基于票据的微支付形式

王海源　王　茜
（重庆大学计算机学院　重庆400044）

MicroPayment based on Scrip

WANG Hai-Yuan　WANG Qian
(Computer School, Chongqing University, Chongqing 400044)

Abstract　This paper describes two kinds of micropayment based on scrip——Millicent and MicroMint, especially analyzes their security, advantages and disadvantages, then prospects the development of micropayment.

Keywords　Micropayment, Millicent, MicroMint

1. 前言

随着电子商务的发展，电子支付正逐渐朝着两个方向发展：一种就是处理大型或中型交易额的宏支付形式，如银行互联网支付系统、电子支票、CyberCash 和 NetBill 等，其每一笔的交易费用都比较大。但是当进行交易额比较小的电子商务支付时，如果再采用宏支付的电子支付协议，每一笔的交易费用将在整个电子商务中占有较大的比重，所以宏支付协议不适合用于交易额比较小的电子支付中。

另外一种就是处理交易额比较小的微支付形式。顾名思义，微支付处理的金额比较小，收取的费用也就很少。所以这种支付机制有着特殊的系统要求，即在满足一定安全性的前提下，要求有尽量少的信息传输、较低的管理和存储需求，就是速度和效率要求比较高。因此，在微支付中一般不用或尽量少用公钥技术，而采用效率比较高的 Hash 函数。

微支付同现实生活的现金交易一样，也存在着盗窃、伪造硬币和重复消费等问题。虽然微支付对于安全问题考虑得不是太多，但是为了防止大规模安全问题的出现（如硬币被大规模地伪造、硬币大量被盗窃或者硬币被某一个消费者或商家经常性地重复消费等），每一种微支付机制都或多或少地有相应的安全机制。

根据微支付的支付类型，可以将微支付主要分为两类：一类是基于票据的微支付形式，如 Millicent，SubScrip，MicroMint；另一类是基于 Hash 链的微支付形式，如 PayWord，Net Card，Paytree。

本文首先简要介绍 Millicent 和 MicroMint 的原理和特点，然后针对两种微支付的安全问题进行分析、找出各自的优缺点以及对缺点的改进办法，并进行相互的比较。

2. Millicent

2.1　工作原理

票据是 Millicent 中一个重要概念。一个票据代表了商家给消费者建立的一个账号，在任何给定的有效期内，消费者都可以利用该票据购买商家的服务。当消费者利用票据在网上购买了商家的服务或商品以后，购买值将自动从票据中扣除，并返回一个具有新的面值的票据，以进行后续交易的平衡。当消费者完成了一系列交易或支付以后，它还可以把票据中剩余的值兑换成现金（同时账号关闭）。

票据的结构如图1所示。

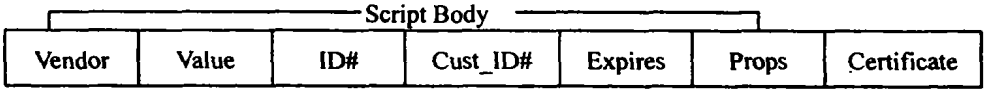

图1

可以看出在票据中给出了商家的标识 Vendor 以及票据的面额 Value，然后就是票据的唯一标识符 ID#（它其中一部分用于产生证书的主票据密钥），Cust_ID# 用于产生消费者密钥（其中一部分用于产生消费者密钥的主消费者密钥），Expires 是票据的过期时间，而 Props 是消费者属性的额外数据（用于给出消费者某些具体的信息），这些是票据的主题，而 Certificate 是票据的签名，防止票据被伪造或破坏。

在 Millicent 中有三个实体，即消费者 C、中间人 B 和商家 V 维持一个不对称的信任关系，中间人最为可信，其次是商家，最后是消费者。一般情况下，中间人都是一些大型的、具有良好信誉的金融机构。中间人是消费者和商家之间的中介。中间人通过大量购买商家的票据，然后再零售给消费者获得

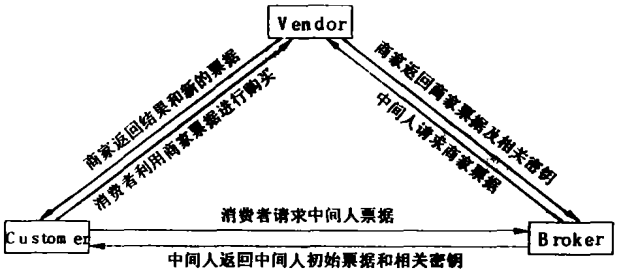

图2

王海源　硕士生，研究方向为电子商务及其安全性。王　茜　硕士生导师，研究方向为电子商务及其安全性、图像压缩等。

利润。中间人获取商家票据主要采取3种方式：票据仓库，票据许可生产，多中间人模式。

Millicent 协议有3种不同的实现形式，以在效率和安全性方面提供平衡，可选择保证交易的认证性和机密性：即：1）明文票据方式；2）私有和安全方式；3）安全但不使用加密的方式。具体的交互过程如图2所示。

2.2 安全性分析

1）伪造票据　在消费者进行消费时，并不以明文传递消费者密钥。消费者将请求、票据以及消费者密钥连接在一起，然后进行 Hash 运算，将结果作为请求的签名。消费者向商家发送请求、票据以及签名。当商家接到请求时，因为商家并没有记录与单个票据相关的密钥，他利用票据中的 Cust_ID# 以及他记录的对应的主消费者密钥来产生消费者密钥。也就是说，主消费者密钥就是用来产生消费者密钥的，它只能被商家和中间人拥有。

主票据密钥是用来认证票据的，防止票据被修改或被破坏，它也只可以被商家和中间人拥有。当商家接到票据时，首先要验证票据的真伪以及是否被破坏。首先，商家利用票据中的 ID# 以及相对应的主票据密钥对重新计算，然后同附加在票据中的证书进行比较，如果票据被修改或者被破坏，两个证书将不匹配。这样商家就可以防止票据被伪造或被修改。

2）偷窃票据　当消费者购买商家的初始票据时，将根据消费者标识产生消费者密钥连同票据返还给消费者，消费者密钥表明消费者对票据的拥有权。即使入侵者获取了票据，因为不知道消费者密钥，也不能用它来消费，即可以有效地防止票据被盗用。

3）重复消费　商家为每一个票据记录了一个唯一的标识符，这样可以防止票据被重复消费。另外为了减轻商家维护该记录的压力，每一个票据都给出了过期时间。当票据过期时，商家就可以直接判断，并将过期的票据记录删除。消费者可以在票据过期之前进行兑换，产生一个新的票据。

4）效率和安全性的平衡　通过在商家站点本地验证票据来减少通信的消耗，不需要中心处理服务器，减少了商家计算的负担。商家只同少数的中间人保持帐务联系，通过中间人减少帐务的处理。在 Millicent 中，由消费者利用票据来维持帐务平衡。由于使用了数字签名，商家不用担心消费者对票据进行修改。

2.3 小结

在 Millicent 中，没有使用公钥技术，而采用了效率更高的 hash 函数，部分采用了对称加密算法。由于部分实现了对称密钥机制，增加了系统的存储和计算负担；另外票据是针对特定商家的，且最终由商家产生和验证（也可由中间人代为产生），所以，消费者不能验证凭据的真伪；因为针对每一个新的商家，消费者都要请求一个新的票据，Millicent 对经常更换商家的消费者效率不高。在票据中采用了标识的方式来代表消费者的身份，具有一定的匿名性，但消费者证书的使用，对消费者的匿名性是一个损害。

3. MicroMint

3.1 工作原理

MicroMint 是基于唯一标识的离线电子现金支付方式，同样涉及到交易的三方：消费者、商家和中间人。MicroMint 货币是由中间人制造并出售给消费者的，消费者作为支付费用提交给商家。一般来说，中间人在每月初发行新的货币，并公布验证货币的方法。在月末，回收还没有被消费的货币，并准备发行新的货币。商家在某个恰当的时间（一般是一天）通过离线方式在中间人处结算。

MicroMint 建立在 hash 函数冲突原理基础之上。单向 hash 函数 h 把 m 位的 x 映射到另一个具有固定长度 n 的 $y=h(x)$ 值。当两个不同的值（如 x1 和 x2）都被 h 映射到同一个值 y 时，即 $h(x1)=h(x2)=y$，则出现了 hash 函数 h 的一个（双向）冲突。在一般情况下，当 k 个不同的输入值 $x_1, x_2, \cdots\cdots x_k$ 都被 h 映射到同一个值 y 时，即 $h(x_1)=h(x_2)\cdots=\cdots=h(x_k)=y$，则会出现一个 k 向 hash 函数。在 MicroMint 中，一个硬币由 k 向 hash 函数冲突来代表，k 一般取4。所以，一个 MicroMint 货币由一个四向 hash 函数冲突来代表，即由四个具有相同 hash 值 y 的输入值 x_1, x_2, x_3, x_4 组成：$C=\{x_1, x_2, x_3, x_4\}$，它代表一定数量的小额钱，如一分等。

如果我们将计算 $y=h(x)$ 比作随机地投掷一个球到 2^n 个箱子中的话，则 MicroMint 货币可用投掷到某一个箱子中的球（即 x）来表示（如果这个箱子中至少有 k 个球）。如果一个箱子中有多于 k 个球的，原则上中间人可以随机抽取 k 个球来组成多个货币。但是如果这样，造假者可能获得同一个箱子中不同货币，然后将他们组合制造假的货币。因此最好的方法是中间人最多可以从一个箱子中产生一个货币。

还有一个问题就是，中间人因为无法准确估计消费者需要的货币，因此必须保存大量实际上用不到的货币，给中间人造成大量的存储浪费。一个好的解决办法就是采用优质货币。如果 n 位 y 的高位（t 位）等于一个中间人特有的值 z，则将计算此 y 的 x 称为优质货币。这样可以大大节省存储空间。

在 MicroMint 中，货币的产生会很困难，但是货币的验证是很简单的。举个例子：假设 $k=4, n=54$，则产生第一个硬币大概需要进行 2^{54} Hash 运算，在一个每秒可以进行 2^{14} Hash 运算的计算机来说，需要大概80年的时间。因此在 McroMint 中的安全性就是基于这种因素而考虑的，即制造前几个硬币的代价会很高，但是随着硬币量的增加，成本会迅速降低。

商家验证货币时，首先保证每个 $x_i (i=1,2,3,4)$ 互不相同，并验证这四个 hash 值都映射到同一个 y，则可以保证货币的真实性。但不能发现重复花费，因此，中间人必须保存每一个已花费过的硬币的副本，以便进行核查。

3.2 安全性分析

下面就针对 MicroMint 中可能出现的安全问题进行分析并介绍相应的措施：

1）伪造货币　通过能动地增加适当的措施，MicroMint 可以有效地防止大规模的硬币伪造（对于小规模的硬币伪造，MicroMint 基本上不予考虑，因为通过上面可以知道，那对造假者而言是得不偿失的）。在 MicroMint 中，中间人在每一个月的月末发行下一个月流通的新的硬币，并收回那些没有消费的硬币，然后在每一个月的月初宣布新的硬币的验证方法，这样就可以有效地防止硬币的伪造。因为对造假者而言，他只有在中间人发布新的验证方法以后才有可能进行硬币的伪造，时间方面显得仓促，而且在每一个月的月末，所有的硬币都将被宣布为无效（当然也包括伪造的硬币）。如果中间人发现有大规模的伪造硬币出现，他可以随时宣布本次硬币无效而发行新的硬币。

2）偷窃货币　由于每一次消费者消费的硬币不会很多，所以此时硬币被偷窃，消费者也不会很在意。但是如果在中间人发行新的硬币给消费者或从商家那里回收硬币时发生硬币

偷窃，因为此时数目相对比较庞大，则不得不考虑安全问题。最简单的考虑就是此时使用公钥加密技术来保证硬币传送的安全性。但是考虑到效率的问题，就尽量避免使用公钥加密技术，并通过使用效率比较高的 Hash 函数来保证安全性。

第一种方法就是组相关，即中间人将消费者分成若干个组。消费者硬币的有效性与他所在的组建立关联。具体地讲就是，中间人分给每一个消费者一个数字 ID 和硬币，商家通过如下的附加条件很容易进行验证：

$h'(x_1, x_2, \cdots, x_k) = h'(ID)$

h′是产生短的散列和的加密散列函数。散列和说明客户从属的组。

还有一种方法就是使消费者的硬币只能用于特定的商家。这对硬币的盗窃者来说就更没有吸引力了。

3）**重复消费** 由于 MicroMint 并不是基于匿名的，所以中间人可以发觉是哪一个商家兑换的重复消费的货币，甚至可以找到是哪个消费者消费的这种货币。因此，中间人可以不兑换这种货币；而且当发生大规模的重复消费时，中间人可以剔除相应的商家或消费者。但是这样做，可能伤害那些诚实的但是被动收到重复消费货币的商家的感情。并且中间人也只是被动地剔除那些有欺骗嫌疑的商家或消费者，而不能主动地对这些欺骗行为进行惩罚。

3.3 小结

MicroMint 没有采用公钥和对称加密技术，整体的安全性不如 PayWord，但由于采用了四向 hash 函数冲突，大规模的欺骗在计算上是不可行的。同 PayWord 不同，MicroMint 货币不是针对某一特定 M 的，所以，可允许 C 高效地和多个 M 交易，这也是 MicroMint 区别于其它微支付的显著特点。另外，与 Millicent 不同，C 可以在本地验证硬币的真伪。

结束语 一般来说，所有的微支付都是建立在效率和安全平衡的基础上的。由于微支付的交易额比较少，所以我们可以寻找效率较高但是又可以保持适当安全的方法进行改进。如利用椭圆曲线来替代现有的 RSA 算法，可以在充分利用公钥技术特性基础上，有效提高系统效率。还有就是对现有的微支付进行局部的改进，以提高微支付的效率和安全性。

随着网络技术及网络应用的发展，微支付的应用将会越来越广泛。例如网络出版或者网络信息提供就可能是微支付应用的一个重要方面。还有就是结合移动通信和移动电子商务中支付的特点，微支付在移动计费中的应用也显得越来越重要，这也是微支付的一个重要发展方向和研究热点。

参考文献

1 Glassman S, et al. The Millicent Protocol for Inexpensive Electronic Commerce. http://www.w3org/Conferences/WWW4/Papers/246/

2 Lang P. Product review MilliCent micropayment system. http://sellitontheweb.com/ezine/millicent.shtml. 1998

3 Glassman S, Jones R, Manasse M. Microcommerce On The Horizon. http://research.compaq.com/SRC/articles/199705/Millicent.html

4 Rivest R, Shamir A. Security Protocols Workshop. http://citeseer.nj.nec.com/rivest-payword.html

5 Puhrerfellner M. An implementation of the Millicent micro-payment protocol and its application in a pay-per-view business model. http://citeseer.nj.nec.com/507471.html. 2000

6 Rivest R, Shamir A. Payword and MicroMint Two simple micropayment schemes. http://theory.lcs.mit.edu/~rivest/Rivest-Shamir-mpay.pdf. 2001

7 Technological Foundation of E-Commerce-chapter5: Digital Payment System. SIMENS AG, CTIC 3. Security/Electronic Commerce

8 李明柱，李志江，杨义先. 微支付机制及应用分析综述. 计算机工程与应用，2002，38(3)

9 林枫等编著. 电子商务安全技术及应用：第6章. 安全电子支付概论. 北京航空航天大学出版社，2001

10 Vesna Hassler 著，钟鸣等译. 电子商务安全基础. 人民邮电出版社，2001

（上接第133页）

```
    THEN { F(k,i)=S(k,i)+L(k,i) * W(k,i);
           S(k,i)=F(k-1,i); }
    IF (该包不属于任何一个已经有的类)
    THEN { F(k,j)=C(t)+L(k,j) * W(k,j)}
    }
队列扫描:SM=非活动类数目;
调度:
  IF (类0队列非空)
      THEN 按实际分配带宽 C0=LBW1调度类0队首数据包
      ELSE IF (类1队列非空)
          THEN 按实际分配带宽 C1=LBW2调度类1队首数据包
      ELSE
          { 扫描后面类队列和缺省类中的不同流子队列的数据包;
            根据 CN,LBW_i,CM 计算实际分配带宽 BW_i;
            选择 C(t)=min(F(k,i))按实际分配带宽 BW_i 调度出队列;}
```

结束语 调度算法是 IP 宽带网络交换机中的重要部分，它为传输实时业务提供 QoS 保证。本文对几种常见的调度算法进行了比较研究，在此基础上进行改进得到 LLQ+CB-WFQ 算法，并给出其实现过程。这些研究成果我们应用于“大唐光通信公司多业务交换平台开发实现项目”中，取得较好的 QoS 性能特征。

参考文献

1 Chen J S, Guerin R. Performance study of an input queueing packet switch witch with two priority classes. IEEE Trans. Commun [J], 1991, 39(1): 117～126

2 Hluchj M G, Karol M J. Queueing in high-performance packet-switching, IEEE J. Sel. Areas Communcation[J], 1998, 6(9): 1587～1597

3 Lee T. A modular architecture for very large packet switches. IEEE Transactions on Communcation [J], 1990, 38(7): 1097～1106

4 戴礼森，洪佩琳. 高速信元交换调度算法研究. 电子学报[J]，2000，28(5)：96～98

5 黄立群. FQLP：ATM 网中一种新的实时业务调度算法. 电子学报[J]，2000，28(4)：20～23