

論文98-35T-9-14

# 상호 관계성을 가진 간접 인증 방법을 이용한 전자 결제 시스템 (Electronic Settlement System Model Using an Indirect Authentication Method by Interrelationship)

朴 泳 鎬 \* , 李 金 石 \*\*

(Young-Ho Park and Keum-Suk Lee)

## 요 약

기존 전자상거래에서는 양단(End-to-End)간의 인증만이 존재한다. 그러나 분산환경에서 사람이나 컴퓨터의 신분을 확인하는 인증은 위조와 사칭으로 인하여 그들의 신분이 고정적이지 않다는 점 때문에 상당히 힘들다. 또한 고객과 판매인을 연결시켜주는 다수의 중개인 (Broker)이 포함된 경우에는 제 3자에 대한 인증이 반드시 필요하다. 이런 인증은 기존 방법으로는 참여하는 구성원이 많아질수록 복잡해지는 한계점이 있다. 본 논문에서는 이를 해결하기 위하여 상호 간접 인증을 이용하며, 또한 이동에이전트를 이용하여 상호 간접 인증에 의해 복잡한 환경에서의 상거래가 성립되는 모델을 제시한다.

## Abstract

There is only authentication of end-to-end in electronic commerce technique. But in a distributed system, an authentication that is ensured users and computers has a big threats which an identity of those is unbelievable because it can be occurred an fabrication or a false personation. Authentication for other elements included customers and vendors is required in the case that there is many a broker connecting customers with vendors as well. In authentication like above, many elements playing different role participated, were performed by the known techniques, processing was more complicated. In this paper, the Mutual Indirect Authentication technique is used to solve and avoid that condition and the generalized electronic commerce using a mobile agent is accomplished with this technique.

## I. 서 론

지금까지 전자 결제 시스템은 요금 지불 등의 제한적인 범위에서만 사용되고 있다. 이러한 전자 결제 시스템에서는 사용자의 인증을 통해 특정 사용자의 정보가 허가 없이 노출되지 않도록 하기 위해 여러 방법들

이 제시되었는데 그 중 가장 대표적인 것이 암호화이다. 메시지를 암호화하는 기법으로는 RSA, PKCS, DES 및 최근 RSA사에서 56-Bit를 사용하는 DES 대신 128-Bit를 사용하는 AES와 같은 기법을 사용한다. 이런 암호화 기법은 양단간의 통신 상황에서만 사용하는 것이 일반적이다. 그러나 양단간의 통신만으로는 경매에서와 같은 다수의 상품 공급자들 중에서 좋은 조건을 갖는 하나를 선택하기에는 상당히 어려운 점이 있다. 또한 암호화를 통한 인증 방법은 암호화하려는 내용에 비례하여 오버헤드가 발생하므로 시간 제약(Time Constraint)이 있는 환경으로의 확장이 힘들다.

실세계에서 높은 시간 제약을 만족해야 하는 경우로

\* 正會員, 富川大學 事務自動化科

(Dept. of Office Automation, Bucheon Technical College)

\*\* 正會員, 東國大學校 컴퓨터工學科

(Dept. of Computer Engineering, Dongguk University)

接受日字: 1998年8月19日, 수정완료일: 1998年9月23日

는 경매시장, 외환시장, 주식시장에서의 거래가 있으며 낮은 시간 제약을 만족해야 하는 경우는 잠지 구독 신청, 하루 단위의 자금 이체 등이 있다.

본 논문에서는 이동 에이전트를 이용하여 사용자와 사용자의 작업을 수행하는 기계사이의 상호인증(Mutual Authentication)이 만족됨과 동시에 상이한 시간 제약이 있는 시스템으로의 확장 가능한 전자 결제 시스템 모델을 제안하려 한다.

## II. 기존의 전자 결제 시스템

기존 전자 결제 시스템에서는 일반적으로 암호화를 이용한 상호 인증 방법을 이용하는데 이러한 시스템에서 사용되는 암호화 기법은 대체로 공유 키, 공유 암호, 공개키를 사용하고 있다. 이런 암호화 방법은 전달하려는 전체를 하나의 방법으로 암호화하는 것이 아니라 메시지 부분들을 암호화를 적용해 사용하는 것이 일반적이는데 그 방법으로 소개된 것으로는 Otway-Rees, Needham-Schröder<sup>[1,2]</sup>, Kerberos<sup>[3]</sup>, Yahalom, CCITT X.509, Wide-mouthed-frog 프로토콜이 있으며<sup>[4]</sup> 이들 방법과는 달리 Millicent<sup>[5]</sup>는 메시지를 부분적인 해싱과 해싱된 부분을 재 암호화하여 전달하는 방법을 사용하고 있다.

이중에서 Kerberos 프로토콜, Needham-Schröder 프로토콜, CCITT X.509 프로토콜은 실제 상업적인 사용에 문제가 없다고 알려지고 있다. 특히 현재 전자 상거래에서 사용되는 시스템은 이 프로토콜을 이용하는 것이 대부분으로 Kerberos 프로토콜은 NetBill로 성장하였고, DigiCash와 NetCash, NetCheque는 PKCS를 이용하거나 RSA를 이용한다.

이들 전자 결제 시스템에서는 메시지를 암호화하여 전송할 때, 메시지가 작동할 시간(Lifetime)을 지정하여 메시지를 받고 전달할 때 발생할 수 있는 교착상태(Deadlock)를 방지하고 메시지의 유효성 검사(validity check)를 할 수 있도록 도와주는 기능을 가지고 있다.

그러나 이 방법들은 경매, 주식 거래, 환 거래와 같이 다양한 서비스를 지원하기에는 부족한 점이 많다. Telescript와 같이 에이전트를 이용한 시스템들은 RSA<sup>[6]</sup>나 앞서 소개한 방법을 함께 이용하여 메시지를 암호화하고, 암호화된 메시지를 에이전트가 전달하여 여러 프로토콜에서 발생가능성이 지적된 교착상

태나 유효성 문제를 발생시키지 않으면서 사용자의 거래를 돕는 방법이다.

## III. 상호 간접 인증을 통한 전자결제시스템

### 1. 에이전트의 분류

본 논문에서는 전자 결제 시스템을 구축하기 위해 이동 에이전트(Mobile Agent)를 이용한다. 최근에 에이전트 기술이 광범위하게 사용되면서 많은 종류의 에이전트가 만들어지고 있으며, 에이전트 에이전트의 속성에 대한 다양한 정의가 존재한다<sup>[7,8]</sup>. 에이전트는 기존 클라이언트/서버 구조나 RPC처럼 복잡한 기능을 모두 구현하여 수행하는 것이 아니라 복잡한 기능중 일부분을 수행할 수 있도록 만든 작은 프로세스라고 할 수 있다<sup>[9]</sup>. 따라서 에이전트는 다음 성격 중에 반드시 하나 이상을 가지게 된다<sup>[10]</sup>.

한편 최근의 에이전트 개발 방향이 이동성과 사교성을 강조하는데 이동성이 특히 강조되는 에이전트를 이동 에이전트라고 하는데 본 논문에서는 네트워크 기반 에이전트이면서 소프트웨어 에이전트인 이동 에이전트를 사용한다.

표 1. 에이전트의 특성

Table 1. Characteristic of agent.

성격	설명
자율성 (autonomous)	사용자나 다른 소프트웨어의 도움 없이도 스스로 판단하여 행동할 수 있는 능력을 말한다.
지능 (intelligence)	지식베이스(Knowledge-Base)와 추론 능력을 갖고서 사용자의 의도를 파악하는 능력으로 계획과 학습하는 성질을 말한다.
이동성 (mobility)	사용자가 요구한 작업을 다른 컴퓨터에서 수행하는 것을 말한다.
사교성 (sociality)	네트워크를 통해 이동하는 에이전트들간의 통신 능력을 의미한다.(혹은 Communicativeness/Collaboration)

### 2. 상호 간접 인증

본 논문에서 제안하는 전자 결제 모델에서는 상호 간접 인증 방법을 이용한다.

상호 간접 인증이란 양단간의 인증이 아닌 3자 관계에서의 인증을 의미한다[그림-1]. 예를 들어, A는 B라는 친구가 있고 B는 C라는 친구가 있을 때, 그것을 friendship이라는 메소드를 이용해 다음과 같

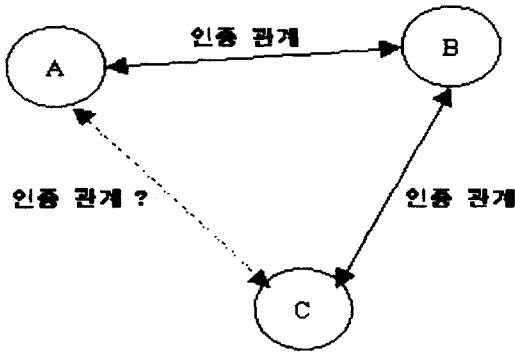


그림 1. 상호 인증 관계  
Fig. 1. Mutual Authentication Relationship.

이 나타낼 수 있다.

A.friendship(B) and B.friendship(A) =>

A <-friend-> B

B.friendship(C) and C.friendship(B) =>

B <-friend-> C

따라서 위의 관계를 정리하면

[ A <-friend-> B ] and [ B <-friend-> C ]

=> [ A <-friend-> B <-friend-> C ]

이 때, A와 C는 서로 친구일 수도 있으며 아닐 수도 있다. 또 서로 알지만 친구로는 여기지는 않을 수 있다. 따라서 아래의 표기는 반드시 성립되지 않는다.

[ A <-friend-> C ] (false)

이를 확장해서 전자 결제 시스템에서 A가 인증하는 B가 있고 B가 인증하는 C가 있을 때, B가 C를 인증한다고 해서 반드시 A가 C를 혹은 C가 A를 인증할 수는 없으므로 상호 간접 인증이 필요하다. 즉, A가 C를 인증하기 위해서는 어떤 조건을 C가 만족해야 하는데 그 조건을 간접 인증 조건이라 한다.

3. 간접 인증 조건을 만족하는 전자 결제 시스템

물건을 판매하는 판매인을 Vendor, 고객을 Customer라 하고 Customer와 Vendor를 엮어주는 중개인 Broker가 존재하는 시스템을 고려해보자.

이 시스템에서는 구매자와 중개인, 중개인과 판매인 사이의 상호인증과 고객과 판매인의 간접 인증이 성립

되는 것을 살펴보자. 그림 2에 보이는 모델에서는 고객의 에이전트만 작동한다고 가정한다.

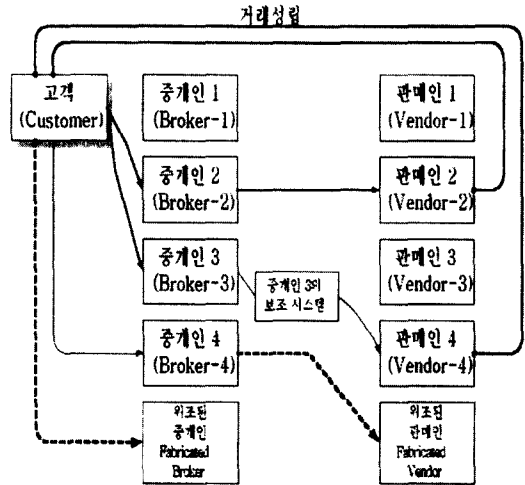


그림 2. 시스템 작동  
Fig. 2. System activity.

그림 2를 보면 고객의 에이전트는 중개인을 통해 판매인 정보를 얻는다. 이 정보로 에이전트는 판매인으로 이동하게 되며, 중개인이 알려준 판매인에 도달하여 중개인이 제공한 가격과 품목이 맞는지 확인하게 된다. 일단 가격이 맞으면 고객의 에이전트는 판매인에게 고객이 소속되어있는 은행을 알려주고 고객에게로 되돌아와 거래가 성립되었음을 알린다. 이 때 고객이 승인 메시지를 보내게 되면 이 메시지는 해당 은행으로 전달한다. 은행은 고객의 승인 메시지에 나타나 있는 같은 판매인의 요청이 들어올 때까지 기다린 다음, 요청 메시지가 도착하면 고객의 가격과 판매인의 가격이 같은지 확인한다. 같은 가격이면 대금이 지불되고 거래가 성립된다. 다음의 그림 3과 그림 4는 고객 에이전트가 중개인과 판매인을 경유하는 동안의 작업 내용을 보여주고 있다.

고객 에이전트(CA)가 중개인 서버에 도달하면 Customer\_ID#, Bank\_ID<sub>Customer</sub>#, PriceList를 전달한다. 중개인 서버는 자신이 인증한 고객이 맞는지 확인한 다음, 고객 에이전트에 Message<sub>Broker</sub> (Broker\_ID#, Bank\_ID<sub>Broker</sub>#, Vendor\_ID#)를 전달한다. 고객의 에이전트는 중개인이 제공한 정보 중에 Vendor\_ID#에 해당하는 판매인으로 이동한다. 이 때 Message<sub>Broker</sub>에는 중개인 서버가 메시지의 시간 정보를 함께 보내준다.

고객 에이전트(CA)가 판매인 서버에 도달하면 Customer\_ID#, Bank\_ID#, Message<sub>Broker</sub>, PriceList를 판매인 서버에 전달한다. 판매인 서버는 고객이 원하는 물품과 가격 정보를 담은 메시지 Message<sub>Vendor</sub>(Vendor\_ID#, Bank\_ID<sub>Vendor</sub>#, Price-List)를 전달한다. 이때, Message<sub>Broker</sub>와 마찬가지로 Message<sub>Vendor</sub>에는 판매인 서버가 메시지의 시간 정보를 함께 보내 준다. 판매인 서버가 제공한 정보(Message<sub>Vendor</sub>)를 고객에게 전달함으로써 거래가 성립한다.

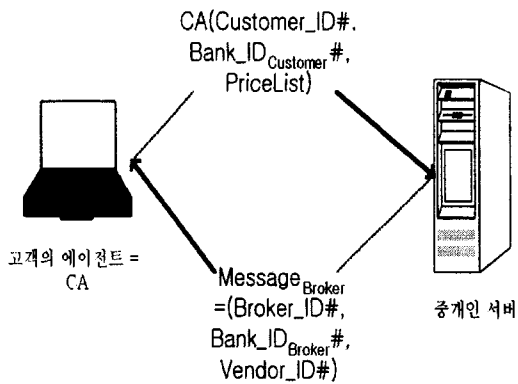


그림 3. 고객에이전트와 중개인 서버와의 관계  
Fig. 3. Relationship between customer agent and broker server.

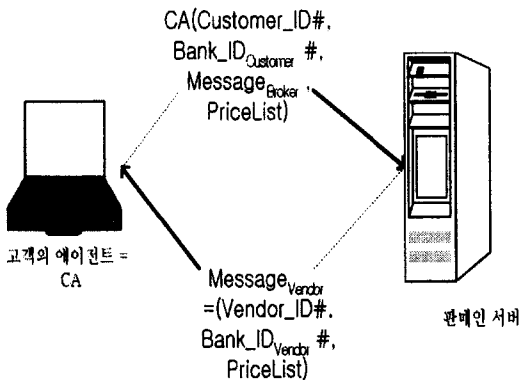


그림 4. 고객에이전트와 판매인 서버와의 관계  
Fig. 4. Relationship between customer agent and vendor server.

그림 5는 그림 2에 은행과 은행 연합이라는 새로운 구성 요소를 추가시킨 것으로서 각각의 구성원들의 작동을 보여주고 있다. 은행 연합을 포함한 구성 요소들은 금융기관에 자신의 신분이 증명되어 있어야 하는데

각 구성원들의 제약 사항과 작동 내용은 다음과 같다.

#### 1) 고객

고객은 은행이나 은행 연합에 의해 신분이 등록되어 있어야 한다. 고객 에이전트가 판매인을 경유하여 돌아왔을 때, 변경된 가격정보를 확인하여 그 물건을 구매한다면 자신의 신분을 인증한 은행에 지급을 의뢰한다. 이때 해당 판매인이 제공한 시간 정보도 함께 전달한다.

#### 2) 중개인

중개인은 신원을 부여받기 위해서는 고객의 신분이 등록되어있는 은행과 은행 연합이 인증하는 지불 기관에 등록되어야 한다. 고객 에이전트가 도달하여 가격정보(PriceList)를 전달하면 자신의 신분을 증명하는 Bank\_ID<sub>Broker</sub>#와 판매인을 연결시켜주고, Bank\_ID<sub>Customer</sub>#로 수수료(Price<sub>Broker</sub>\$)를 청구한다.

#### 3) 판매인

판매인은 신원을 부여받기 위해서는 고객의 신분이 등록되어있는 은행과 은행 연합이 인증하는 지불 기관에 등록되어야 한다. 고객 에이전트와 거래를 시작하면 시간정보를 기록하고 작업이 완료되면 고객 에이전트에게 시간 정보(TimeStamp<sub>Vendor</sub>\$)를 제공해야 한다.

#### 4) 은행

은행은 고객의 신분과 연합 은행의 상호간의 신분을 인증해야 한다. 고객이 지불을 요청하면 중개인과 판매인이 소속된 은행을 은행 연합에서 확인한 다음 이들 은행에 Broker\_ID#와 Vendor\_ID#를 갖는 고객이 있는지 확인을 요청한다. 고객이 소속된 은행의 확인 요청이 오면 해당은행은 지불 대상인 판매인과 중개인이 고객 에이전트와 거래를 하였는지를 Message<sub>Broker</sub>와 Message<sub>Vendor</sub>의 시간 정보를 이용하여 확인한다. 이 확인에서 문제가 없다는 통보가 전달되면 고객이 소속된 은행은 이들 은행으로 지급을 전송한다. 지급이 전송되면 판매인과 중개인의 계좌로 이체한다. 고객 신분의 확인 요청에 대해서는 존재 여부만을 확인하여 요청한 은행에 통보한다.

#### 5) 은행 연합

은행 연합은 이 시스템에 존재하는 금융기관에 의해 인증 받은 상태로 간주한다.

6) 고객 에이전트

고객 에이전트는 자신이 특정 고객이 발행한 에이전트임을 밝히는 내용을 가지고 있어야 하며 고객이 아닌 경우에도 이 정보를 제공할 수 있다.

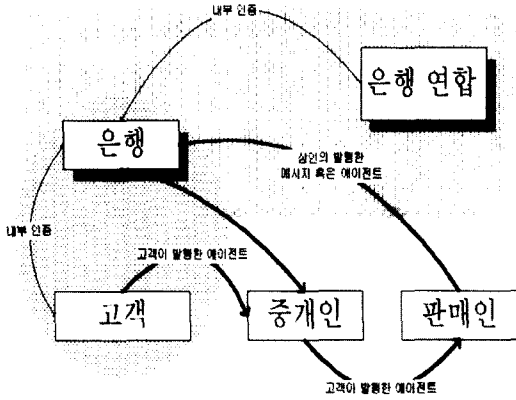


그림 5. 상호 간접 인증에서 고객 에이전트의 작동  
Fig. 5. Activity of customer agent in the mutual indirect authentication.

만약 이 시스템에 위조된 중개인이나 판매인이 존재하는 경우, 고객은 자신을 인증해준 은행에 Vendor\_ID#(혹은 Broker\_ID#)와 Bank\_IDVendor#(혹은 Bank\_IDBroker#)로 해당 판매인(혹은 중개인)이 있는지 확인해 줄 것을 의뢰한다. Bank\_IDBroker#와 Bank\_IDVendor#는 은행연합을 통해 인증 받은 상태로 Bank\_IDCustomer#를 은행 연합에 소속된 것인지를 확인한다.

Bank\_IDVendor#에 Vendor\_ID#가 존재하지 않는 경우에는 위조로 판별되지만 Vendor\_ID#를 갖는 판매인이 있는 경우에는 그것을 위조하여 사용하는 경우가 있으므로 Bank\_IDBroker#의 은행은 Vendor\_ID#를 갖는 판매인을 찾아 MessageBroker와 MessageVendor의 시간 정보를 이용하여 두 개 이상의 질의를 하게 된다. 이 질문은 다음과 같은 내용을 포함하고 있어야 한다.

- ① 메시지를 만들었을 때의 고객의 신분은?
- ② 이 고객에게 어떤 물품 정보를 제공하였는가?
- ③ Vendor\_ID#와는 거래하지 않은 다른 고객의 시간 정보는 무엇인가?

이 질문 중 마지막 질의에는 '모른다'의 의미인 Null이 답해야 정상적인 판매인으로 인식된다. 이 방법에는 질의에 답한 내용이 모두 맞는 경우에는 물건을 판

매하지도 않은 판매인에게 돈이 지급되는 경우가 발생할 수도 있으므로 판매인의 은행은 고객의 이의 제기 에 대비해 일정 기간 이후에 판매인에게는 지급되도록 해야한다.

또 고객이 위조된 경우를 피하기 위해 중개인과 판매인은 고객이 발행한 에이전트의 진위여부를 확인하여야 하는데 이를 위해 중개인이 발행한 에이전트(BA)와 판매인이 발행한 에이전트(VA)는 Bank\_IDCustomer#에 해당하는 은행에 Customer\_ID#가 존재하는지를 확인한다. 확인이 완료되면 BA와 VA는 고객 에이전트가 제공한 Bank\_IDCustomer#에 해당하는 은행이 존재하는지를 각각이 소속된 은행에 의뢰한다. 의뢰를 받은 은행은 진위 여부를 은행 연합에서 확인한다. 이 시스템에 참가하는 구성원들간의 자금 이동에 필요한 단계는 그림 6과 같다.

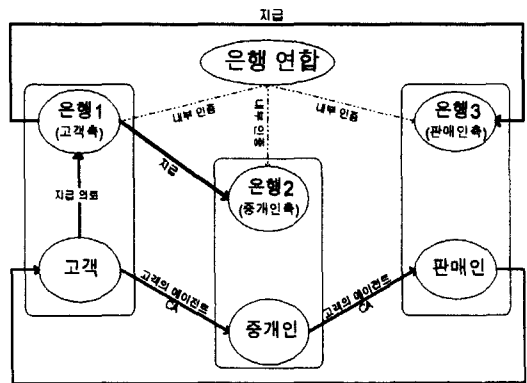


그림 6. Bank\_IDCustomer#의 지급 과정  
Fig. 6. Payment processing of Bank\_IDCustomer#.

고객이 소속된 은행이 중개인과 판매인에게 대금을 지급하기 위해서는 Request\_payment와 Payment\_Commit이라는 에이전트를 이용하여 안전한 지급을 유도해야한다. 고객이 지불을 의뢰한 중개인과 판매인을 찾아 일정 금액을 지불하기 위해 Bank\_IDCustomer#에 해당하는 은행은 Bank\_IDBroker#와 Bank\_IDVendor#에 PriceBroker\$와 PriceVendor\$를 시간 정보(TimeStampVendor\$)와 함께 전송한다.

```
Bank_IDCustomer# → Bank_IDBroker#:
Request_payment( Bank_IDCustomer#,
Broker_ID#, PriceBroker$, TimeStampVendor$ )
Bank_IDCustomer# → Bank_IDVendor#:
Request_payment( Bank_IDCustomer#,
```

Vendor\_ID#, Price<sub>Vendor</sub>\$, TimeStamp<sub>Vendor</sub>\$)  
 지급이 완료되면 Bank\_ID#를 갖는 은행들은 은행연합에 “지급 완료” 에이전트를 통해 통보한다. 각 은행들이 은행연합에 통보하는 지급 완료 에이전트는 다음과 같이 구성된다.

```
Bank_IDCustomer# → Bank_IDSuperBank# :
Payment_Commit (Bank_IDCustomer#,
  Bank_IDBroker#, Bank_IDVendor#, PriceBroker$,
  PriceVendor$, TimeStampVendor$)
Bank_IDBroker# → Bank_IDSuperBank# :
Payment_Commit( Bank_IDBroker#,
  Bank_IDCustomer#, PriceBroker$, Broker_ID#)
Bank_IDVendor# → Bank_IDSuperBank# :
Payment_Commit ( Bank_IDVendor#,
  Bank_IDCustomer#, PriceVendor$,
  Vendor_ID#, TimeStampVendor$)
```

은행 연합은 “지급 종료”라는 에이전트를 Bank\_ID<sub>Customer</sub>#에 전송한다.

```
Commit(Bank_IDSuperBank#, ResultBroker$,
  ResultVendor$)
```

반면 고객을 인증한 은행은 자신을 확인한 판매인이 위조된 것인지 확인하기 위해 은행에 전달된 에이전트가 전달한 Broker ID#로 중개인을 찾는다. 이 은행이 보낸 에이전트 BA는 판매인에 접근한 고객의 에이전트가 해당 중개인을 경유하였는지를 확인한다. 경유하지 않은 경우라면 해당 판매인으로서의 지급이 이루어지지 않는다. 이 시스템에서 간접 인증 조건은 중개인이 고객과 판매인을 알고 있어야하며 각각의 구성 요소들은 그 신분을 인증받은 상태여야 한다.

#### IV. 결 론

본 논문에서 제안한 상호 관계성을 고려한 전자 결제 시스템은 현재 단일 중개인만이 참여한 상태를 보인 것이다. 많은 전자 결제 시스템에서 양단간의 인증을 언급하였으나 고객, 중개인, 판매인이라는 3자 관계나 그 이상의 참여자를 고려한 것은 아니었다. 이 시스템에서는 항상 신뢰받는 신분이란 존재하지 않는다는 점을 고려하였다. 따라서 거래를 시작할 때는 항상

자신의 신분이 아닌 관계성을 밝힐 수 있어야한다. 여기서 관계성이란 고객-중개인, 중개인-판매인, 판매인-고객, 고객-은행, 은행-은행연합의 인증 관계를 말한다. 이러한 인증관계가 성립할 때 비로소 거래가 성립하게되며, 결제가 이루어지도록 함으로써 전자상거래에서 안전한 결제가 이루어지도록 하였다.

이 시스템에서는 교착상태를 방지하기 위하여 사용자가 원하는 시간 안에 에이전트가 거래를 완료할 수 있어야만 한다. 만약 해당 시간 안에 거래가 완료되지 않는다면 에이전트는 자신의 신분을 증명하는 내용을 제외한 모든 내용을 삭제한다. 이는 고객 에이전트를 통하여 이 시스템에 참가하는 구성원들의 정보가 침입자에게 한정된 시점에 대하여만 참조 가능한 것으로 만들기 위해서이다. 또 이러한 제한 사항으로 인해 중개인, 판매인, 그리고 그들이 소속된 은행과 고객이 소속된 은행이 보다 빠른 거래를 완료할 수 있는 방법이다.

본 논문에서 제안한 전자결제 시스템은 기존의 전자결제 시스템에서 이용되는 방법들의 문제점으로 제시된 양단간의 인증만 가능한 것과, 시간제약을 갖는 시스템으로의 확장성의 결여등의 문제를 해결할 수 있는 방법으로서 전자상거래에서의 결제방식의 새로운 방향을 제시한 것이다.

#### 참 고 문 헌

- [ 1 ] Needham, R.M. & Schröder, M.D. 1978 Using Encryption for authentication in Large networks of Computers. CACM Vol. 21, No 12, pp. 993~999.
- [ 2 ] Needham, R.M. & Schröder, M.D. 1987 Authentication Revised. Operating Systems Review Vol. 21, No.1, pp. 7.
- [ 3 ] NetBill Security and Transaction Protocol, B. Cox, J.D. Tygar, M. Sirbu., In Proceedings of the USENIX Workshop on Electronic Commerce, July 1995.
- [ 4 ] Marín Abadi, Michael Burrows and Roger Needham, A Logic of Authentication, Proceedings of the Royal Society, Series A, 426, 1871 (December 1989), 233-271. Also appeared as SRC Research Report 39 and, in a shortened form, in ACM

- Transactions on Computer Systems 8, 1 (February 1990), 18-36.
- [ 5 ] Martín Abadi and etc., The Millicent Protocol for Inexpensive Electronic Commerce, World Wide Web Journal, Fourth International World Wide Web Conference Proceedings, O'Reilly & Associates, Inc. (December 1995), 603-618.
- [ 6 ] Rivest, R.L., Shamir, A. & Adelman, L. 1978 A method for obtaining Digital Signatures and Public-Key Cryptosystems. Communication of the ACM Vol. 21, No.2, pp. 120~126.
- [ 7 ] Bhandaru N. and Coft W., "An architecture for supporting goal-based cooperative work," in Gibbs S. and Verrijn Stuart A., eds., Multi-User Interfaces and applications, Elsevier Science Publisher B.V., North-Holland, pp 337-354, 1990.
- [ 8 ] E.S.LEE "An Application of Agent for Next generation Computing Using," Int'l Biannual Conf. on Industrial Survival Strategy for Next Generation Software Technology, pp 115-140, 1996.
- [ 9 ] Dag Johansen, Nils P. Sudmann and Robbert van Renesse, Performance Issues in TACOMA, 3rd. Workshop on Mobile Object Systems, 11th European Conference on Object-Oriented Programming, Jyvaskyla, Finland, 9-13 June 1997.
- [ 10 ] 최중민, 에이전트의 개요와 발전 방향, 한국 정보 과학회, 에이전트 시스템, pp 7-16, 1997년 3월, ISSN 1015-9908

---

 저 자 소 개
 

---

## 朴 泳 鏞(正會員)

1983년 동국대학교 전자계산학과 졸업(학사). 1985년 동국대학교 대학원 컴퓨터공학과 졸업(공학석사). 1986년 12월 ~ 1991년 8월 통계청 전산사무관. 1992년 ~ 현재 부천대학 사무자동화과 조교수. 관심분야: 운영체제, 컴퓨터보안, 컴퓨터통신등

## 李 金 石(正會員)

1971년 서울대학교 공과대학 응용수학과 졸업(학사). 1973년 한국과학기술연구소 전산개발센터 전산기술과 근무. 1978년 한국과학기술원 전산학과 졸업(이학석사). 1981년 ~ 현재 동국대학교 컴퓨터공학과 교수. 관심분야: 운영체제론, 컴퓨터성능평가, 소프트웨어공학, 컴퓨터보안등