Secure Mobile Agents in Electronic Commerce by Using Undetachable Signatures from Pairings

Yang Shi, Xiaoping Wang, Liming Cao, Jianxin Ren
Department of Computer Science and Technology, Tongji University, Shanghai 200092, China
 cnshiyang@yahoo.com.cn

ABSTRACT
It is expect that mobile agents technology will bring significant benefits to electronic commerce. But security issues, especially threats from malicious hosts, become a great obstacle of widespread deployment of applications in electronic commerce based on mobile agents technology. Undetachable digital signature is a category of digital signatures to secure mobile agents against malicious hosts. An undetachable signature scheme by using encrypted functions from bilinear pairings was proposed in this paper. The security of this scheme base on the computational intractability of discrete logarithm problem and computational Diffie-Hellman problem on gap Diffie-Hellman group. Furthermore, the scheme satisfies all the requirements of a strong non-designated proxy signature i.e. verifiability, strong unforgeability, strong identifiability, strong undeniability and preventions of misuse. An undetachable threshold signature scheme that enable the customer to provide n mobile agents with ‘shares’ of the undetachable signature function is also provided. It is able to provide more reliability than classical undetachable signatures.

Keywords: electronic commerce, mobile agents, undetachable digital signatures, bilinear pairings

1. INTRODUCTION
Mobile agents technology are attracting a great deal of interest from both industry and academia since middle of 1990’s. Compared with traditional computing models, e.g. client/server, mobile agents technology has following advantages[1][2][3]:

- Autonomous mobile agents strive to achieve a given goal without permanent observation by its owner. As a matter of consequence, the user is free to take care of other tasks, saving time in the process.
- If a host is being shut down, all mobile agents executing on that machine are warned and given time to dispatch and continue their operation on another host in the network.
- Users may dispatch mobile agents over a temporary network connection to a target network. After dispatching, the temporary network link may be brought down until a later point in time.

Mobile agents technique brings significant benefits to electronic commerce because of these advantages. But on the other hand, there are also some problems. The most important one is security.

Threats to the security of mobile agents generally fall into four comprehensive classes[4]:

- Agent against agent platform
- Agent platform against agent
- Agent against other agents
- Other entities against agent system

Hohl[5] identified the following attacks: spying out code; spying out data; spying out control flow; manipulation of code; manipulation of data; manipulation of control flow; incorrect execution of code; masquerading of the host; denial of execution; spying out interaction with other agents; manipulation of interaction with other agents; returning wrong results of system calls issued by the agent.

Thus, security issues, especially threatens from potentially malicious hosts become a great obstacle of widespread deployment of applications in electronic commerce based on mobile agents technique.

2. PREVIOUS WORKS ABOUT UNDETACHABLE DIGITAL SIGNATURES
2.1 The preliminary idea
Before 1998, many researchers believed that, on malicious hosts, mobile agents were impossible to prevent tampering unless trusted and tamper-resistant hardware is available. Follow points[6] are considered by them:

- Cleartext data can be read and changed.
- Cleartext programs can be manipulated.
- Cleartext messages can be faked.

But Sander and Tschudin[7] pointed out that this belief is incorrect because mobile agents do not have to be executed in cleartext form. They proposed the idea of undetachable digital signatures that allows a mobile agent to effectively produce a digital signature inside a remote and possibly malicious host without the host being able to deduce the agent’s secret or to reuse the signature routine for arbitrary documents. Here is a brief introduction of the idea.
Let $\text{Sig}$ be a rational function used by $C$ (a customer) to produce the digital signature $\text{Sig}(m)$ of an arbitrary message $m$. Furthermore suppose the message $m$ is the result of a rational function $f$ applied to some input data $x$. Finally the verification function $\text{Ver}$ that $C$ publishes in order to let others check the validity of the digital signature $z$ is regarded to be a valid signature of $m$ if and only if:

$$z = \text{Sig}(m)$$

For letting the customer’s mobile agent create “undetachable” signatures, he computes:

$$f_{\text{Signed}} = \text{Sig} \circ f$$

Then he sends $f_{\text{Signed}}$ and $f$ to $S$ (a shop) with his mobile agent. $S$ evaluates:

$$m = f(x) \quad (3)$$

$$z = f_{\text{Signed}}(x) \quad (4)$$

Though the signature function, $\text{Sig}$, is not known by others, every one can verify the validity of a message $m$ by testing:

$$\text{Ver}(z) = m \quad (5)$$

### 2.2 The first implementation

Although Sander and Tschudin tried to give a outline of undetachable digital signatures by using birational functions based on Shamir’s work[8]. Unfortunately no secure undetachable digital signatures scheme has been proposed until 2000. In 2000 Kotzanikolaou, Burmester and Christopoulous presented an RSA implementation[9] of undetachable digital signatures. But this scheme does not provide server’s non-repudiation because it does not contain server’s signature[10].

### 2.3 Strong proxy signatures and other implementations

In 2001 Lee, Kim and Kim provided an RSA based construction of undetachable digital signatures called "Strong Non-designated Proxy Signature" [10]. Their scheme enhanced [9] and often be acronymized as "LKK-SPS" scheme. A scheme of undetachable threshold signature[13] was proposed by Borselius, Mitchell and Wilson in the same year. In 2002, a strong proxy signature scheme with proxy signer privacy protection[11] was given and a pragmatic alternative to undetachable signatures[12] was also proposed.

### 3. BILINEAR PAIRINGS AND SIGNATURE SCHEMES BASED ON THEM

#### 3.1 Mathematical preliminaries of Bilinear Pairings

Let $G_1$ be a cyclic group generated by $P$, whose order is a prime $q$, and $G_2$ be a cyclic multiplicative group of the same order $q$. The discrete logarithm problems in both $G_1$ and $G_2$ are hard. Let $e : G_1 \times G_1 \rightarrow G_2$ be a pairing satisfies the following conditions:

- **Bilinear**: (6) and (7) or (8)
  
  $$e(P_1 + P_2, Q) = e(P_1, Q)e(P_2, Q) \quad (6)$$
  $$e(P_1Q_1 + Q_2, Q_2) = e(P_1, Q_2)e(P_1, Q_2) \quad (7)$$
  $$e(aP, bQ) = e(P, Q)^{ab} \quad (8)$$

- **Non-degenerate**: There exists $P \in G_1$ and $Q \in G_1$ subject to (9).
  $$e(P, Q) \neq 1 \quad (9)$$

- **Computability**: There is an efficient algorithm to compute $e(P, Q)$ for all $\{P, Q\} \subseteq G_1$.

We note that the Weil and Tate pairings associated with supersingular elliptic curves or abelian varieties can be modified to create such bilinear maps. Suppose that $G$ is an additive group. Four mathematical problems is defined as follow [14][15].

- **Discrete Logarithm Problem (DLP)**: Given two group elements $P$ and $Q$, and an integer $n$, such that (10) is satisfied whenever such an integer exists.
  $$Q = nP \quad (10)$$

- **Decision Diffie-Hellman Problem (DDHP)**: For $(a, b, c) \subseteq Z_q^*$, given $P, aP, bP, cP$ decide whether:
  $$c \equiv ab \pmod{q} \quad (11)$$

- **Computational Diffie-Hellman Problem (CDHP)**: For $(a, b) \subseteq Z_q^*$, $P, aP, bP$ compute $abP$.

- **Gap Diffie-Hellman Problem (GDHP)**: A class of problems where DDHP is easy while CDHP is hard.

We assume through this paper that CDHP and DLP are intractable, which means there is no polynomial time algorithm to solve CDHP or DLP with non-negligible probability. When the DDHP is easy but the CDHP is hard on the group $G_1$, $G_1$ is called a Gap Diffie-Hellman (GDH) group. Such groups can be found on supersingular elliptic curves or hyperelliptic curves over finite fields, and the bilinear pairings can be derived from the Weil or Tate pairing $e : G_1 \times G_1 \rightarrow G_2$. Our schemes of this paper can be built on any GDH group. More mathematical background can be found in [14][16][17][18]. Now, some system parameters should be defined for Sec. 4 as following: Let $P$ be a generator of $G_1$, the bilinear
paring is $e: G_1 \times G_1 \rightarrow G_2$. Moreover, to hash functions are given here: $H_1: \{0,1\}^* \rightarrow \mathbb{Z}_q$ and $H_2: \{0,1\}^* \rightarrow G_1$. The implementation of these hash functions can be referred to works such as [25]. But the choice of conventional hash functions should be very carefully because many hash functions were cracked recently [26].

3.2 A Survey of Signature Schemes Based on Bilinear Pairings


4. UNDETACHABLE SIGNATURES BASED ON BILINEAR PAIRINGS

4.1 Settings

Assume that all participants have the common system parameters: $(G_1, G_2, e, q, P, H_1, H_2)$.

Settings about the customer:

- Let $C$ be an identifier for the Customer.
- Let $s_C$ be the signature key of $C$.
- Let $PU_C$ be the verification key of $C$ where:
  \[ PU_C = s_C P \] (12)
- Let $req\_C$ be the constraints of the Customer.
- Let $T_C$ be the timestamp of $req\_C$.

Settings about shops:

- Let $S$ be an identifier for a shop.
- Let $s_S$ be the signature key of $S$.
- Let $PU_S$ be the verification key of $S$ where:
  \[ PU_S = s_S P \] (13)
- Let $bid\_S$ be the bid information of $S$.
- Let $T_S$ be the timestamp of $bid\_S$.
- Let $\varphi: G_1 \rightarrow \{0,1\}^*$ be a function mapping a point of $G_1$ into a binary string.

4.2 Security scheme

The Customer does following operations:

- Computes:
  \[ H_C = H_2(C \| req\_C \| T_C) \] (14)
- Computes:
  \[ K_C = s_C \cdot H_C \] (15)
- Gives $(C \| req\_C \| T_C)$ to the Agent.
- Gives to the Agent as part of its executable code the undetachable signature function pair:
  \[ f(\cdot) = (\cdot)H_C \] (16)
  and
  \[ f_{\text{Signed}}(\cdot) = (\cdot)K_C \] (17)

Suppose:

\[ \text{Sig}(x) = s_C(\cdot) \] (18)

It turns out the following proposition:

**Proposition 1.** The functions above construct an undetachable digital signature scheme.

**Proof.**

\[
\begin{align*}
\text{Sig} \circ f(\cdot) &= \text{Sig}(f(\cdot)) \\
&= \text{Sig}(f(\cdot)H_C) \\
&= s_C(\cdot)H_C \\
&= (\cdot)s_CH_C \\
&= (\cdot)K_C \\
&= f_{\text{Signed}}(\cdot)
\end{align*}
\]

After the Agent migrated to the host of $S$, the validity of the mobile agent should be verified first by checking:

\[ e(K_C, P) = e(H_C, PU_C) \] (20)

If it is a valid signature of $C$, then $S$ does following operations:

- Computes:
  \[ H_S = H_2(C \| req\_C \| T_C \| S \| bid\_S \| T_S) \] (21)
- Computes:
  \[ Y = s_SH_S \] (22)
Computes:

\[ x = H_1(\phi(Y)) \quad (23) \]

Computes:

\[ z = f_{\text{Signed}}(x) \quad (24) \]

Gives \((x, z, Y, H_S, S, \text{bid}_S, T_S)\) to the mobile agent.

After the mobile agent goes back to \(C\). The correctness of (21) and (23) should be verified first. Then the verification should be performed by formula (25):

\[ e(Y, P)^\gamma = e(H_S, PU_S) \quad (25) \]

### 4.3 Security Analysis

The following proposition can be obtained because \(G_1\) is a GDH group.

**Proposition 2.** A transaction is valid if and only if (20), (21), (23) and (25) are true.

**Proof.** If the mobile Agent is not detached before it migrated to the host of \(S\), then:

\[
e(K_C, P) = e(s_C H_C, P) = e(H_C, P)^{s_C} = e(H_C, s_C P) = e(H_C, PU_C) \tag{26}
\]

If an opponent Oscar want to modify \(C\)'s bid information when \(C\)’s agent mobiles to his host, he has to construct a new undetachable digital signature pair \((H_C, K_C)\) of the Customer which will include modified constraints \(\text{req}_C\)' of the Customer. But this needs to solve the computational difficult problems mentions in section 3.1.

Furthermore, if a transaction is valid:

\[
e(Y, P) = e(s_S H_S, P) = e(H_S, P)^{s_S} = e(H_S, s_S P) = e(H_S, PU_S) \tag{27}
\]

Similar to equation (20), the security of (23) and (25) also relies upon the difficulty of the problems computational infeasible to solve at present.

### 5. UNDETACHABLE THRESHOLD SIGNATURES

#### 5.1 Basic Idea

The notion of undetachable threshold signatures was introduced in [27]. An undetachable threshold signature scheme will enable the customer, \(C\), to provide \(n\) mobile agents with 'shares' of the signature key (where the shares will be a function of \(\text{req}_C\) and \(T_C\)). For more details about undetachable threshold signature such as their usage and application can be found in [27].

### 5.2 Security Scheme

Based on undetachable signature scheme proposed in section 4, a variety of secret sharing scheme can be used to construct undetachable threshold signatures by converting \(s_C\) into signature shares. An example using Lagrange Polynomial Interpolation is given here:

Suppose \(f\) is a polynomial over \(Z_q\) with degree \(t-1\) subject to (28).

\[ s_C = f(0) \quad (28) \]

Operations in equations (29) to (32) and (38) are performed over \(Z_q\).

Let:

\[ s_i = f(i), i = 1, 2, \ldots n \quad (29) \]

Then any one has \(t\) shares can obtain \(s_C\) by calculating (30)

\[
s_C = \sum_{j=1}^{t} S_j \prod_{h=1 \atop h \neq j}^{t} \left( \frac{i_h}{i_h - i_j} \right) \quad (30)
\]

To simplify the description we suppose the \(t\) shares are \(s_1, s_2, \ldots s_t\), thus:

\[
s_C = \sum_{j=1}^{t} (s_j l_j) \quad (31)
\]

Where:

\[
l_j = \prod_{h=1 \atop h \neq j}^{t} \left( \frac{h}{h - j} \right) \quad (32)
\]

Then \(C\) does following extra operations

- Computes:

\[ PC_i = s_i P, i = 1, \ldots, n \quad (33) \]

and (34) instead of (14).

\[ H_C = H_2(C \parallel \text{req}_C \parallel T_C \parallel PC_1 \parallel \cdots \parallel PC_n) \quad (34) \]

- Computes:

\[ K_i = s_i \cdot H_C, i = 1, \ldots, n \quad (35) \]
Finally, $C$ gives (36) to the $i$-th agent as part of its executable code instead of (17):

$$f_{\text{Signed},i}(\cdot) = (\cdot)K_i$$  \hspace{1cm} (36)

After the $i$-th Agent migrated to the host of $S_i$, the validity of the mobile agent should be verified first the by checking:

$$e(K_i, P) = e(H_i, PC_i)$$  \hspace{1cm} (37)

It is clear that a shop has $t$ or more than $t$ shares can reconstruct $f_{\text{Signed}}$ by (38). But anyone cannot reconstruct $f_{\text{Signed}}$ from less than $t$ shares.

$$f_{\text{Signed}}(\cdot) = \sum_{i=1}^{t} [i f_{\text{Signed},i}(\cdot)]$$  \hspace{1cm} (38)

After a shop has reconstruct $f_{\text{Signed}}$ successfully, other operations are similar to undetachable signatures proposed in section 4. So redundant words are omitted.

6. CONCLUSION

In this paper, we have presented a novel implementation of undetachable digital signatures and a correspondent security scheme. Compared to [9][10], our scheme uses a different cryptosystem to construct undetachable signatures. This implementation of undetachable digital signatures is based on non-interactive CEF (Computing with Encrypted Functions) from bilinear pairings to protect the original signature function $\text{Sig}$ by encrypting it with a function $f$ to obtain the encrypted function $f_{\text{Signed}}$ defined as the composition of $\text{Sig}$ and $f$. Furthermore, the scheme satisfies all the five requirements of a strong non-designated proxy signature proposed by Lee et al. [10] as follow: First, verifiability: A proxy signer can create a valid proxy signature for the original signer. But the original signer and any third party cannot create a valid proxy signature with the name of proxy signer. Second, strong unforgeability: Anyone can determine the identity of the corresponding proxy signer form a proxy signature. Third, strong identifiability: Once a proxy signer creates a valid proxy signature on behalf of an original signer, the proxy signer cannot repudiate his signature creation against anyone. Fourth, strong undeniability: It should be confident that proxy key pair cannot be used for other purpose. In the case of misuse, the responsibility of proxy signer should be determined explicitly. Finally, prevention of misuse: From a proxy signature a verifier can be convinced of the original signer's agreement on the signed message.

As to the undetachable threshold signature scheme proposed in the paper, it has following features: First, each agent can use their share to sign a message $M$, e.g. a piece of bid information, of their choice to obtain a ‘signature share’. Second, The ‘correctness’ of a signature share can be verified independently of any other signature shares. Third, any shop, when equipped with $t$ different signature shares restricted with the same request of $C$ for the same message $M$, can construct a signature on $M$ which will be verifiable by any party with a trusted copy of public key of $C$, and which will also enable the corresponding request $\text{req}_C$ and $T_C$ to be verified. Finally, knowledge of less than $t$ different signature shares for the same message $M$ cannot be used to construct a valid signature on $M$ and knowledge of any number of different signature shares for messages other than the message $M$ will not enable the construction of a valid signature with associated request $\text{req}_C$ or with a different time stamp will not enable the construction of a valid signature with associated request $\text{req}_C$ and $T_C$. So our scheme satisfied all the requirements of undetachable threshold signatures defined in [27] and provide more reliability than classical undetachable signatures.

ACKNOWLEDGEMENT

This work is supported by the National Science Foundation of China under grant 70171061.

REFERENCES