Abstract: Nowadays, there are multiple solutions for exchanging electronic currencies which will most probably replace physical money. This paper presents a platform for electronic currency system implemented within a Java smart card for accessing different types of products or services. The user interacts with an “electronic currency bank” and with a merchant through a secure protocol.

Key-Words: Smart Cards, Electronic Currency, Security Protocol

1. Introduction

Smart cards improve the experience of the users and the security in any transaction. They use tamper-proof storage of account identity and applications. These systems are reliable and prove to have a lower cost than other machine-readable cards, as for example magnetic stripe and barcode.

Smart cards protect against many security threats and became a cost-effective solution in many environments. The technology has developed and nowadays people are using worldwide smart cards for a variety of fields, which include: Government, Healthcare, IT, Mobile Communications, Banking or loyalty programs.

Java Card Technology is a technology that it is under Oracle license. Smart cards that are using it are compatible with all of the existing smart card standards, so this secured platform is trusted by all the manufactures.

Electronic currency is represented by money which exists only in banking computer systems and that don’t have a physical form. Most of the currencies that are in circulation today are created by government in order to make money on demand. This gives them the power of unlimited spending, although the values of the money people already have will be reduced. This is the main reason why a safe currency should not depreciate very quickly and as simple as physical money.

2. Problem Formulation

The main objective of this software solution is to create an environment that will allow users to make payments with their smart card and safely transaction an electronic currency, in order to remain in close contact with the growth and development of the electronic market.

The impressive growth of the Internet is making electronic shopping at least a real possibility, if not a habit, among computer users. However, the business model used in current electronic commerce applications still cannot enjoy the full potential of the electronic medium.

The efficiency of this solution is determined by its simplicity and rapidness in which the user can interact with his java card and simply make any personal purchase he needs, without worrying about the security of his transactions. It is also very fast for developers too, to build and deploy this kind of applications.

Transportation tickets, taxis, parking, cafeterias, and other types of vending machines are only some of the purchases that often do not reach amounts to justify the hassle of using a credit card (a cash card reader does not require a permanent phone connection with a host computer). Radio-read smart cards will allow the free flow of people through transportation...
systems, avoiding the need of ticketing machines or validation gates. The most important assets are very easy to identify: the PIN of the smart card, user's Paypal account details and the electronic currencies that are being transferred. There are at least two reasons for considering user's Paypal account details assets. First, knowing somebody's account will eventually lead to stealing his money there; then, this may be sufficient for some attacks in the electronic exchange protocol. In both situations, the most important properties are confidentiality and integrity.

- **Threats:** The threat consists of things that can be done when the assets are known, which includes the possibility of making payments in someone else's name.

- **Countermeasures:** The first countermeasure is to secure the PIN and to make sure that the information in the card is not available without it being verified. The next countermeasures consist in protecting the integrity and confidentiality of the stored data on the smart card. Then, the communication between the participants of the electronic currency protocol must be properly analyzed.

- **Impact of countermeasures:** When deciding the countermeasures that we will consider, it is important to take into consideration the (possibly negative) impact of them on the real user experience. Here, we can consider many different issues: the loss of the smart card PIN or the loss of the card itself. Moreover, there are also the details of the PayPal account that can be lost. In these situations, the issue is to back-up the data as required.

Advantages of using java smart cards are:
- The consumers can securely store cash purchasing and information on their smart card, so that make is very portable;
- Multiple standardized applications allowed to run in the same time on the card;
- The card holder is not attached to a single vendor, because he can use the smart card for any purchases at any vendor he wants that support POS technology;
- A card can carry, first of all, a personal account. The card holder can also store some credit and buying preferences information so there is no need to fill out some papers or form to do so;
- Another advantage of using a smart card is that the transaction fees associated with credit cards are eliminated.

Advantages of using electronic currencies:
- Effective control on cybernetic attacks in the money transfer systems;
- More secured system without the problem of making duplicate, copying unique code number;
- Participants of the electronic system will have a complete parallel record of every transaction;
- Online payments will be secure and easier.

3. **Problem Solution**

The software solution combines comprehensive security and payment features with an easy-to-use interface and centralized management capabilities.

3.1 **Application architecture**

3.1.1 **Java Card application**

Smart cards have a very limited memory associated so they can only store and run small applications called applets, but not only one at a time. The application in this solution consists of only one applet that is installed on the card which is PIN required.

After the authentication is done, the applet permits different actions like getting the balance of existing credits, increasing or decreasing of the this balance when requested by the host application.
3.1.2 Java Swing host application
This application (client application) runs on a laptop/pc and it is the center of the whole solution. It is a swing application that handles the requests from user and interrogates the Java Card, Electronic bank and Merchant individually. The swing application has the following functionalities:
- Reads the status of the wallet;
- Communicates with the Electronic Bank for acquiring new electronic currency in exchange of real money payments. It is updating also the balance of smart card account;
- Communicates with the Merchant for acquiring different products in exchange of electronic currency.
A secured connection is established when the user tries to pay with his smart card wallet a product from the Merchant platform. If the transaction is accepted and finalized, the swing application must decrease the balance of the account and the user is receiving ordered product or service.

3.1.3 Electronic Bank server (server application)
This server has as its main functionality the exchange of electronic currency using a secure protocol, that we will generically call SecCoin. The payments are, at first, secured by Paypal API, so the user has the confirmation that the payment it is made according to current security standards. Then, after the payment is validated, the exchange of the electronic coins will be made through SecCoin protocol. The electronic bank has two balances to manage: the electronic currency balance (to give to the user when requested), and the real money balance where all the payments from users go.

3.1.4 Merchant server (Java web application)
This is a platform that has products for sale, and in exchange of an electronic currency, has the purpose to sell products. The user chooses the right product to buy and proceeds with order. After the payment is completed, the user gets the product.
This server has multiple tasks:
- It keeps a balance of products, with possibilities of increasing and decreasing the stock.
- It has a balance of electronic currency from previous orders, and it’s being updated when a new one is taking place.
- Has a history of previous orders.

3.2 Data flows
Step 1: This is the first step a user should make within application. With step 1 the user can buy electronic currency from the
**electronic bank.**

**Step 2:** This step is available after the user is already authenticated on his smart card. An information screen is displayed with data regarding the card balance, and other useful information regarding the history of transactions and user preferences read from a database.

**Step 3:** The last step from the Figure 2 is also the most complex. It requires information and data exchange from all sides (smart card, swing application and merchant server).

![Figure 2. Application data flow of the application](image)

### 3.3 SecCoin protocol

The protocol is a simplified version of the Bitcoin Protocol. In order to explain main ideas behind the protocol, we will use classical Alice and Bob character interaction, which will be considered participants.

Alice has some digital money which she wants to spend. Well consider this money is represented by a string of bits and we should prevent her from spending the same bit string more than one time. Then, we should prevent someone else forging and using that string of bits to steal from Alice. We will use the digital signature of Alice to sign the message using a private cryptographic key. After that, Alice will announce she signed that particular string of bits to all of the participants of the protocol. Anyone can verify that Alice signed the message using her public key.

Then Alice can give her electronic coin to Bob, in order to pay for some products or services he provided for her.

In order to uniquely identify the coin she already spent, we will use serial numbers for each coin in the protocol. One possibility would be to have a central bank that will have a centralized list of all uniquely identified coins and their owners. When a coin is given from a participant to another, the bank should be informed and should update its recordings.

Another way of doing things would be to make every participant of the protocol have a complete list of the coins. In this case, when a transaction is made, everyone has to be informed and has to update his local list.

The second situations would protect the protocol from being compromised by an attacker that will take advantage of the authority of the central bank.

An important problem to take into consideration would be the double spending of the same coin. The solution to this problem is the verification of each transaction in the protocol by some or by all of the participants. In order that a transaction can be approved and accepted by Bob, is has to be verified by a minimum specified number of people or by everyone and collectively accepted.

After this is done, everyone will update his local list of the coins and their owners. Then, we should consider another important threat, the case when an attacker will take advantage of his power and pretend he represents more than one participant. This way, he can and he will verify and accept invalid transactions in order to make everyone incorrectly update his list.

Also, there comes the most important element of the protocol. If anyone wants to validate a transaction, he previously has to solve a computational problem that will take time (which is the key of the
entire protocol). This way, the attacker will not have the chance to resolve all of the puzzles in time and other participants will do it. As a compensation of the participants that use their computational power to confirm transactions, each of them will receive some electronic coins after they solved the problem.

### 3.4 Setting input data in accordance with outputs information

Nowadays we can notice that the smart cards are turning out to be a fundamental piece of the transformation of retailing into electronic commerce. One of the main concern of working with smart cards is security. And security can only be achieved with well formatted and standardized input and output data. The **input data** within software application are one of the most important things to set right from the beginning of the project. A well formatted input reduces dramatically the chance to get an unexpected result.

For the current project, as the input data we have 5 categories of data:

- Currency data: The money used to acquire virtual credit coins is USD, and any other currency is not permitted;
- Products: Fields for products to be displayed within the application are well known (ID, Name, Price, Availability, Description);
- Transaction stored: All transaction described at previous chapter and stored in databases for further use and verifications;
- Card holder information;
- Bank and Merchant information.

With the input data well formatted, the **output results** will be as expected:

- All the output data will be displayed through solution interface only, not unauthorized access;
- Card balance to be correctly managed
- Products balance to be correctly managed;

During application flows, the following **rules** regarding data are respected:

- Feedback during data entry;
- Fast response;
- Single method for entering data;
- Define display areas for data entry;
- Consistent Method for Data Change;
- Feedback for completion of data entry;
- Errors displayed when needed.

### 4. Conclusion

The suggested secure architecture is based on concepts like Java Card security, electronic signature, an electronic currency exchange protocol and secure transmission on the internet using Hypertext Transfer Protocol Secure (HTTPS).

The need to respond in optimum time to market requirements has led to preoccupation on a better security and a better user experience. The efficiency of this solution is determined by its simplicity and rapidness in which the user can interact with his java card and simply make any personal purchase he needs, without worrying about the security of his transactions.

The solution shows that the Smart Card payments and electronic currencies have become long future markets.

### Acknowledgement

Parts of this paper were presented at The 7th International Conference on Security for Information Technology and Communications (SECITC 2014), Bucharest, Romania, 12-13 June 2014.

### References

5. MIT Open Courses, http://web.mit.edu