client identity are verified, the administrator forwards the order and credit card information to the wallet purchasing application to start the SET protocol.

9. The wallet purchasing application, using its wallet setting components, constructs the payment request and forwards it to the PSS through the established connection.
10. On the PSS, the constructed wallet listener is waiting now for the wallet purchasing request, to forward it to the PSS purchasing application.
11. The PSS purchasing application components process the received request and forward the payment instruction data to the PSS authorization application.

12. The PSS authorization application establishes the authorization protocol with the PGWS according to the SET specification and as explained earlier and returned the response to the purchasing application.
13. Upon receiving the authorization response, the PSS constructs the purchasing response and forwards it to the wallet\PSS listener.
14. The wallet\PSS listener forwards the response to the wallet application and terminates the connection.

9. System Performance

During the experimental verification of the system, the system had been subjected to different scenarios of security attacks. The system had successfully withstood these attacks with a high robustness and a good level of performance. It is acceptable that the higher security services provided by the system pose some price that must be paid. Lag times up to 60 seconds have been noticed for the typical cardholder initiated purchase request to the approved response from the acquirer and the finalization of the transaction by the merchant server. The heavier processing load may be due to the 1024-bit RSA cryptography required in the processing of messages exchanged. Elliptic curves approach can be suggested as an alternative public key cryptography that appears to offer an equal security for a smaller key size, thereby reducing the processing overhead. However, the confidence level in elliptic curves cryptography is not yet as high as in RSA. Smart cards can also be used as an alternative of the disk storage of the certificates, which would free the cardholder from his local computer and make the system portable.

10. Conclusions

The design of the system follows the three-tiered architecture, which makes the system more flexible and easier to expand and scale out because any modification performed on any tier does not affect the other. The system also has the following two features:

- Multithreading; which permits multiple and simultaneous requests processing in a way that increase the efficiency and the speed of the system.
- Portability; because of the cross platform capability supported by the Java language, the system can be built on any platform and run on any other without recompiling the code.

The dual signature technique used by the SET protocol distinguishes it from the rest of the security payment protocols and more, solves the confidentiality and integrity problems associated with the SSL protocol. The cardholders do not have to rely on the security of the merchant server containing their credit card information, nor concerning how trust the merchant is in dealing with these data. On the other hand the defined SET certificate used can completely detect any fraud that can acquire between the merchant and cardholders and hence solves any future non-repudiation. However, the SET protocol do not give any privacy to the order information, such condition may be required by some clients, also if the web store

request any private information from their clients, e.g. for a registration purpose, all the supplied information will be in clear. The suggestion here is that to apply the SSL protocol during the initiation phase, and when the customer agrees to buy, the SET protocol begins its operation.

11. References

1. Chan, H., Lee, R., Dillon, T., and Chang, E., ***"E-Commerce Fundamentals"***, Wiley Pub., 2001.

2. Schenider, G., and Perry, J., ***"Electronic Commerce Course Technology"***, Thomson Learning, 2001.

3. Garfinkel, S., and Spafford, G., ***"Web Security Privacy & Commerce"***, O'Reilly Pub., 2002.

4. Whiteley, D., ***"E-Commerce Strategy"***, *Technologies and Applications*, McGraw Hill, 2000.

5. Ghosh, A., ***"E-Commerce Security"***, Weak Links, Best Defense, Wiley Pub., 1998.

6. Stallings, W., ***"Cryptography and Network Security Principles and Practice"***, Third Edition, Prentice Hall, 2003.

7. Thomas, S., ***"SSL and TLS, Essentials: Securing the Web"***, Wiley Pub., 2000.

8. Merkow, M., Breithaupt, J., and Wheeler, K., ***"Building SET Application for Source Transactions"***, Wiley 1998.

9. SET Co. Secure Electronic Transaction Standard Glossary, *SET Specification Book1: Business Description*, http://www.setco.com.

10. SET Co. Secure Electronic Transaction Standard Glossary, *SET Specification Book2: Programmer's Guide,* http://www.setco.com.

11. SET Co. Secure Electronic Transaction Standard Glossary, *SET Specification Book3: Formal Protocol Definition*, http://www.setco.com.

12. SET Co. Secure Electronic Transaction Standard Glossary, *SET External Interface Guide*, 1998, http://www.setco.com.