International Journal of Research in Advent Technology (IJRAT) Special Issue
E-ISSN: 2321-9637
Available online at www.ijrat.org
National Conference on "Role of Information Technology in Social Innovations"
26th & 27th February 2019

An Online Signature Exchange Protocol for Contract Signing Using Dynamic Signature

Rajasree RS
Assistant Professor
Department of Information Technology D.Y.Patil College of Engineering Pune, Maharashtra, India
rajasreecse@gmail.com

Abstract-Security administrations end up significant to numerous applications, for example, web based business installment conventions, Online contract marking. The Protocol is reasonable as it enables two clients to trade their computerized marks in a verified way with the end goal that both the clients stay faithful to the exchange. Convention isn't not out of the question, likewise idealistic, since the believed outsider is included just in the circumstances where one gathering is deceiving or the correspondence channel is interfered. Additionally, if the convention is executed fruitlessly, none of the two parties can demonstrate the legitimacy of middle outcomes to other people. Moreover, both will have a similar measure of data about one another amid a specific session. As more business is directed over the Internet, the reasonable trade issue is increasing more noteworthy significance

Index Terms-Fair-exchange protocols, TTP, digital signatures, security

1. INTRODUCTION

Contracts play an important role in many business transactions. Traditionally, paper-based contracts were signed by the transacting parties who need to be present at the same venue and at the same time. Each party signs a copy of the contract for every contracting party so that every party has a copy of the signed contract.
Along with this one contract copy needs to be submitted to the legal authority. If the parties, however, are not able to meet to sign the paper-based contract, then the transaction delays which may cause time as well as financial loss.
An alternative found to the above problem came up with rising usage of internet and every field going online. An electronic contract is an alternative. Electronic contracts deals with active usage of digital signatures as a token for authentication of users. A digital signature is a piece of information that is sent along with the message and can be generated only by the sender. Everyone (including the receiver) can verify this digital signature and make sure about the origin of the message. By this way, the sender cannot later repudiate sending the message. Therefore, non-repudiation is achieved by digital signatures[5].

Online Contract signing works in a very efficient way as follows [2]:

Let's say that the contract is to be signed between two users, User1 and User2, We are dealing with the authentication of the users and then exchanging the contract. User1 registers itself with the

Certificate Authority (CA) and then with the Trusted Third Party (TTP). User1 will encrypt the partial signature and then send the encrypted signature to user2. User2 will then verify the encrypted signature and if it is correctly verified, send his signature to User1. If User1 finds that User2's signature is correct then she will send the decryption key to User2 to decrypt her encrypted signature. If User1 fails to send the decryption key, User2 will contact the TTP to recover the decryption key.

A certificate authority (CA) is an authority in a network that issues and manages security credentials and public keys for message encryption. As part of a public key infrastructure, a CA checks with a registration authority to verify information provided by the requestor of a digital certificate. If the registration authority verifies the requestor's information, the CA can then issue a certificate. Depending on the public key infrastructure implementation, the certificate includes the owner's public key, the expiration date of the certificate, the owner's name, and other information about the public key owner [5].

.

2. LITERATURE SURVEY

Compared to a protocol using an on-line third party, the optimistic approach greatly reduces the load on the third party, which in turn reduces the cost and insecurity involved in replicating the service in order to maintain availability

A. Verifiable Escrows Based Protocol

The verifiable escrows based protocol is a fair protocol that allows two users to exchange digital signatures so that either each user gets the other's signature, or neither user does[3] . This protocol ensures timely termination for fair exchange. A trusted third party is needed only in cases where one user crashes or attempts to cheat. Here the trusted third party is used as an "escrow service". The basic idea is that Alice, the initiator, encrypts her signature under the public key of the trusted third party. So Bob, the responder, can have

it decrypted by the trusted third party. Together with this escrow scheme a standard "cut-and-choose" interactive proof is used which make it verifiable. In the sense that the user who receives this escrow can verify that it is indeed the escrow of a signature of the desired form with a correct condition attached. This protocol makes use of three sub-protocols: an abort protocol for the initiator, a resolve protocol for the receiver, and a resolve protocol for the initiator. The protocol can also be used to encrypt data for maintaining data integrity while it is exchanged through the internet.

B. *Park et al.'s RSA-Based Multisignature Protocol*

For e-commerce applications the fair exchange must be assured. In this protocol a method of constructing an efficient fair-exchange protocol by distributing the computation of RSA signatures is described. By using the features of multisignature model [2], the protocol is constructed that require no zero-knowledge proofs in the exchange protocol, so the computation can be reduced. Only in the protocol setup phase, the use of zero knowledge proofs are needed. In this approach fairness is ensured by splitting an RSA private key into two parts. The signer holds both parts while the TTP holds just one of the parts [10].

C. *Generic Fair Non-Repudiation Protocols with Transparent Off-Line TTP*

In non-repudiation service evidences need to be generated, exchanged, and validated via
computer networks. After the completion of such a transaction, each involved party should obtain the expected items. If any dishonest party denies his/her participation in a specific transaction, others can refute such a claim by providing electronic evidences to a judge. This non-repudiation protocol is a generic fair protocol with transparent off-line TTP [10] using the same principle. At the end of this protocol execution, either both parties obtain their expected items or neither party does, hence it is said to be fair.

D. *Bao et al's Fair Contract Signing Protocol*

In contract signing protocol, two mutually distrusted parties exchange their commitments to a contract in a
fair way such that either each of them can obtain the other's commitment, or neither of them does. A practical and efficient approach for fair contract signing is using an invisible trusted third party. This contract signing protocol preserves fairness while remaining optimistic in the sense that the trusted party need not be involved in the protocol unless a dispute occurs. The protocol is a generic scheme since any secure digital signature scheme and most of secure encryption algorithms can be used to implement it.

E. *An Abuse-free Fair Contract Signing Protocol Based on the RSA Signature*

. A fair contract signing protocol [9], [5] allows two mistrusted parties to exchange their digital signatures to an agreed contract. Here for achieving fairness the private key of the initiator is split into two parts and the TTP hold one part which is kept secret. The initiator holds both parts of the private key. This digital contract signing protocol is based on the RSA signature and it is optimistic since the trusted third party is involved only in the situations where one party is cheating or the communication channel is interrupted. Furthermore, if the protocol is executed unsuccessfully, none of the two parties can show the validity of intermediate results to others.

3. ALGORITHMS

A .RSA (Rivest, Shamir &Adleman)

RSA algorithm [7] is a cryptosystem for public key encryption, and is widely used for securing sensitive data particularly when being sent over an insecure network such as internet.
Algorithm:

1. Enter p and q such that p and q are co-primes to each other.
2. Calculate n such that n=p*q
3. Calculate value of z which is totient of n, z= (p-1)*(q-1)
4. Calculate e such that, 1<e<z,
5. Calculate d such that,

   (d *e)mod z=1
6. Public key is (e ,n)
7. Private key is (d, n).

B . Hash Algorithm

Hash algorithm is an algorithm which is used to compute a data fingerprint of a data block. It is a one-way function which satisfies the following conditions:

1) Can receive data with any length;
2) Can produce abstract with fixed length;
3) Can compute abstract easily;
4) Cannot compute message from abstract;
5) It is impossible to find two different messages which have same abstract.

Hash function can make short abstract with fixed length for the binary data with any length. The popular hash algorithms are MD5, Secure Hash Algorithm (SHA, having all kinds of security level.).

C. SHA(Secure Hash Algorithm)

SHA encryption is a series of five various cryptographic functions and this presently has three generations: SHA-1, SHA-2, and SHA-3.

The first SHA generation is SHA-1 and it is the fundamental 160-bit hash function. SHA-1 appears similar to the former algorithm MD5

D. MD5(Message Digest)algorithm

Algorithmic steps:

1. *Append padding bits*

The input message is "padded" (extended) so that its length (in bits) equals to 448 mod 512. Padding is always performed, even if the length of the message is already 448 mod 512. Padding is performed as follows: a single "1" bit is appended to the message, and then "0" bits are appended so that the length in bits of the padded message becomes congruent to 448 mod 512. At least one bit and at most 512 bits are appended

2. *Append length*

A 64-bit representation of the length of the message is appended to the result of step1. If the length of the message is greater than 2^64, only the low-order 64 bits will be used. The resulting message has a length that is an exact multiple of 512 bits. The input message will have a length that is an exact multiple of 16 (32-bit) words.

3. *Initialize MD buffer*

A four-word buffer (A, B, C, D) is used to compute the message digest. Each of A, B, C, D is a 32-bit register. These registers are initialized to the following values in hexadecimal, low-order bytes first:
word A: 01 23 45 67

word B: 89 ab cd ef

word C: fe dc ba 98

word D: 76 54 32 10

4. *Process message in 16-word blocks*
Four functions will be defined such that each function takes an input of three 32-bit words and produces a 32-bit word output.

F (X, Y, Z) = XY or not (X) Z
G (X, Y, Z) = XZ or Y not (Z)
H (X, Y, Z) = X xor Y xor Z
I (X, Y, Z) = Y xor (X or not (Z))

4. COMPARISON TABLES OF DIFFERENT ALGORITHMS

The table [8] shows the comparison of basic features, security, and efficiency between the protocols. In the category of basic features, the properties such as transparent TTP [10] or not, off-line or on-line TTP are considered.
Here two main security requirements are compared: fairness and timeliness In the efficiency evaluation; the costs of communication is compared. Various types of TTP can be considered according to their involvement in the protocol. Online TTP - A TTP involved during each session of the protocol but not during each message's transmission, is said to be online [6]

Parameters	Verifiable Escrows Based Protocol	Park et al.'s RSA-Based Multisignature Protocol	Generic Fair Non-Repudiation Protocols	Bao et al.'s Fair Contract Signing Protocol	An Abuse-free Fair Contract Signing Protocol
Fairness	Yes	Yes	Yes	Yes	Yes
Timeliness	Yes	Yes	Yes	Yes	Yes
Transparent TTP	No	Yes	Yes	Yes	Yes
No. of messages	4	3	3	3	7
TTP's statelessness	No	Yes	No	No	Yes

5. PROPOSED METHOD

Signature Generation

The signature generation algorithm takes as input the secret key d[i], generated as a result of key generation protocol and a message m and outputs a signature σ.
In our proposed protocol we employ RSA algorithm for signature generation. The signature σ is generated by

$$\sigma [i] = M^{d[i]} \bmod n \quad (1)$$

Thus the signature generated for each user in the group

Signature Verification

The legitimacy of the made mark by every client is confirmed utilizing the check algorithm. The signature confirmation calculation takes as input the open key e[i] message m and the signature σ.

$$\text{If } \sigma^{d[i]} \bmod \Phi(n) \quad (2)$$

at the point when determined for every client gives indistinguishable result from that of the message then the clients are substantial.

The most grounded purpose of this calculation is that some other client who has the information of the open key just as the esteem of Φ(n) can't sign the message .Because the solid factorization property of RSA calculation it isn't feasible for him to compute the estimation of p and q regardless of whether the estimation of Φ(n) is known.

After the confirmation of the marks of gathering individuals, the bunch supervisor expels the marks from the message and joins his own signature. Now the first message is send alongside the mark of the gathering director. This convention likewise gives traceability of the gathering individuals by the gathering director.

At the less than desirable end the mark of the gathering director is confirmed by the collector with the assistance of the key gpk that is shared between the gathering administrator and the collector. Notwithstanding this calculations, dynamic gathering marks offer a convention called Join. This convention is executed between the gather chief and the individual from the gathering.

Join Protocol

The gathering manager keeps up an enrollment list which is at first empty. The join convention is executed for each member of the gathering who wishes to join the group. It takes as info the personality of the client. After execution of this protocol, the aggregate part gets the mystery key for marking and the gathering chief gets some mystery data that is expected to open the signature. A trapdoor responsibility conspire as clarified in [7][8]enhances the quality of the join convention wherein the client is registered by the administrator however he himself does not know the private key of the client. The upper headed for the quantity of clients is additionally characterized by the .gathering administrator

Revoke Protocol

In the event that any part needs to pull back from the group, group supervisor executes the REVOKE method and updates the data in the data table. The renounced part can no longer utilize the mark to sign the documents. Every time at whatever point the gathering supervisor receives the marked message from the gathering individuals the gathering director initially confirms the identity of the client by checking in the enrollment list. If the character of the client turns out to be legitimate then he considers the signature as substantial. A client who is disavowed from the gathering can't utilize his old character to approve himself.

6. FUTURE WORK

In this paper we have proposed a dynamic computerized mark conspire that enables the individuals to join and renounce the group. In this strategy a message can't be send to the recipient specifically by the gathering member. Thus it guarantees security.

Besides it keeps up the detectability of the individuals by the branch manager. The security of the plan is improved by giving diverse private keys to various clients and is troublesome to break in light of the solid factorization problem. Thus the plangive to be progressively secure and effective one

REFERENCES

[1] Abdullah M. Alaraj "Optimizing One Fair Document Exchange Protocol "International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.1, January 2012 DOI : 10.5121/ijnsa. 4101 1 , 2012.

[2] Alptekin Kupcu and Anna Lysyanskaya, "Optimistic Fair Exchange with Multiple Arbiters", Brown University, Providence, RI, USA,2008.

[3] H.Jayasree1 and Dr. A.Damodaram "A Novel Fair Anonymous Contract Signing Protocol for E-Commerce Applications" 2012 International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.5, September 2012.

[4] Alfin Abraham, "An Abuse-Free Optimistic Contract Signing Protocol with Multiple TTPs", IJCA Special Issue on "Computational Science – New Dimensions & Perspectives" NCCSE, 2011.

[5] Guilin Wang "An Abuse-Free Fair Contract-Signing Protocol Based on the RSA Signature", IEEE Transactions On Information Forensics And Security, Vol. 5, No. 1, March 2010.

[6] Alfin Abraham "A Survey on Optimistic Fair Digital Signature Exchange Protocols" , International Journal on Computer Science and Engineering (IJCSE),Feb 2011

[7] K.P. Thooyamani, R. Udayakumar and V. Khanaa "A Novel Ruin Gratis Fair Digital Contract Signing Protocol Based on Rsa Signature",
School of Computing Science, Bharath University, Chennai-73, India

[8] Abdullah M. Alaraj "Simple and Efficient Contract Signing Protocol " (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 3, No. 3, 2012 67

[9] N. Asokan Victor Shoup Michael Waidner "Optimistic Fair Exchange of Digital Signatures", IBM Zfirich Research Laboratory, S~umerstr. 4, 8803 Rfischlikon, Switzerland.

[10] V.SWAPNA KUMARI, C. SREEDHAR
"Efficient and Fair Exchange of Digital Signatures Based on RSA Algorithm", V Swapna Kumari et al ,Int.J.Computer Technology & Applications,Vol 3

[11] Albert Levi and M. Ufuk Çaðlayan
"The Problem of Trusted Third Party in Authentication and Digital Signature Protocols"

[12] Sumit Kumar Pandey1, Umesh Lilhore2 "A Review on Various Contract Signing Protocol" , International Journal of Emerging Technology and Advanced Engineering, Issue 8, August 2014

[13] Rajasree RS,Shailaja V Pede, "An Abuse-Free Optimistic Signature Exchange Protocol Using Block Cipher" 2015 International Conference on Computing Communication Control and Automation