Computers and Electrical Engineering 37 (2011) 169–173

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Contract signature in e-commerce ☆

Lein Harn [a,*], Chu-Hsing Lin [b]

[a] Department of Computer Science Electrical Engineering, University of Missouri-Kansas City, United States
[b] Department of Computer Science, Tunghai University, Taiwan

ARTICLE INFO

Article history:
Available online 19 February 2011

ABSTRACT

In this paper, we propose a notion of contract signature used in e-commerce applications. We propose a contract signature scheme based on the discrete logarithm assumption. The contract signature scheme adopts a digital multi-signature scheme in public-key cryptography to facilitate fair signature exchange over network. This proposed solution allows multiple signers of a contract signature to exchange their partial signatures which are *fully ambiguous* for any third party (i.e., *1 out of* ∞ *ambiguity*) to construct a valid contract signature. In case any signer releases the partial signature to others, the signer does not bind to the contract.

1. Introduction

As we are rapidly heading into an era of computer communications, cryptographic protocols are pressingly needed to facilitate secure communication. Cryptographic digital signatures have been used to provide non-repudiation services, such as signing business contract. One category of multi-party protocols in the future can be the exchange of digital signatures. One of the most important concerns in exchange signatures is the unfair exchange, which occurs when one party obtains the signature of another party without giving out his own.

The notion of fairness is well established in a traditional secret exchange, in which all parties involved obtain secrets simultaneously. However, in network environment, secrets are exchanged over a network that does not provide simultaneously exchange of messages, and, therefore, adequate cryptographic protocols are needed to facilitate secrets exchange and guarantee fairness to all parties involved.

Fair secret exchange protocols go as early as 1980s [1,2]. Earlier protocols are based on gradually exchange their secret bit-by-bit in an interleaving manner. This process continues until both of them have received and released all the bits; but only at the end of the protocol, both parties can discover that the received bits are real secrets and are not gibberish.

In this paper, we propose a fair protocol to let multiple parties to interact with each other over network to construct a *contract signature* (i.e., we will define this term in Section 3). Our scheme adopts the digital multi-signature [3] in public-key cryptography to facilitate fair signature exchange. A contract signature is a special form of digital multi-signature that involves multiple signers. A protocol allows multiple parties to produce and exchange their ambiguous signatures which are *fully ambiguous* (i.e., *1 out of* ∞ *ambiguity*). The combination of these ambiguous signatures forms the contract signature. There is no "*keystone*" (i.e., a secret key used in the *concurrent signature*). In case anyone releases the contract signature to a verifier, all signers bind to the contract signature.

☆ Reviews processed and proposed for publication by Associate Editor Pro. N. Sklavos.

* Corresponding author. Address: University of Missouri-Kansas City, Computer Science Electrical Engineering, 5100 Rockhill Rd., Kansas City, MO 64110, United States. Tel.: +1 816 235 2358; fax: +1 816 235 5159.
E-mail address: harnl@umkc.edu (L. Harn).

0045-7906/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compeleceng.2011.02.001

The paper is organized as follows. Next section gives an overview of related works. We define the contract signature and present a modified ElGamal signature scheme in Section 3. We use this modified signature scheme to construct contract signature. Security proof under the random oracle model of this modified signature scheme is included. An interactive protocol to let multiple signers to construct a contract signature is introduced in Section 4. Security discussion of the protocol is presented in Section 5. We draw a conclusion in Section 6.

2. Related works

A number of protocols have been proposed up to date to achieve fair secrets exchange. Historically, the protocol design has been evolved from two-party approach, without the involvement of any trusted third party (TTP), to the TTP-based approach, where fairness is guaranteed with the involvement of a TTP.

The two-party protocols go as early as 1980s. Protocols [1,2] and, more recently, [4,5] are based on gradually exchange of small parts of the items to ensure that the exchange of the items occurs pseudo-simultaneously. This is achieved by having the parties release their secret items bit-by-bit in an interleaving manner- one party releases a bit of his item (together with the proof of correctness of the bit), and in return, receives a bit of his counterpart's item (and its proof of correctness). This process continues until both of them have received and released all the bits. Obviously, to achieve fairness, both secret items must be of the same length.

A major disadvantage associated with two-party approach is that a large number of exchange rounds are needed to ensure an acceptable level of fairness, thus the overhead of communication is very high. In addition, gradual secret release protocols require that participating parties have approximately equal computational power in order to guarantee fairness. Otherwise, the party with larger computational capabilities can launch a brute-force attack after receiving the first several bits, and work out the remaining bits of his counterpart's secret. Although reasonably convincing in theory, this approach is too impractical for real life applications. Additionally, this kind of protocols provides no guarantees of the quality of secrets exchanged, i.e., parties can fairly exchange bits of their own, but only at the end of the protocol to discover that the received bits are gibberish.

In order to overcome these weaknesses, a TTP is introduced in the protocols to assist the participants with the exchange and to achieve fairness. In early TTP-based protocols, the TTP is on-line, i.e., it mediates every exchange process, even when participating parties are not attempting to cheat [6–9]. Basically, both parties send their items to the TTP, which verifies the correctness of the items and forwards them to the rightful recipients. The TTP is also responsible for generating and storing security sensitive transactional records, making it a focal point of security attacks. In on-line TTP-based approach, a protocol cannot be performed without the TTP, that makes it a potential communicational and performance bottleneck and susceptible to denial-of-service attacks. In addition, as all the exchanged data is exposed to the on-line TTP, it has to be fully and unconditionally trusted. Clearly, it is desirable to design protocols where the involvement of and security/storage requirements placed on the TTP are reduced. In off-line TTP-based protocols [10–19], the TTP is not involved in the protocol run under normal circumstances, i.e., when the participants do not attempt to cheat or are willing to resolve possible disputes themselves. Only when the exchange process fails to complete due to a network failure or a party's misbehavior, the TTP is invoked to assist the exchange finally come to a fair completion. Wang [20] has proposed an abuse-free fair contract-signing protocol based on the RSA signature very recently. In Wang's scheme, the TTP is involved only in the situations where one party is cheating or the communication channel is interrupted.

Recently, a new category of off-line TTP-based contract signing protocols, capable of making the role of the TTP transparent, have been proposed based on a cryptographic primitive called as verifiable and recoverable encrypted signature (VRES) [10–15,17,19]. The special primitive, e.g. VRES, makes the fair exchange protocols more complicate and restricts the employment of these protocols.

Misbehavior penalization can motivate all participants to behave honest so as to deter any party from misbehaving. Zhang et al. [1] propose a fair signature exchange protocol with such property. However, in their protocol, TTP remains off-line during secrets exchange between two parties. TTP will be called upon whenever some party misbehaved. Since TTP cannot determine who the misbehaved party is and the misbehavior penalization can only be applied to one of the parties, the dishonest party can frame the honest one to be punished.

Chen et al. [21] proposed the concept of concurrent signature, CS for short. Such signature scheme allows two parties, without TTP, to produce and exchange two ambiguous signatures which are ambiguous for any third party until an extra piece of information (called keystone) is released by one of the parties. Concurrent signature is very efficient and requires neither a TTP nor a high degree of interaction between parties. Later in this section, we will address two problems associated with the concurrent signature. In order to strength the ambiguity of the signature before keystone is released, there are papers [22,23] proposed a strong notion, called perfect concurrent signatures. Later, asymmetric concurrent signature [24], tripartite concurrent signature [25], are proposed.

2.1. Problems associated with concurrent signature

The keystone is in the hand of only one signature signer (i.e., the initiator). The owner can determine whether to release the keystone or not.