A Novel Fair Anonymous Contract Signing Protocol for E-Commerce Applications

H.Jayasree[1] and Dr. A.Damodaram[2]

[1]Assoc. Prof, Dept. of IT, ATRI, Hyderabad.
jayahsree@yahoo.com
[2]Director – Academic Audit Cell & Prof. of CSE Dept, JNTUH, Kukatpally, Hyderabad.
damodarama@rediff.com

Abstract

With the economy developing and popular Internet, the general concept of contract signing has changed. In the past, people usually sign a contract at the same time and same place face to face, but actually each party involved in contract may live in different part of earth, they want to sign something for business or some other things in economic, efficient, secure and fairway. A fair contract signing protocol allows two potentially mis-trusted parities to exchange their commitments (i.e., digital signatures) to an agreed contract over the Internet in a fair way, so that either each of them obtains the other's signature, or neither party does. Based on the LUCAS signature scheme, a new digital anonymous contract signing protocol is proposed in this paper. Like the existing LUCAS-based solutions for the same problem, our protocol is fair, anonymous and optimistic. Furthermore, the proposed protocol satisfied a new property, i.e., it is abuse-free. That is, if the protocol is executed unsuccessfully, either of the two parties can not show the validity of intermediate results to others.

Keywords

Contract signing, fair-exchange, digital signatures, LUCAS, e-commerce, cryptographic protocols, security.

1. Introduction

Services of E-business or E-commerce can be categorized in two classes, namely, E-contracting and E-trading. Signing a contract, where the parties are distributed geographically, is one of the major activities in today's commerce, business and governance. E-contracting is a major activity of Ecommerce or E-business or E-governance. But the success of E-commerce (either E-contracting or E-trading) faces a major challenge of information security. In the paper-based scenario, contract signing is simple due to the existence of "simultaneity". The parties involved generally sign two hard copies of the same contract at the same place and at the same time. After that, each party keeps one copy as a legal document that shows the proof that both of them have committed to the contract. If one party does not abide by the contract, the other party could provide the signed contract to a judge in court [6].In today's world contracts are vital for all kind of businesses, including electronic commerce. Any agreement between two or more parties involves a contract of some kind example on-line purchases imply a contract that promise to exchange money for some goods or service. Electronic contracts make electronic commerce possible.

DOI : 10.5121/ijnsa.2012.4509

2. Previous Work

In this section we discuss briefly some related work in providing fair-exchange in E-commerce, particularly in contracting [1]. To obtain non-repudiation of origin and delivery of an email message the idea of using a trusted third party in online mode was proposed by Deng et al. [3] and Zhou and Gollmann [5]. In essence, these protocols are similar. In these protocols, the dispute resolution is outside the scope of the protocol. However, the protocols specify the evidences which are to be stored and the way of collection of these evidences for the dispute to be resolved in a fair manner. Franklin and Reiter [7] propose a set of fair-exchange protocols that verify the consistency of a document before the exchange takes place. These protocols require a semi-trusted third party. A semi-trusted third party is one that can misbehave on its own but will not collude with any of the participating parties. There are several fair exchange protocols that use third party in off-line mode, when it is required and hence they are optimistic fair exchange protocol. These protocols are designed either to sign a contract [8,9] or to purchase a digital product [10,11,12]. Asokan, Shoup and Waidner[8] designed An Optimistic Contract Signing Protocol to provide a service to Originator and Responder for obtaining each other's commitment on a previously agreed content. The protocol consists of three interdependent sub-protocols, viz., Exchange sub-protocol, Abort sub-protocol and Resolve sub-protocol. This asynchronous protocol, in essence, a fair exchange protocol involves three participating parties, viz., originator (O), Responder (R) and trusted third party (T). As it is a contract signing protocol, the protocol does not consider the anonymity property for any transacting party.

In this paper we propose a new digital anonymous contract signing protocol based on LUCAS signature scheme. Like the existing LUCAS-based solutions for the same problem, our protocol is not only fair, but also optimistic.

3. Analysis of Contract Signing Features

3.1 Contract Protocol Properties

Following are the properties required in a contract signing protocol taken from [2].

One necessary property is *fair contract signing*. A protocol is said to be fair if either party could not gain any advantage by terminating the protocol in the middle. The common resolution for fair exchange is to use a TTP. The TTP approves the contract only after the signing of all parties involved is done. Conventionally this solution requires that all the communication go through the TTP, i.e. an in-line TTP.

Another important property in a contract scheme is *accountability*. Accountability means that if the TTP misbehaves in any way, then misbehave of TTP need to be proven. The contract between the parties is then ruled invalid. At the same time it must be infeasible for Alice and Bob to frame the TTP if it does not misbehave. A misbehave executed by any party for ex. Alice or Bob or TTP needs an accountability to ensure that the contract is valid.

Non-repudiation is an important security service that gives more trust to e-commerce. Non-repudiation is a security service that creates, collects, validates, and maintains cryptographic evidence, such as digital signatures, in electronic transactions to support the settlement of possible disputes. If the Internet had more non-repudiation services, it would be more difficult to commit fraud and other malicious activities.

Completeness is also a desired property in contract signing protocols. A protocol should execute in completeness. A protocol should be robust against adversaries attempting to cause to abort without the consent of either party.

3.2 Role of Trusted Third-Party

TTP works as a third participant in scenarios where two parties need additional trust between them, for instance when it comes to contract signing. This trusted third party can in real life be compared to the post office receiving registered mail, and getting a receipt from the receiver before delivering the registered mail. In this case both parties trust the post office.

There are three different kinds of TTPs:
• In-line TTP where the TTP acts as an intermediary between the two participants.
• On-line TTP where the TTP is actively involved in every instance of the non-repudiation service.
• Off-line TTP where the TTP provides non-repudiation without being involved in each instance of the service. It is only involved when needed.

3.3. Desirable Properties of the proposed protocol

(1) *Fairness* : Our protocol guarantees the two parities involved to obtain or not obtain the other's signature simultaneously. This property implies that even a dishonest party who tries to cheat cannot get an advantage over the other party.

(2) *Optimism* : The third trusted party (TTP) is involved only in the situation where one party is cheating or the communication channel is interrupted. So it could be expected that the TTP is only involved in settling disputes between users rarely, due to the fact that fairness is always satisfied, i.e., cheating is not benefitial to the cheater.

(3) *Abuse-Freeness* : If the protocol is not executed successfully, any of the two parties cannot show the validity of the intermediate results generated by the other to an outsider. As we mentioned before, the unique known abuse-free contract signing protocol is based on the discrete logarithm problem, instead of the RSA cryptosystem.

(4) *Provable Security* : Under the standard assumption that the RSA problem is intractable, the protocol is provably secure in the random hash function model, where a hash function is treated as if it were a "black box" containing a random function.

(5) *Timely Termination* : The execution of a protocol instance will be terminated in a predetermined time. This property is implemented by adding a reasonable deadline t in a contract. If one party does not send his/her signature to the other party after the deadline t, both of them are free of liability to their partial commitments to the contract and do not need to wait any more.

(6) *Non-Repudiation* : Our protocol will not allow any party involved in contract to arbitrarily deny or withdraw signing this contract without another party's permission after the contract signing succeed.

(7) *Zero-Knowledge* : The Signature Exchange sub-protocol in our protocol use the interactive proof properties to achieve the proof , and it is also zero-knowledge, means that during the signature exchange, nothing can be leak to use to forge signatures.

(8) *Completeness* : This property unites with Fairness and non-repudiation to make sure the atomicity. An attacker can launch two instances of the protocol, one with Alice, the other one with Bob, but the attacker cannot get any benefits with just half signature. Both parties get the other's full signature at the last step.

(9) *Confidentiality* : This property means that the contract signing should not reveal the contents of the contract during the signing process. In our protocol, we use a DES session key to encryption every data of transmission, that DES session key is produced from X.509 two-way authentication.

(10) *Anonymity* : Each party authenticates anonymously using a pseudonym in order to avoid a scenario of impersonation. The pseudonym used by each party is created by TTP and sent securely to both parties before initiation of the protocol. This property ensures anonymity throughout the transaction.

(6) *Compatibility* : In our protocol, each party's commitment to a contract is a standard digital signature. This means that to use the protocol in existing systems, there is no need to modify the signature scheme or message format at all. Thus, it will be very convenient to integrate the contract signing protocol into existing software's for electronic transactions.

4. LUC CryptoSystem

4.1 The LUC Public Key System

It is based on a different trapdoor function from the RSA and El Gamal systems, which is defined by Lucas functions. Because the properties of Lucas functions mirror those of exponentiation, public key and private key processes can be developed in an exactly analogous manner to the RSA system. This enables us to prove that any successful attack on the LUC system would give a successful attack on the RSA system. Since the weakness of the RSA system does not occur for the LUC system, the LUC system is cryptographically stronger than RSA.

4.2 LUC

Suppose N and e are two chosen numbers, with N the product of two different odd primes, p and q. The number e must be chosen so that it is relatively prime to (p-1)(q-1)(p+1)(q+1). Let M be a message which is less than N and relatively prime to N, we define $f_{LUC}(M) = Ve(M,1) \bmod N$, where Ve is a Lucas function. This is the LUC public key process, giving an encrypted message, M'. To define the matching private key process, we need a number d such that $de=1 \bmod S(N)$, where $S(N) = lcm((p-(D/p)), (q- (D/p)))$ where $D =(M')^2 - 4$ and (D/p) , (D/q) are the Legendre symbols of D with respect to p and q. We assume that D is relatively prime to N, so the Legendre symbols are either +1 or -1. The private key process is then the same as the public key process with e replaced by d. Therefore the decrypted message M, $M = Vd(Ve(M,1) \bmod N,1) \bmod N$ and the private key process and public key process are inversions of each other by the symmetry between e and d.

4.3 Cryptographic Strength of LUC

Just as for the RSA method, the LUC private key process can be discovered only if there is a way of computing $V_d (M ' , 1) \bmod N$ without knowledge of d , or if there is a way of finding d from e and N . The second problem is harder than the corresponding problem for the RSA method, because there are four different values of d for each pair of e and N , only

one of which will work for an arbitrary M ' . The first problem is really no different from the problem of computing powers; trial of all possible values seems to be the only way.

The fact that Lucas functions are a generalization of powers makes it certain that any successful attack on the LUC public key system would automatically lead to a successful attack on the RSA public key system. Because of the additional complications with Lucas functions, however, the reverse may not be true; successful attack on RSA may not lead to a successful attack on LUC. For example, the weakness of RSA, due to its multiplicative nature, is not shared by LUC. Thus we say, with confidence, that LUC is cryptographically stronger than RSA.

The *advantage* of implementing the Contract Signing Protocol based on LUC Cryptosystem is that, the computational effort required for the LUC public key process is about the same as that required for the RSA public key process, while the LUC private key process involves less than double the computational effort of the RSA private key process(and may take considerably less, if parallel computation is available). Some signature formation time could be saved by omitting the hashing process, since LUC is not susceptible to adaptive chosen-message attacks, but in practice a simple hashing method would be used to compress the message before signing.

5. Proposed Protocol

The basic idea is that Alice first splits his/her private key k into k1,k2 so that k=k1+k2 mod ϕ(n). Only k2 is delivered to the TTP, while Alice keeps (k,k1,k2) as secrets. To exchange the signature with Bob, Alice first sends partial signature to Bob, and proves that the partial signature is prepared correctly in an interactive zero-knowledge way. After that, Bob sends his signature to Alice, since he is convinced that even if Alice refuses to reveal the second partial signature, the TTP can do the same thing. The protocol can be subdivided into 3 sub protocols

5.1 Registration Protocol

To use our protocol for exchanging digital signatures, only the initiator needs to register with the TTP. That is, initiator is required to get a voucher V_A from the TTP besides obtaining a certificate from CA. Though the above registration protocol is a little complicated, we remark that this stage needs to be executed only once for a sufficiently long period, for example, one year. In this period, initiator can fairly sign any number of contracts with all potential parties. Furthermore, it seems reasonable in the real world to require users to first register with the TTP before they are served. Further TTP assigns pseudonyms for the users to ensure anonymity. The reason is that the TTP is usually unlikely to provide free service for settling disputes between users.

The user has to first register to TTP, to ensure that he is an authenticated party, and to satisfy the requirement of *optimism* so that TTP can take care in the scenario of dispute or in the scenario of *non-repudiation*. TTP generates pseudonyms to user so that the user remains *anonymous* in order to avoid any impersonation by an attacker.

5.2 Signature Exchange Protocol

We assume that a contract M has been agreed between Initiator and Responder before they begin to sign it. In addition, it is supposed that the contract explicitly contains the following information: a predetermined but reasonable deadline t, the anonymous identities of Alice, Bob and the TTP, to avoid disclosure of the identities of Alice and Bob. This anonymous

information further reduces the risk of some x party impersonating as either Alice or Bob. Then the signature exchange starts where in the partial signature of Initiator is sent to the responder which in turn is verified and then the responder in turn provides a challenge to Initiator. The initiator fulfills the challenge and then executes the commit wherein the given parties have almost agreed to the contract and then the signature of Bob is delivered to the initiator.

The signature generated using LUC method is extremely strong enough to satisfy the property *provably secure*.

5.3 Dispute Resolution Protocol

In case the initiator does not reply after getting the signature from the Responder, then the signature can be obtained from TTP .The TTP ensures that the contract has been signed properly without any intermediate results coming out. TTP takes care of any disputes in order to ensure that he satisfies the property of *Zero-knowledge*, that apart from the parties nothing is leaked to any other party, to achieve *optimism* , *confidentiality* , *abuse –free* and *completeness*.

6. Conclusion

In this paper, based on the LUC signature scheme, we proposed a new digital contract signing protocol that allows two potentially mistrusted parties to exchange their digital signatures on a contract in an efficient and secure way. Like the existing RSA-based solutions, the new protocol is fair and optimistic, i.e., two parties get or do not get the other's digital signature simultaneously, and the trusted third party is only needed in abnormal cases that occur occasionally. However, different from all previous RSA-based contract signing protocol, the proposed protocol is further abuse-free. That is, if the contract signing protocol is executed unsuccessfully, each of the two parties cannot show the validity of intermediate results generated by the other party to outsiders. In other words, each party cannot convince an outsider to accept the partial commitments coming from the other party. This is an important security property for contract signing, especially in the situations where partial commitments to a contract may be beneficial to a dishonest party or an outsider.

7. References

[1] Debajyothi Konar and Chandan Mazumdar, "A Novel Fair GSR Contract Signing Protocol against Earnest Money", IJCSNS, VOL.7 No.11, November 2007, pages 55-64.

[2] Ole Kasper Olsen ,Ole Martin Dahl ,Torkjel Søndrol, Fredrik Skarderud , "Contract Signing Using PGP", Gjøvik University College, NISlab December 17, 2004

[3] R.H. Deng, L. Gong, A.A. Lazar, W. Wang, "Practical protocols for certified electronic mail", Journal of Network and System Management, Vol. 4 (3), (1996).

[4] Luan Haiying, Lin Wentao, " Simultaneous Contract Signing", M.Sc. in Computer Security and Forensic Computing Dublin City University.

[5] Jianying Zhou and Dieter Gollmann. "A Fair Nonrepudiation Protocol". Proceedings of 1996 IEEESymposium on Security and Privacy, pages 55--61,Oakland, USA, May 1996.

[6] Gulin Wan, " An abuse Fair Contract Signing Protocol Based on the RSA Signature". Infocomm Security Department Institute for Infocomm Research (I2R) 21 Heng Mui Keng Terrace, Singapore 119613

[7] M.K. Franklin, M.K. Reiter, “Fair exchange with a semi-trusted third party”, Proceedings of the 4th ACM Conference on Computer and Communications Security, Zurich, Switzerland, Association for Computing Machinery, New York, 1997 (April), pp. 1-6.

[8] N. Asokan, Victor Shoup, and Michael Waidner. "Asynchronous Protocols for Optimistic Fair Exchange". Proceedings of 1998 IEEE Symposium onSecurity and Privacy, pages 86--99, Oakland, USA,May 1998.

[9] J.A. Garay, Mjackobsson and P. MacKenzie, “Abusefree Optimistic Contract Signing”, Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology, Springer-Verlag, 1999, pp.449-466.

[10] Indrakshi Ray and Indrajit Ray. “An Anonymous Fair-Exchange E-Commerce Protocol.” Proceedings of the First International Workshop on Internet Computing and E-Commerce, San Francisco, CA, April, 2001.

[11] Indrajit Ray, Indrakshi Ray, and N. Natarajan. "An Anonymous and Faliure Resilient Fair-exchange Ecommerce Protocol". Decision Support Systems, 39(2005):267-292, 2005.

[12] Yusuke Okada, Y. Manabe and T. Okamoto."Optimistic Fair Exchange Protocol for E-Commerce".Proceedings of Symposium on Cryptographic and Information Security, SCIS 2006 , Hiroshima, Japan, January 17-20, 2006.

Authors

Dr Avula Damodaram obtained his B.Tech. Degree in CSE in 1989, M.Tech. in CSE in 1995 and Ph.D in Computer Science in 2000 all from JNTUH, Hyderabad. His areas of interest are Computer Networks, Software Engineering, Data Mining and Image Processing. He has successfully guided 6 Ph.D. and 2 MS Scholars apart from myriad M.Tech projects. He is currently guiding 9 scholars for Ph.D and 1 scholar for MS. He is on the editorial board of 2 International Journals and a number of Course materials. He has organized as many as 30 Workshops, Short Term Courses and other Refresher and Orientation programmes. He has published 35 well researched papers in national and International journals. He has also presented 45 papers at different National and International conferences. On the basis of his scholarly achievements and other multifarious services, He was honored with the award of DISTINGUISHED ACADAMICIAN by Pentagram Research Centre, India, in January 2010.

H.Jayasree obtained her B.E. in CSE from Bangalore University and M.Tech. in CSE from JNTUH, Hyderabad in 2001 and 2006 respectively. She is currently a Research Scholar of CSE JNTUH, Hyderabad. She is working as Associate Professor, for Aurora’s Technological and Research Institute and has 10yrs of teaching experience in various colleges of Hyderabad and Bangalore. Areas of research interest include Computer Networks and Network Security.