소액지불시스템 MilliCent에서의 빈번하지 않은 거래에 대한 효율성 개선

이석준°, 이문규, 박근수
{sjlee, mklee, kpark}@theory.snu.ac.kr
서울대학교 컴퓨터공학과

Two Methods to Improve Efficiency of Infrequent Purchases in the MilliCent System

Sokjoon Lee, Munkyu Lee, Kunsoo Park
Department of Computer Engineering, Seoul National University

요 약

전자상거래에서 거래 비용을 최소화하여 소액의 거래에 대한 지불이 가능해지도록 만든 시스템을 소액지불시스템(Micropayment System)이라고 한다. 소액지불시스템으로 제안된 많은 시스템 중 MilliCent는 고객과 상점의 거래가 자주 이루어지고, 그 관계가 오래 지속된다는 가정 하에 매우 효율적인 시스템이다. 그러나, MilliCent는 고객과 상점의 거래가 자주 이루어지지 않는 경우에는 고객과 상점의 거래에 필요한 계산 및 통신에 비해 상대적으로 은행과의 통신이 지나치게 많이 필요하므로, 효율이 떨어지는 단점이 있다. 본 논문에서는 이 경우에 대해 은행과의 통신량을 가능한 줄여 은행 서버의 부담을 줄이는 두 가지 방법을 제시한다

1. 서론

고객이 매장을 직접 방문하여 물품을 구매하는 기존의 상거래와는 달리 인터넷을 이용하여 구매하는 전자 상거래가 급속도로 발달하고 있다. 전자 상거래에서 이용하는 인터넷은 그 개방성으로 인하여 구매 비용을 지불하는데 있어서 안전성에 문제가 생길 수 있으며, 고객과 상점이 직접 대면하여 물품을 구매하는 것이 아니므로 사기 등의 위험성이 매우 크다고 할 수 있다. 이를 위하여 고객과 상점이 믿고 사용할 수 있는, 안전한 전자지불시스템이 필요하다.

전자지불시스템이란, 전자 상거래에서는 기존의 상거래에서 사용하던 현금, 수표, 어음 등의 장표 결제수단을 사용할 수 없으므로, 이에 대신하여 전자적 매체를 통하여 지급 결제 기능을 수행하도록 하는 모든 거래흐름을 포괄하는 개념이다. 이는 거래시 은행과의 통신 여부에 따라 온라인(On-Line)과 오프라인(Off-Line) 시스템으로 분류할 수 있으며, 지불 방법에 따라 전자 현금 시스템, 신용카드 기반 시스템, 수표 시스템, 직불 기반 시스템 등이 있다.

인터넷상에서 신문 기사의 구독, 음악 화일의 구입 등을 위해 소액의 금액을 지불해야 할 경우, 전자지불시스템의 거래 비용이 매우 적어야 한다. 거래 비용이라 함은, 전자지불시스템이 유지, 보수되기 위한 비용, 거래시 통신 비용, 저장 비용 등 하나의 거래가 이루어지기 위해서 부수적으로 들어가는 수수료 비용을 말한다. 거래 비용이 적지 않으면 거래 비용이 구매 비용을 초과하거나 매우 큰 부분을 차지하여 그 시스템을 사용하기에 경제성이 없어지기 때문이다. 이렇게 거래 비용을 최소화하여 1달러 미만의 소액을 거래할 수 있도록 하는 전자지불시스템을 소액지불시스템(Micropayment System)이라고 한다.

현재까지 제안된 소액지불시스템은 MilliCent[1], PayWord[2], MicroMint[2], NetCard[3], Micro-iKP[4], MiniPay[5] 등이 있다. PayWord, NetCard. Micro-iKP는 고객이 해쉬 체인을 생성하여 상점에 체인의 일부를 알려 주는 방법을 통해 지불을 하고, MicroMint는 해쉬 함수의 충돌을 이용하여 화폐를 발행하며, MilliCent는 해쉬 함수, 비밀키를 이용하여 보증서와 서명을 만드는 방법 등을 사용한다.

이 대부분의 소액지불시스템들은 안정성을 최소화하면서 거래 비용을 최소화하기 위해, 공개키 암호 알고리즘의 사용과 은행과의 통신량을 최소화하는 방법을 사용하고 있다.

공개키 암호화기법은 일반적으로 시간이 오래 걸리고 키의 길이가 길기 때문에, 대부분 온라인 상으로 거래되어 빠른 응답시간을 요구하는 소액거래에는 부적합하다. 또한, 은행 서버에 대한 통신량의 증가는 은행 서버에 대한 고비용의 투자를 발생시키고, 이는 곧 소액지불시스템의 거래 비용을 높이는 결과를 초래한다. 특히, MilliCent는 고객이 상점과 자주 거래를 할 경우 상대적으로 높은 성능을 보이는 반면, 고객과 특정 상점이 빈번하지 않은 거래를 할 경우 한 거래당 은행과 통신량의 비율이 매우 높아 이러한 거래가 이 소액지불시스템에서 다수를 차지할 경우 시스템의 거래 비용을 증가시킬 수 있는 단점이 있다. 그러므로 거래가 이루어질 때 은행과의 통신량을 줄이는 것은 매우 중요한 일이다.

본 논문은 2장에서 MilliCent 시스템에 대한 설명을 하고, 3장에서는 MilliCent 시스템에 대한 평가와 함께 고객이 한 상점과 자주 거래를 하지 않을 경우 생기는 은행과의 통신을 줄이기 위한 두 가지 방법을 제안한다. 그리고 4장에서 결론을 내린다.

2. MilliCent

MilliCent[1]는 1995년 DEC SRC사에서 Mark Manasse 등에 의해 만들어진 소액지불시스템이다. MilliCent에서는 Scrip이라는 전자 현금이 쓰이며, 직접 Scrip을 구매해야 사용할 수 있는 직불 기반 시스템이다

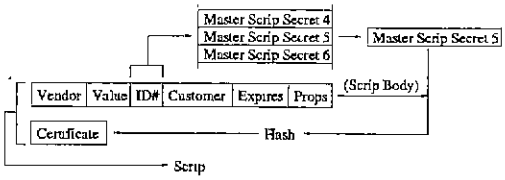
2.1 Scrip

Scrip은 상점, 혹은 상점과 발행 계약을 체결한 은행에서 발행하게 되며, 고객은 은행으로부터 이 Scrip을 구매하게 된다. 만약 은행이 상점으로부터 발행 계약을 체결하지 않았을 경우, 은행은 (발행 계약을 체결한) 다른 은행을 소개해주거나, 그 상점으로부터 미리 Scrip을 대량 사 놓아야 한다.

이 때 고객이 Scrip의 대가로 은행에 지불하는 돈은 MilliCent 방식이 아닌, 보다 상위의 전자지불시스템(Macropayment system)을 이용하게 된다 그리고 은행은 고객에게 보안이 유지되는 상태에서 Scrip과 함께 Customer Secret를 건네준다.

각 Scrip은 특정 상점에게만 지불할 수 있으며, 고객은 상점에서 물건을 사는 대가로 Scrip을 지불하고 거스름돈으로 새로운 Scrip을 받게된다.

Scrip의 구조는 [그림 1]과 같다.

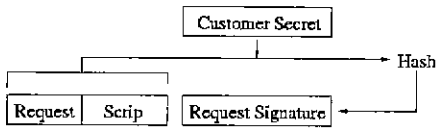


[그림 1] MilliCent의 Scrip

- Vendor Scrip을 사용할 수 있는 상점의 ID
- Value : Scrip의 액면가
- ID # : 이 Scrip을 구별할 수 있는 ID로, Scrip의 복사에 의한 재사용 문제를 방지하기 위한 것이다. 또한, 이 Scrip의 Master Scrip Secret를 선택할 때 사용된다.
- Customer 이 Scrip을 사용하는 고객의 ID
- Expires : 이 Scrip의 만료일
- Props . 기타 정보를 담는다.
- Certificate : H(Scrip Body || Master Scrip Secret) (단, H는 일방향 해쉬 함수이다) 에 의해서 얻어지며, 이를 Scrip에 대한 보증서로 사용한다. 즉, Master Scrip Secret를 모르는 어떤 사람이 Scrip의 다른 부분을 고치려 하거나(예를 들어 돈의 액수나 ID #부분 등) Scrip을 새로 만드는 경우, 올바른 Certificate를 만들 수 없다. 그러므로 Master Scrip Secret는 돈을 발행하는 곳(상점, 상점과 계약한 은행)만이 알고 있어야 한다.

2.2 물품 구매 요구

물품을 구매하기 위하여 고객은 상점에게 Scrip과 함께 물품 구매 요구서를 보낸다. 이 요구서의 구조는 다음 [그림 2]와 같다.



[그림 2] 물품 구매 요구서

- Request : 물품 구매 정보를 담고 있다
- Request Signature : H(Scrip || Request || Customer Secret)에 의해서 만들어지며, 이를 전자서명으로 사용한다. Customer Secret는 고객과 상점만 알고 있으므로, 다른 사람은 Request Signature를 위조할 수 없으며 해쉬 함수 H가 일방향 함수일 경우, 이 서명을 가지고 역으로 Customer Secret을 알아낼 수도 없다.

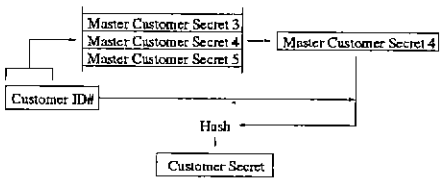
상점은 이 요구서를 받으면, Request와 Scrip, Customer Secret를 해쉬해서 Request Signature와 같은지, 그리고 Scrip이 올바른 Scrip인지를 확인한다. 그리고 요구서와 Scrip이 올바르다고 증명이 되면, 상품과 함께 새로운 거스름돈을 생성하여, 다음과 같은 담장을 고객에게 넘겨준다.

Scrip', Reply, H(Scrip' || Cert || Reply || Customer Secret) (단, Scrip'는 거스름돈에 해당하는 Scrip이며, Cert는 H(Scrip Body || Master Scrip Secret)로 이 담장이 실제로 자신의 요구서에 의한 것인지를 확인할 수 있는 정보가 된다.)

고객은 이를 받으면 이 거스름돈에 대한 서명이 맞는지를 확인하고, 다음 거래에는 이 Scrip을 사용하도록 한다.

2.3 보안

- Master Scrip Secret Scrip의 보증서를 만들기 위해 필요한 정보로, Scrip을 발행하는 구성원만 알고 있어야 한다. 여러 개의 Master Scrip Secret를 가지고 있으며, Scrip의 ID에 따라 선택하게 된다.
- Master Customer Secret : Customer Secret를 만들기 위해 필요한 정보이며, Scrip을 발행하는 구성원만 알고 있어야 한다. 상점은 미리 여러 개의 Master Customer Secret를 가지고 있도록 한다.
- Customer Secret 고객이 상점에서 물품을 구매하려 할 때, 물품 구매 요구에 대한 서명에 필요한 정보로, [그림 3]과 같이 Customer의 ID의 앞부분을 이용하여 Master Customer Secret를 선택한 다음 선택한 Master Customer Secret와 Customer의 ID를 해쉬하여 구한다. 이 정보는 상점과 고객만 알고 있어야 하며, 상점은 고객별로 Customer Secret를 따로 저장해 둘 필요가 없이, 필요할 때마다 다시 Customer ID로 위의 과정을 통해 구하면 된다.
- 해쉬 함수 : Scrip에 대한 보증서를 만들 때, 혹은 고객이 구매 요구에 대한 서명을 할 때 사용되며, MD6[7]나 SHA[8]와 같은 일방향 함수(one-way function)이어야 한다



[그림 3] Customer Secret의 생성

3. 개선

3.1 MilliCent의 특징

MilliCent는 다음과 같은 장점을 가지고 있다.

- Master Scrip Secret를 모르면 Scrip을 위조할 수 없으므로, Master Scrip Secret의 보안을 유지하면 위조 문제가 해결된다.
- Scrip의 ID와 Master Scrip Secret를 통해 재사용 문제를 해결할 수 있다.
- 고객과 상점이 거래를 시작하면, 그 이후부터는 공개키 암호화 기법이 사용되지 않는다.
- Scrip이 올바른 화폐인지 확인하기 위해 상점이 은행과 접속할 필요가 없다.

그러나, 이와 함께 다음과 같은 단점을 가지고 있다.

- Scrip이 올바른 Scrip인지를 고객이 확인할 방법이 없다.
- 고객은 각 상점에 대한 Customer Secret를 모두 저장하고 있어야 한다
- 고객과 상점이 계속해서 거래를 할 때만 효율적이며, 고객이 상점과 한 번만 거래를 하려 하거나, 자주 거래를 하지 않을 때 비효율적이다. 그 상점에 대한 Scrip이 없을 경우, 은행과의 통신을 통해 새로운 Scrip을 구매해야 하며, 잔돈 Scrip이 남고 이 Scrip에 대한 상점과의 거래가 앞으로 이루어지지 않을 것이라고 예상되는 경우, 이를 가지고는 다른 상점과는 거래를 할 수 없으므로, 다른 상점의 Scrip이나 은행의 전자화폐로 바꾸기 위해서 은행에 다시 한번 접속해야 하는 문제가 있다.

다음 3.2에서 세 번째 단점을 개선하는 방법을 제시한다

3.2 개선

3.2.1 확률적 반환

MilliCent에서 자주 거래하지 않는 경우, 은행과의 통신이 한번 더 생기는 원인은 불필요한 잔돈 Scrip이 생기기 때문이다. 만약, 이 잔돈 Scrip을 고객과 상점이 공평하게 하면서 없앨 수 있으면 이후에 생길 수 있는 고객과 은행과의 통신을 없앨 수 있다.

R. L. Rivest는 97년 전자복권방식을 이용한 확률적 지불 방식[6]을 제안한 바 있다. 그는 고객이 상점에 거래를 할 때, 확률적으로 대금을 지불하도록 함에 의해서, 고객과 상점이 추가 계산, 통신량을 부담하는 대신, 상점이 은행과 대금 정산을 훨씬 적게 하도록 하였다. 여기서는 이를 응용하여 다음 방식을 제안하고자 한다.

① 고객은 물품 구매 요구서에 확률적 반환 요구 내용을 포함시킨다.
② 상점은 잔돈 Scrip을 보낼 때, 무작위로 w를 선택하여 해쉬한 다음 H(w)를 Scrip에 포함시킨다.
③ 고객도 무작위로 w'를 선택하여 해쉬한 다음, H(w')와 기타 정보를 포함한 내용에 공개키를 이용한 서명을 하여 보낸다.
④ 상점이 먼저 w를 공개한다.
⑤ 만약 w ≠ w' (mod 100)이면 고객은 잔돈을 포기하는 Reply를 보낸다.
⑥ 만약 w = w' (mod 100)이면 고객은 w'를 상점에게 보내고 전자 화폐 지불을 요구한다. 상점은 고객의 서명을 확인한 다음, 잔돈 액수의 100배에 해당하는 전자 화폐를 MilliCent보다 상위의 전자지불시스템(Macropayment System)을 이용하여 보낸다. 이 단계는 1/100의 확률로 일어나게 된다.

상점이 손해를 보게 될 경우 잘못된 w를 공개함과 동시에 Scrip에 포함된 H(w)의 값을 부인할 수 있다. 혹은 고객이 잘못된 w와 H(w)를 받았다고 주장하며 잔돈 지불을 요구할 수 있다. 이렇게 되면 상점의 부인 방지, 혹은 고객의 거짓 방지를 위해 은행의 개입이 필요할 수 있다. 하지만, 이런 일들을 통해서 상점 혹은 고객이 얻을 수 있는 이익은 크지 않고, 처벌에 대한 부담 등을 고려하면 이런 일은 일어나기 어렵다고 가정할 수 있다.

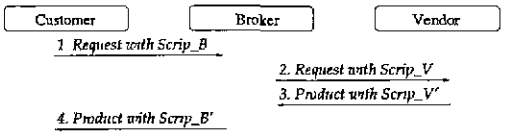
이 방법은 일반 거래에 비해 고객과 상점과의 통신량이 증가하고, 공개키 알고리즘을 한 번 사용하게 되지만 잔돈을 처리하기 위한 은행과의 통신이 없어지므로, 그만큼 은행 서버의 부담이 줄어드는 효과가 있다.

한편, 고객과 상점 모두 자신이 원하지 않은 금전적인 손해가 생길 수 있으며 반대로 이익이 생길 수 있다. 그러나, 거래되는 액수가 적고 고객과 상점은 서로 공평한 기대값을 가지고 있으므로 이 프로토콜에 따르는 경우가 많아질수록 통계적으로 큰 손해나 큰 이익을 볼 확률은 적어진다.

3.2.2 중개인 이용

R. Hauser 등은 처음 거래하는 상점 혹은 자주 거래하지 않는 상점과 거래를 할 경우 생기는 많은 량의 통신과 은행에의 접속을 줄이기 위해 중개인을 이용하는 방법을 [4]에서 제안한 바 있다. 이와 비슷한 방법을 다음과 같이 사용할 수 있다. 각 고객은 중개인의 Scrip을 가지고 있으며, 중개인 들은 또한 상점들과 계약을 맺고 각 상점들의 Scrip을 미리 구매해 가지고 있도록 한다.

거래에서 이루어지는 각 과정은 [그림 4]와 같다.



(단, *Scrip_V'*, *Scrip_B'*는 각각 *Scrip_V*, *Scrip_B*의 거스름돈)
[그림 4] 중개인을 이용한 거래

이 방법은 중개인의 개입으로 인한 통신 시간의 증가와 중개인에 대한 수수료의 부담이 생기는 단점이 있다. 그러나 개념이 간단하고, 자주 거래하지 않는 상점과 거래하기 위해서 은행을 통해 그 상점의 Scrip을 구매해야 하는 단점을 없애므로, 은행과의 통신량 자체를 크게 줄일 수 있다는 장점이 있다. 또한 MilliCent와는 다른 소액지불시스템만을 쓰는 상점이 있다면, 중개인이 그 시스템을 사용하여 상점과 거래를 하여, 고객이 다른 시스템을 추가적으로 사용할 필요가 없는 이점도 지닌다.

4. 결론

본 논문에서는 MilliCent 소액지불시스템에서 고객이 자주 거래하지 않는 상점과 거래해야 할 경우, 상대적으로 많은 은행과의 통신을 줄이는 방법에 의해 MilliCent의 효율성을 개선시키는 두 가지 프로토콜을 제안하였다.

첫 번째 프로토콜인 잔돈 Scrip의 확률적 지불에서는 상점이 고객에게 잔돈을 확률적으로 지불하게 함으로써, 잔돈 Scrip을 처리하기 위해 생기는 은행과의 추가 통신량을 없애도록 하였고, 두 번째 프로토콜에서는 중개인을 두어 고객이 새로운 상점에 대한 Scrip을 구하기 위해 생기는 은행과의 통신을 없애도록 하였다. 은행과의 통신을 줄이는 것은 은행 서버에 대한 병목 현상을 줄일 수 있으므로 은행 서버에 대한 고비용의 투자를 할 필요가 없어, MilliCent 시스템의 거래 비용을 줄이는 효과가 있다.

은행과의 통신을 줄이고, 대신 고객과 상점이 추가 계산, 통신량을 부담하는 것과 중개인의 개입이 MilliCent 시스템에서 수수료의 부담을 어느 정도 줄일 수 있는지는 실제 시스템에서 특정 상점과의 빈번하지 않은 거래가 얼마나 많이 일어나는지에 따라 달라질 것이다. 앞으로 이에 대한 실험을 통해 실제로 기존의 MilliCent, 더 나아가 다른 소액지불시스템에 비해 어느 정도의 효용성을 얻을 수 있는 지에 대한 평가가 있어야 하겠다.

참고 문헌

[1] S. Glassman, M Manasse, M. Abadi, P. Gauthier, and P Sobalvarro, "The MilliCent Protocol for Inexpensive Electronic Commerce", Fourth International World Wide Web Journal, 1995

[2] R. L. Rivest and A Shamir, "PayWord and MicroMint - Two simple micropayment schemes", http://theory.lcs.mit.edu/~rivest/publications.html, 1996

[3] R. Anderson, C. Manifavas, and C. Sutherland, "Netcard - a practical electronic cash system", Fourth Cambridge Workshop on Security Protocols, 1996

[4] R. Hauser. M. Steiner, M Waidner, "Micro-payments based on iKP", 14th Worldwide Congress on Computer and Communication Security Protection, 1996

[5] A. Herzberg, H. Yochai, "Mini-Pay - Charging per Click on the Web", Sixth International Wolrd Wide Web Conference, 1997, available from http://atlanta.cs.nchu.edu.tw/www/PAPER99.html

[6] R L. Rivest, "Electronic Lottery tickets as Micropayments", Proceedings of Financial Cryptography '97, LNCS series 1318, pp. 306-314

[7] R L. Rivest, "The MD5 message-digest algorithms", Internet Request for Comments, 1992, RFC 1321

[8] National Institute of Standard and Technology (NIST), FIPS Publication 180 Secure Hash Standard (SHS), 1993