Electronic Commerce in Java
A Glimpse at the Future

By Qusay H. Mahmoud
The author is a graduate student in Computer Science at The University of New Brunswick, Saint John campus, Canada. Currently, he is working on his masters' thesis which concentrates on the Web and Java. His most recent article, Sockets programming in Java, was published in December's issue of JavaWorld Magazine.

Email: k3is@unb.ca

The exponential growth of the Web has offered new opportunities for doing commerce on the Web. However, commerce on the Web has been held hostage due to security concerns. This article reviews the Java Electronic Commerce Framework which uses a security model based on digital signatures to enable application programming interfaces to authenticate their caller.

Why Credit Card Purchasing isn't secure?

If you have been surfing on the Web long enough, you would have come across some Web sites that try to do business on the Internet. In order to purchase something over the Internet you would have to submit, usually by filling a form, your credit card number to them. Your credit card number might be seen by a third party if the underlying protocol is not encrypting the messages before condusting the transfer. On the other hand, there are sites that give you instant access to their information once you submit your Credit Card Number; however, since the algorithm for validating credit card numbers is widely known, it is possible to easily generate valid credit card numbers that could be used to get access to information from such Web sites.

The Java Electronic Commerce Framework - JCEF?

As commercial use of the Internet grows, the need for a secure mechanism for conducting commercial transactions become greater. Java creates ways to enhance electronic commerce beyond credit card purchasing. Java adds components to support emerging technologies of sophisticated payment instruments such as Smart cards, electronic cash, and electronic checks. The Java Electronic Commerce Framework (JECF) - a secure, extensible framework for creating financial applications on the Internet, is Java's solution for the growing need for a secure mechanism for conducting transactions on the internet.

Using JECF, a transaction goes into five phases as follows:

- **A shopper selects items for purchase:** An online shopper using a Java-enabled browser (e.g., Netscape) downloads a Web page containing a *shopping cart applet*. The shopper selects the items she wants to purchase. Once all the items are placed in the shopping cart, she clicks on a button to initiate the payment processing using JECF.
- **A Shopper's private database is opened:** After pressing the button at the end of the above phase, the shopper's identity is identified and her private transaction database is opened.
  The software that performs the payment on the shopper's machine is called a **Cassette** . Cassettes are similar to applets, in that they are downloaded from servers to client machines; however, unlike applets, cassettes are retained on the customer's system when the user quits the browser. Cassettes store information in a database provided by JECF, and they provide long term relationships between the customer and the financial institution.

Examples of Cassettes include, brokerage account and home banking.

- **The seller payment page with three applets is opened:** The seller payment page has three applets: one applet is the identity applet of the seller, a second applet is the tally applet that contains information about the goods and services being purchased and the total price, and the third applet helps the user in the selection of a payment instrument accepted by the seller.
- **A confirmation page appears:** This page appears after the shopper reviews all the information on the payment window and clicks on the button that dismisses that page. The confirmation page is displayed by the JEFC to ensure that the amount of $$ seen by the JEFC matches the amount on the tally applet on the seller's page. The shopper at this point has the opportunity to confirm the transaction. Once she does that the Cassette will perform the actual payment by transmitting its data to the appropriate server. While the Cassette is performing the actual payment, information about the purchase is saved in the *pending transaction list*. This information can be used to **back out** of a transaction in the case of systems crash during transactions.
- **A Verification page appears:** Once the transaction has been completed successfully, a verification page is displayed to the shopper indicating so, and information about the purchase is removed from the pending transaction list and saved in a permanent *transaction register* which allows the user to view her past purchases.

The JEFC framework support payment instruments such as: Smart cards, electronic checks, coupons, credit cards using the Secure Electronic Transaction (SET) protocol, and some other instruments. Since JEFC is an extensible framework, it can be extended to provide other types of financial service, such as: accounting, tax reporting . . . etc.

The JEFC Framework sounds like a great step forward towards Secure Electronic Commerce and hopefully when it is released to the public it will increase the volume of commercial transactions on the Internet. Once the JEFC is available, all is needed is a very high speed network to handle all the transactions. :-)

There is so much information to be covered about The Java Electronic Commerce Framework, and space doesn't allow all the information to be covered in just one article, so please web to the following URLs.

Resources:

- The Java Commerce Homepage
- The Java Electronic Commerce Framework
- The Java Language - An Overview