UNIVERSIDADE DO MINHO

Escola de Engenharia
Departamento de Sistemas de Informação

Digital Cash

Segurança e Privacidade em Sistemas de
Armazenamento e Transporte de Dados

Curso de Mestrado e de Especialização em Sistemas de Dados e
Processamento Analítico

Braga, 2007

Índice

1. Introdução

Hoje em dia, a noção de anonimato ou a sua ausência, numa transacção comercial é tido por um número cada vez maior de pessoas como um aspecto muito importante na sua vida.
Até à algum tempo atrás considerava-se que a aquisição anónima de um bem e/ou serviço só era possível utilizando o dinheiro no seu suporte físico (o papel).

Porque a sua utilização é hoje muitas vezes impraticável dada a forma como hoje se realizam muitas transacções (por exemplo as transacções resultantes do comércio on-line), é necessário encontrar um seu equivalente electrónico. É a este seu equivalente, baseado em *tokens* anónimos, que é conhecido como *digital cash* (*electronic cash, e-cash, d-cash)*.

Este pode ser dividido em 2 categorias:

- Baseados em *Tokens/Coins*
- Baseados em Contas

Como exemplos de sistemas baseados em *Tokens/coins* destacamos: o tradicional dinheiro em papel, cartões de telefone pré-pagos ou mesmo os selos do correio. Estes sistemas permitem manter o anonimato dos seus utilizadores, sendo virtualmente impossível a sua identificação.

Por outro lado, os meios de pagamento tais como cheques, cartões de crédito e/ou transferências bancárias, que se baseiam no conceito de contas, na génese do seu funcionamento necessitam da completa identificação dos seus intervenientes (utilizadores) e das operações (transacções) por estes realizadas.

As pessoas gostam do dinheiro na sua forma física (em papel), porque se tem a noção física de deter "algo", porque ele é "fácil" de transportar e porque não é necessário recorrer a uma instituição financeira para depositar o dinheiro recebido antes de o voltar a utilizar em nova transacção.

No entanto, este tipo de dinheiro transporta germes, pode ser roubado ou perdido, não existindo nenhum tipo de compensação nestas condições.

Já os cartões de crédito apresentam como grande vantagem comparativa com o dinheiro em papel, o facto de reduzirem o risco da sua perda, mas como principal desvantagem, igualmente comparativa, a perda de privacidade na sua utilização.

O advento da internet estimulou o forte crescimento de serviços electrónicos, com especial destaque para o comércio on-line. Com este, é criada a necessidade de meios de pagamento cada vez mais eficientes, sofisticados e seguros.

Transmitir um número de cartão de crédito por uma rede aberta, como é hoje a internet, de forma a efectuar uma compra on-line, é de facto uma acção com um risco elevado. Quer pelo facto de existir a possibilidade de a comunicação ser escutada com sucesso, bem como pelo facto dos sistemas informáticos do fornecedor poderem ser alvo de um ataque, possibilitando assim que os números de cartão de crédito por ele conhecido possam ser utilizados de forma fraudulenta.

2. Digital Cash o que é?

Uma vez que o *digital cash* resulta de uma modelação do sistema de pagamentos baseado em papel, veremos neste capítulo que algumas das propriedades deste último se encontram igualmente no seu "*irmão*" presente no mundo virtual.

No fundo, o *digital cash* mais não é do que uma *moeda/Token*, ao qual lhe é atribuído um determinado valor monetário e que é transmitida entre duas partes segundo um determinado protocolo. Esse protocolo reveste-se de alguma complexidade e deve permitir a troca de mensagens autenticadas, sem que no entanto seja possível a rastreabilidade dessas mensagens.

Para melhor entender os objectivos e também como se controlam alguns dos riscos inerentes ao *digital cash* vejamos o seguinte exemplo:

" A *Alice* dona de uma de uma empresa de construção civil e obras públicas, stands automóveis e parques de estacionamento, quer transferir dinheiro sobre a forma de *digital cash* para *Bob*, vereador de uma câmara com quem *Alice* tem já diversos negócios, a fim de este mover influências para que um novo negócio de *Alice* envolvendo terrenos camarários seja aprovado. No entanto, *Alice* e *Bob* não podem correr o risco de serem descobertos por *Eve,* jornalista que se dedica à investigação de casos de corrupção. Desta forma *Bob* pode depositar o *digital cash* recebido na sua conta, sem que o banco saiba quem é *Alice*. No entanto, se *Alice* subornar um outro vereador com o mesmo "*pedaço*" de *digital cash* utilizado para subornar o vereador *Bob,* o banco irá detectar e saberá quem é *Alice*. Se *Bob* tentar depositar duas vezes o mesmo suborno de *Alice*, mas em contas diferentes, o banco irá detectar, mas *Alice* irá permanecer anónima."

É esta possibilidade de permanecer anónimo que melhor distingue o *digital cash* do *digital money*, uma vez que este último guarda um conjunto de pistas que permitem a rastreabilidade.

Alguns dos protocolos utilizados pelo *e-cash* pressupõe a utilização de um dispositivo de armazenamento, onde são colocados os valores, controlado pelo utilizador. Fazendo uma analogia com o dinheiro em formato papel, o conjunto do dispositivo juntamente com a informação que representa o *e-cash* é conhecido como *carteira electrónica* (*electronic wallet*).

2.1. Propriedades

Reconhecimento/Aceitação

Isto significa que o dinheiro é facilmente reconhecido como verdadeiro pelo(s) outro(s) interveniente(s) na transacção..

Segurança

Isto significa que o digital cash não pode ser copiado e/ou reutilizado, bem como negada a sua utilização. De forma a atingir estes objectivos, existe um conjunto de riscos relacionados com a falsificação que devem ser acautelados. Esses riscos dividem-se em duas classes:

1. Falsificação do *token* que representa a *moeda*.
   Neste caso o falsificador, sem efectuar nenhum levantamento bancário, simula o *token* emitido por um determinado banco.
2. Utilização múltipla da mesma *moeda*/*Token*
   Uma vez que uma *moeda/token* mais não é do que um conjunto caracteres, onde se encontra, entre outra informação, o número de série da *moeda/token; o seu valor;* Assinatura digital da instituição financeira, esta é facilmente duplicada, triplicada, etc…

De forma a combater o risco de falsificação aplicam-se técnicas de autenticação das partes envolvidas e de integridade da mensagem. Como exemplo das primeiras temos a assinatura digital e das segundas as funções de hash de sentido único.

As técnicas de autenticação utilizadas visam pois 3 aspectos fundamentais existentes nas comunicações electrónicas (e consequentemente nas transacções efectuadas neste meio)

1. Identificação das partes
   As partes devem ter a certeza que quem está do "outro lado" é quem diz ser.
2. Integridade da Mensagem
   As partes necessitam certificar-se que as mensagens que enviam chegam inalteradas ao seu destinatário. Imaginemos uma mensagem que contêm uma ordem de pagamento de 100€. Se a integridade da mensagem não for garantida, o

destinatário da ordem de pagamento poderia alegar que esta teria o valor de 10.000€

3. Não repudiação

   Para protecção das partes envolvidas de forma a que nenhuma delas possa ter negado a participação na transacção.

Como forma de combater a utilização abusiva de uma *moeda/token* as entidades financeiras mantêm uma base de dados com todas as *moedas/tokens* já utilizados.

Em situações em que o protocolo utilizado recorre a carteiras electrónicas, estas assentam em plataformas *tamper-resistance* onde o utilizador não pode manipular a informação que lá se encontra armazenada.

Transmissível

Isto significa que a moeda/token pode ser reutilizado no momento seguinte à transacção, sem necessidade de previamente ser "trocado"/depositado numa instituição financeira, ou seja, diz-se que um sistema de pagamento é transmissível, sempre que quem recebe uma dada moeda/token a pode utilizar de seguida em outra transacção sem primeiro ter a necessidade de a depositar numa instituição financeira.

O normal ciclo de vida e uma moeda/Token transmissível, terá o aspecto que se representa de seguida:

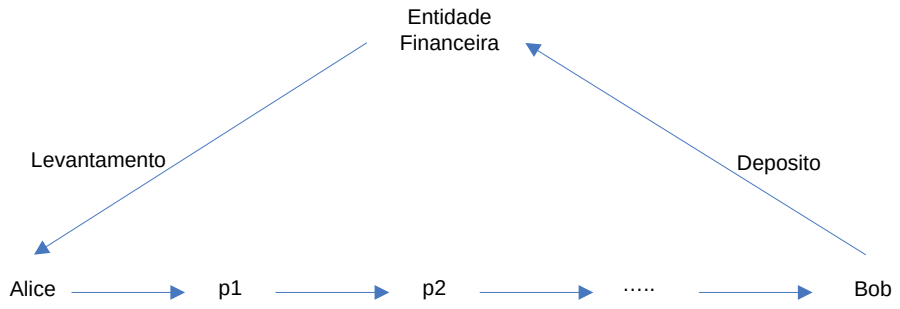

Figura 1 - Ciclo de vida de uma *moeda/token* transmissível

Privacidade

Esta é uma definição que não é consensual na indústria. Para uns, privacidade é sinónimo de protecção da comunicação, tal como a sua encriptação, a fim de evitar que esta possa ser "escutada", para outros, como David Chaum, é sinónimo de anonimato do cliente perante o

fornecedor e a não rastreabilidade do pagamento por parte do banco, de forma a que este não consiga saber qual o dinheiro que foi utilizado num pagamento específico.

Não rastreável

A impossibilidade de "desfazer" uma transacção com o intuito de chegar ao indivíduo originador da transacção.

Anonimato

É a impossibilidade do originador da transacção ser conhecido quer pelo vendedor de bens/serviços, quer pela instituição financeira emissora moeda/token utilizada na transacção, ou pela entidade financeira utilizada pelo vendedor para "trocar" a moeda/token.

Portabilidade

A segurança e utilização do dinheiro digital é independente da sua localização física. O dinheiro digital pode ser transferido entre diferente redes de computadores e diferentes sistemas de armazenamento.

Divisível/Indivisível

No contexto do digital cash nem sempre as moedas/tokens detidas representam a exacta importância necessária na transacção. Muitas vezes representam um valor em excesso, pelo que a capacidade de dividir as moedas/tokens existentes em valores mais pequenos de forma a proporcionar troco é conhecida como Divisibilidade. Segundo T. Okamoto, nem sequer o dinheiro em papel consegue satisfazer esta propriedade.

Esta propriedade alcança-se utilizando moedas/tokens que podem ser subdivididos em importâncias mais pequenas, mas cujo total representa o valor da moeda/token original.

Formas de Pagamento

Os sistemas de pagamento baseados em Digital cash implementam uma de duas estratégias:

Off-line

Esta estratégia significa que o Bob só submete a moeda/token da Alice à instituição financeira para verificação e depósito, depois da transacção ter lugar.

Isto significa que a transacção pode ter lugar a qualquer hora sem existir necessidade da intervenção de uma terceira parte (por exemplo, uma instituição financeira) na transacção.

Apesar de este ser o sistema mais prático e mais barato para os seus utilizadores, é também aquele que é mais susceptível ao problema da múltipla utilização de moedas/tokens.

Devido a este problema de segurança, este sistema está vocacionado para pagamentos de baixo valor.

On-line

Esta estratégia significa que o Bob, antes de aceitar o pagamento de Alice, submete para validação a uma terceira parte (por exemplo, uma instituição financeira), a *moeda/token* fornecida por esta.

A segurança adicional desta implementação comparativamente à implementação off-line vocaciona-a para pagamentos onde seja necessário elevados níveis de segurança.

Porque estes sistemas devem ser capazes de verificar a credibilidade dos pagadores perante as lojas, é quase impossível manter a anonimato dos utilizadores.

2.2. Vantagens

Hoje em dia, com a profusão de utilização de cartões de crédito, transferências de dinheiro on-line e com o advento dos débitos direitos, é muito fácil às instituições financeiras e aos comerciantes on-line traçarem o perfil dos seus clientes, e inclusive utilizar essa informação para aumentar, muitas vezes de forma moralmente reprovável, os seus rendimentos.

Neste contexto, o anonimato e a privacidade numa compra, ou utilização de um bem/serviço, é cada vez mais considerado uma necessidade. É pois com este objectivo que se desenvolveram muitos dos protocolos de *digital cash*.

Para além desta necessidade de anonimato, a questão da segurança põem-se também muitas vezes quando falamos de comércio on-line. O roubo de informação relativa a cartões de crédito é um dos problemas que se colocam a este nível. A cifra das mensagens onde esta informação circula, o seu armazenamento igualmente cifrado, bem como a criação de cartões

de crédito/débito virtuais são algumas das preocupações que os protocolos nesta área pretendem resolver.

2.3. Desvantagens

Esta ideia da transmissibilidade é uma ideia que parece simples, uma vez que esta característica é inerente à forma de utilização habitual do dinheiro em papel. No entanto, na versão digital do dinheiro, são vários os problemas que se levantam:

- Por cada "transmissão" da *moeda/token* esta cresce em tamanho, devido à necessidade de armazenar informação relativa a cada uma das transmissões. Só desta forma é possível as instituições financeiras protegerem-se contra o fenómeno da utilização múltipla da mesma *moeda/token*.

  Esta necessidade tem como consequência a limitação do número máximo de vezes que uma mesma *moeda/token* é transmissível.
- A lavagem de dinheiro e a fuga aos impostos são ainda mais difíceis de detectar, uma vez que não existem disponíveis registos das transacções.
- Cada transmissão atrasa a detecção de falsificações da *moeda/token* bem como da múltipla utilização da mesma *moeda/token.*
- Um utilizador pode reconhecer uma *moeda/token* se já a viu em outro pagamento. Neste caso existe uma perda, mesmo que parcial, do anonimato.

Comparativamente com o sistema em papel, o anonimato que se pretende obter com um sistema de *digital cash* tem vários outros lados da "moeda", potencialmente mais extensos que no seu similar em papel:

- Lavagem de dinheiro
- Chantagem
- Falsificação

Vejamos um exemplo, em que a chantagem

3. Alguns algoritmos/técnicas de cifra

3.1. RSA

Este é um algoritmo que implementa o conceito de criptografia de chave pública definido por *Diffie & Hellman*. Isto significa que as chaves são formadas aos pares, uma para cifrar e outra para decifrar, não sendo exequível a geração de uma chave a partir da outra.

Foi implementado em 1978 por Rivest, Shamir e Adleman e a flexibilidade e a velocidade de processamento tornam este algoritmo adaptado quer para *assinaturas digitais* quer para *encriptação*.

O conceito matemático por detrás do RSA é o problema de *factorização de inteiros* e em [Almeida, '07] conseguimos obter uma descrição do funcionamento deste algoritmo:

1. Inicialização (produção do par de chaves)
   - Geram-se dois números primos grandes p,q (faz-se n=p*q, logo $\varphi(n)=(p-1)*(q-1)$)
     Para máxima segurança, p e q devem ser do mesmo tamanho.
   - Considera-se um valor *e* tal que seja primo relativo a $\varphi(n)$ (i.e. $mdc(e,\varphi(n))=1$).
   - Calcula-se *d* como a inversa de *e* no grupo multiplicativo $Z^*\varphi(n)$, i.e. $e*d=1 \bmod \varphi(n)$.

   Chave para cifrar: (n,e) **Chave para decifrar**: (n,d)
2. Utilização (operações de cifra e decifragem)
   - Ambas as operações são a exponenciação modular.

   Cifra do texto limpo x ($0 \leq x < n$) com chave (n,e):

   x^e **mod n**

   Decifragem do criptograma y ($0 \leq y < n$) com chave (n,d)

   y^d **mod n**

Segurança RSA

Segundo alguns autores, como por exemplo [Farsi, '97] defendem que a segurança deste algoritmo se baseia na dificuldade em factorizar 2 primos suficientemente grandes. Segundo [Schneier, '96], tecnicamente esta afirmação não é completamente verdade. De facto, conjectura-se que a segurança deste algoritmo se baseia neste problema, mas matematicamente ainda não se provou que é necessário factorizar *n* para calcular *m*, a partir de *c* e *e*.

A segurança deste algoritmo está também directamente relacionada com o tamanho da chave. Quanto maior for esta, melhor é o nível de segurança, mas também é menor a performance do algoritmo. Segundo os RSA Labs, hoje em dia considera-se seguro, para fins corporativos uma chave com o tamanho 1024 bits, e para aplicações onde o par de chaves necessita atingir um nível de segurança particularmente exigente, recomenda-se a utilização de chaves com o tamanho 2048 bits. Um exemplo de um par de chaves desta dimensão, poderá ser aquele utilizado pelo topo da hierarquia (*root)* de uma entidade de certificação. Para informação menos valiosa, a utilização de chaves de 768 bits revela-se suficiente, pois não são ainda conhecidos algoritmos capazes de "quebrar" uma chave desta dimensão.

3.2. Assinaturas Digitais

As assinaturas digitais foram propostas em 1976 por Whitfield Diffie [Farsi, '97]. Estas aparecem devido à necessidade de encontrar um equivalente às assinaturas em papel existentes. Estas só devem poder ser produzidas por uma pessoa/instituição e verificadas por qualquer pessoa/instituição.

Tal como as assinaturas em papel, a sua equivalente digital deve cumprir as seguintes funções:

- Autenticação

  Como as assinaturas digitais são compostas por um par de chaves, o provado utilizado para cifrar e outro, o publico utilizado para decifrar. Como só uma mensagem assinada com uma chave privada, pode ser decifrada e verificada utilizando a chave pública. Desta forma, só o detentor da chave privada é que pode criar a mensagem.

- Integridade

Seja K_c a chave privada e Alice, e K_d a sua chave pública. Quando Alice envia uma mensagem assinada a Bob, ela envia o par *(M,E(K_C(M))*, sendo *M* a mensagem e *E()* uma função de hash de sentido único.

Bob verifica a integridade da mensagem, comparando o resultado da decifragem do criptograma (utilizando K_d com chave) com a mensagem recebida.

- Não repúdio

  Suponhamos que Bob recebe uma mensagem assinada por Alice e que esta mais tarde nega ter enviado essa mensagem. Bob pode recorrer a uma *terceira parte*, e esta obtendo a chave pública de Alice, pode verificar a validade da assinatura. Desta forma, a assinatura pode ser utilizada para comprovar a origem da mensagem.

3.3. Blind signatures

Este conceito de assinatura mascarada foi introduzido por David Chaum, e é um tipo especial de assinatura digital, onde o conteúdo da mensagem é mascarado/dissimulado antes de ser assinado.

A forma mais fácil de explicar o processo de assinar digitalmente um documento mascarado/dissimulado, será fazendo uma analogia com o tradicional sistema em papel:
A Alice leva um documento até ao notário Bob. A Alice pretende que o conteúdo do documento permaneça desconhecido de toda a gente, incluindo Bob. Para atingir esse propósito, Alice coloca o documento dentro de um envelope e fecha esse envelope. Uma pequena parte do documento é visível pela janela desse envelope. Bob coloca o seu selo na parte visível do documento comprovando a autenticidade do mesmo. Quando se utiliza uma assinatura mascarada, são utilizadas técnicas de criptografia para substituir o envelope e o selo. Alice encripta o documento digital (análogo ao processo de colocar o documento dentro do envelope) e Bob coloca uma assinatura digital no documento que se encontra no envelope. Para verificar a autenticidade do documento, é verificada a assinatura digital.

Na realidade, e voltando a um raciocínio mais formal, estes sistemas podem ser pensados como 2 sistemas de assinatura digital, que funcionam de forma comutativa.

Segundo [Chaum, ‘98], existem três funções que estes sistemas devem respeitar:

1. A função *a'* que efectua a assinatura, conhecida unicamente pela entidade que irá colocar a assinatura digital, e a correspondente inversa *a* de domínio público, de tal forma $a(a'(x)) = x$, sendo que *a* não dá nenhuma pista sobre *a'*.
2. Um função comutativa *c* e a sua inversa *c'*, sendo estas funções unicamente do conhecimento do dono do documento a assinar, de tal forma que $c'(a'(c(x))) = a'(x)$ sendo que *a'* não fornece nenhuma pista sobre x.
3. Um predicado de verificação *r*, redundante, que verifica se existe redundância suficiente para tornar impraticável procura de uma assinatura válida.

Graças a estas propriedades, quem assina não conhece, ou conhece unicamente de uma forma parcial, a mensagem original. A assinatura mascarada pode ser verificada sobre a mensagem original, não mascarada, da mesma forma que um assinatura digital "normal".

Este tipo de assinaturas utiliza-se normalmente quando quem origina a mensagem e quem a assina são partes distintas e onde é necessário preservar a privacidade do originador da mensagem. Neste caso o receptor sabe que a mensagem recebida é autêntica e fidedigna, apesar de não saber quem é o seu originador.

As blind signatures são também utilizadas para impedir que quem assinou uma mensagem mascarada seja capaz de reconhecer a respectiva mensagem desmascarada, no caso de ser necessário verificá-la. Esta característica é extremamente importante em situações em que o cliente de um banco quer preservar a sua privacidade perante o próprio banco.

As blind signatures podem ser implementadas utilizando esquemas de criptografia assimétrica de chave pública tais como o RSA e o DSA. Utilizando o RSA, a implementação de este tipo de assinatura, pode ser descrito da seguinte forma:

1. Alice escolhe o factor *f* utilizado para mascarar/dissimular a mensagem *m* de forma que $mdc(f, n) = 1$, onde (n, e) é a chave pública do banco e (n,d) a sua chave privada.
2. Alice envia a seguinte mensagem ao banco:

   $m' = mf^e \pmod n$, sendo que *m* representa a mensagem original
3. O banco de Alice assina a mensagem recebida da seguinte forma:

   $s' = (m')^d \bmod n = (mf^e)^d \bmod n$
4. Alice retira o factor multiplicativo *f*

   $s = s'/f \bmod n$

5. Alice utiliza, para pagar as suas contas:

$$s = m^{d}$$

Uma vez que f é um valor aleatório, o banco de Alice não é capaz de determinar m. Desta forma não é capaz de ligar a assinatura ao pagamento de Alice.

3.4. Cut-and-Choose

Este protocolo foi utilizado pela primeira vez, em criptografia, por Michael Rabi no ano de 1978, e baseia-se em conceitos estatísticos.

A razão para o seu nome, prende-se com a similaridade com o protocolo clássico utilizado para dividir igualmente qualquer coisa [Schneier, '96].

1. Alice divide "a coisa" a meio
2. Bob escolhe uma das metades para ele
3. Alice fica com a restante metade.

É pois de todo o interesse para Alice dividir equitativamente, uma vez Alice não sabe qual a metade que Bob vai escolher.

Mais tarde, este tipo de protocolos, será também conhecido como *Zero-knowledge Proofs.*, pois permitem a Bob provar a Alice que ele conhece um segredo sem no entanto o ter de revelar.

3.5. Divisão de Segredo (Secret Splitting)

Existe um conjunto de situações em que a divisão de um segredo em partes "mais pequenas", é a melhor situação. A mensagem só consegue ser reconstruída se todas as m partes estiverem reunidas.

Para melhor entender a necessidade de um protocolo deste género façamos mais uma vez uma analogia com o mundo que todos nós conhecemos.

Suponhamos que a Coca-Cola, numa perspectiva de mercado global, necessita de distribuir pelas suas fábricas o segredo da receita que está na base do famoso refrigerante. Se essa receita for conhecida na sua totalidade por algum dos seus funcionários, em qualquer uma das

fábricas, para que no dia em que esse funcionário abandonar a empresa ou se transferir para a sua arqui-rival, o segredo passar a ser por esta conhecido.

Se a entrega do segredo completo a uma pessoa não resolve o problema, então se a Coca-Cola tiver a capacidade de dividir este segredo por duas pessoas, de forma a que nenhuma delas por si só detenha o segredo e só com junção da parte detida por cada uma delas seja possível reconstruir o segredo, então não existe mais o perigo de o segredo ser revelado quando uma delas sair da empresa.

Na realidade, o segredo não será "partido" a meio. Aquilo que será feito é uma operação matemática de XOR, conforme [Schneier, '96] nos descreve, com a personagem *Trent* a dividir o segredo com *Alice* e *Bob*:

1. *Trent* gera uma sequência aleatória de bits *R*, com o mesmo comprimento da mensagem *M.*
2. *Trent* efectua um XOR entre a mensagem *M* e a sequência *R* obtendo assim *S*

   $M \oplus R = S$
3. *Trent* entrega a *Alice* a sequência *R* e a *Bob* a sequência *S.*

Para *Alice* e *Bob* reconstruírem o segredo, só necessitam de efectuar um XOR entre as sequências que possuem.

$R \oplus S = M$

Esta técnica se aplicada correctamente é absolutamente segura. Cada "pedaço" por sí só é absolutamente inútil. Na verdade, *Trent* quando cifra a mensagem está no fundo a aplicar o conceito de uma cifra *on-time-pad* (a sequência aleatória *R*).

Este protocolo tem ainda a vantagem de ser facilmente escalável para três, quatro, cinco, …, n partes. Bastará no fundo fazer *n-1* XORs. Vejamos então também em [Schneier, '96] como é que este protocolo funcionaria, para $n = 4$:

1. *Trent* gera as sequências aleatórias de bits *R*, *S* e *T,* todas elas com o mesmo comprimento da mensagem *M.*
2. *Trent* efectua três XORs com as sequências geradas com o objectivo de gerar *U.*

   $M \oplus R \oplus S \oplus T = U$
4. *Trent* entrega a *Alice* a sequência *R*, a *Bob* a sequência *S,* a *Carol* a sequência *T,* e a *Dave* entrega *U*.

Para reconstruir a mensagem, *Alice*, *Bob*, *Carol* e *Dave* devem trabalhar em conjunto, efectuando igualmente o XOR das suas mensagens.

$$R \oplus S \oplus T \oplus U = M$$

Na realidade, *Trent* tem o poder absoluto e pode fazer aquilo que quiser. Pode entregar a qualquer um dos presentes uma qualquer "algaraviada" e afirmar que é uma parte do segredo, sendo que ninguém o saberá até que tentem reconstruir o segredo.

No entanto, este protocolo padece de um problema. Imaginemos que se perde alguma das partes e o *Trent* não está disponível. Numa situação como esta, a fabrica da Coca-Cola não poderia voltar a produzir até *Trent* estar novamente disponível…

3.6. Protocolo Bit Commitment

Este protocolo foi desenvolvido para as situações em que é necessário garantir que uma das partes não altera a resposta dadas previamente.

Por exemplo, uma das partes *(Alice)* tem o poder de, no inicio da sessão de bolsa, saber qual a cotação de uma determinada acção no final dessa mesma sessão. Para o provar, *Alice* coloca a cotação que essa acção virá a ter no final da sessão num ficheiro e cifra esse ficheiro. Entrega então a cifra à outra parte (*Bob*) e no final da sessão de bolsa entrega igualmente a chave para decifrar a cifra. Dessa forma *Bob* sabe se *Alice* lhe mentiu ou não.

O conceito deste protocolo é este, o problema é que *Alice* pode fazer batota, e possuir mais do que uma chave, entregando a *Bob* chave que irá dar a cotação correcta.

De forma a corresponder a esta necessidade e a garantir que não é feita "batota" por uma das partes intervenientes, foram desenvolvidos 3 tipos diferentes de protocolo [Schneier, '96].

a. *Usando criptografia simétrica*

    1. *Bob* gera uma sequência aleatória de bits, *R* e envia-a a *Alice*
    2. *Alice* gera uma mensagem com a resposta pretendida *b* e a sequência *R* enviada por *Bob.* Cifra a mensagem assim obtida e envia de volta a *Bob.*

$$E_k(R, b)$$

Aqui acaba a parte de *commitment* do protocolo. *Bob* é incapaz de decifrar a mensagem, uma vez que *Alice* não lhe enviou a chave, não tendo por isso a possibilidade, até *Alice* considerar oportuno, de saber a resposta de *Alice*.

3. *Alice* envia a chave a *Bob.*
4. *Bob* decifra a mensagem. *Bob s*ó considera válida a resposta de *Alice*, se encontrar a sua sequência *R* na mensagem.

b. Usando funções de sentido único

1. *Alice* gera duas sequências aleatórias de bits, R_1 e R_2.
2. *Alice* gera uma mensagem que contêm R_1 e R_2, bem como a resposta pretendida *b*.

   (R_1, R_2, b)

3. *Alice* submete a mensagem à função de sentido único, e envia o resultado bem como uma das sequências aleatórias geradas a *Bob*

   $H(R_1, R_2, b), R_1$

A mensagem enviada por *Alice* funciona como o *commitment* do protocolo. *Bob* não tem possibilidade de determinar a resposta *b* de *Alice,* uma vez que a função aplicada por *Alice* é uma função de sentido único, por isso não invertível.

4. *Alice* envia a *Bob* a mensagem original

   (R_1, R_2, b)

5. *Bob* submete a mensagem recebida à função de sentido único, e compara o resultado e R_1 com a mensagem recebida em 3.

   Se os resultados coincidirem, a resposta enviada é válida.

A vantagem comparativa de este protocolo em relação ao anterior é que neste *Bob* não tem necessidade de enviar nenhuma mensagem.

c. Usando gerados de sequências pseudo-aleatória de bits

1. *Bob* gera uma sequência aleatória de bits, $R_{\boldsymbol{B}}$ e envia-a a *Alice.*

2. *Alice* utiliza a sua resposta como inicialização do gerador de sequências pseudo-aleatórias. Então, para cada bit de R_B, *Alice* envia a *Bob* uma de duas possibilidades:
    a. O output do gerador no caso do bit, na posição *i* de R_B ser 0, ou
    b. O resultado do XOR do output do gerador com a posição *i* da sequência de *Alice*, no caso de a posição *i* de R_B ser 1

Quando *Alice* considera ser a altura de revelar a sua resposta, o protocolo continua.

3. *Alice* envia a *Bob* a sequência utilizada par a inicialização do gerador.
4. *Bob*, completa o passo 2. de forma a confirmar que *Alice* está a agir de boa fé.

3.7. Entidade Certificação

Como a utilização de um certificado digital implica a utilização de criptografia assimétrica, sempre que *Bob* receber uma mensagem de *Alice* assinada digitalmente, *Bob* necessita de obter, de forma segura, a chave pública de *Alice*. No entanto, se *Bob* não conhecer *Alice* nem a sua chave pública, necessita que lhe seja fornecida por alguém que lhe garanta a integridade da chave pública de *Alice*, bem como que aquela chave é de facto a chave de *Alice*.

O conceito de Entidade de Certificação nasce então destas necessidades. É pois ela a responsável por fornecer, de forma fiável, as chaves públicas de uma dada entidade. Cabe ainda à entidade de certificação a emissão dos certificados digitais, onde se associa a uma dada entidade a sua chave pública. É também da responsabilidade da entidade de certificação e revogação dos certificados digitais.

Um certificado digital é um documento electrónico assinado digitalmente, emitido por uma terceira parte considerada universalmente como sendo de confiança, a tal entidade certificadora, e que tem como principal função fornecer uma forma de verificar a identidade de uma entidade. Pode-se considerar que os certificados são os equivalentes digitais à carta de condução ou ao cartão de crédito.

4. Protocolos

4.1. A proposta de Schneier

A descrição do protocolo aqui apresentado, segundo [Schneier, '96] é apenas um dos protocolos existentes para *digital cash*.

Neste protocolo são tidos em consideração aspectos como a segurança e a privacidade descritos no capítulo 2.1. Curioso é que este protocolo se preocupe principalmente em evitar a fraude sobre entidade financeira emissora/receptora do *digital cash*, não protegendo de igual forma os restantes intervenientes da transacção. Afinal, este protocolo em algumas aspectos não passa de um mimetismo daquilo que os sistemas de cartões de débito/crédito fornecem no dia à dia da maioria das pessoas.

4.1.1. Descrição

a) A *Alice* prepara *n* ordens de pagamento anónimas, para um determinado montante.
   Cada uma destas ordens contém uma sequência de caracteres X, aleatória, única e de um tamanho suficientemente grande para tornar negligenciável a hipótese de encontrar outra sequência Y idêntica.
   Em cada uma das ordens de pagamento, devem ser gerados *n* pares de *bit strings* de identidade, I_1, I_2, … , I_n. Cada um destes pares, deve ser gerado da seguinte forma:
   Alice cria uma sequência de caracteres, onde coloca o seu nome, morada e qualquer outro tipo de informação requerida pelo seu banco. Depois *Alice* procede à divisão em duas partes dessa sequência utilizando o protocolo de partilha de segredo. Para finalizar aplica a cada uma das partes da sequência o protocolo *Bit-Commitment*.
   Por exemplo, I_{25} consiste em 2 partes: I_{25_E} e I_{25_D}. No caso de ser pedido à Alice para "abrir" cada uma das partes, o conteúdo dessa parte pode ser imediatamente verificado, pois foi previamente submetido ao protocolo *bit-commitment*.
   Qualquer um dos pares, i.e., I_{25_E} e I_{25_D} e não I_{25_E} e I_{26_D} revela a identidade de Alice.
   Cada uma das ordens de pagamento, deve respeitar a seguinte estrutura:

Montante

Sequência única: X

Bit strings de identidade: $I_1 = (I_{1_E}$ e $I_{1_D})$

$I_2 = (I_{2_E}$ e $I_{2_D})$

…

$I_n = (I_{n_E}$ e $I_{n_D})$

b) *Alice* mascara todas as *n* ordens de pagamento, utilizando o protocolo para as blind signatures entregando de seguida todas as *n* ordens ao banco.

c) O banco pede à Alice para desmascarar *n-1* das ordens de pagamento (aleatoriamente) e confirma que elas se encontram bem formadas. O banco verifica o montante, a sequência única e pede à Alice para revelar as *Bit Strings* de identidade.

d) Depois do banco confirmar que Alice não o tentou enganar, assina a ordem de pagamento que ainda se encontra mascarada, devolve esta ordem à Alice e desconta o montante da conta desta.

e) A Alice desmascara a ordem de pagamento e utiliza-a com o comerciante.

f) O comerciante verifica a assinatura digital do banco, assegurando-se desta forma que a ordem de pagamento é legítima.

g) O comerciante pede a Alice que para aleatoriamente revelar quer o lado direito quer o lado esquerdo de cada um dos pares que constituem a ordem de pagamento. Para o fazer, o comerciante gera aleatoriamente uma sequência de *n-bits* (b_1, b_2, …, b_n).

h) Alice revela o lado esquerdo ou o lado direito de I_i, consoante b_i seja 0 ou 1.

i) O comerciante deposita a ordem de pagamento no banco.

j) O banco verifica a assinatura e confirma na sua base de dados se a sequência única de caracteres X já foi depositada em algum momento anterior. Se o não foi, é depositado na conta do comerciante o montante da ordem de pagamento apresentada e coloca a sequência única de caracteres X e toda as *bit*-strings de identidade na sua base de dados.

k) Se a sequência única de caracteres X já existe na base de dados do banco, este recusa o pagamento da ordem de compra. Depois compara as *bit-string* de identidade existentes na ordem de pagamento com aquelas que se encontram armazenadas na base de dados. Se são iguais, o banco sabe que o comerciante falsificou a ordem de pagamento copiando-a. Se forem diferentes, o banco sabe que a pessoa que comprou a ordem de pagamento a copiou, utilizando-a múltiplas vezes. Uma vez que o segundo comerciante, que aceitou a ordem de pagamento em questão, gerou, de uma forma igualmente aleatoria, uma sequência de *n-bits* à qual Alice teve de submeter as suas *bit-strings* de identidade, e esta sequência gerada pelo segundo comerciante é diferente da sequência gerada pelo primeiro

comerciante, pelo menos numa posição *k*, então o banco é possuidor pelo menos da b_{k_E} e da b_{k_D} identificando assim a Alice.

4.1.2. Segurança

Este é de facto um protocolo espantosamente simples, mas ao mesmo tempo igualmente espantoso na forma como evita a fraude.

Para melhor percebermos o funcionamento deste protocolo em termos de segurança, vejamos a sua construção passo a passo.

Por cada ordem de pagamento assinada digitalmente pelo banco, este pede à Alice que preencha *n* ordens de pagamento. Em todas estas ordens de pagamento, Alice sabe que deve preencher o mesmo montante em todas elas. De forma a manter o anonimato da ordem de pagamento, o banco terá de assinar digitalmente uma da *n* ordens sem a verificar, devolvê-la à Alice e no final descontar esse valor na conta da Alice (passos *a.* a *d.*).

O problema põe-se em como é que a banco sabe que a Alice não o enganou dizendo que a ordem de pagamento era no montante de 100€ quando aquela que o banco assinou digitalmente, era de facto de 10000€.

Na realidade o banco confia que a ordem de pagamento que assinou sem abrir é no montante de 100€, e baseia esta sua convicção no método *Cut-and-Choose*. Como o banco escolhe de forma aleatória as *n-1* ordens de pagamento que vai verificar, a Alice tem unicamente *1/n* por cento de hipóteses de enganar o banco.

Para resolver o problema da utilização múltipla do mesmo *digital cash* é pedido pelo banco à Alice, que esta gera uma sequência de caracteres única com um determinado tamanho. Desta forma o banco estará capacitado para verificar se aquele *digital cash* em particular foi ou não utilizado anteriormente (passos *a.*,*b.*e *j.*).

De forma a identificar quem é o *"batoteiro"*, o banco pede à *Alice* que crie *n bit-strings* de identidade para cada uma das ordens de pagamento que vai submeter ao banco (passos *a.*). Posteriormente essas *strings* serão utilizadas para identificar quem está de má fé na transacção. (passos *g., j.* e *k.)*

Se a Alice estiver de má fé com o comerciante, então o protocolo pode terminar logo no passo *h)* no caso de esta se recusar a responder ao desafio colocado pelo comerciante.

Se a *Alice* ou o comerciante tentarem a falsificação do *token* que corresponde ao *digital cash* em questão, a assinatura digital do banco deixa de estar válida e o banco (no caso de ser o comerciante o falsificador) ou o comerciante (no caso de ser Alice a falsificadora) podem detectá-lo.

A forma como o anonimato da Alice se mantêm baseia-se nas propriedades das blind signatures. A Alice mascara as ordens de pagamento que envia para o banco, e o banco assina digitalmente uma delas, sem saber o seu conteúdo e assina "por cima da máscara" (passo *b.* e *c.*). Quando a Alice utiliza a ordem de pagamento, desmascara esta primeiro (passo *e.*). Quando a ordem de pagamento chega ao banco, ele não tem forma de estabelecer o relacionamento com Alice.

4.1.3. Pontos Fortes

Este protocolo funciona *off-line*.

Apesar de este protocolo não impedir a falsificação dos *tokens* utilizados, ou a sua utilização múltipla, os prevaricadores sabem que serão apanhados, residindo precisamente aí uma das vantagens deste protocolo.

4.1.4. Pontos Fracos

Apesar de este protocolo ter a possibilidade de ser utilizado *off-line* ele funcionará melhor se for utilizado *on-line*, apesar da degradação de performance que possa daí advir.
Outro dos seus pontos fracos reside na *Eve*. Se ela *"escutar"* a comunicação entre *Alice* e o comerciante e se conseguir chegar ao banco antes do comerciante, então ela será a primeira a depositar *"aquele" digital cash*. Sendo *Eve* a primeira, o banco aceita o *digital cash* por ela entregue, e pior, quando o comerciante tentar depositar o dinheiro, será identificado como "batoteiro". Se *Eve* roubar a Alice o *digital cash* que ela possui e o gastar primeiro que ela, então é Alice que passa por "batoteira". Dado que este protocolo tenta preservar o anonimato de Alice, não existe solução para prevenir este problema.
A segurança deste protocolo reside na segurança com que Alice e o comerciante protegem as ordens de pagamento que estão em seu puder.

A crescente utilização deste tipo de protocolo pode colocar problemas de eficiência/performance do lado das instituições financeiras. Como estas têm de armazenar os dados relativos a todas as ordens de pagamento já utilizadas, o espaço de armazenamento utilizado pelas bases de dados será um problema cada vez maior. Será também cada vez

menos eficiente a verificação por parte dos bancos, da validade da ordem de pagamento que lhe é apresentada.
Este protocolo não respeita a transmissibilidade do *digital cash* uma vez que no final de cada transacção, este tem de ser devolvido ao banco.

4.2. A proposta de Chaum

A aproximação proposta por David Chaum tem por objectivo respeitar um dos principais objectivos do *digital* cash: o anonimato. Para isso o banco deve desconhecer a identidade do originador da transacção.
Para o conseguir, é utilizada uma implementação RSA do protocolo *blind signatures* bem como o protocolo *Cut-and-choose*.
Desta forma, o banco colocará o seu certificado digital numa ordem de pagamento, sem na realidade conhecer o conteúdo dessa ordem. Desta forma, o banco não será capaz de reconhecer a ordem de pagamento quando esta for depositada para pagamento, não sendo por isso capaz de relacionar o levantamento com o depósito. A única coisa que será capaz de reconhecer é que esta é uma ordem de pagamento válida.
De forma a prevenir a utilização múltipla do *digital cash*, o banco controla o identificador do *token* em questão.
Esta proposta é muito semelhante aquela proposta por Schneier em 4.1., diferindo em:

- No matemática aplicada no desafio a *Alice*, o que permitirá identificar *Alice* no caso de ser ela a batoteira e tentar utilizar mais do que uma vez o mesmo *token*.
- No facto de o banco só pedir para revelar *n/2* das ordens de pagamento geradas por *Alice*.

4.2.1. Pontos Fortes

Se o utilizador não utilizar

4.2.2. Pontos Fracos

É o valor *n/2* que determina a possibilidade de fraude, pois ao contrário da primeira proposta em que *n-1* das ordens de pagamento são inspeccionadas pelo banco, aqui unicamente o são *n/2*.

Tal como o protocolo anterior, este protocolo poderá ter problemas de performance, bem como não respeita a propriedade da transmissibilidade.

5. Produtos existentes

Já é possível encontrar um vasto leque de soluções no mercado, mas aquelas que mais sucesso tem obtido são soluções baseadas em cartões de crédito virtuais, ou em carteiras electrónicas.

Muitas das soluções apresentadas baseiam-se também no conceito de pré-pagamento.

5.1. Portugal

5.1.1. MBnet

É uma solução que replica para o meio digital a forma de funcionamento de um cartão de crédito, não fosse esta solução fornecida por um conjunto de entidades financeiras. È uma solução estritamente nacional e são aceites detentores de Cartões Visa, Mastercard e American Express.

O conceito foi pensado especialmente para disponibilizar pagamentos seguros no comércio on-line. Nesta situação, substitui com vantagem o cartão de débito/crédito dos seus utilizadores.

Estas vantagens centram-se todas na área de segurança, sendo de destacar:

- A emissão de cartões virtuais, com prazos de validade curtos e montantes pré-definidos, de uma só utilização

- Definição de um montante máximo, diário, disponível para as compras on-line.

Esta solução funcionam por intermédio de um TPA virtual (Terminal de Pagamentos Automáticos Virtual) . Esta interface é uma interface segura, baseada num "*browser*" e usando sessões *SSL*.

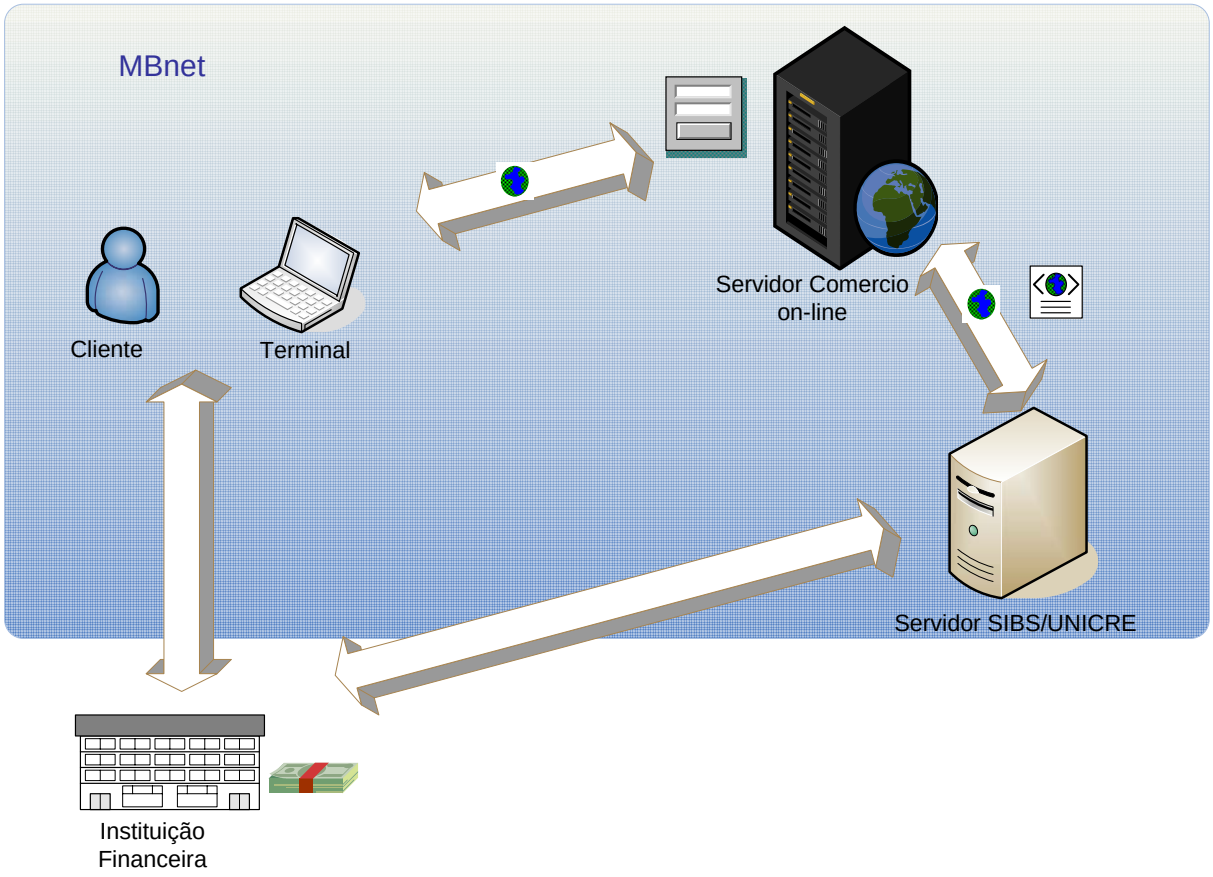

Figura 2 – Esquema de funcionamento do MBNet (Baseado em [Cruz et al.. '98])

Os passos para a utilização do sistema de MBNet, podem ser descritos da seguinte forma:

- Um cliente que pretenda utilizar o sistema MBNet, deve primeiro registar-se num ATM (*Automatic Teller* Machine) ou na instituição financeira onde lhe foi atribuído para cada cartão de débito e/ou crédito (Vias, Mastercard, American Express). Desta forma é atribuido um *user* e uma *password*, através dos quais passa a ser possível realizar compras on-line, sem nunca utilizar os dados dos seus cartões convencionais.
- A validação dos dados de cada transacção é efectuada em *Realtime/On.Line* pelo Servidor
- UNICRE/SIBS, e os montantes envolvidos nas compras efectuadas ficam cativos na conta do cliente.
- Cabe ao comerciante gerir as Ordens de crédito e débito aos Titulares de cartão.
- É uma Plataforma de comércio electrónico totalmente segura, pois é efectuado pelo Servidor SIBS/UNICRE a autenticação do cliente e a validação dos dados do cartão.
- A ligação entre a Loja Virtual do Comerciante e o Servidor UNICRE/SIBS é efectuada através da constituição por parte do Comerciante de um form em https com os elementos do carrinho de compras "apontado" para um URL do Servidor UNICRE/SIBS.

A entidade de certificação utilizada entre a Loja Virtual do Comerciante e o Servidos da UNICRE/SIBS é a Multicert.

5.1.2. PMB

Este é um sistema que foi descontinuado pela SIBS no final de 2005, sendo que se tratava de um cartão pré-pago, que pelas suas características entra dentro da categoria das *carteiras electrónicas*, pois os *tokens* encontravam-se armazendos num dispositivo inviolável (neste caso o *smartcard*) não tendo o utilizador nenhuma forma de interagir directamente com o seu dispositivo e/ou software lá instalado.
Na realidade, este sistema apesar de permitir um anonimato em relação ao detentor de um bem e/ou serviço perante o comerciante, tal como o sistema de dinheiro em papel, já o mesmo não se verificava em relação à

5.1.3. Yo! Card

O Yo! Card é um cartão pré-pago recarregável, podendo ser utilizado dentro da rede Visa, (inclusivé ATM em qualquer parte do mundo) e no comércio electrónico com toda a segurança.

O carregamento deste cartão pode ser efectuado num ATM, via *homebanking* ou no balcão de uma agência bancária, através de um pagamento de serviços ou por transferência bancária.

Este cartão não necessita de estar associado a nenhuma conta, funcionando igualmente sob um conceito de *carteira electrónica*..

Ao contrário de vários protocolos de *Digital Cash*, neste caso é possivel a transferência de valores *pessoa-a-pessoa (P2P)*, sem ser necessário que esse montante seja primeiro depositado no banco e depois novamente convertido a *digital cash*. A realidade, no entanto é bem diferente daquilo que é percepcionado, pois tratando-se de uma instituição financeira que gere este cartão (Banco Espírito Santo), na realidade ocorre transferência de dinheiro entre contas, e o banco pode sempre rastrear a forma como o dinheiro é gasto, não existindo assim anonimato e/ou privacidade na utilização deste cartão. De lembrar que um dos objectivos do *digital cash* é precisamente preservar o anonimato e a privacidade de quem o utiliza…

5.2. Resto do mundo

5.2.1. Paypal

Este é um sistema que permite enviar e receber dinheiro por

Como vantagens apresentadas por este sistema, destacam-se:

- A possibilidade de pagar com cartão de crédito, directamente da conta bancária ou por intermédio do saldo *Paypal* sem expor os números (cartão, conta, etc…) ao comerciante.
- Facilidade de utilização, uma vez que não é necessário reescrever a informação financeira ou endereços.
- Isenta de comissões nas compras ou transferências de dinheiro efectuadas via e-mail.
- Os dados relativos à entidade financeira nunca são enviados ao comerciante.
- Quando é efectuada uma compra on-line, o cliente só necessita de introduzir o seu e-mail Paypal, não circulando por isso informação financeira na net, na altura da transacção.

Partindo do princípio que tanto o cliente como o comerciante se encontram registados no Paypal, os passos seguidos numa transacção são:

1. O cliente acede ao site do comerciante e escolhe os produtos que deseja colocando-os no carrinho de compras.
2. Clica no logo Paypal e o seu browser é redireccionado para o site da Paypal, mantendo no entanto o *look&feel* do site do comerciante. O cliente introduz a informação necessária à transacção (o seu endereço e-mail)
3. O Cliente volta a ser redireccionado para o site do comerciante e o comerciante vê transferido para a sua conta o montante respeitante à transacção.

Como o cliente é perfeitamente anónimo para o comerciante, este não tendo informação do cliente também não possui os custos de a armazenar de forma segura.

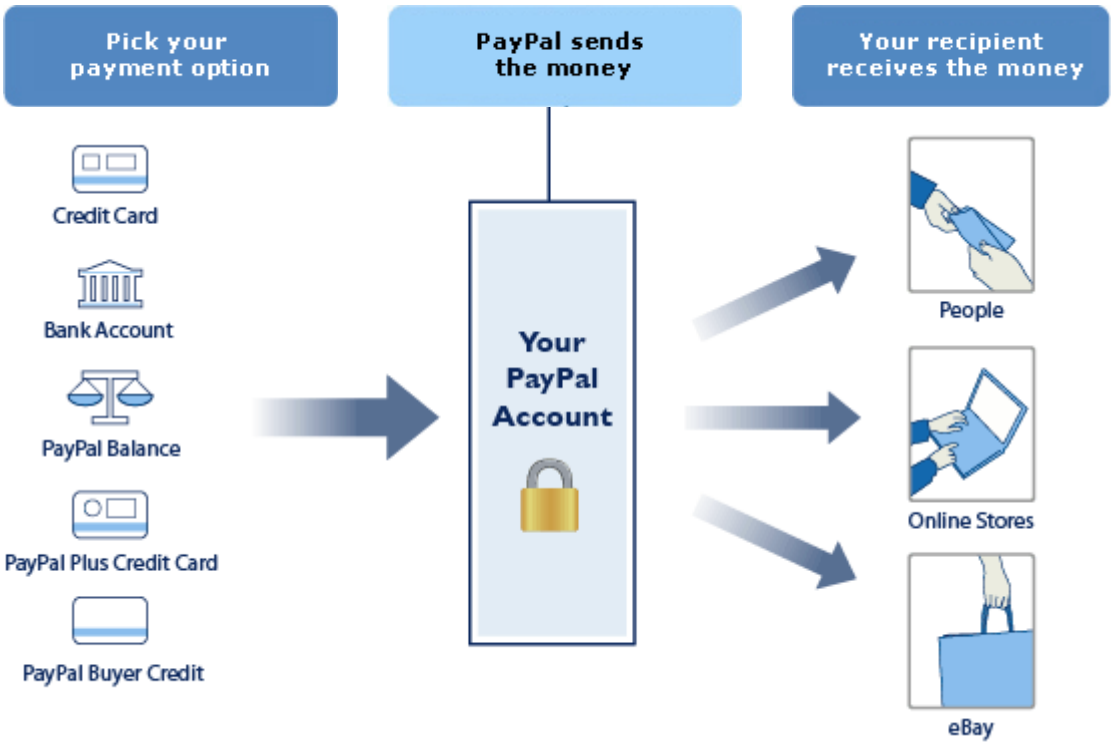

Figura 3 – Esquema de funcionamento do sistema Paypal

Uma vez que o email é uma das formas preferênciais de contacto com o cliente, a Paypal autentica os mails fazendo uso de certificados digitais emitidos por Entidades de Certificação reconhecidas.

Para utilizar a Web, o cliente deve utilizar um browser que implemente, pelo menos, o protocolo Secure Sockets Layer 3.0 (SSL).

Todas as mensagens trocadas serão encapsuladas no protocolo SSL utilizando uma chave de 128 bits.

5.2.2. Digicash

Esta é a empresa de David Chaum, responsável pela proposta de protocolo apresentada em 4.2. e que pretende possibilitar o pagamento completamente anónimo de transacções, não possibilitando desta forma, nem aos bancos, a rastreabilidade da transacção até ao seu originador.

O funcionamento desta solução gira à volta das *blind signatures*. Esta aproximação permite a rastreabilidade do utilizador de uma forma controlada por este, uma vez que sé ele pode revelar informação que o permita ligar a uma dada transacção.

O sistema montado por esta empresa permite então que o utilizador defina se revela ou não a sua informação. Para que o utilizador revela a sua informação, basta que este não utilize o

factor de *blinding*, transformando assim a sua assinatura numa normalíssima assinatura digital.

Para que um cliente (a nossa famosa *Alice*) possa utilizar este sistema, é necessário que ele possua uma conta aberta num banco que emita *ecash*. Os passos a seguir serão:

1. O banco fornece à Alice o software necessário, a *carteira electrónica (ewallet),* de forma a que esta proceda à sua instalação no seu computador.
2. Alice inicia o processo para levantar o *ecash*, dando instrução à sua *ewallet* para produzir séries de 100 números aleatórios e assinando-os todos recorrendo a processos de criptografia de chave pública. Os números são "multiplicados" pelo factor de *blinding* e são enviados para o banco
3. O Banco verifica a assinatura de forma a certificar-se que o pedido foi efectuado pela Alice e que pode debitar a sua conta. O banco adiciona aos números enviados pela *Alice* a sua assinatura digital, criando assim um *token* que representa uma *moeda* no meio electrónico.
4. *Alice* retira o factor de *blinding* e armazena os *tokens* no sua *ewallet*
5. *Alice* efectua uma transacção junto de *Bob,* e a sua *ewallet* envia a este, o número de *tokens* necessários para prefazer o montante da transacção.
6. *Bob* entrega o ecash no banco a fim de este verificar se o(s) número(s) série(s) do *ecash* já foi utilizado anteriormente. Se o(s) número(s) já existir(em) na base de dados o(s) *token(s)* é(são) inválido(s). Se não existir(em), o(s) *token(s)* é(são) considerado(s) válido(s) e o(s) seu(s) número(s) de série adicionado(s) à base de dados e *Bob* vê o dinheiro a ser creditado na sua conta.

Sem a *blind signature* o sistema seria seguro, mas sem anonimato/privacidade, uma vez que as transacções efectuadas pela *Alice* poderiam ser a ela ligadas.

Este sistema não é completamente anónimo, uma vez que o comerciante responsável por uma dada transacção é sempre identificado. A *Alice* pode ser identificada, sempre que ela o pretenda, bastando para isso identificar o factor de *blinding* utilizado para um daod *token*.

A desvantagem primordial a apontar a este sistema reside no facto de ele ter sido pensado para funcionar *on*-line, apesar de ser possível fazê-lo *off-line*. Com a utilização crescente deste sistema e a necessidade de manter todos os identificadores alguma vez utilizados no *ecash* emitido armazenados na base de dados do banco, a sua performance degrada-se e a sua capacidade de processamento necessita continuamente de ser actualizada de forma a não degradar a sua performance do sistema.

Uma outra desvantagem pode ser apontada a este sistema A Alice deve sempre ter o *Tokens* (dinehiro) na *ewallet* na quantia exacta para a transacção. Isto pode significar que a *Alice* pode ter de armazenar na sua *ewallet* um número elevado de *tokens* nos mais variados montantes.

5.2.3. Mobipay

Este é um sistema muito interessante, onde um telemóvel e o seu cartão é utilizado para efectuar pagamentos. Esta é uma forma interessante de efectuar pagamentos, não tendo ainda nenhum dos operadores móveis em Portugal seguido esta moda, apesar de já disporem de todas as condições para isso, o cartão SIM/USIM necessário para utilizar o serviço.

Neste caso o telemóvel poderá funcionar como um potente e flexível carteira electrónica, permitindo efectuar pagamentos em restaurantes, lojas de rua, taxistas, comércio electrónico, máquinas de *vending*, recarregamentos de cartões pré-pagos de telemóveis, bilhetes de cinema, facturas de electricidade, etc…

É de facto impressionante a quantidade de diferentes serviços que é possível comprar, bem como o facto de ser um meio de pagamento vocacionado tanto para pagamentos de pequenos montantes como para montantes de dimensão mais elevada.

As transacções são rápidas e têm lugar em menos de 15 segundos.

Para utilizar este serviço disponibilizado pela Mobipay, os utilizadores deverão associar os seus meios de pagamento (cartões de crédito/débito, contas bancárias, cartões de telefone…)

Um cliente registado no Mobipay que queira efectuar o pagamento de uma compra realizada numa loja, só tem de seguir os seguintes passos:

1. Dar o seu número de telemóvel, ou em caso de querer manter o seu anonimato e/ou revelar o seu número de telemóvel, pode fornecer um aliás ao número ou um código de barras fornecido pela Mobipay no visor do seu telemóvel.
2. O comerciante insere o identificador do cliente no seu sistema e o cliente recebe um SMS a pedir autorização para ser efectuado o pagamento.
3. O cliente autoriza o pagamento e o dinheiro é transferido para a conta do comerciante, sendo debitado do meio de pagamento associado pelo cliente na Mobipay.

O caso de um pagamento on-line, o processo é diferente. Neste caso o cliente diz que o meio de pagamento é o Mobipay em lugar de dizer Visa ou Master Card, e a loja virtual envia um número de referência que será apresentada no computador do cliente e o número de cliente Mobipay para que o pagamento possa ser efectuado.

O pagamento em máquinas de *vending* funciona de forma similar ao pagamento via internet.

O pagamento de reservas, facturas, transferências, e recargas funciona sempre de forma similar. È unicamente necessário saber a referência Mobipay de quem vai receber o pagamento (em alguns casos basta saber o número de telefone do destinatário) e o processo segue sempre a mesma filosofia.

No final e em jeito de autenticação e não repudio do pagamento, o cliente Mobipay deve sempre introduzir a sua *password* Mobipay.

A segurança deste sistema baseia-se nos seguintes elementos:

- O número Mobipay, que é um número pessoal
  Este número assegura que a operação é conduzida pelo cliente Mobipay.
- Sistema passivo de ser bloqueado
  O cliente falhando a introdução do seu número Mobipay, o sistema é bloqueado
- As comunicações encontram-se cifradas pelos protocolos GSM, CDMA, TDMA
- Número de telefone associado ao número Mobipay
- Pin do telemóvel associado ao Mobipay
  É necessário introduzir este número para iniciar uma transacção Mobipay

Este é sem dúvida um sistema bastante interessante, onde o cliente consegue obter o anonimato perante o comerciante, mas onde continua a ser possível às entidades bancárias rastrearem a utilização do serviço.

6. Conclusões

Esta é uma área onde as propostas para protocolos para um verdadeiro sistema de *digital cash* existem, mas que na prática estes protocolos não se têm conseguido implementar.

Uma das dificuldades será sem nenhuma dúvida a dificuldade legal, pois este tipo de soluções facilita em muito, com total impunidade, actuações à margem da lei. Imaginemos o que seria os bancos puderem abrir contas completamente anónimas...

Por outro lado, do ponto de vista financeiro, a implementação de um sistema destes é caro. Não existindo entidades financeiras dispostas a financiar o sistema, este não conseguirá obter massa crítica para ser implementado.

Um dos campos onde de facto pode ser interessante este tipo de soluções, e onde o problema da utilização com fins ilegais não se coloca com tanta acuidade, é o campo dos micro-pagamentos. A compra de músicas, por exemplo, poderia ser um campo a explorar e onde a massa critica poderia facilmente aparecer.

Influenciados em conceitos do *digital cash*, vemos o aparecimento de soluções que visam a segurança e o anonimato parcial para transacções efectuadas on-line. Aqui apontam-se algumas soluções como o MBNet, o Paypal e uma num campo que tem começado a despontar nos últimos 5 anos, que é a área do m-commerce, com a Mobipay.

Na maioria das soluções que encontrei, mesmo aquelas não descritas neste relatório, a maioria das implementações recorre sempre ao conceito de *ewallets*.

Em relação a Portugal, e tirando a SIBS, o panorama é de uma verdadeira ausência de ofertas neste domínio do *digital cash*. Mesmo a proposta do MBNet não pode ser considerada uma verdadeira proposta nesta área, isto porque existe um conjunto de propriedades, referente ao *digital cash*, que não são respeitadas. Entre elas destaco a privacidade perante as instituições financeiras do originador das transacções (*Alice*), pois todas as transacções com este sistema são rastreáveis.

Esta ausência de propostas é reconhecida pelo Banco de Portugal no seu relatório de 2007 referente aos meios de pagamento utilizados em Portugal, onde se consegue notar a completa ausência de meios de pagamento na área do *e-money*. O mesmo relatório afirma que o meio

de pagamento que mais se aproximava a este conceito foi o PMB, mas que este meio de pagamento foi descontinuado, por falta de adesão, durante o ano de 2005.

Para finalizar esta minha conclusão, deixo aqui a minha insatisfação sobre o resultado final deste trabalho. Foi um trabalho que se estendeu durante demasiado tempo e onde mesmo assim não tive oportunidade de fazer um estudo comparativo com um conjunto de protocolos. Penso mesmo que esse poderia ser a grande mais valia de um trabalho destes, para além de perceber a problemática inerente a este tipo de área. Soube a pouco.

7. Bibliografia

[Almeida, '07] José Carlos Bacelar Almeida: "Cifras Assimétricas", Material bibliográfico da disciplina de Segurança e Privacidade de Sistemas de Armazenamento e Transporte de Dados, Curso de Mestrado em Sistemas de Dados e processamento Analítico, 2007
URL:
http://wiki.di.uminho.pt/twiki/pub/Education/Criptografia/CriptografiaMestrados/CriptoMICEI-6-PKey.pdf, acedido em 2007/03/14

[Bellare et al. '05] Mihir Bellare, Phillip Rogaway: "Introduction to Mordern Cryptography", Department of Computer Science and Engineering, University of California at San Diego http://www.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf, acedido em 2007/10/06

[BP, '07] Banco de Portugal: "Sistemas de Pagamentos em Portugal", Núcleo de Desenvolvimento de Sistemas de Pagamento, Banco de Portugal, 2007.
http://www.bportugal.pt/bank/payments/bbk2007_p.pdf, acedido em 2007/10/07

[Chaum. '98] David Chaum: "Blind Signatures for Untraceable Payments", Department of Computer Science, University of California, Santa Barbara. Springer-Verlag
http://dsns.csie.nctu.edu.tw/research/crypto/HTML/PDF/C82/199.PDF, acedido em 2007/09/06

[Cruz et al.. '98] Armando Cruz, Gil Costa, Paulo Guedes, Pedro Ferreira: "REDUNICRE – Plataformas Tecnológicas de Pagamento", Tecnologias de Comércio Electrónico, Mestrado em Sistemas de Informação, Universidade do Minho, 2004
http://papadocs.dsi.uminho.pt:8080/retrieve/84/Relatorio+TCE.pdf, acedido em 2007/09/23

[Farsi, '97] Mandana Jahanian Farsi: "Digital Cash", Master's Thesis ir Computer Science, Department of Mathematics and Computing Science, Göteborg University, 1997
URL: http://www.simovits.com/archive/dcash.pdf, acedido em 2007/09/06

[Froomkin, '96] A. Michael Froomkin: "Flood Control on the Information Ocean: Living With Anonymity, Digital Cash, and Distributed Databases", Pittsburgh Journal of Law and Commerce, 1996
URL: http://osaka.law.miami.edu/~froomkin/articles/oceanno.htm#ENDNOTE0, acedido em 2007/09/06

[Schneier, '96] Bruce Schneier: "Applied Cryptography: Protocols, Algorithms and Source Code in C", John Wiley & Sons, 1996.

REFERÊNCIAS WEB

http://en.wikipedia.org/wiki/Digital_cash

Página da wikipedia com informação relativa a e-cash. Consultada em 2007-10-05

http://en.wikipedia.org/wiki/Blind_signature

Página da wikipedia com informação relativa a blind signatures. Consultada em 2007-10-05

http://en.wikipedia.org/wiki/Digital_signature

Página da wikipedia com informação relativa a assinaturas digitais. Consultada em 2007-10-05

http://www.rsa.com/rsalabs/node.asp?id=2218
Página RSA Laboratories, onde se apresentam os valores recomendados para o tamanho das chaves para o algoritmo RSA, Acedido em 2007-10-05

http://dn.sapo.pt/2006/01/30/economia/sibs_e_bancos_acabam_portamoedas_mul.html~
Página notícias sapo sobre o desaparecimento do Porta Moedas Multibanco, Acedido em 2007-10-07

http://www.bportugal.pt/bank/payments/bbk2007_p.pdf
Relatório referente aos meios de pagamento. Acedido em 2007-10-07

http://www.mobipay.com/en/home.htm
Página inicial da Mobipay. Acedido em 2007-10-07

https://www.paypal.com/
Página inicial da Paypal. Acedido em 2007-10-07