Security Threats in Electronic Currency Exchange Protocols

Marek R. Ogiela* and Piotr Sułkowski
Department of Automatics and Biomedical Engineering,
AGH University of Science and Technology, Poland
mogiela@agh.edu.pl, piotr.sulkowski@o2.pl

Abstract

In this paper will be presented secure protocol improvement to the known David Chaum anonymous currency exchange procedure. We show its vulnerability to possible frauds after a digital coin is spent several times. In such case, the system cannot successfully determine how many times the coin was spent and how many times the seller may faked the transaction. Finally in such attack the bank is not able to charge the real abuser. This limitation leads to the conclusion that the original system cannot be securely used for irreversible off-line transactions. In this paper we try to solve this problem and propose an improved method which allows this vulnerability to be overcome.

Keywords: Digital Coin, Electronic Cash, Anonymous Transaction

1 Introduction

Today, even though the existence of ubiquitous online banking and electronic payment infrastructure, there is no chance to completely secure transfer money in a way guaranteeing anonymity. This means that no electronic transaction can be made in secret, as is made when traditional banknotes are used. However, for many reasons the exchange of regular cash is not a convenient method, so a way of anonymously transferring cash over the Internet would be very useful. There are at least two ways in which such a system can be built. One of them is to create a virtual currency, and the second way is to create an electronic cash system. The first method enabled cheques to be anonymously exchanged is David Chaum's protocol described in paper [4]. Despite it was developed a long time ago and has been improved many times, it has so far been the basic solution and a large proportion of protocols developed later were founded on it [1, 2, 5, 6, 7].

In this paper we try to present a possible attack on David Chaum's basic protocol, and later propose an important modification which allow to eliminate fraud attempts. Such secure and improved procedure will be described in next sections.

Before its presentation we can describe some basic properties of systems processing anonymous transactions. In an electronic cash exchange system, the transactions may be realized in three stages.

At the first, the client contacts the bank and downloads an electronic cheque. Then, the client may contacts the seller and spends the cheque obtained from the bank. At the third stage, the seller contacts the bank and presents it with a certificate of concluding a transaction with the client, in return for which it receives its monetary equivalent.

If transaction protocols are executed like this, the main problem is that the bank must not be able to associate the transaction certificates presented to it by the seller with the cheque which it had issued to the client.

In general the main features of electronic cash exchange systems are following:

*Corresponding author: Department of Automatics and Biomedical Engineering, AGH University of Science and Technology, 30-059 Krakow, Al. Mickiewicza 30, Poland

- Anonymity – the system must guarantee that the bank cannot trace the transactions.
- Transaction processing may occur when the banknotes are spent, or may take place later.
- Transaction should be reversible.
- Requirements concerning secure access to data.

Below we present an improved protocol based on such requirements.

2 Properties and limitations of Chaum's protocol for electronic cash exchange

The first and best known method of implementing an electronic cash system which meets all the assumptions presented above was firstly outlined in [3] and then proposed in [4] by David Chaum. This scheme has become the reference example for implementing an electronic cash exchange. In this solution, the banknote is composed of a signed n element sequence of pairs $P_i(i \in 1,2,\dots n)$ having the following structure:

$$P_i = (h(a_i, c_i), h(a_i \bigoplus u, d_i)) \tag{1}$$

where:
u – a unique client ID also known to the bank
a_i - a number randomly selected by the client to hide the value of u
c_i, d_i - numbers randomly selected by the client to create the hash function with a password
n is a certain constant of the system and determines its security (at the cost of its possible efficiency).

The way in which electronic cash is transferred between bank, seller and client is presented in more detailed way in [11].
The basic properties of this protocol are following:

- A transaction can be concluded without the need to contact the bank at the time of its conclusion.
- No need to use any special tamper-proof devices or cards.
- Transactions are anonymous if the client is behaving honestly.
- The security of every party is guaranteed even if the remaining two parties are acting in collusion against it.

Beside these properties, some possible risks to the parties using such protocol may arise. The main ones are: the same banknote being spent many times, counterfeit banknotes being made and the loss of the client's anonymity.
Let us then consider a case in which the client tries to spend a banknote several times. In this case, the risk to the bank and the seller depends greatly on the strategy of action chosen. There are at least two different options:

1. **Strategy A** In exchange for every banknote correctly presented by the seller, the bank pays money out, even if the client has spent this banknote several times. The client is then obliged to pay for all transactions concluded.

2. **Strategy B** If it is detected that a banknote has been used several times, no funds are credited to the seller's account. The seller is only given the personal data of the dishonest client. It is then the seller who must reverse the transaction and possibly claim damages.

In strategy B all the risk rests with the seller. If the seller incurs any costs of the transaction, it must charge them to the client itself. The limitations resulting from selecting strategy B are a reason to consider strategy A. This solution implies that the bank is responsible for prosecuting dishonest clients. However, it is difficult to claim the amount due from clients. One can imagine a situation in which a person with an average income spends one banknote worth one dollar one million times. The bank has to pay a million dollars to sellers. However, it stands no realistic chance to recover this amount due from the client. Another problem is banknote theft. The thief can spend the stolen banknotes many times, each time charging the account of the client it has stolen them from. Consequently, strategy A gives rise to a greater risk of the bank and the client. Under the B strategy, the risk is mainly borne by the seller, and this strategy restricts the functionality of the system. If a system fulfils all the conditions for strategy A to be adopted, it can also operate according to strategy B. In the opposite case, it is necessary to introduce a mechanism for proving the sale.
Finally the presented system is only capable of executing fully reversible transactions without a guarantee that damages will be received if a fraud is committed.

3 An improved protocol supporting multiple transaction detection

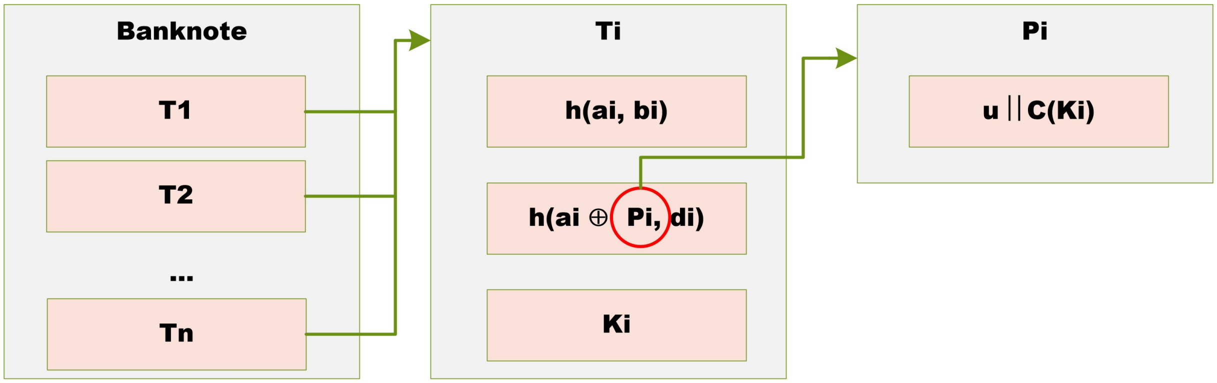
In order to enhance Chaum's system towards offline transactions can be concluded, it is necessary to introduce the ability to prove all transactions. This will make it possible to claim compensation if the same banknote is spent more than once. To achieve it, it's necessary to change the certificates issued by the clients in such a way that sellers cannot generate new ones based on any number of those already held.
We therefore propose a modification of the protocol in which clients sign all the challenges received from sellers and send them together with certificates. However, if they used their own key for signing, they would cease to be anonymous. It is therefore necessary to apply a one-time key which should be tied to the real key of the client somehow. One of the possible ways is to attach it to the banknote together with its certificate signed by the client (the structure of the banknote is presented in Figure 1). Instead of pairs Pi, the client then sends triplets:

$$T_i = (h(a_i, c_i), h(a_i \bigoplus (u \parallel C(K_i)), d_i), K_i) \tag{2}$$

where:
a_i, c_i, d_i - random numbers chosen by the client
u – the unique ID of the client
K_i – public part of a one-time RSA key generated by the client. This can be written e.g. as $(e \parallel n)$, where e is the public exponent of this key, while n is its module
C – the client's certificate employed to sign all K_i keys. This can be e.g. the number $h(K)^d mod\ m$, where the numbers (d, m) constitute the private part of the RSA key published by the client (the so-called main key)

Before starting to create banknotes, the clients must register their main public keys with the bank [8]. It is best if clients use keys certified by a certain certification authority. The keys are registered only once for each client, at the time its account is created.

The protocol of creating the banknote, in which the bank and the client are involved, is similar as in original version. The difference is the banknote itself, which consists of n triplets T_i, and not pairs P_i as



Ti – triplet containing hashed parameters and public one-time key
Pi - client's unique identifier and a certificate of the one-time key included in the same triplet
Ki – public one-time key used to verify signature of the challenge

Figure 1: Modified digital note

in the previous case. The client sends the bank 2*n obfuscated banknotes, from which the bank chooses one half and asks the client to remove the obfuscation from them. Having received all the necessary coefficients, the bank checks the regularity of the banknotes just as before. In addition, it must assure itself that all certificates $C(K_i)$ contained in the sent banknotes are the correct certificates of keys K_i, i.e. they apply to the key Ki contained in the subsequent part of the banknote and they have been signed with the client's main key (which the bank holds in its database). If everything is correct, then the bank sends the signed banknote to the client just as in the previous version of the protocol. Thus the client, having removed the obfuscation, has the following number, which represents modified banknote:

$$z = \prod h(T_i)^d mod\ n, i \in L \tag{3}$$

where:
d – the private exponent of the bank key
n – the bank signature module
L – a set of indexes of banknotes selected for signing by the bank

The idea of this entire improvement is that every transaction executed by the client should leave a unique trace that cannot be faked. To obtain this functionality of the protocol, the client signs the challenge sent to it by the seller using the Ki keys contained in the banknotes. The protocol for the banknote exchange between the client and the seller (presented in Figure 2) thus looks as follows:

1. The client sends the banknote Z signed by the bank.

2. The client also sends the Ti triplets (i.e. the values $h(a_i, c_i)$, $h(a_i \bigoplus (u \parallel C(K_i)), d_i)$ and K_i).

3. The seller checks whether banknote Z is the correct signature of the signed triplets.

4. The seller sends the challenge Y to the client.

5. The client provides the seller with the value of the challenge Y signed with all one-time keys K_i:

$$R = K_1(K_2(\dots K_n(Y)\dots)) \tag{4}$$

where: $K_i(x)$ – the signature of the value x with the use of the key K_i

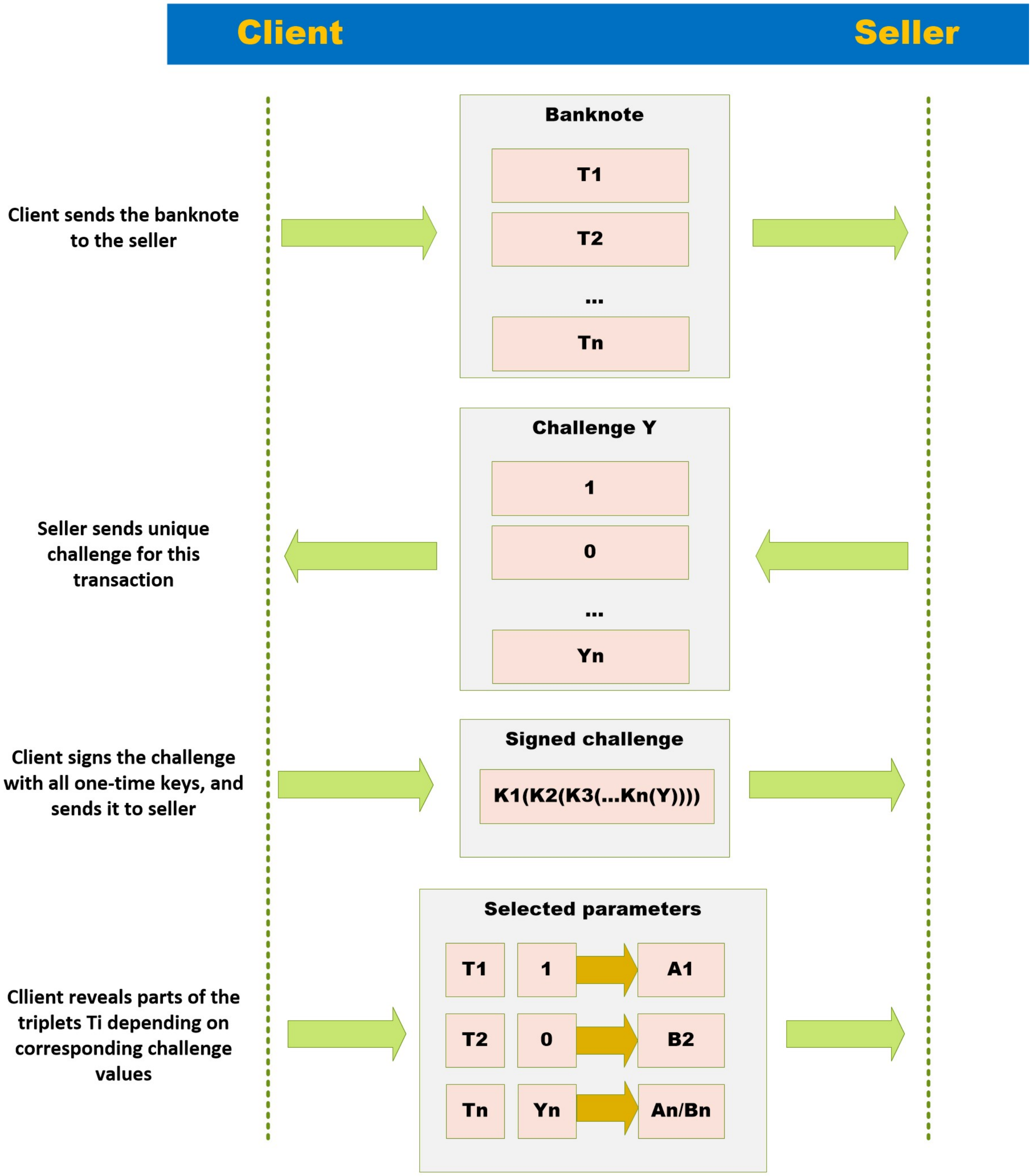
6. The seller verifies the validity of the signature R.

7. The client provides the seller, respectively, with the values (a_i, c_i) or $(a_i \bigoplus (u \parallel C(K_i)), d_i)$ depending on the value of the ith bit of challenge Y (just as in the previous version of the protocol)

8. The seller checks whether the data sent corresponds to the hashes contained in triplets Ti. If everything is correct, the payment is accepted.

In order to cash the banknote, at each moment, the seller presents the bank with the signed banknote Z, the sequence of triplets T_i, the generated challenge Y together with the signature R and all the values dependent on this challenge sent by the client. The bank is able to check the regularity of all data in the same way as the seller was able to when it exchanged the banknote with the client.
Just as in the standard version of the protocol, after spending the banknote once, the client reveals only one half of each liability. At the same time, if the same banknote has been spent at least twice, the bank is highly likely to possess two complementary halves, but thanks to the modification it will learn not only the client's identity, but also at least one of the certificates $C(K_i)$.

Thus the bank becomes able to prove to the client that it has spent a banknote more than once. This is because every transaction is signed by the client with all keys K_i. At the time of the fraud, the bank not only knows the identity of the client, but also holds at least one set containing the liability signed with a certain one-time key and the client's certificate authenticating this key. If the bank is able to provide the client with the signature R of a given challenge Y and to prove to it that the signature belongs to the client, this transaction can be considered proven. This is because no one other than the client can create the certificate for the key K_i used to sign the challenge. Neither can anyone fake this signature as the banknote only contains the public part of it.
If the client spends banknotes only once, then it is impossible to learn either its ID u, or any of the certificates $C(K_i)$. This certificate, together with the key Ki, could also be used to identify the client. It is enough that the bank tries to verify this certificate by using all main keys of clients it holds in its database. One of them would probably be correct. This could be the basis for discovering the identity of the client, so certificates must also be kept secret until a fraud occurs. What is, however, overt is the key K_i itself. It is created randomly by the client and contains no information that could identify it.
After this solution is implemented, the bank is able to prove exactly how much the client has spent. Consequently it can adopt the strategy of paying funds to all sellers who present correct transaction certificates. Thus the system makes offline transaction conclusion possible.
What still remains is the problem of banknote theft. If a banknote ends up in the hands of an unauthorised person, it can be used to overdraw the owner's account without any limitation. To prevent this, once the client learns of the theft, it can report it to the bank so that the latter publishes a list of void banknotes. In addition, the bank itself, once it detects a double payment, can publicly report this banknote as stolen. On the other hand, if we assume that transactions are concluded without contacting the bank, we can never eliminate this problem completely. However, the same difficulty arises in the digital signature scheme itself. The problem can be eliminated if, every time before we start receiving a digital signature, we refer to a public database to check if the signature has not been stolen. However, if we decide to build a system which accepts signatures offline - without contacting a public database of stolen signatures – we can never be certain that the signature has not been stolen.
If the risk of theft is considered to be too high, it is always possible to fall back on the strategy of concluding only irreversible transactions. Apart from capacity issues, the proposed modification does not weaken the original system in any way.


Client
Seller
Banknote
T1
T2
...
Tn
Client sends the banknote to the seller
Challenge Y
1
0
...
Yn
Seller sends unique challenge for this transaction
Signed challenge
K1(K2(K3(...Kn(Y))))
Client signs the challenge with all one-time keys, and sends it to seller
Selected parameters
T1
1
A1
T2
0
B2
Tn
Yn
An/Bn
Cllient reveals parts of the triplets Ti depending on corresponding challenge values

Ai – parameters *ai* and *bi* (hash of them are stored in the first part of each triplet)
Bi – parameters *ai* ⊕ *Pi* (hash of them are stored in the second part of each triplet)

Figure 2: The electronic coin exchange protocol between the client and the seller

4 Conclusion

The origin system proposed Chaum is the first system capable of processing anonymous transactions offline and represented a ground-breaking discovery in the field of anonymous electronic cash exchange. However, if the same banknote is spent many times by the client, the system becomes susceptible to attacks both by cheated sellers and the client itself. This weakness means it cannot be used to conclude irreversible transactions. The same problem also applies to other systems based on a similar protocol. The introduction of the modification proposed by the authors allows such attacks to be prevented and makes it possible to conclude hard transactions as well.

It seems that, sooner or later, electronic cash technology will gain in popularity and will start replacing traditional credit cards and bank transfers making electronic money exchange more reliable and secure [9, 10, 12].

Acknowledgments

This work has been supported by the AGH University of Science and Technology research Grant No 11.11.120.329

References

[1] S. Brands. Untraceable off-line cash in wallets with observers. In *Proc. of the 13th Annual International Cryptology Conference on Advances in Cryptology(CRYPTO'93), Santa Barbara, California, USA*, volume 773, pages 302–318. Springer-Verlag, August 1994.

[2] S. Brands. Off-line electronic cash based on secret-key certificates. In *Proc. of the 2nd Latin American Symposium on Theoretical Informatics(LATIN'95), Valparaíso, Chile*, volume 911, pages 131–166. Springer-Verlag, April 1995.

[3] D. Chaum. Blind signatures for untraceable payments advances in cryptology. In *Proc. of the 2nd Annual International Cryptology Conference on Advances in Cryptology(CRYPTO'82), Santa Barbara, California, USA*, pages 199–203. Plenum Press, New York, 1983.

[4] D. Chaum, A. Fiat, and M. Nao. Untraceable electronic cash. In *Proc. of the 8th Annual International Cryptology Conference on Advances in Cryptology(CRYPTO'93), Santa Barbara, California, USA*, volume 403, pages 319–327. Springer-Verlag, 1990.

[5] R. H. Deng, Y. Han, A. Jeng, and T. Ngair. A new on-line cash check scheme. In *Proc. of the 4th ACM conference on Computer and communications security(CCS'97), Zurich, Switzerland*, pages 111–116. ACM, April 1997.

[6] N. Ferguson. Single term off-line coins. In *Proc. of the 12th workshop on the theory and application of cryptographic techniques on Advances in cryptology(EUROCRYPT'93), Lofthus, Norway*, volume 765, pages 318–328. Springer-Verlag, May 1994.

[7] S. Kim and H. Oh. Single term off-line coins. *International Journal of Information Security*, 1(3):175–188, 2002.

[8] W. Mao. *Blind Certification of Public Keys and Off-line Electronic Cash*. Hawlett-Packard Laboratories, 1996.

[9] M. R. Ogiela and U. Ogiela. Dna-like linguistic secret sharing for strategic information systems. *International Journal of Information Management*, 32(2):175–188, 2012.

[10] M. R. Ogiela and U. Ogiela. *Secure Information Management using Linguistic Threshold Approach, Advanced Information and Knowledge Processing*. Springer-Verlag, 2014. DOI 10.1007/978-1-4471-5016-9, ISSN 1610-3947, ISBN: 978-1-4471-5015-2.

[11] M. R. Ogiela and P. Sułkowski. Protocol for detection of counterfeit transactions in electronic currency exchange. In *Proc. of the 3nd International Conference on Cryptography and Security Systems(CSS'14), Lublin,*

Poland, volume 448 of *Communications in Computer and Information Science*, page 145–152. Springer-Verlag, September 2014.

[12] B. Schneier. *Secrets and Lies: Digital Security in a Networked World.* Wiley, 2004.

Author Biography

Marek R. Ogiela is a professor of Computer Science, cognitive scientist and cryptographer, head of Cryptography and Cognitive Informatics Laboratory. Professor Marek R. Ogiela works at the AGH University of Science and Technology and Pedagogical University in Krakow. In 1992 graduated from the Mathematics and Physics Department at the Jagiellonian University. In 1996 for his honours doctoral thesis on syntactic methods of analysis and image recognition he was awarded the title of Doctor of Control Engineering and Robotics at the Faculty of Electrical, Automatic Control, Computer Science and Electronic Engineering of the AGH University of Science and Technology. In 2001 he was awarded the title of Doctor Habilitated in Computer Science for his research on medical image automatic analysis and understanding. In 2005 he received a professor title in technical sciences. Member of numerous world scientific associations as well as of the Forecast Committee 'Poland 2000 Plus' of the Polish Academy of Science and member of Interdisciplinary Scientific Committee of the Polish Academy of Arts and Sciences (Bio cybernetics and Biomedical Engineering Section in years 2003-2011). Author of more than 290 scientific international publications on pattern recognition and image understanding, artificial intelligence, IT systems and biocybernetics. Author of recognised monographs in the field of cryptography and IT techniques; author of an innovative approach to cognitive medical image analysis, and linguistic threshold schemes. For his achievements in these fields he was awarded many prestigious scientific honors, including Prof. Takliński's award (twice) and the first winner of Prof. Engel's award.

Piotr Sułkowski graduated from the Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering at the AGH University of Science and Technology in 2013. The title of his thesis was "Cryptographic Protocols for Anonymous Currency Exchange". Since then he develops his interests in the area of electronic currency exchange as well as systems security and software architecture. Currently, he works as a programmer in LGBS Polska Sp. z o.o. and simultaneously continues his scientific research. He is a co-author of several international publications, inter alia, in "Soft Computing" and in the Proceedings of the Third International Conference CSS 2014.