Sistemas de Pagamento Eletrônico:
Classificação, Análise e Implementação

Lucas de Carvalho Ferreira

Dissertação de Mestrado

Sistemas de Pagamento Eletrônico: Classificação, Análise e Implementação

Este exemplar corresponde à redação final da Dissertação devidamente corrigida e defendida por Lucas de Carvalho Ferreira e aprovada pela Banca Examinadora.

Campinas, 5 de Novembro de 1998.

Prof. Dr. Ricardo Dahab (Orientador)

Dissertação apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Instituto de Computação
Universidade Estadual de Campinas

Sistemas de Pagamento Eletrônico: Classificação, Análise e Implementação

Lucas de Carvalho Ferreira

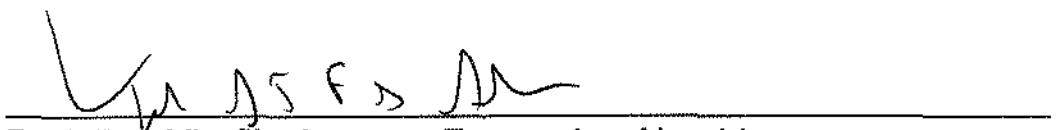
Novembro de 1998

Banca Examinadora:

- Prof. Dr. Ricardo Dahab (Orientador)
- Prof. Dr. Virgílio Augusto F. Almeida
  Depto. de Ciência da Computação, Universidade Federal de Minas Gerais
- Prof. Dr. Arnaldo Vieira Moura
  Instituto de Computação - UNICAMP
- Prof. Dr. Cláudio L. Lucchesi (suplente)
  Instituto de Computação - UNICAMP

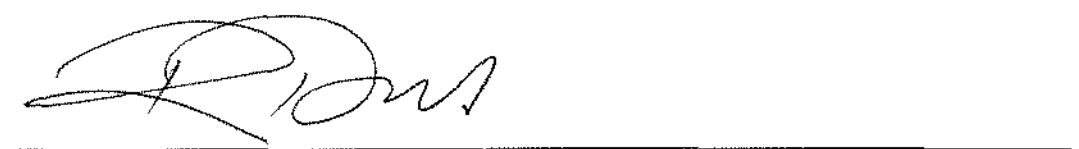
TERMO DE APROVAÇÃO

Dissertação defendida e aprovada em 05 de novembro de 1998, pela Banca Examinadora composta pelos Professores Doutores:



Prof. Dr. Virgílio Augusto Fernandes Almeida
DCC - UFMG

Prof. Dr. Arnaldo Vieira Moura
IC - UNICAMP



Prof. Dr. Ricardo Dahab
IC - UNICAMP

Prefácio

Esta dissertação trata de sistemas projetados para permitir a transferência de valor através de comunicação eletrônica, ou, simplesmente, *sistemas de pagamento eletrônico*. Estes sistemas terão fundamental importância no desenvolvimento do comércio na Internet e fora dela. Três aspectos desses sistemas são abordados: classificação, análise formal e implementação.

A classificação consiste de um esquema que descreve um sistema de pagamento através da sua tipificação, dos requisitos desejáveis, seu funcionamento e fatores de implementação. Este esquema permite analisar, comparar e melhor compreender os sistemas de pagamento. Durante o processo de análise, pudemos encontrar similaridades em diversas classes de tais sistemas.

A análise formal é feita pela aplicação de dois métodos adequados à análise de sistemas criptográficos. Em conjunto com a aplicação destes métodos formais, é apresentado um modelo que generaliza o funcionamento dos sistemas de pagamento baseados em cartões de crédito, e que facilita a compreensão deste tipo de sistema.

Utilizando o esquema de classificação proposto, implementamos um protótipo capaz de automatizar parte do processo de construção de sistemas de pagamento.

Agradecimentos

Dedico este trabalho à memória de meu avô Orlando, que, tenho certeza, o consideraria como mais um fruto de suas inúmeras realizações.

Agradeço a meu filho Pedro, maior felicidade de minha vida, e a minha esposa Gisele, que sempre me apoiou nesta empreitada.

Gostaria de agradecer a meu orientador, prof. Ricardo Dahab, pela paciência e compreensão quando precisei sair de Campinas por causa do nascimento do Pedro, e por ter me guiado no desenvolvimento deste trabalho.

Agradeço a meus colegas do Instituto de Computação da Unicamp e, especialmente, aos membros do Movimento dos Sem Tese (MST-4).

Sempre terei uma divida de amizade e gratidão para com os irmãos da República Falcão que me acolheram e sempre me ajudaram durante minha estada em Campinas.

Agradeço a minha família e, em especial, a meus pais e irmãos.

Sou grato ao Sr. Rodolfo M. Bacarelli pelas importantes opiniões a respeito do andamento deste trabalho.

Agradeço ao CNPq, à CAPES e à Comissão de Pós-Graduação do Instituto de Computação da Unicamp pela alocação de bolsas de estudo para a realização desta dissertação.

Conteúdo

Lista de Tabelas

Lista de Figuras

Capítulo 1

Introdução

A penny for your thoughts.

A Internet vem crescendo muito nos últimos tempos, principalmente por causa da popularização de seu segmento multimídia, chamado de *World Wide Web*. A facilidade de uso por parte do usuário final é um dos fatores que tem alavancado este crescimento, embora a grande publicidade que tem sido feita em relação à Internet fique por conta do uso da rede para fins comerciais, notadamente o varejo eletrônico.

Outros fatores que vem levando a um crescimento no uso da Internet são a possibilidade de disponibilização de bancos de dados pela rede e a diminuição do custo das comunicações, sejam elas pessoais ou profissionais. Pela Internet, uma empresa pode tornar disponível um banco de dados de produtos ou serviços, permitir que seus clientes tenham acesso a dados de suporte, onde estariam os problemas conhecidos no uso dos produtos da empresa, ou até permitir que fornecedores e clientes obtenham informações atualizadas de seus estoques. A comunicação pessoal via Internet tem se tornado muito popular por causa de seu baixo custo. A rede mundial permite que as pessoas usem recursos de correio eletrônico e teleconferência, que substituem as chamadas telefônicas com um custo menor.

O uso da Internet para realização de comércio eletrônico ainda não atingiu todo seu potencial por causa da preocupação, por parte dos seus usuários, com a segurança das transações que se realizariam na rede. As outras aplicações citadas já estão tendo um desenvolvimento maior porque existe uma boa gama de situações em que essas aplicações não requerem mecanismos muito seguros. Por exemplo, uma empresa que coloca um catálogo de produtos na Internet deve querer que todos sejam capazes de usá-lo, ou a comunicação pessoal entre dois indivíduos pode não requerer privacidade. Ainda assim, existem usos de teleconferência que requerem um nível de segurança maior do que seria possível hoje.

Na situação atual, são comuns notícias de invasões de sistemas de computadores ligados

à Internet e há consenso entre os especialistas de que os mecanismos de comunicação usados na rede não são suficientemente seguros para aplicações relativas a comércio eletrônico.

As estimativas de que a Internet tem potencial para que as cifras do comércio eletrônico atinjam a casa das centenas de bilhões de dólares por ano vem levando diversas organizações a se preocuparem em pesquisar e propor protocolos que garantam a segurança das transações no ambiente Internet.

1.1 Segurança de Informações

Ao longo dos anos, diversos métodos foram desenvolvidos para garantir a *segurança de informações*. Estes métodos consistiam, na maioria das vezes, em esconder as informações através da criptografia, ou ciframento, de mensagens ou documentos escritos em papel. O objetivo da segurança de informações limitava-se, portanto, a garantir a confidencialidade destes documentos e mensagens.

Com o advento dos computadores e a predominância de métodos digitais para armazenamento e transferência de dados, surgiram novas necessidades de proteção de informações. Assim, o que se entende hoje por criptografia e segurança de informações abrange um espectro muito mais amplo.

Por criptografia, entende-se todo um conjunto de técnicas (conceitos, algoritmos, protocolos), que tem base matemática e objetiva atender a uma série de requisitos da segurança de informações, que são tratados abaixo.

1.1.1 Requisitos da Segurança de Informações

A lista de requisitos considerada abaixo não é exaustiva. De fato, à medida que novas aplicações surgem, novos requisitos deverão ser considerados e atendidos. O objetivo principal de um sistema de provisão de segurança de informações é o atendimento destes requisitos (Esta discussão é inspirada em [20]):

Privacidade ou Confidencialidade: Ocultação de informações, via ciframento, de toda entidade sem autorização de acesso.

Integridade de Dados: Garantia de que as informações não tenham sido modificadas sem autorização (acidentalmente ou não).

Certificação de Origem: Garantia da identidade do remetente de uma informação.

Certificação dos Dados: Obtenção de um endosso oficial à existência daquela informação.

Validação Temporal: Verificação de que uma informação esteja dentro de seu prazo de validade.

Identificação: Obtenção da identidade correta de uma entidade.

Atribuição de Direitos: Maneira segura de atribuir formas de acesso ou propriedade sobre serviços ou recursos.

Confirmação da Prestação de Serviços: Emissão de um atestado de prestação de um serviço. Incluem-se aqui recibos de recepção de mensagens.

Autorização: Permissão oficial para prestação de um determinado serviço.

Autenticação de Dados: Garantia de autenticidade, sob vários aspectos, da informação. Um dos mecanismos usados para prover este requisito é a assinatura digital.

Anonimato: Ocultação da identidade de uma entidade envolvida em algum processo.

Controle de Acesso: Restrição do acesso a recursos/serviços apenas a entidades autorizadas.

Não-repúdio: Prevenção da negação por parte das entidades de sua participação em ações passadas. Este requisito também pode ser provido por assinaturas digitais.

Revogação: Retirada ou anulação de certificados ou autorizações.

Testemunho: Verificação, por terceiros, de algum aspecto da informação.

1.1.2 Soluções Criptográficas

No Apêndice A encontra-se uma introdução aos conceitos e técnicas criptográficos mais comuns, com os quais pode-se garantir quatro requisitos básicos da segurança de informações:

1. **Confidencialidade:** consiste em impedir que o conteúdo da informação chegue ao conhecimento de pessoas não autorizadas.
2. **Integridade de dados:** consiste na capacidade de detecção da modificação não autorizada dos dados.
3. **Autenticação:** certificação da identidade de uma entidade ou da origem de dados.
4. **Não Repúdio:** consiste em impedir que uma entidade negue sua participação em uma determinada ação.

Protocolos mais sofisticados para garantir todos os requisitos descritos na seção anterior podem ser construídos a partir das técnicas que garantem estes quatro pontos básicos. Um exemplo onde podemos perceber como a junção destas técnicas criptográficas pode ser usada para garantir outros requisitos da segurança de informações seria um sistema de dinheiro eletrônico. Neste caso, os quatro objetivos da criptografia são necessários. O sistema deve garantir confidencialidade nas transações, integridade dos dados, autenticação dos participantes da transação e não repúdio quanto à participação numa transação. As técnicas criptográficas usadas podem também garantir a existência de um recibo, que dependeria de técnicas de integridade e autenticação, além de garantias da impossibilidade de repúdio pelo emitente.

1.2 Comércio Eletrônico

Comércio eletrônico pode ser definido como: "Qualquer forma de transação de negócios em que as ações se dão por meio eletrônico". Dentro desta definição se encaixam o EDI (*Electronic Data Interchange*), a comunicação por fax, os códigos de barras, cartões de crédito e *smart cards* entre outros processos de negócios usados atualmente.

Apesar das técnicas tradicionais de comércio eletrônico terem evoluído e melhorado ao longo dos anos, e ainda haver uma grande perspectiva de melhora, o comércio via Internet tem sido a grande vedete, não só na imprensa, mas também nos planos das grandes empresas.

Nosso interesse será focalizado no uso da Internet como meio de comunicação e negociação entre consumidores, empresas e governos, sem contudo abandonar o uso das técnicas tradicionais de comércio eletrônico, como redes proprietárias. Focalizaremos também outras tecnologias que afetam nossa maneira de tratar o comércio de forma eletrônica sem ter de usar redes proprietárias diretamente. Assim, podemos identificar as seguintes categorias de comércio eletrônico, quanto às entidades participantes [36]:

1. **Empresa - Empresa:**
   Quando as empresas se comunicam usando computadores para fazer pedidos, receber faturas ou realizar pagamentos. Esta categoria tem estado operacional há vários anos através de EDI por redes proprietárias.

2. **Empresa - Governo:**
   Engloba toda a interação entre empresas e governos via computadores. Nos EUA, a Internet já é usada em concorrências públicas, e no Brasil para a entrega das declarações do Imposto de Renda. Esta categoria ainda está incipiente, mas deve se desenvolver rapidamente.

Oportunidades para os fornecedores	Benefícios para os consumidores
Presença a nível mundial	Escolha a nível mundial
Maior competitividade	Maior qualidade
Customização em massa	Produtos ou serviços personalizados
Encolhimento da cadeia de distribuição	Maior rapidez na resposta
Redução de custos	Redução de preços
Novas oportunidades de negócios	Novos produtos e serviços

Tabela 1.1: Oportunidades e benefícios do comércio eletrônico

3. **Empresa - Consumidor:**
   Corresponde ao varejo eletrônico. Esta categoria encontra-se em expansão graças à WWW.

4. **Consumidor - Governo:**
   Começa a ser uma alternativa para que os cidadãos possam manter em dia suas declarações de impostos, entre outras atribuições. Bons exemplos são as declarações de Imposto de Renda de 1997 e 1998, que puderam ser entregues pela Internet.

A Tabela 1.1 [36] mostra as oportunidades e benefícios possibilitados pelo uso do varejo eletrônico. Considerando apenas o varejo eletrônico, existem diversos modos de uso da Internet para fins de comércio:

1. Estabelecimento de contato inicial entre cliente e fornecedor;
2. Troca de informações entre cliente e fornecedor;
3. suporte pré e pós venda;
4. venda direta;
5. pagamento eletrônico;
6. distribuição de produtos sob forma eletrônica ou acompanhamento do processo de distribuição;
7. empresas virtuais.

Dentre os modos apresentados acima, apenas o pagamento eletrônico exige técnicas criptográficas específicas e será tratado em detalhes nos próximos capítulos. Os outros modos podem ser considerados como casos de comunicação multimídia entre usuários ou acesso a bancos de dados, se considerarmos apenas os requisitos de segurança.

1.2.1 O Comércio Eletrônico Hoje e no Futuro

Atualmente, já estão disponíveis diversas formas de comércio eletrônico, como sistemas de pagamento por cartões de crédito/débito, compras pela Internet, entre outros. Entretanto as técnicas de comércio eletrônico ainda não estão disponíveis para a maior parte de população, e têm portanto impacto social limitado.

Os fornecedores de tecnologias e serviços de comércio eletrônico direcionados ao varejo têm concentrado suas forças em duas áreas principais:

- Transações comerciais na Internet
- Sistemas de carteira eletrônica

Estas duas áreas tem potencial para revolucionar a maneira como é realizado o comércio, principalmente varejista, tradicional.

Já estão disponíveis diversas tecnologias que permitem utilizar a Internet para os diversos modos de comércio eletrônico que esta rede proporciona, incluindo aí sistemas de pagamento eletrônico. O grande problema hoje consiste em aumentar a aceitação e disponibilidade destas tecnologias e torná-las corriqueiras para uma grande parcela da população. Os sistemas de carteira eletrônica também já existem, mas enfrentam os mesmos desafios.

1.2.2 Pagamento Eletrônico

Já existem em funcionamento diversas maneiras de efetuar pagamentos usando apenas comunicação eletrônica. Entretanto, estas maneiras foram projetadas para transferências de grandes importâncias, o que aumenta os requisitos de segurança. Além disto, muitos dos sistemas existentes para transferência de valores eletronicamente foram implantados antes que o uso da Internet se tornasse uma coisa corriqueira. Por isso, estes sistemas fazem uso de redes proprietárias e especializadas, as quais têm um custo elevado.

Neste trabalho, focalizaremos os sistemas de pagamento adequados à realização de transações em redes abertas, como a Internet, ou sistemas que possam ser usados para o varejo sem o auxílio de redes especializadas.

Uma aplicação de comércio eletrônico desta natureza pode apresentar problemas de implementação ou segurança em diversos componentes do sistema de computação, como o software cliente ou servidor, sistema operacional, protocolos de rede, protocolo de pagamento, entre outros. O foco deste trabalho será no protocolo de transferência de valores, normalmente chamado de sistema de pagamento.

1.3 Notação

As convenções de notação usadas ao longo deste trabalho dizem respeito principalmente ao uso de técnicas criptográficas, como cifras, assinaturas digitais e *hashings* ou funções de espalhamento, que são detalhados no apêndice A.

As convenções são:

1. K_{ab}: é uma chave criptográfica para comunicação segura entre A e B; ou seja, a chave K pode ser usada para cifrar as mensagens trocadas entre A e B, supondo-se, portanto, que K seja do conhecimento de A e B somente.

2. K_a: é a chave pública de A em um par de chaves assimétricas; ou seja, a chave K_a pode ser usada para cifrar mensagens endereçadas a A.

3. K_a^{-1}: é a chave privada de A, fazendo par com K_a. A chave K_a^{-1} só é conhecida por A e pode ser usada para assinar as mensagens enviadas por A.

4. $\{M\}_K$: a mensagem M foi cifrada usando a chave K.

5. $H(M)$: corresponde ao resultado da aplicação da função de espalhamento H sobre a mensagem M.

As notações específicas serão apresentadas nos respectivos capítulos.

1.4 Resumo

A principal preocupação deste trabalho foi o estudo dos sistemas que permitem a realização de pagamentos por redes abertas, como a Internet. Estes sistemas tentam suprir as necessidades de segurança encontradas em redes deste tipo.

Muitos sistemas deste tipo já foram propostos ou implementados e, embora não exista um esquema amplamente aceito para classificar este tipo de sistema, já surge a necessidade de compará-los. As empresas têm de escolher um sistema de pagamento quando começam a implantar sistemas de comércio eletrônico e a realizar vendas pela Internet. Desta forma, aparece uma demanda por métodos de classificação que permitam estudar, classificar e comparar sistemas de pagamento. Um esquema com estes objetivos é apresentado no Capítulo 2. Este esquema é composto de quatro partes: tipificação, requisitos, funcionamento e aspectos de implementação.

O esquema do Capítulo 2 apresenta um modelo de funcionamento de sistemas de pagamento, que tenta generalizar as diferentes etapas das transações de pagamento. Este modelo de funcionamento é abrangente o suficiente para permitir a implementação de um gerador semi-automático de sistemas de pagamento, como é mostrado no Capítulo 2.2.

Dentro desta linha de análise e estudo dos sistemas de pagamento, é possível fazer uso de métodos que permitam provar que um método cumpre determinados objetivos, como autenticação ou responsabilização dos participantes. Dois métodos formais são apresentados no Capítulo 4: a lógica BAN e o *framework* de Kailar. A primeira permite estudar as características de autenticação em protocolos criptográficos e permite verificar o funcionamento daqueles sistemas em que os participantes são identificados. O *framework* permite verificar a capacidade de responsabilizar os participantes de uma transação.

Como principais resultados deste trabalho, ressaltamos o esquema de classificação e análise do Capítulo 2, o gerador de sistemas de pagamento da Seção 2.2, as características de alguns sistemas específicos ressaltadas pela análise formal do Capítulo 4 e as similaridades encontradas em algumas classes de sistemas de pagamento apresentadas no Capítulo 3.

Capítulo 2

Um Esquema para Análise de Sistemas de Pagamento Eletrônico

Nos últimos anos sugiram diversos sistemas de pagamento eletrônico, sendo que grande parte destes sistemas já foi implementada e está em fase de testes ou em funcionamento comercial. Não há, no entanto, uma metodologia que permita comparar as características destes sistemas. Alguns trabalhos foram desenvolvidos neste sentido [14, 2, 6, 32, 24, 1, 7, 25], mas não tratam todas as facetas dos sistemas, incluindo as suas características, seu modo de funcionamento e aspectos de implementação.

Neste capítulo, apresentamos um esquema que possibilita a análise e comparação de sistemas de pagamento eletrônico a partir de sua tipificação, requisitos, modo de funcionamento e aspectos de implementação. Apresentamos, no Capítulo 3, análises de alguns dos sistemas de pagamento eletrônico, escolhidos por serem considerados importantes ou representativos.

As definições apresentadas neste capítulo serão usadas no decorrer deste trabalho, mesmo que conflitem com as definições usadas por outros autores.

2.1 Descrição do Esquema de Classificação

Esta seção apresenta o esquema proposto para classificação de sistemas de pagamento eletrônico, primeiro caracterizando o tipo do sistema e enumerando seus requisitos. Em seguida, é apresentado o modelo de funcionamento destes sistemas, que permite uma melhor compreensão dos protocolos que compõem cada sistema e um resumo dos principais aspectos de implementação. A Figura 2.1 apresenta este esquema graficamente.

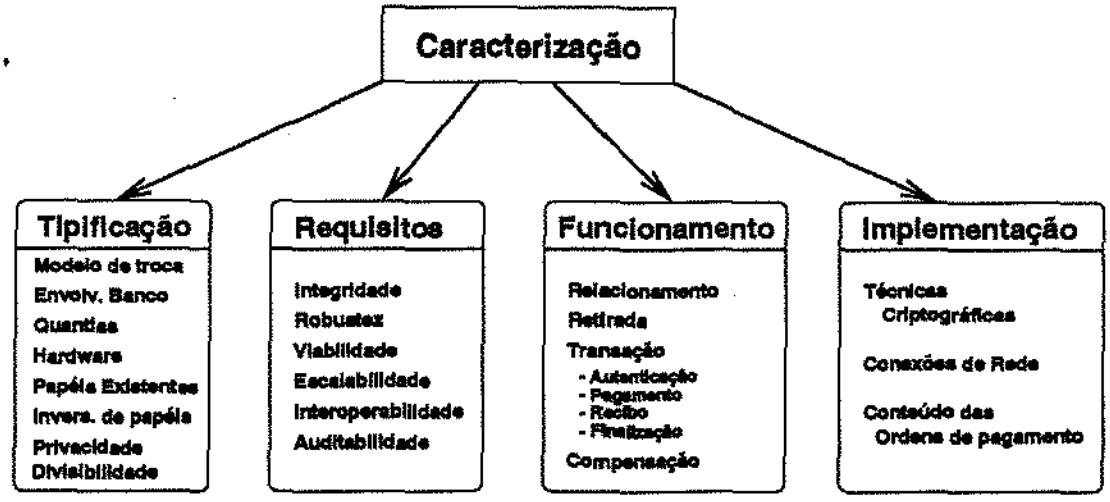


Figura 2.1: Os aspectos da caracterização dos sistemas de pagamento

2.1.1 Tipificação

Os sistemas de pagamento eletrônico apresentam certas características que nos permitem classificar e compreender seu funcionamento e aplicações. Estas características são:

- **Modelo de Troca** [7]
  - **cupons (troca direta)**: Sistemas em que as transações ocorrem pela transferência de cupons de valor predeterminado. Estes cupons funcionam como num sistema de notas e moedas. Num sistema de cupons, o usuário deve "comprar" seus cupons de uma entidade emissora antes de poder realizar transações. Sistemas baseados em cupons são algumas vezes chamados de sistemas *cash-like*.
  - **notacional (troca indireta)**: São sistemas em que as transações ocorrem através da atualização de saldos em contas mantidas junto a uma instituição financeira. Nestes sistemas, os usuários trocam documentos que autorizam a transferência de valores entre suas contas, o valor da transferência sendo determinado pelo usuário. O cheque é um bom exemplo de um sistema notacional, assim como os cartões de crédito. Estes sistemas são algumas vezes chamados de sistemas de débito-crédito.
  - **híbrido**: Sistemas que usam cupons e atualização de saldos. Um exemplo é o CAFE, que será estudado mais adiante.
- **Envolvimento da Entidade Controladora**

  - **on-line**: são os sistemas que necessitam do envolvimento da entidade controladora do sistema durante a realização da transação de transferência de valores. São sistemas adequados para a Internet, mas que incorrem em custos devidos às conexões adicionais necessárias. Alguns exemplos são os sistemas projetados especificamente para a Internet (e-cash, First Virtual, SET etc) e os sistemas de cartões de débito (Visa Electron, Cheque Eletrônico, B.I.S.).
  - **off-line**: alguns sistemas não necessitam que a entidade emissora seja contactada no momento da transação. Esta entidade deverá ser contactada em algum momento do futuro para que a transação seja efetivada; o recebedor do pagamento é capaz de verificar a validade da transação.

- **Quantias envolvidas**
  - **micropagamentos**: pequenos valores são trocados, desde milésimos de dólares (ou reais) até alguns poucos dólares (reais), tipicamante até US$ 5,00.
  - **pequenos e médios pagamentos**: valores que podem transitar na Internet com certa segurança, em geral de US$ 1,00 a US$ 500,00.
  - **grandes pagamentos**: grandes quantias, geralmente acima de US$ 500,00, que exigem um nível de segurança maior do que se consegue hoje na Internet.

- **Hardware necessário**
  - **dedicado**: faz uso de hardware especial, como *smart cards*.
  - **uso geral**: usa apenas computadores de uso geral. O usuário que já possui um computador não precisa adquirir nenhum equipamento especial.

- **Papéis envolvidos**: Os papéis desempenhados pelos participantes das transações. Em geral são pagadores ou recebedores (usuários) ou bancos, embora alguns sistemas façam distinção entre compradores e vendedores. Nos sistemas em que não há distinção entre usuários, usaremos os termos pagador e recebedor. Os termos comprador e vendedor serão usados nos sistemas em que for clara a distinção de quais participantes podem assumir determinado papel. As entidades controladoras instalam servidores para realizar as operações que lhes cabem.

- **Inversibilidade dos papéis**
  - **papéis fixos**: cada participante tem seu papel definido, seja vendedor, comprador ou banco. Para poder assumir dois papéis, o usuário tem que se cadastrar duas vezes, uma para cada papel desempenhado.

- **papéis variáveis:** o sistema permite que o usuário desempenhe papéis diferentes dependendo da situação, podendo assumir o papel de pagador ou recebedor de acordo com sua conveniência. Estes esquemas permitem naturalmente a transferência de valores entre usuários. Em geral, os sistemas não permitem que um usuário assuma o papel de banco.

- **Privacidade**
  - **existente:** o sistema permite preservar a privacidade dos participantes, em situações como compras anônimas, transmissões seguras ou proteção de informações críticas.
  - **inexistente:** o sistema não faz uso de técnicas criptográficas que garantam a privacidade das informações transmitidas. Caso seja necessário, protocolos externos devem ser usados.

- **Divisibilidade** Capacidade de substituir um cupom de valor alto por diversos cupons de menor valor. Estabelecemos quatro níveis de divisibilidade:
  1. o usuário é capaz de dividir os cupons.
  2. a entidade emissora pode ser contactada para trocar cupons pelo valor equivalente em cupons de menor valor.
  3. o sistema permite a devolução de troco.
  4. não há divisibilidade, ou trata-se de um sistema notacional.

2.1.2 Requisitos

Identificamos alguns requisitos desejáveis nos sistemas que analisamos. É preciso ter em mente, porém, que o contexto em que os sistemas serão usados deve ser levado em conta, para determinar quais destas características podem ser relevadas.

Integridade O sistema deve ser capaz de impedir que informações sejam alteradas por descuido ou de forma não autorizada. Em geral, os sistemas assumem que os servidores (se existirem) garantem a integridade das informações que armazenam.

Robustez O sistema deve ter características transacionais [7], que podem ser traduzidas pelas características ACID:

- Atomicidade: a transação acontece completamente ou não acontece. Existem três níveis de atomicidade:

1. Atomicidade monetária: a transferência monetária acontece sem a possibilidade de criação ou destruição de valor.
2. Atomicidade de entrega: garante a atomicidade monetária e garante que o pagamento só ocorre se ocorrer a entrega dos bens.
3. Entrega certificada: garante a atomicidade de entrega e permite aos participantes provar qual o contéudo da mercadoria (informação) entregue.

- Consistência: O sistema nunca deve estar num estado ilegal ou num estado em que os participantes tenham visões conflitantes do sistema.
- Isolamento: uma transação não interfere em outras.
- Durabilidade: o sistema deve ser capaz de retornar ao último estado consistente, mesmo em caso de falha em equipamentos.

Viabilidade econômica O custo das transações deve ser compatível com os valores trocados através do sistema.

Escalabilidade A inclusão de novos usuários no sistema não deve trazer uma queda de desempenho acentuada. O sistema deve permitir aumento do número de usuários e da quantidade de moeda envolvida sem que haja uma degradação acentuada de seu desempenho.

Interoperabilidade O sistema deve permitir a troca de moeda com outros sistemas. Por exemplo, deve ser possível trocar um cheque por papel moeda.

Auditabilidade O sistema deve permitir a realização de auditorias para detecção de falhas, fraudes ou comportamento inadequado de seus usuários.

2.1.3 Funcionamento

O funcionamento dos sistemas de pagamento eletrônico consiste, de forma geral, das seguintes etapas, ilustradas na Figura 2.2. É importante ressaltar que, em determinados sistemas, algumas etapas podem não existir.

1. **Relacionamento**
   Consiste do estabelecimento do relacionamento entre o usuário e a entidade do sistema financeiro responsável pela administração do sistema de pagamentos. Esta entidade é chamada de banco ou entidade emissora/controladora.

   O primeiro passo para que um usuário possa fazer uso de um sistema de pagamento eletrônico é se cadastrar junto à entidade que controla o sistema, o que pode envolver a abertura de uma conta, a aquisição de *software* ou a inicialização de mecanismos de autenticação, tais como a geração de chaves públicas.

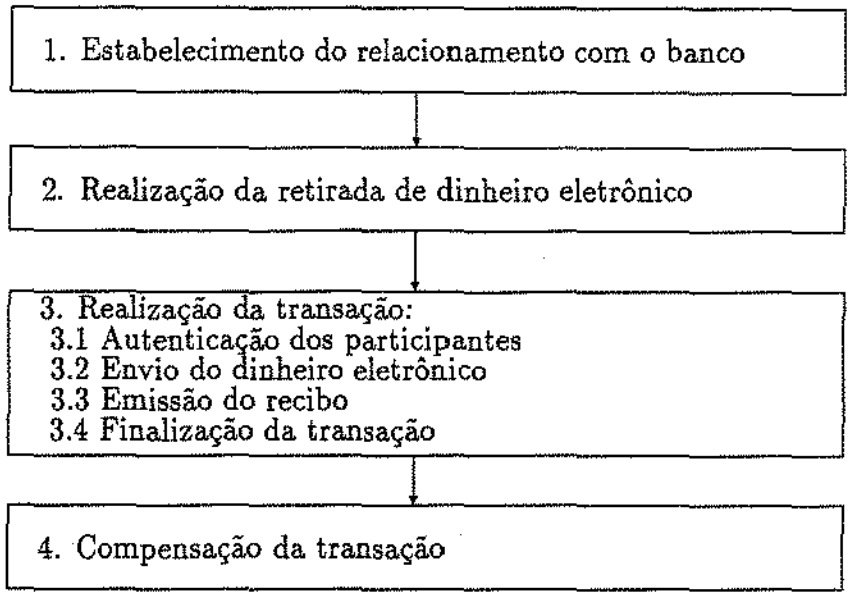


Figura 2.2: Esqueleto geral de funcionamento dos sistemas de pagamento

2. **Retirada**
   Nesta etapa, o usuário deve contactar a entidade emissora que o habilitará a realizar transferências de valores. Isto pode se dar pela transferência de moedas eletrônicas, carga de um *smart card* ou pela emissão de um certificado de crédito.

3. **Transação**
   Esta é a etapa principal do funcionamento dos sistemas, na qual ocorre a transferência de valor monetário.

   (a) **Autenticação**
   Pode ser necessário aos participantes ter a certeza de que estão em contato com outro usuário autorizado do sistema.

   (b) **Pagamento**
   O pagador envia o dinheiro eletrônico ao recebedor.

   (c) **Recibo**
   O recebedor envia ao pagador um recibo correspondente à transação. Em alguns protocolos, a entrega de bens de informação é realizada nesta etapa.

   (d) **Finalização**
   Estando os participantes de acordo quanto ao resultado da transação, esta pode ser finalizada sem problemas. Caso contrário, algum mecanismo de resolução

de disputas pode ser acionado.

4. **Compensação**
   O recebedor deve levar o dinheiro eletrônico ao banco para que este seja convertido em moeda real ou depositado em sua conta. Em alguns casos, o recebedor pode optar por receber a quantia em dinheiro eletrônico do mesmo tipo ou solicitar sua conversão em outro tipo de dinheiro eletrônico.

2.1.4 Aspectos de Implementação

Alguns aspectos do projeto de um sistema de pagamento eletrônico estão intimamente ligados à sua implementação. Na literatura, alguns sistemas foram especificados em detalhes, incluindo seus aspectos de implementação, enquanto outros não incluem este tipo de informação. Os aspectos de implementação mais importantes para a análise de um sistema de pagamento eletrônico são:

Técnicas Criptográficas Descreve os algoritmos criptográficos a serem usados no sistema, detalhando os seguintes itens:

- Cifras simétricas
- Cifras assimétricas
- Assinatura digital
- Certificados
- *hashings*

Mais detalhes a respeito de técnicas criptográficas podem ser encontrados no Apêndice.

Conexões de rede Quantidade e tipo das conexões de rede necessárias ao funcionamento do sistema. Qualquer estabelecimento de um canal de comunicação entre dois participantes é contado como uma conexão de rede, mesmo que seja usado para transmitir diversas mensagens.

Formato dos cupons ou ordens de pagamento Conteúdo e organização dos cupons ou ordens de pagamento usados no sistema devem ser apresentados.

2.2 Implementando um Gerador de Sistemas de Pagamento

Tendo um esquema para classificar e estudar sistemas de pagamento eletrônico que inclui um modelo de funcionamento, é necessário mostrar que este modelo é útil. Uma das

formas de fazê-lo é construir um programa que, usando do modelo, permita automatizar a geração de sistemas de pagamento.

O gerador de sistemas de pagamento descrito a seguir é capaz de combinar primitivas construídas para cada uma das etapas de funcionamento de um sistema de pagamento e gerar sistemas completos. A possibilidade de construção deste gerador, demonstrada através da implementação de um protótipo, é um bom indício de que o modelo é capaz de representar uma boa gama de sistemas de pagamento e também que o modelo tem utilidade não só para a análise mas também para o projeto de sistemas de pagamento.

2.2.1 Descrição do Gerador

O objetivo final do gerador é implementar os três componentes de um sistema de pagamento: os módulos para o pagador, o recebedor e o banco. Este módulos podem se utilizar dos serviços de outros módulos, permitindo assim a montagem de uma biblioteca de primitivas para sistemas de pagamento. O uso de módulos independentes possibilita também ao projetista do sistema implementar suas próprias primitivas.

Para o desenvolvimento do protótipo, foi escolhida a linguagem Java [15], cuja especificação atual inclui o JCE [15], ou *Java Cryptographic Extensions*, que é uma biblioteca de classes que fornecem acesso a diversas primitivas criptográficas, como cifras, métodos de assinatura, geração de chaves etc. Devido às restrições norte americanas quanto à exportação de *software* criptográfico, foi usada uma implementação alternativa desenvolvida pelo IAIK da Universidade de Viena [13].

O gerador consiste de uma aplicação que serve de interface com o usuário, recebendo as instruções relativas ao sistema a ser gerado e fornecendo as classes-base para cada um dos módulos. Estas classes-base realizam suas tarefas através de chamadas a funções disponibilizadas por classes contidas na biblioteca de primitivas.

Estas classes-base seguem o modelo de funcionamento de sistemas de pagamento descrito na Seção 2.1.3 e realizam uma chamada para cada um dos passos descritos no modelo. A classe da biblioteca a ser chamada é escolhida de acordo com as opções do usuário.

Se o usuário (projetista do sistema) desejar incluir primitivas que não façam parte da biblioteca, ele deverá editar o código gerado pelo programa e incluir as declarações de suas classes.

2.2.2 Biblioteca de Classes

Para este protótipo, implementamos uma biblioteca de classes bastante reduzida, embora significativa, já que possibilita a construção de sistemas simples que representam as principais categorias dos sistemas descritos na Seção 2.1.3. As classes implementadas

	Primitivas para		
	Pagador	**Recebedor**	**Banco**
Retirada	Withdrawal TokenWithdrawal		BankWithdraw TokenBankWithdraw
Autenticação	Authentication PKAuthentication	PayerAuthentication PKPayerAuthentication	
Pagamento	Payment TokenPayment NotationalPayment	ReceivePayment TokenReceivePayment NotationalReceivePayment	BankTransaction TokenBankTransaction NotationalBankTransaction
Recibo	Acknowledgement PKAcknowledgement	SendAcknowledgement PKSendAcknowledgement	
Finalização	Conclusion	PayeeConclusion	BankConclusion
Depósito		Settlement TokenSettlement NotationalSettlement	BankSettlement TokenBankSettlement NotationalBankSettlement

Tabela 2.1: Tabela de classes implementadas

são apresentadas na tabela 2.1. Estas classes e o protótipo se encontram disponíveis no endereço http://www.dcc.unicamp.br/~cripto/lucas/gerador.zip.

Foram implementados sistemas notacionais e sistemas baseados em cupons, mais especificamente as etapas de Retirada, Pagamento e Depósito. As etapas de autenticação dos usuários, emissão de recibo e finalização da transação fazem uso de implementações comuns, independente do tipo de sistema a ser gerado. Para permitir maior flexibilidade, foram implementadas também classes sem funcionalidade para cada uma das etapas do modelo de funcionamento, o que permite ao projetista decidir se deseja ou não incluir uma determinada etapa em seu sistema.

A biblioteca de classes para sistemas baseados em cupons é composta por:

Retirada Foram implementadas duas classes, uma para o cliente (pagador) e outra para o banco. A classe do banco constrói um cupom, assina este cupom e o envia ao cliente, que armazena o cupom para uso a posteriori.

Pagamento São necessárias classes para o pagador, o recebedor e o banco. A classe do pagador recupera o cupom armazenado e o envia ao recebedor, que por sua vez o repassa ao banco para verificação. Se o cupom estiver correto, o banco informa ao recebedor que informa ao pagador. O recebedor deve também armazenar o cupom para posterior depósito.

Depósito Os módulos envolvidos são o recebedor e o banco. O recebedor envia o cupom ao banco, que deve verificar sua validade e informar ao recebedor. O banco pode então atualizar o saldo do recebedor.

Para os sistemas notacionais, as etapas são:

Retirada Não é comum em sistema notacionais e não foi implementada.

Pagamento São necessárias classes para pagador e recebedor. O pagador constrói uma ordem de pagamento assinada e a envia ao recebedor, que verifica a assinatura do pagador e o informa se o pagamento foi aceito. O recebedor deve armazenar a ordem de pagamento para uso futuro.

Depósito Foram implementadas duas classes, uma para o recebedor e outra para o banco. O recebedor recupera a ordem de pagamento e envia ao banco, que vai verificar a assinatura do pagador e, caso esteja correta, realizar a transferência entre as contas.

As implementações comuns existentes hoje são:

Autenticação Foi implementada uma classe que realiza a autenticação pelo uso de criptografia de chaves públicas: a entidade que deseja se autenticar assina sua identificação usando sua chave privada e envia o resultado à entidade que requisitou a autenticação. Assim, basta verificar a assinatura com a chave pública correspondente à identificação recebida.

Recibo A implementação realizada permite à entidade que recebeu o pagamento a emissão de um recibo assinado com sua chave privada.

Finalização Não foi implementada. Esta etapa aparece apenas em alguns poucos sistemas e não teria muita utilidade no contexto das biblioteca apresentada acima.

Aspectos de implementação

Durante o desenvolvimento do protótipo, algumas decisões foram tomadas para simplificar ou facilitar a implementação:

1. O protótipo atual utiliza apenas assinatura digital como primitiva criptográfica.
2. Para fins de armazenamento e transmissão um objeto e sua assinatura são tratados com entidades distintas, ocupando arquivos diferentes e sendo transmitidos em separado. Isto leva a uma simplificação do tratamento de assinaturas.
3. Não foi usado nenhum sistema de gerência de chaves, embora tenha sido implementada uma classe capaz de simular a existência de um repositório seguro de chaves. Esta classe simplesmente supõe que as chaves se encontram no diretório onde os módulos são executados.
4. Os módulos devem ser executados na mesma máquina, embora usem o mecanismo de comunicação por *sockets*, que permitiria que estivessem em máquinas diferentes. Isto facilitou a simulação do repositório de chaves.

5. O sistema só é capaz de gerar um único cupom ou ordem de pagamento, embora o valor deste cupom ou ordem de pagamento possa variar. Esta decisão eliminou a necessidade de controle de números de série para cupons ou ordens de pagamento.

É importante ressaltar que estas simplificações ocorreram por problemas de tempo e não por limitações inerentes ao gerador de sistemas de pagamento e que é possível reverter cada uma delas. Considerando que o protótipo foi desenvolvido com o intuito de demonstrar a viabilidade de implementação do gerador de sistemas de pagamento, não se considera necessário que a implementação seja feita de maneira completa, mas que dê indícios da possibilidade de geração de sistemas mais complexos.

2.2.3 Uma Aplicação do Gerador

Vamos agora mostrar como o esquema de análise e, por conseqüência, o gerador de sistemas de pagamento podem ser úteis quando do projeto e implementação de um sistema de pagamento. Como exemplo vamos mostrar como seria o projeto de um sistema de pagamento para cotas de impressão numa rede de computadores.

O Ambiente alvo

Suponhamos uma rede de computadores com diversos usuários compartilhando uma única impressora. Para evitar desperdícios, ficou estabelecido que os usuários seriam cobrados por cada página impressa. Assim, é necessário que o sistema de impressão seja capaz de cobrar de cada usuário quando este desejar imprimir.

Usando o Esquema de Análise

Usando o esquema descrito na Seção 2.1.3, concluímos que o sistema de pagamentos necessário tem as seguintes características:

Modelo de troca: *qualquer.*

Envolvimento da entidade controladora: *qualquer.*

Quantias envolvidas: *micropagamentos.*

Hardware necessário: *uso geral.*

Papéis existentes: *usuário* e *servidor central.*

Inversibilidade de papéis: *existente.* O sistema deve permitir que os usuários troquem cotas de impressão entre si.

Privacidade: *desnecessária*, já que o servidor terá acesso a todos os dados do sistema.

Integridade: *garantida.*

Robustez: *não-crítica.*

Viabilidade econômica: *viável.* O sistema deve ser de baixo custo de manutenção.

Escalabilidade: *desnecessária.* O ambiente não requer aumentos significativos de usuários ou de valor envolvido.

Interoperabilidade: *desnecessária.*

Auditabilidade: *necessária.* Arquivos de *log* são suficientes.

Da Geração do Subsistema de Pagamento

De acordo com as características apresentadas acima, podemos concluir que é necessário um sistema de micropagamentos do tipo MILLICENT, PAYWORD ou MICROMINT. O ideal seria implementar uma variação do MILLICENT, ou seja, um sistema baseado em cupons que fizesse uso de uma função de *hashing* para autenticar os cupons. Neste sistema, as funções de vendedor e corretor seriam realizadas pelo servidor de impressão.

Dentre as classes implementadas para a biblioteca descrita acima, não se encontra uma que tenha tais características. Assim, foram usadas as classes que implementam os cupons assinados e as operações associadas a estes, como retirada, depósito e validação. O protótipo resultante poderia ser aperfeiçoado para uso em um ambiente real.

2.3 Conclusões

Este capítulo apresenta um esquema que permite sistematizar a análise e comparação dos sistemas de pagamento eletrônico, com base na tipificação, requisitos, funcionamento e implementação destes sistemas.

Tendo em vista que não deve haver apenas um único sistema de pagamento no mercado, mesmo no futuro, é necessário encontrar critérios que nos permitam decidir qual o sistema de pagamento mais adequado a nossas necessidades. Assim, acreditamos que uma das importantes contribuições deste trabalho seja oferecer uma sistemática que permita, após um levantamento detalhado dos requisitos da aplicação, analisar a adequabilidade de um sistema ou comparar, dentre os sistemas existentes, qual é o mais indicado. Assim, acreditamos que a análise e comparação de sistemas de pagamento eletrônico devem sempre ocorrer tendo em vista a aplicação desejada, por exemplo: venda de informações, venda de bens físicos, *pay-per-view* etc.

Outro ponto que consideramos importante neste trabalho foi a sistematização do funcionamento dos sistemas de pagamento de uma maneira bastante ampla, o que pode assistir o projeto de novos sistemas. As diferentes etapas de funcionamento que caracterizamos para os sistemas de pagamento devem ser levadas em conta no projeto de novos sistemas, o que permite ao projetista verificar se deixou de incluir algum ponto importante ou eliminar algum item cuja inclusão revelou-se desnecessária em vista da aplicação desejada.

Como não acreditamos na possibilidade de construção de um sistema universal, ou seja, que possa substituir com eficiência qualquer outro sistema de pagamento, acreditamos que a análise e comparação levando em conta o contexto de aplicação do sistema são imprescindíveis a um bom entendimento e ao uso correto dos sistemas de pagamento.

A possibilidade de implementação de um gerador de sistemas de pagamento eletrônico baseado no esquema de funcionamento descrito na Seção 2.1.3, demonstra que este esquema de funcionamento é adequado para descrever uma grande gama de sistemas de pagamento.

Embora a utilidade comercial de um tal gerador de sistemas de pagamento seja duvidosa (acreditamos que a padronização dos sistemas é necessária), sua utilidade didática é clara. É possível facilitar a compreensão a respeito de sistemas de pagamento eletrônico usando um modelo bem definido de funcionamento e possibilitando aos estudantes uma análise de como são feitas as composições das diversas primitivas existentes para sistemas deste tipo, além do desenvolvimento de novas primitivas.

Capítulo 3

Análise dos Sistemas de Pagamento Eletrônico

Neste capítulo, apresentamos alguns dos mais importantes e representativos sistemas de pagamento eletrônico, analisados de acordo com sua tipificação, requisitos, funcionamento e aspectos de implementação. As descrições destes sistemas estão resumidas nas tabelas que se encontram no final do capítulo.

3.1 GREEN COMMERCE MODEL da FIRST VIRTUAL

A FIRST VIRTUAL HOLDING COMPANY foi uma das primeiras empresas a oferecer uma solução que permite a realização de transações pela Internet. O principal objetivo do sistema é eliminar a necessidade da transmissão de números de cartão de crédito pela rede. O sistema foi batizado de GREEN COMMERCE MODEL [34], sendo o único sistema encontrado a não fazer uso de criptografia.

O GREEN COMMERCE foi projetado para a venda de informações pela rede, sendo que o vendedor assume os riscos de não receber o pagamento. O sistema funciona pela troca de mensagens por correio eletrônico, embora outros tipos de comunicação, como telnet, possam ser usados.

Como não faz uso de criptografia, os usuários do sistema não necessitam de *software* especial, e podem usar os programas navegadores e de envio de correio eletrônico que já possuírem. Os vendedores podem construir sistemas que automatizem o processo de venda ou usar os *softwares* prontos fornecidos pela FIRST VIRTUAL. A segurança do sistema está parcialmente baseada na segurança do sistema de correio eletrônico usado pelos participantes das transações.

3.1.1 Tipificação

Modelo de troca: notacional.

Sendo uma extensão ao método tradicional de compras por cartões de crédito, o sistema define contas especiais vinculadas a um cartão e que podem ser usadas para a realização de transações na Internet.

Envolvimento da entidade emissora: *on-line.*

Numa transferência, o comprador informa ao vendedor o número de sua conta FIRST VIRTUAL, que tem de ser validado no momento da transação.

Quantias envolvidas: micropagamentos.

O sistema foi projetado para venda de informações na Internet sem mecanismos de segurança, possibilitando uma redução nos custos. Assim, apenas micropagamentos podem ser efetuados com alguma segurança. O servidor central acumula diversas transações de cada usuário antes de processá-las junto à empresa de cartões de crédito, diminuindo assim o custo por transação.

Hardware necessário: uso geral.

O GREEN COMMERCE foi projetado para fazer uso apenas de *hardware* e *software* disponíveis na maior parte das máquinas usadas para acesso à Internet.

Papéis envolvidos: comprador, vendedor e servidor central.

O servidor central realiza as transferências de valores e pertence à entidade controladora. Compradores e vendedores devem estabelecer contas junto ao servidor central antes de participar do sistema.

Inversibilidade de papéis: inexistente.

O sistema distingue claramente entre contas de compradores e contas de vendedores.

Privacidade: inexistente.

O sistema não prevê mecanismos para garantia de privacidade.

Divisibilidade: nível 4, o sistema é notacional.

3.1.2 Requisitos

Integridade: O sistema confia nos mecanismos de integridade do TCP/IP e do SMTP, que são reconhecidamente fracos. Não há garantia da integridade das mensagens trocadas.

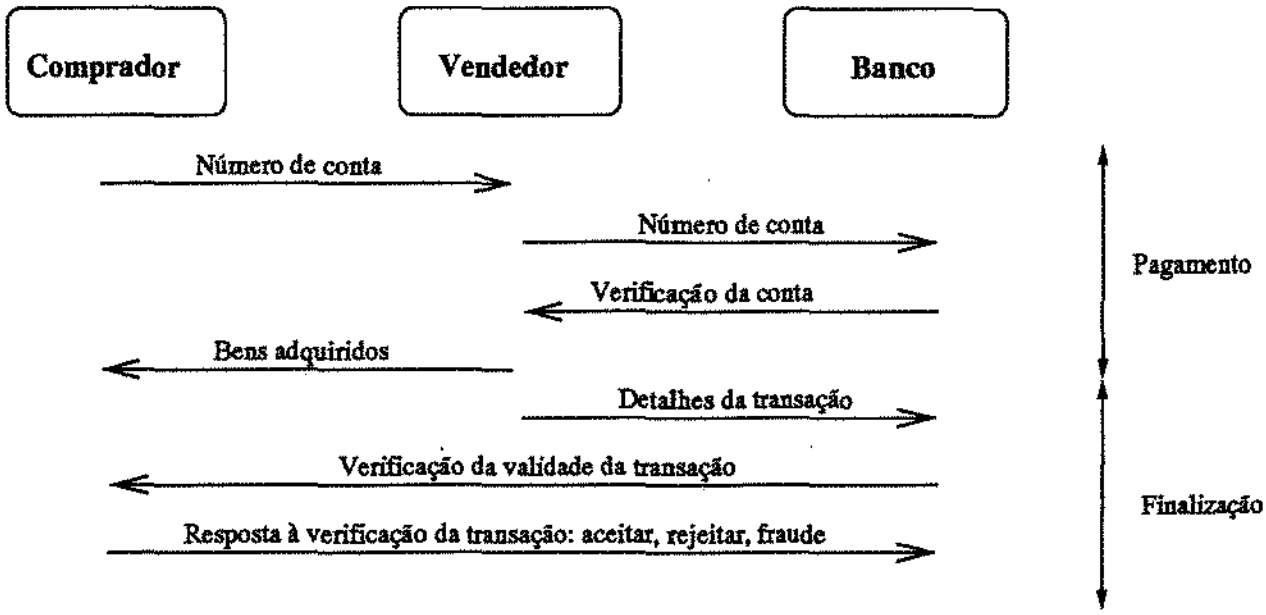


Figura 3.1: Fluxo de mensagens no GREEN COMMERCE MODEL

Robustez: O servidor central tenta garantir a consistência do sistema. Em caso de dúvida ou disputa, este servidor resolve a questão. O sistema não tem mecanismos para garantir as características ACID das transações.

Viabilidade econômica: O sistema está baseado nos protocolos mais usados na Internet, não agregando novos serviços de segurança. Sendo assim, apresenta um baixo custo de implantação e manutenção e pode ser considerado viável à realização de micropagamentos.

Escalabilidade: O sistema é bem escalável, apesar de apresentar um gargalo no servidor central da entidade emissora. Uma frequência muito grande de requisições pode comprometer o desempenho do sistema. Uma extensão ao sistema com o uso de várias entidades emissoras poderia melhorar este aspecto.

Interoperabilidade: O sistema não prevê interoperabilidade.

Auditabilidade: Todas as transações passam pelo servidor central que provê mecanismos de *logs*, extratos etc.

3.1.3 Funcionamento

A Figura 3.1 apresenta o fluxo das mensagens trocadas entre os participantes da fase de transação, como descrito a seguir:

1. **Relacionamento**
   O usuário deve entrar em contato com o servidor central e estabelecer uma conta neste servidor. O usuário deve também fornecer o número de seu cartão de crédito por telefone ou fax, o que garante maior segurança. A conta será liberada para a realização de transações após a verificação do cartão de crédito. Os vendedores devem solicitar a transformação de suas contas em contas especiais para vendedores.

2. **Retirada**
   Não há etapa de retirada.

3. **Transação**

   (a) **Autenticação**
   Não há autenticação dos participantes.

   (b) **Pagamento**
   Após a negociação do valor a ser pago, o comprador envia o número de sua conta FIRST VIRTUAL ao vendedor. O vendedor repassa este número, o número de sua conta, o valor da transação e a moeda usada na transação ao servidor central. Após esta etapa, o vendedor entrega as mercadorias.

   (c) **Recibo**
   Não há emissão de recibo.

   (d) **Finalização**
   O servidor central envia um pedido de confirmação ao comprador, que pode responder:

   SIM: se concorda com a transação. Neste caso, o servidor envia uma confirmação ao vendedor e vai descontar o valor do cartão de crédito do comprador.

   NÃO: se não concorda. O servidor informa então ao vendedor que a transação não se completou.

   FRAUDE: se o comprador não iniciou a transação. O servidor central inicia uma investigação para apurar a fraude.

4. **Compensação**
   Após um prazo de 91 dias, a FIRST VIRTUAL deposita o valor na conta do vendedor. Este é o prazo que o comprador tem para receber a fatura do cartão de crédito e verificar se houve algum problema.

3.1.4 Aspectos de Implementação

Técnicas Criptográficas: não são usadas.

Conexões de rede: 3 conexões.
O sistema prevê:

1. uma conexão entre comprador e vendedor, por onde se dará a transação;
2. uma conexão do vendedor com o servidor central;
3. uma conexão do servidor central com o comprador.

Todas as conexões podem ser substituídas pela troca de correio eletrônico.

Formato das ordens de pagamento: As ordens de pagamento consistem apenas do número de identificação do comprador. O vendedor anexa a este sua identificação, valor e unidade monetária usada na transação.

3.1.5 Comentários e Análises

O GREEN COMMERCE MODEL é uma das maneiras mais simples e baratas de construir um sistema de pagamentos para a Internet: a necessidade de *software* especializado é mínima e o funcionamento do sistema é de fácil compreensão. Entretanto, o sistema apresenta o problema de deixar o vendedor desprotegido contra fraudes: um usuário poderia adquirir uma grande quantidade de informação e responder sistematicamente NÃO ou FRAUDE quando o servidor requisitar a confirmação da transação.

Este foi o primeiro sistema para micropagamentos a funcionar na Internet e foi projetado de tal maneira a diminuir a necessidade de verificação de honestidade dos vendedores, permitindo assim que pequenas ou micro empresas e particulares pudessem entrar sem dificuldades na era do comércio digital.

Numa comparação com outros sistemas de micropagamentos, o GREEN COMMERCE leva vantagem pelo fato de poder ser usado em qualquer computador já ligado à Internet e ser de fácil implantação. Entretanto, como não apresenta mecanismos de segurança, este sistema enfrenta muita desconfiança por parte dos consumidores, o que pode explicar o fraco desempenho que a FIRST VIRTUAL vem apresentando. Nos últimos meses, a FIRST VIRTUAL anunciou que está abandonando o mercado de sistemas de pagamento.

3.2 GLOBEID da GCTECH

A empresa francesa GCTECH desenvolveu o GLOBEID [27] para ser um sistema completo para comércio eletrônico, e que deve tratar desde a negociação do preço até o pagamento.

O sistema foi projetado para ser usado em conjunto com métodos tradicionais de pagamento, através de uma conta especial vinculada a uma conta bancária ou a um cartão de crédito.

O ponto que distingue o GLOBEID dos outros sistemas estudados é que seus projetistas tentaram fazer dele mais que um sistema para a realização de pagamentos, incluindo nele um serviço de cartório, onde ficam armazenados os dados referentes às transações, possibilitando assim a prestação de serviços de não-repúdio.

3.2.1 Tipificação

Modelo de troca: notacional.

O sistema trabalha com contas especiais gerenciadas por uma entidade central, com transferências feitas entre estas contas. Os projetistas argumentam que a criação de moeda eletrônica poderia ser ilegal em alguns países, além de apresentar problemas técnicos quanto ao armazenamento destas moedas com segurança.

Envolvimento da entidade emissora: *on-line*.

Para que as transferências possam acontecer, a entidade que gerencia as contas deve ser acionada. O cliente repassa uma proposta de negócio ao banco que valida os dados e realiza a transação.

Quantias envolvidas: pequenos pagamentos.

Os projetistas indicam um limite inferior de cerca de US$ 0,05 para as transações por este sistema, indo até algumas centenas de dólares na outra ponta. Valores da ordem de US$ 0,05 seriam classificados como micropagamentos, mas acreditamos que este sistema seja realmente eficiente na faixa dos pequenos pagamentos, a partir de US$ 1,00 .

Hardware necessário: uso geral.

O sistema não faz uso de *hardware* específico e pode ser implementado em qualquer tipo de computador. Os projetistas prevêem o uso futuro de *hardware* para autenticação dos participantes.

Papéis envolvidos: comprador, vendedor e servidor.

O sistema consiste de compradores, vendedores e de uma rede de servidores, que mantém as contas especiais dos usuários fornecendo os serviços de movimentação financeira e de cartório. Os compradores possuem uma senha, enquanto que os vendedores devem ter um par de chaves para uma cifra assimétrica. O servidor deve conhecer todas as senhas e todas as chaves públicas. O servidor deve atuar como intermediário em todas as transações. Podem existir vários servidores participando do sistema.

Inversibilidade de papéis: inexistente.

Embora os projetistas não façam distinção entre contas de compradores e contas de vendedores, os programas usados são diferentes, assim como o tipo de informação que a entidade controladora deve ter a respeito de cada participante. Por exemplo, os vendedores devem registrar chaves públicas enquanto os compradores usam uma senha.

Privacidade: existente

Algumas mensagens são cifradas. É possível a um observador descobrir as mercadorias e valores envolvidos na transação. O banco tem informações sobre toda a operação. O cliente sabe quem é o vendedor embora o vendedor não saiba quem é o cliente. As mensagens cifradas e o mecanismo de *challenge-response* (ver seção A.5.1) usados impedem que um observador obtenha informações sobre a conta do usuário.

Divisibilidade: nível 4, o sistema é notacional.

3.2.2 Requisitos

Integridade: Assinaturas digitais e *hashings* são usados para garantir a integridade das mensagens.

Robustez: O servidor central tenta garantir a consistência do sistema. Em caso de dúvida ou disputa, este servidor resolve a questão.

Atomicidade: monetária. Depois que o comprador autoriza uma transação, não há mais possibilidade de revertê-la. Até este momento o comprador pode cancelar a transação.

Consistência: O vendedor apresenta uma proposta que não pode ser alterada pelo comprador, sendo que esta proposta é o documento usado pelo banco para realizar a transferência. Assim, todos tem uma visão consistente da transação.

Isolamento: Desde que o comprador tenha crédito, as transações são independentes. Se o comprador ultrapassar o limite que pode ser pré-estabelecido, as transações passam a ser negadas em função de transações anteriores.

Durabilidade: A entidade controladora pode ser consultada sobre o estado do sistema e seu servidor deve ser capaz de se recuperar de falhas.

Viabilidade econômica: Este sistema apresenta um custo de manutenção significativo, o que deve torná-lo eficiente para valores acima de US$ 1,00. No entanto, os projetistas afirmam que este sistema é adequado para transações acima de US$ 0,05.

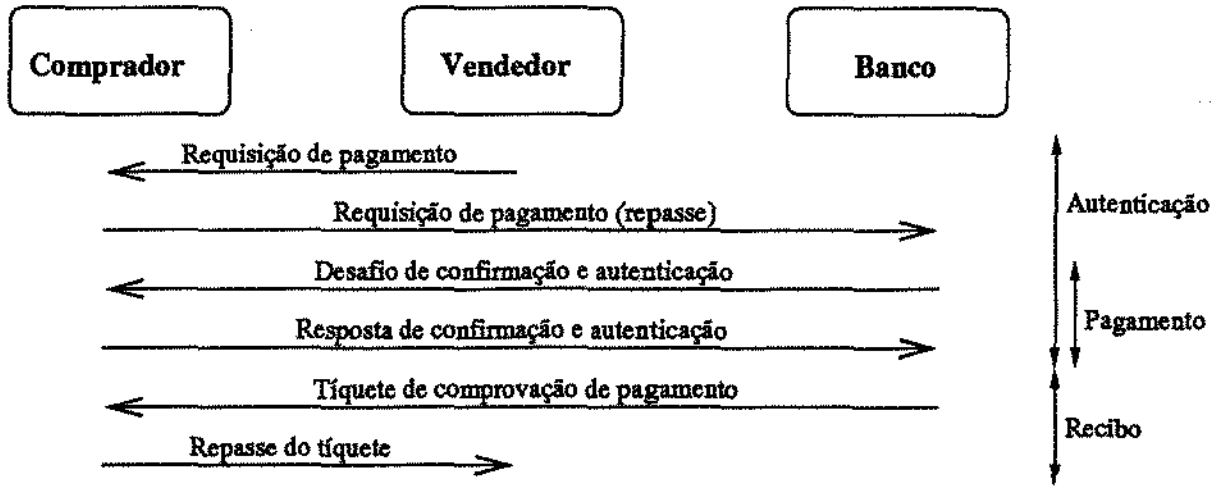


Figura 3.2: Fluxo de mensagens no GLOBEID

Escalabilidade: O sistema tem como gargalo o servidor da entidade controladora, embora sejam possíveis extensões para permitir a coexistência de diversas destas entidades.

Interoperabilidade: O sistema não prevê interoperabilidade.

Auditabilidade: Todas as transações passam pelo servidor central que deve prover mecanismos de *logs*, extratos etc. Além disto todas as transações são autenticadas com base em assinaturas digitais ou senhas.

3.2.3 Funcionamento

As etapas do funcionamento estão descritas abaixo; a Figura 3.2 apresenta as mensagens trocadas na etapa da transação.

1. **Relacionamento**
   O usuário deve estabelecer um relacionamento com um dos servidores de intermediação, fornecendo os dados do seu cartão de crédito, conta corrente ou de algum outro meio de pagamento que o servidor aceite. O servidor irá fornecer uma senha e um número de identificação ao usuário, assim como o *software* necessário à sua participação no sistema.

2. **Retirada**
   Não há retirada já que o sistema atua pela transferência de valores entre contas.

3. **Transação**

(a) **Autenticação**
O comprador deve entrar em contato com o vendedor e pedir uma cotação de preços. O vendedor enviará a cotação assinada com sua chave privada. O comprador irá repassar esta cotação ao servidor GLOBEID, que irá efetuar a autenticação do comprador, via *challenge-response* (seção A.5.1), o que garante que a senha não trafega na rede. O vendedor é identificado pela assinatura na cotação.

(b) **Pagamento**
Ao enviar o pedido de autenticação, o servidor informa ao comprador os detalhes da transação. Na resposta ao pedido de autenticação, o comprador deve confirmar seu desejo de continuar com a transação. Se for o caso, o servidor irá realizar a transferência entre a conta do comprador e a conta do vendedor.

(c) **Recibo**
Não há emissão de recibo.

(d) **Finalização**
O servidor emite uma prova de pagamento e a envia ao comprador, que a repassa ao vendedor. O vendedor então entrega a mercadoria ou informação adquirida.

4. **Compensação**
Não há etapa de compensação, já que a transferência ocorre quando o servidor recebe a confirmação de que o comprador aceita a transação.

3.2.4 Aspectos de Implementação

Técnicas Criptográficas: são usados os seguintes itens:

Cifras assimétricas: São usadas, embora o algoritmo não seja especificado.

Assinatura digital: O vendedor deve assinar a oferta de produtos/serviços.

Hashings: São usados como MACs no processo de *challenge-response*.

Conexões de rede: 2 conexões.
Uma conexão deve ser estabelecida entre o vendedor e o comprador e outra é necessária entre o comprador e o servidor do banco.

Formato das ordens de pagamento: Não especificado.

3.2.5 Comentários e Análises

O GLOBEID enfrenta dois problemas para ser um sistema de micropagamentos bem sucedido: primeiro, é um sistema caro e centralizado; segundo, é um sistema pouco conhecido e surgiu num momento em que empresas já famosas começam a se interessar por este mercado. Por ser um sistema caro e centralizado, o GLOBEID pode ter dificuldades em enfrentar concorrentes como o GREEN COMMERCE (seção 3.1) e o MILLICENT (seção 3.9), que apresentam estruturas mais enxutas.

O sistema foi apresentado por uma empresa francesa pouco conhecida na Internet, especialmente no Brasil, e pode ter dificuldade de aceitação por parte do mercado. Entretanto, ao que tudo indica, o mercado alvo do sistema não é prioritariamente a Internet, que ainda não tem na França a mesma penetração que nos Estados Unidos, mas o sistema MINITEL da telefônica francesa, que já apresenta índices de comércio eletrônico bastante razoáveis.

3.3 SECURE ELECTRONIC TRANSACTIONS

O SET [37] é hoje o mais importante protocolo projetado para permitir a realização de pagamentos seguros pela Internet. Isto porque surgiu dos esforços conjuntos de grandes empresas da área financeira (VISA, MASTERCARD) e da área de informática (IBM, NETSCAPE, MICROSOFT). É um protocolo destinado à realização de transações com cartões de crédito na Internet e pressupõe o uso da infra-estrutura já existente para compensação de transações com cartões de crédito. Assim, existem diversas empresas capacitadas a fornecer cartões e outras capacitadas a compensar as transações realizadas. O sistema pressupõe também a existência de uma hierarquia de certificação de chaves públicas.

O estudo apresentado a seguir está baseado na variação mais simples possível para uma transação no SET. A Figura 3.3 apresenta todas as possíveis mensagens deste sistema [17].

3.3.1 Tipificação

Modelo de troca: notacional

O SET é uma extensão dos sistemas tradicionais de processamento de transações de cartões de crédito. Foi projetado como uma maneira segura de transferir números de cartões pela Internet. Portanto, usa o modelo notacional.

Envolvimento da entidade controladora: *on-line*

É necessária a participação da entidade que processa as transações de cartões de crédito para o vendedor, já que apenas esta entidade está habilitada a validar a

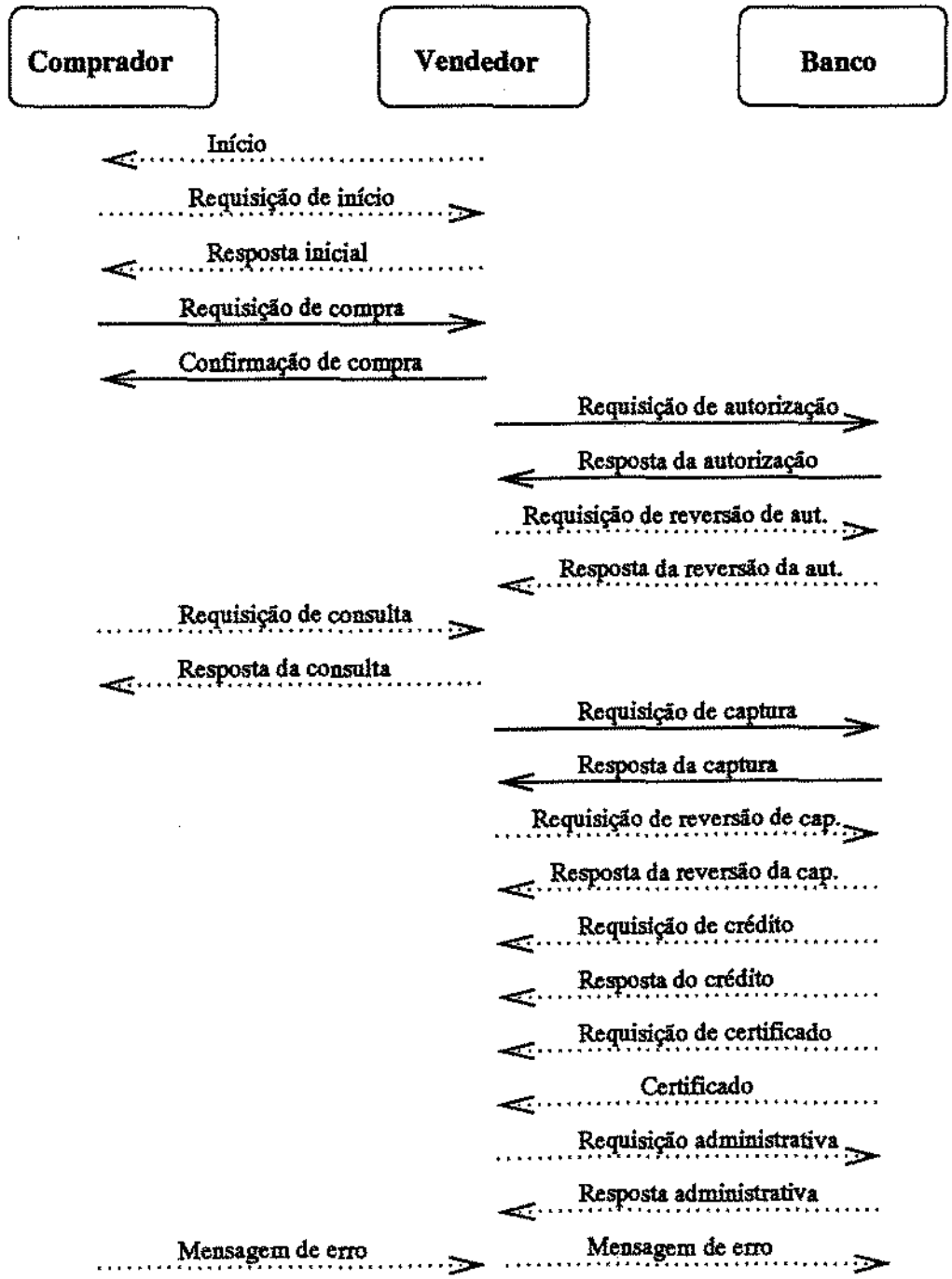


Figura 3.3: Todas as mensagens existentes no SET

transação. Assim como no sistema tradicional de cartões de crédito, as tarefas da entidade controladora é repartida entre diversas entidades, congregadas sob a mesma marca (eg. VISA).

Quantias envolvidas: pequenos a grandes pagamentos

O custo das transações com cartões de crédito torna o sistema inadequado para a realização de micropagamentos. O limite estabelecido para cada cartão indica o teto para o valor das transações.

Hardware necessário: uso geral

O sistema não necessita do uso de nenhum tipo de maquinário dedicado ou especial.

Papéis envolvidos: comprador, vendedor e entidade controladora.

O protocolo pressupõe um comprador, que deve ter um cartão de crédito, um vendedor e uma entidade controladora, que realiza as compensações para o vendedor.

Inversibilidade dos papéis: inexistente

Os papéis desempenhados no sistema são fixos, assim como em transações normais com cartões de crédito.

Privacidade: existente

O sistema não revela a identidade do comprador (número do cartão de crédito) ao vendedor e não revela o conteúdo do pedido à entidade controladora. Certificados e cifras são usados para garantir autenticação e privacidade das mensagens trocadas, onde necessário. A entidade controladora deve ter acesso ao número do cartão de crédito do comprador, o que pode ser usado para identificá-lo.

Divisibilidade: nível 4, o sistema é notacional.

3.3.2 Requisitos

Integridade: O sistema usa assinaturas digitais e certificados para proteger as mensagens e as implementações devem garantir a integridade dos dados críticos que sejam armazenados localmente.

Robustez: O sistema é capaz de preservar as seguintes características das transações:

Atomicidade: monetária. A transação só é realmente efetivada pela entidade controladora escolhida pelo vendedor. Assim, se as informações chegam até ela corretamente, a transação ocorre; senão, a transação não acontece. As outras entidades envolvidas serão informadas ou poderão verificar mais tarde se a transação foi processada.

Consistência: As mensagens trocadas pelos participantes permitem garantir que todos estejam de acordo quanto aos detalhes da transação, principalmente a quantia e o objeto ou serviço adquirido.

Isolamento: A interferência entre transações só ocorre quando o usuário ultrapassar o limite de seu cartão; a partir daí novas transações não podem ser efetuadas.

Durabilidade: O sistema tem um mediador, que é a entidade que processa as transações, e este é quem indica o estado do sistema. Assim, todos podem consultar o mediador e tomar conhecimento do estado atual. O sistema tradicional de processamento de transações de cartões de crédito provê mecanismos para garantir a durabilidade que devem ser usados quando do processamento através do SET.

Viabilidade econômica: O uso da infra-estrutura tradicional de cartões de crédito aumenta o custo de operação deste sistema e o torna eficiente apenas para transações médias a grandes. A complexidade do protocolo também deve elevar o custo de torná-lo operacional.

Escalabilidade: O sistema é tão escalável quanto o sistema tradicional de cartões de crédito.

Interoperabilidade: O SET não prevê integração com outros sistemas de pagamento eletrônico.

Auditabilidade: O uso de logs e assinaturas digitais permite verificar com bastante eficiência o funcionamento do sistema e detectar problemas, falhas ou uso indevido.

3.3.3 Funcionamento

As etapas do funcionamento são descritas abaixo e a etapa de transação é detalhada também na Figura 3.4. A seta tracejada indica uma mensagem que pode ser adiada e as setas pontilhadas indicam mensagens opcionais.

1. **Relacionamento**
   O usuário já deve ter um cartão de crédito, o que indica uma relação com uma empresa autorizada a emitir cartões. Esta entidade deve indicar a autoridade certificadora que emitirá os certificados de seus clientes. Já os vendedores devem estar habilitados a receber pagamentos com cartões de crédito e devem obter um certificado junto à autoridade certificadora indicada pela empresa que realiza as compensações de suas transações.

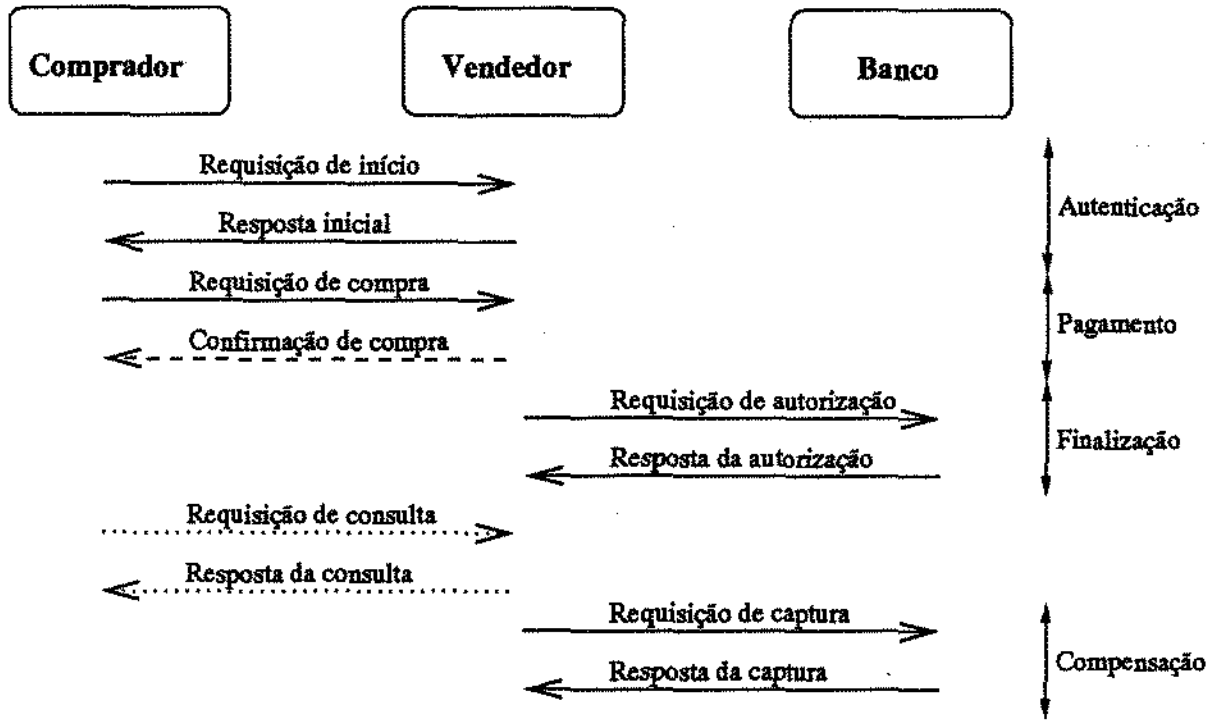


Figura 3.4: Fluxo de mensagens no SET

2. **Retirada**
   não há.

3. **Transação**

   (a) **Autenticação**
   Já tendo o comprador e o vendedor chegado a um acordo quanto às mercadorias em questão e ao preço, o comprador envia uma requisição de início de transação contendo a marca de seu cartão (Visa, MasterCard etc) e seu certificado. O vendedor responde com seu certificado, que indica ser habilitado a participar do sistema, e o certificado da entidade escolhida para realizar a compensação da transação.

   (b) **Pagamento**
   O usuário gera então uma requisição de compra, contendo: uma ordem de pagamento e uma requisição de produtos ou serviços cifrada com a chave pública do vendedor. A ordem de pagamento é cifrada com a chave pública de uma entidade controladora. Estas duas mensagens são assinadas pelo processos de assinaturas duais (ver Seção A.4.1), que liga as duas informações. As assinaturas são enviadas junto com as mensagens para o vendedor. O vendedor deve

repassar a ordem de pagamento à entidade controladora escolhida.

(c) **Recibo**
não há emissão de recibo.

(d) **Finalização**
A entidade controladora vai receber a ordem de pagamento e realizar sua compensação pelo processo tradicional de compensação de transações com cartões de crédito, obtendo uma resposta indicativa do crédito do usuário. Esta resposta, seja ela afirmativa ou não, será repassada ao vendedor. Se o usuário tiver crédito, o vendedor recebe também um cupom para que possa realizar a compensação da transação.

Após receber a confirmação, o vendedor deve avisar ao comprador o resultado da transação. Isto pode ocorrer *off-line*.

4. **Compensação**
Quando tiver entregue os bens ou serviços adquiridos pelo comprador, o vendedor deve pedir a compensação (captura) da transação, enviando uma requisição à entidade controladora junto com o cupom de compensação recebido anteriormente.

Observações:

1. O vendedor pode finalizar a transação assim que receber a ordem de pagamento, indicando ao comprador que a transação será autorizada mais tarde, e que o comprador deve verificar mais tarde se a autorização foi positiva.

2. O vendedor pode pedir a realização da compensação junto com a autorização da transação e, neste caso, as etapas de compensação e pagamento são condensadas numa única.

3.3.4 Aspectos de Implementação

Criptografia: as primitivas usadas são:

Cifras simétricas: DES padrão de 56 bits.

Cifras assimétricas: RSA de 1024 bits para usuários e certificadores de usuários e RSA de 2048 bits para entidades de mais alto nível, que devem certificar chaves a serem usadas para emissão de certificados.

Assinatura digital: RSA com chaves de mesmo tamanho que aquelas usadas para cifrar, embora os participantes tenham chaves diferentes para cifrar e assinar.

Certificados: devem estar no formato X.509 versão 3.

Hashings: O algoritmo usado é o SHA.

Conexões de rede: 2 conexões.

Uma conexão é necessária entre o comprador e o vendedor, e uma entre o vendedor e a entidade controladora.

Formato das ordens de pagamento: As ordens de pagamento consistem de duas partes, chamadas de OI (*order information*), que contém dados que identificam os produtos serviços negociados, e PI (*payment instructions*), que autoriza a entidade controladora a efetuar o pagamento. O OI consiste do identificador da transação, identificador da marca do cartão, data e *nonces*. O PI consiste dos dados do cartão, identificação da transação, valor e de um *hashing* da descrição dos produtos/serviços.

OI e PI são assinados usando o processo de assinatura dual para garantir autenticidade e correspondência dos dois.

3.3.5 Comentários e Análises

O SET é certamente o sistema de pagamento que mais esteve em evidência nos últimos dois anos, em parte por causa do peso das empresas que participam de seu desenvolvimentos e em parte porque vem sendo apontado como o futuro padrão de fato para sistemas baseados em cartões de crédito.

A descrição acima apresenta apenas a variante mais simples do sistema, que é extremamente complexo e tenta abarcar todas as possibilidades das transações com cartões de crédito na Internet. É portanto um sistema grande e de difícil compreensão e implementação, levando também ao desenvolvimento de módulos grandes.

O SET vem sendo adotado em diversos países, inclusive o Brasil (através do VISA MALL e em parcerias com bancos como BRADESCO, REAL e BANCO DO BRASIL). A força das empresas responsáveis pelo SET certamente está ajudando o sistema a se tornar o mais difundido sistema de pagamento da atualidade.

Apesar do sucesso, o SET tem sido bastante contestado devido exatamente a sua complexidade. Informações não oficiais indicam que uma transação usando o sistema poderia durar até 20 segundos em ambientes de teste, e que portanto o sistema não seria adequado para uso comercial. Outro fato que pode se constituir num empecilho à ampla difusão do SET é a necessidade de certificação de cada um de seus usuários, o que pode levar algum tempo e ter custos não desprezíveis na implantação do sistema.

3.4 PAYWORD

O PAYWORD [30] é um sistema de micropagamentos projetado por R. Rivest e A. Shamir e é bem eficiente para a realização de pagamentos repetidos. O objetivo do sistema é reduzir o número de aplicações de cifras assimétricas, usando cifras simétricas e *hashings* (veja o Apêndice A) sempre que possível.

O sistema é baseado em cadeias de *hashings* (Seção A.4). Cada elemento da cadeia representa uma unidade monetária do sistema.

3.4.1 Tipificação

Modelo de troca: cupons

Cada comprador tem um certificado que o habilita a gerar séries de cupons (cadeias de *hashings*) que servem como forma de pagamento. As séries são específicas para um determinado vendedor.

Envolvimento da entidade emissora: *off-line*

A entidade emissora fornece a cada comprador um certificado que permite a confecção de séries de cupons. A emissão deste certificado e a aceitação e verificação das séries emitidas por cada usuário podem acontecer *off-line*.

Quantias envolvidas: micropagamentos

O sistema foi projetado para venda de informações na Internet. Para minimizar o custo devido ao uso de algoritmos criptográficos, principalmente algoritmos assimétricos, seu nível de segurança é reduzido, adequado apenas para micropagamentos.

Hardware necessário: uso geral

O sistema é todo baseado em *software*, não fazendo uso de *hardware* especial.

Papéis envolvidos: comprador, vendedor e corretor.

Os corretores gerenciam as contas dos usuários e emitem certificados aos usuários. Os vendedores devem se cadastrar com os corretores para poderem compensar os cupons recebidos. A entidade controladora do sistema deve cadastrar os corretores para agirem em seu nome.

Inversibilidade de papéis: inexistente

O sistema distingue claramente compradores de vendedores. O relacionamento com o corretor é diferente em cada caso.

Privacidade: inexistente

O sistema não prevê mecanismos para garantia de privacidade, sendo que o usuário é identificado em cada transação. As informações que trafegam na rede não são protegidas.

Divisibilidade Nível 4. O sistema não prevê divisibilidade, embora os projetistas sugiram que as cadeias possam ter um valor negociado entre comprador e vendedor.

3.4.2 Requisitos

Integridade: O sistema usa assinaturas digitais para garantir a integridade dos certificados. Os cupons não usam mecanismos de proteção de integridade.

Robustez: Os protocolos para envio de cupons não são especificados, o que não permite análise deste item.

Viabilidade econômica: Os projetistas tiveram a preocupação de diminuir os custos do sistema, reduzindo o número de operações de aplicação e verificação de assinaturas digitais. Assim o protocolo tem custo compatível com a realização de micropagamentos.

Escalabilidade: O sistema é bem escalável, principalmente porque as operações que envolvem a entidade controladora podem ser realizadas *off-line*.

Interoperabilidade: O sistema não prevê interoperabilidade.

Auditabilidade: As operações do sistema são identificadas, sendo necessário o uso de certificados para poder realizar transações.

3.4.3 Funcionamento

As etapas do funcionamento são, ilustradas na Figura 3.5:

1. **Relacionamento**
   O comprador deve escolher um corretor, que lhe emitirá um certificado. Este certificado dá direito ao usuário de gerar cadeias de *hashings* para a realização de pagamentos. O comprador se compromete a pagar o valor correspondente às cadeias que gerar.

2. **Retirada**
   O comprador deve gerar uma cadeia de *hashings* e assinar o último valor gerado. Esta assinatura é um comprometimento em pagar por cada um dos valores da

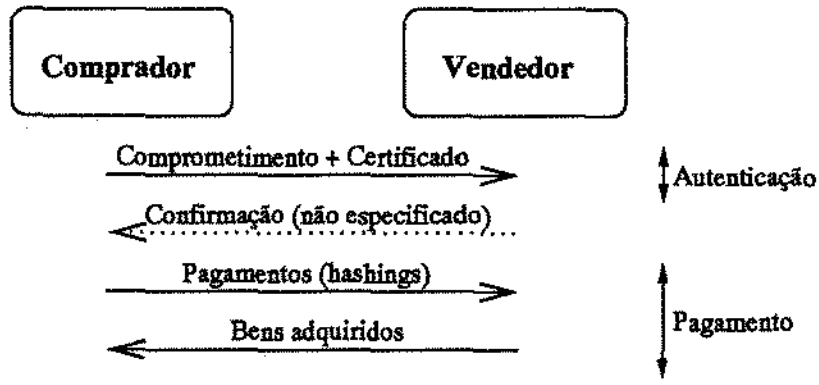


Figura 3.5: Fluxo de mensagens no PAYWORD

seqüência, que são gastos na ordem contrária àquela em que foram gerados. Os comprometimentos são específicos para cada vendedor.

3. **Transação**

   (a) **Autenticação**
   O comprador deve se identificar, enviando o comprometimento (último valor da cadeia assinado) e seu certificado, para que o vendedor possa verificá-lo.

   (b) **Pagamento**
   No ato do pagamento de x unidades monetárias, o comprador envia ao vendedor o par de valores (j, n_j), tal que

   $$k - j = x,$$

   onde (k, n_k) é o último par utilizado numa compra com este vendedor. O vendedor certifica-se do valor do pagamento recalculando a sequência $n_j, n_{j+1}, ..., n_k$.

   (c) **Recibo**
   Não há emissão de recibo.

   (d) **Finalização**
   Não existe etapa de finalização neste sistema.

4. **Compensação**
   O vendedor deve enviar ao corretor o comprometimento assinado pelo usuário e o último valor da cadeia que tenha recebido, indicando a posição deste valor na cadeia. O corretor irá gerar os *hashings* desta cadeia a partir do último valor recebido e verificar se a cadeia nunca foi usada. Caso seja válido, o vendedor receberá o valor correspondente. O corretor então registra esta cadeia como já tendo sido usada.

3.4.4 Aspectos de Implementação

Técnicas Criptográficas: as primitivas necessárias são:

Assinatura digital: é usada embora os projetistas não especifiquem qual algoritmo em particular.

Certificados: As chaves públicas dos usuários devem ser certificadas.

Hashings: São usados na geração da cadeia. O algoritmo não é especificado.

Conexões de rede: 1 conexão.

Apenas a comunicação entre comprador e vendedor precisa ocorrer durante cada transação.

Formato dos cupons: cadeias de *hashings* (ver seção A.4) com raiz assinada. Cada um dos valores da cadeia é um cupom.

3.4.5 Comentários e Análises

O PAYWORD é um dos poucos sistemas apresentados aqui do qual não conhecemos uma implementação. Este é um sistema bastante interessante, que explora novas possibilidades no uso de funções de espalhamento no comércio eletrônico. Sistemas similares foram propostos independentemente por outros autores (veja em [30]). Este sistema é muito eficiente para pagamentos repetidos a um mesmo vendedor, mas é ineficiente para a realização de pagamentos para vendedores diferentes por requerer uma operação de assinatura digital para cada novo vendedor. Sendo um sistema *off-line*, o PAYWORD reduz bastante o custo de operação do corretor, o que pode levar a uma redução do custo de cada transação.

O alvo principal deste sistema é a venda de informações *on-line*, e, em especial, para transações de baixo custo como venda de artigos de jornais, imagens e *pay-per-use*. Poderia ser uma alternativa ou complemento a serviços baseados em assinaturas por permitir acesso a não-assinantes, na base do *pay-per-need*.

3.5 MICROMINT

Apresentado junto com o PAYWORD (seção 3.4), o MICROMINT [30] também é um sistema para a realização de micropagamentos baseado em *hashings* (veja a Seção A.4). O MICROMINT tem a vantagem de não fazer uso de nenhum outro tipo de função criptográfica e permitir grande eficiência em pagamentos isolados a diferentes vendedores.

O sistema é baseado em colisões de *hashings* (ver seção A.4). Para gerar as moedas, a entidade emissora deve escolher valores aleatórios e aplicar a função nestes valores, se

possível usando *hardware* dedicado, e ir armazenando os valores e o resultado da função. Quando, para um dado resultado, a entidade emissora já tiver armazenado valores suficientes, esta pode gerar um cupom válido. O fato de gerar colisões em larga escala permite à entidade emissora fazê-lo de forma econômica.

3.5.1 Tipificação

Modelo de troca: cupons

O sistema é baseado em cupons gerados por uma entidade emissora. Os cupons correspondem a k-colisões de uma função de *hashing*.

Envolvimento da entidade emissora: *off-line*

Os cupons usados podem ser identificados e um usuário que gastar um cupom mais de uma vez pode ser identificado. Assim, a entidade emissora não precisa estar *on-line* durante a transação.

Quantias envolvidas: micropagamentos

O sistema foi projetado para venda de informações na Internet e sua segurança está em parte baseada no baixo valor unitário dos cupons, o que elimina o estímulo à fraude no sistema.

Hardware necessário: uso geral

O sistema é baseado em software e não faz uso de *hardware* específico. Os projetistas sugerem que a entidade emissora use *hardware* especial para poder gerar cupons com mais eficiência e aumentar o nível de segurança do sistema.

Papéis envolvidos: pagador, recebedor e corretor.

O corretor emite cupons e os vende aos usuários que podem resgatá-los junto ao mesmo, e permite aos usuários realizar ou receber pagamentos. A entidade emissora deve cadastrar os corretores para agir em seu nome.

Inversibilidade de papéis: existente

Este sistema permite que qualquer usuário receba cupons de outros usuários, embora a entidade emissora possa limitar a capacidade dos usuários de trocarem cupons entre si, já que a troca indiscriminada de cupons entre os usuários pode dificultar a identificação dos usuários que cometerem fraudes. Uma das maneiras de limitar a troca de cupons é estabelecer cupons identificados para cada vendedor.

Privacidade: inexistente

A entidade emissora é capaz de saber qual usuário comprou um determinado cupom, embora a troca indiscriminada de cupons entre os usuários possa aumentar o grau

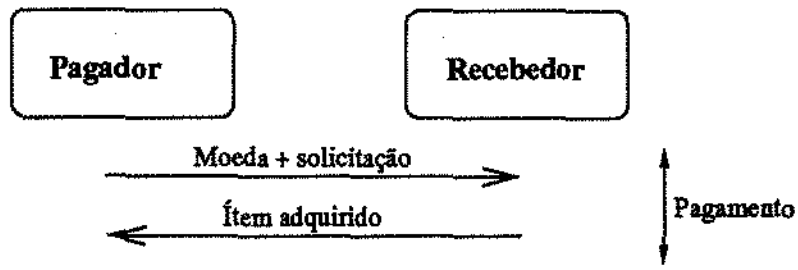


Figura 3.6: Fluxo de mensagens no MICROMINT

de privacidade do sistema. Nenhum tipo de proteção para as transmissões de cupons pela rede é especificado.

Divisibilidade: nível 4.
O sistema não prevê divisibilidade e os projetistas sugerem o uso de cupons cujo valor seja 1 centavo.

3.5.2 Requisitos

Integridade: Não são especificados mecanismos de integridade.

Robustez: O protocolo para envio de moedas não é especificado, o que não permite analisar este item.

Viabilidade econômica: A viabilidade econômica do sistema está baseada na economia de escala, ou seja, se a entidade emissora gerar muitos cupons, o sistema pode ser viável, já que o custo por cupom diminui.

Escalabilidade: O número de cupons em circulação depende da capacidade da entidade emissora de gerar cupons a tempo e de gerenciar estes cupons. Assim, dados os custos e a tecnologia disponível, existe um limite na quantidade de cupons ou usuários do sistema.

Interoperabilidade: O sistema não prevê interoperabilidade.

Auditabilidade: Os cupons poder ser identificados, o que permite à entidade emissora identificar usuários que estejam tentando fraudar o sistema.

3.5.3 Funcionamento

A Figura 3.5 apresenta o fluxo de mensagens na etapa de transação do MICROMINT, cujo funcionamento é descrito a seguir:

1. **Relacionamento**
   O usuário deve entrar em contato com um corretor, que lhe venderá cupons, e negociar a forma de pagamento a ser usada.

2. **Retirada**
   Fazendo uso do sistema negociado na fase de estabelecimento de relacionamento, o usuário paga ao corretor, que lhe envia cupons já prontos para uso. Os cupons podem posteriormente ser reconhecidos pelo corretor, que consegue verificar se o usuário está usando cada cupom mais de uma vez e lhe impor sanções.

3. **Transação**

   (a) **Autenticação**
   Não há autenticação.

   (b) **Pagamento**
   Os cupons são enviados pelo comprador em claro pela rede. Se for necessário cifrar os cupons, os usuários devem fazer uso de métodos externos ao sistema. Ao receber um cupom $(x_1, x_2, ..., x_k)$, o vendedor verifica se $h(x_1) = h(x_2) = ... = h(x_k)$.

   (c) **Recibo**
   Não há emissão de recibo.

   (d) **Finalização**
   Não existe etapa de finalização neste sistema.

4. **Compensação**
   O vendedor envia os cupons que recebeu ao corretor, que os reembolsará. O corretor irá verificar se os cupons não foram gastos duas vezes para poder tomar as providências cabíveis, como expulsar do sistema o comprador ou vendedor que estiver cometendo fraude. O depósito dos cupons pode acontecer *off-line*.

3.5.4 Aspectos de Implementação

Técnicas Criptográficas: a primitiva necessária é:

Hashings: Este sistema usa *hashings* como o único tipo de algoritmo criptográfico.

Conexões de rede: 1 conexão.
Apenas a comunicação entre comprador e vendedor precisa ocorrer durante cada transação.

Formato dos cupons: colisões de uma função de espalhamento (*hashing*, ver seção A.4). Os projetistas consideram viável o uso de 4-colisões, ou seja, quatro valores que a função mapeie sobre o mesmo valor. Os cupons tem valor monetário único fixado pelo corretor.

3.5.5 Comentários e Análises

Assim como o PAYWORD, não temos conhecimento de uma implementação do MICROMINT e muito menos de intenções de uso comercial deste sistema. O MICROMINT também apresenta novas possibilidades no uso de funções de espalhamento em sistemas de pagamento. Este sistema permite a realização de transações com diversos vendedores sem perda de eficiência, embora o custo de operação do servidor central seja bastante grande, principalmente por causa da geração de novas moedas. Este custo na geração de moedas é a base da segurança do sistema, já que possíveis fraudadores não conseguiriam obter lucro.

Assim como o PAYWORD, o MICROMINT também é adequado à venda de informações e à substituição de sistemas baseados em assinaturas.

3.6 CAFE

O projeto CAFE [3] foi um projeto europeu que tinha como objetivo a geração de tecnologia na área de permissões de usuários e controle de acesso. Seu resultado mais importante foi um dos mais avançados sistemas de pagamento eletrônico já propostos. Trata-se de um complexo sistema *off-line*, anônimo e seguro. É um sistema que faz uso de *smart cards* ou de sistemas de carteiras eletrônicas.

3.6.1 Tipificação

Modelo de troca: híbrido

Os cupons deste sistema são chamados de *slips*, cujo valor valor é determinado pelo usuário no momento da transação. Os *smart cards* contém contadores que indicam o valor armazenado no dispositivo. O valor total dos *slips* não pode ser superior a este valor armazenado.

Envolvimento da entidade emissora: *off-line*

No momento da transação, apenas o pagador e recebedor precisam se comunicar.

Quantias envolvidas: pequenos pagamentos

O sistema foi projetado para substituir a carteira de dinheiro tradicional, sendo

adequado para valores pequenos a médios.

Hardware necessário: específico.

O sistema usa dois tipos de dispositivos *tamper-proof*: *smart cards* e carteiras eletrônicas. Os *smart cards* são cartões plásticos semelhantes a cartões de crédito que têm um pequeno processador embutido, enquanto as carteiras eletrônicas são aparelhos mais sofisticados que possuem pequenos teclados e visores de cristal líquido. As carteiras usadas no CAFE são capazes de transferir valor para os *smart cards*.

O CAFE funciona com o uso de observadores [8], que são dispositivos em que o banco confia e que ficam dentro da carteira ou *smart card* do usuário. O observador impede que o usuário faça algo que seja contrário aos interesses do banco, enquanto que o *software* da carteira verifica se o observador está fazendo algo que seja contrário aos interesses do usuário. O observador é responsável, por exemplo, pelo controle do saldo do dispositivo.

Papéis envolvidos: pagador, recebedor e banco.

Os pagadores usam seus dispositivos para carregar as ordens de pagamento em branco e para realizar pagamentos. Os recebedores possuem dispositivos capazes de receber as ordens de pagamento, armazená-las e depois depositá-las em um banco. Os bancos assinam as ordens de pagamento emitidas pelos usuários e realizam a compensação dos pagamentos efetuados através do sistema.

Inversibilidade dos papéis: papéis variáveis.

Os usuários do sistema podem tanto atuar como pagadores quanto como recebedores. O único papel fixo é o papel do banco, ou entidade emissora.

Privacidade: existente

O sistema garante que o banco não será capaz de identificar as transações de um usuário se este respeitar as regras do sistema. Os dispositivos *tamper-proof* tornam a fraude mais difícil.

Divisibilidade: nível 1.

O usuário é quem determina o valor dos *slips*.

3.6.2 Requisitos

Integridade: O banco assina todos os *slips*, garantindo a integridade destes. O uso de dispositivos *tamper-proof* ajuda a manter a integridade do sistema.

Robustez: As seguintes características são observadas:

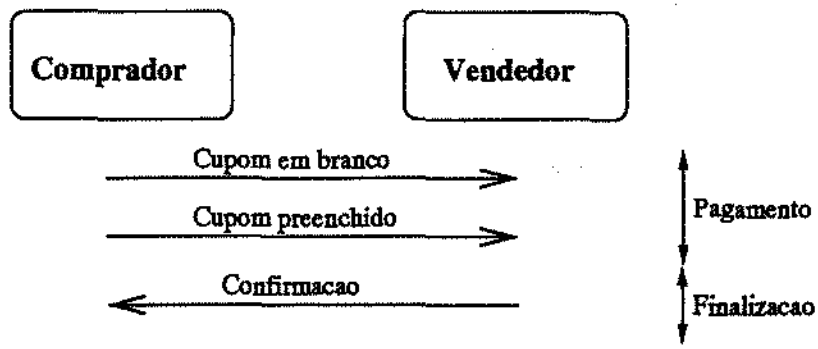


Figura 3.7: Fluxo de mensagens no CAFE

Atomicidade: monetária, garantida pelos protocolos de comunicação.

Consistência: O recebedor e o pagador negociam os parâmetros, garantindo que os participantes estão de acordo quanto aos parâmetros da transação.

Isolamento: As transações só interferem em outras se o usuário tentar usar o *slip* duas vezes. Neste caso a segunda tentativa é ilegal e deve ser impedida pelo observador.

Durabilidade: O sistema implementa um mecanismo para recuperação de *slips* perdidos.

Viabilidade econômica: O sistema evita a necessidade de uma conexão *on-line* durante a transação, o que reduz os custos. Apesar disto, não é adequado para micro-transações por ser bastante complexo e necessitar de *hardware* especial, o que leva a um aumento dos custos de operação do sistema.

Escalabilidade: O sistema é escalável já que os servidores centrais podem atuar *off-line*. Apenas a carga dos dispositivos deve ser feita *on-line*.

Interoperabilidade: O sistema prevê a existência de vários bancos e pode funcionar com moedas de diferentes países.

Auditabilidade: Os depósitos são registrados e o sistema é capaz de detectar múltiplos gastos da mesma moeda.

3.6.3 Funcionamento

As etapas do funcionamento do sistema estão descritas abaixo e o fluxo de mensagens na etapa de transação é apresentado na Figura 3.7.

1. **Relacionamento**
   O usuário deve manter uma conta corrente num dos bancos que participa do sistema e obter deste banco um observador para sua carteira ou um *smart card.*

2. **Retirada**
   O usuário deve (re)carregar seu dispositivo junto a um caixa eletrônico especialmente adaptado ao sistema. O dispositivo do usuário entrará em contato com o seu banco através deste caixa eletrônico e irá gerar as ordens de pagamento em branco, que serão assinadas pelo banco. O dispositivo então armazenará estas assinaturas e o mínimo de informação necessária para gerar as ordens de pagamento novamente quando estas forem preenchidas. O dispositivo indica ao banco o valor total máximo que estas ordens de pagamento podem atingir, e este valor é adicionado ao contador de saldo do observador e retirado da conta do usuário.

3. **Transação**

   (a) **Autenticação**
   Apenas o recebedor deve ser identificado.

   (b) **Pagamento**
   O pagador envia uma ordem de pagamento em branco ao recebedor, que verifica sua validade confirmando a assinatura do banco. O pagador então envia a ordem de pagamento preenchida ao recebedor, que é capaz de verificar sua validade comparando-a com a mesma ordem de pagamento em branco.

   (c) **Recibo**
   O sistema não prevê a emissão de recibo, já que o pagador é capaz de provar que participou de um pagamento se assim o desejar.

   (d) **Finalização**
   O recebedor verifica se a ordem de pagamento está de acordo com as informações recebidas anteriormente, e envia uma confirmação de finalização da transação.

4. **Compensação**
   Após ter recebido a ordem de pagamento, o recebedor pode, a qualquer momento, enviar esta ordem de pagamento ao seu banco, que enviará a ordem de pagamento ao banco do pagador que verificará a validade da ordem de pagamento e se esta já não havia sido usada, e então realizará a transferência do valor correto.

3.6.4 Aspectos de Implementação

Técnicas Criptográficas: é necessário o uso de

Assinatura digital: O sistema usa assinaturas de Schnorr [20], que requerem processamento menos intensivo que os métodos tradicionais.

Conexões de rede: 1 conexão.

Este sistema não foi desenvolvido para a realização de compras numa rede, e presume que os dispositivos do comprador e do vendedor se comuniquem por contato direto ou infra-vermelho.

Formato das ordens de pagamento: Os *slips* em branco consistem de:

- PK_{Blind}: uma chave pública gerada a partir da chave secreta do usuário, mas que não permite sua identificação.
- Duas chaves públicas auxiliares, PK_1 e PK_2, que são desconhecidas do banco.

O *slip* preenchido consiste de:

- As chaves públicas PK_{Blind} e PK_i, onde i indica se o *slip* está sendo usado pela primeira ou segunda vez.
- Identificação do vendedor
- Data
- Valor
- Valor aleatório para garantir que a mensagem é única.

Este *slip* é assinado pelo comprador usando sua chave secreta e um string aleatório, o que impede que o usuário seja identificado.

3.6.5 Comentários e Análises

O CAFE é certamente o sistema mais inovador apresentado neste trabalho e é o único baseado em tecnologias menos conhecidas como observadores e assinaturas de Schnorr.

Sendo fruto de um projeto da Comunidade Europeia, é um projeto internacional que contou com a participação de empresas e centros de pesquisa, o que levou a um sistema que, embora tenha objetivos comerciais, apresenta em seu projeto tecnologias de ponta.

Os primeiros protótipos das carteiras eletrônicas do CAFE já estão em funcionamento, assim como alguns testes piloto. Acreditamos que este será um dos mais importantes sistemas de pagamento fora da Internet no futuro, capaz de rivalizar com sistemas do tipo VISA ELECTRON, VISA CASH ou outros tipos de carteiras eletrônicas em uso no mundo.

3.7 O E-CASH da DIGICASH

O E-CASH [11] da DIGICASH é um dos mais conhecidos sistemas de pagamento na Internet. Este sistema permite a realização de transações na qual o pagador permanece anônimo perante o recebedor e a entidade emissora. O uso de assinaturas às cegas (ver Seção A.3.1) na validação dos cupons garante o anonimato deste sistema. É um sistema em que cada banco participante opera de maneira independente, emitindo e resgatando suas próprias moedas.

Este sistema já vem sendo utilizado na Internet por bancos dos Estados Unidos, Alemanha, Finlândia, Austrália e Suíça. Até o momento da redação deste documento, no entanto, a DIGICASH não havia publicado a especificação do sistema, e a análise abaixo baseou-se em informações extra-oficiais [26].

3.7.1 Tipificação

Modelo de troca: cupons

Os cupons deste sistema são chamados de moedas, sendo numerados e com data de validade. O valor do cupom é indicado pela chave usada pelo banco para assiná-lo, já que o banco não terá acesso aos dados que está assinando por causa do processo de assinaturas às cegas.

Envolvimento da entidade emissora: *on-line*

O banco que emite cada moeda deve ser contatado para confirmar a validade da mesma.

Quantias envolvidas: pequenos pagamentos

O sistema foi projetado para uso na Internet, e as moedas devem ser armazenadas nos computadores dos usuários. Apesar do sistema de pagamento fazer uso de criptografia forte o suficiente para permitir a realização de pagamentos de centenas de dólares ou mais, os computadores nos quais este deve ser usado não apresentam, em geral, segurança suficiente para tanto.

Hardware necessário: uso geral.

O sistema não faz uso de *hardware* especial como *smart cards*, sendo passível de implementação em qualquer computador.

Papéis envolvidos: pagador, recebedor e banco.

O banco valida as transações entre pagadores e recebedores.

Inversibilidade dos papéis: papéis variáveis.

Os usuários do sistema podem tanto atuar como pagadores quanto como recebedores. O único papel fixo é o do banco, ou entidade emissora.

Privacidade: existente

O sistema está baseado em assinaturas às cegas (veja A.3.1) e garante o anonimato do pagador, embora este deva se identificar em caso de disputa. O banco e o recebedor são identificados. O sistema não especifica que os pedidos devam ser enviados criptografados pela rede, o que permite que um observador possa obter algumas informações sobre as transações.

Divisibilidade Nível 2.

Para dividir uma moeda, é necessário trocá-la no banco, realizando um depósito e retirando moedas de valor menor.

3.7.2 Requisitos

Integridade: A integridade das moedas é garantida pelo uso de assinaturas. Algum esquema com redundância deve ser usado para evitar que as moedas assinadas pelo banco com uma chave correspondente a um valor baixo sejam verificadas por uma chave de valor alto e sejam consideradas válidas.

Robustez: O pagador e o recebedor devem concordar quanto ao conteúdo do pedido e preço. As transações são independentes se não há uma tentativa de gastar a mesma moeda duas vezes. O sistema permite a recuperação de moedas perdidas. O sistema apresenta atomicidade monetária.

Viabilidade econômica: A necessidade de contato com o banco durante a transação e o uso de cifras assimétricas para o processo de assinatura tornam o sistema inviável para micropagamentos. Apesar disto, o sistema é perfeitamente adequado à realização de pequenas ou médias transações.

Escalabilidade: O sistema prevê que os usuários que queiram realizar transações devem ter contas no mesmo banco. Assim, os servidores dos bancos são os pontos críticos do sistema. A lista de moedas usadas pode ser outro problema se crescer muito.

Interoperabilidade: O sistema não prevê interoperabilidade, sendo que cada banco opera independentemente, emitindo e recebendo apenas suas próprias moedas.

Auditabilidade: É possível ao usuário provar sua participação em transações, sacrificando seu anonimato. O banco mantém registros das moedas depositadas e pode identificar o usuário que as depositou ou trocou, embora seja incapaz de identificar qual usuário realizou a retirada de uma determinada moeda.

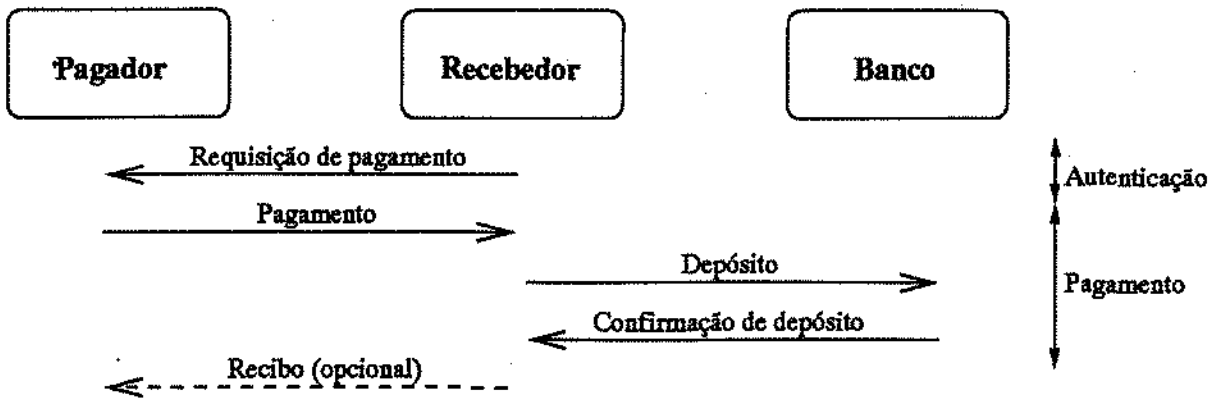


Figura 3.8: Fluxo de mensagens no E-CASH

3.7.3 Funcionamento

A Figura 3.8 apresenta o fluxo de mensagens na etapa de transação do protocolo, que é descrito a seguir:

1. **Relacionamento**
   O usuário deve manter uma conta corrente num dos bancos que participa do sistema e requisitar a este banco a instalação de uma conta E-CASH. Desta conta E-CASH, o usuário poderá realizar retiradas e depósitos de E-CASH. Além disto, o usuário precisa receber e instalar o *software* de carteira eletrônica em seu computador.

2. **Retirada**
   O usuário deve gerar as moedas eletrônicas e enviá-las ao banco para que este as valide pela aplicação de uma assinatura às cegas. O banco recebe as moedas e uma indicação do valor que o usuário deseja associar a cada uma delas. O banco aplica a assinatura com a chave correspondente ao valor solicitado e debita este valor da conta do usuário. A seguir o banco devolve as moedas ao usuário, que retira delas o *blinding factor* e já pode utilizá-las.

3. **Transação**

   (a) **Autenticação**
   O recebedor deve enviar uma requisição de pagamento contendo: valor, descrição dos bens e sua identificação. Esta informação não é cifrada.

   (b) **Pagamento**
   O pagador envia ao recebedor as moedas e um *hashing* das instruções de pagamento cifrados com a chave pública do banco e as instruções de pagamento. O recebedor envia os cupons ao banco para verificação. O recebedor deve proceder então à etapa de depósito dos cupons.

(c) **Recibo**
Não há

(d) **Finalização**
Não existe etapa de finalização neste sistema.

4. **Compensação**
Após o banco ter confirmado a validade dos cupons, o recebedor tem duas alternativas: depositar os cupons em sua conta ou gerar novos cupons no mesmo valor total, que serão validados pelo banco.

3.7.4 Aspectos de Implementação

Técnicas Criptográficas: as pimitivas necessárias são:

Cifra simétrica: Triplo-DES, usado em conjunto com a cifra assimétrica para tornar o processo mais eficiente, pelo modelo do envelope seguro.

Cifra assimétrica: RSA.

Assinatura digital: RSA, usado para assinar *hashings* das mensagens.

Hashing: SHA.

Conexões de rede: duas conexões.
Uma conexão é necessária entre pagador e recebedor, e uma entre o recebedor e o banco para verificação da validade das moedas.

Formato dos cupons: Os cupons usados consistem de um número de série, uma data de validade e da assinatura do banco. O número de série tem um formato especial para permitir a identificação dos números válidos.

3.7.5 Comentários e Análises

O E-CASH é o sistema de pagamento que mais protege a privacidade do usuário, oferecendo transações completamente anônimas. Os pagamentos realizados por um usuário não podem ser identificados nem rastreados sem a conivência deste. É possível diminuir o nível de privacidade do sistema pela interceptação dos pedidos de compra e durante a fase de negociação de bens e preços, que não faz parte do protocolo.

O sistema está baseado num servidor central que deve assinar *on-line* todas as moedas, o que pode levar a uma sobrecarga do servidor se o número de usuários é muito grande. O mesmo servidor deve verificar todas as moedas ao serem depositadas, o que implica num aumento de carga. Juntamente com a possibilidade de ter de tratar um banco de

dados de números de série das moedas que pode crescer enormemente, estes fatos limitam a escalabilidade do sistema.

Apesar disto, o E-CASH é um sistema interessante que pode se tornar bastante popular se a demanda por privacidade na Internet aumentar. É possível extender o sistema para tratar uma rede de servidores que se comuniquem e, sendo um sistema *on-line*, existe a possibilidade de implantação de servidores de câmbio, para permitir que os usuários comprem de vendedores foram do domínio do servidor local.

3.8 INTERNET KEYED PROTOCOL

A divisão de pesquisas da IBM desenvolveu o IKP [14] para ser um sistema adequado à realização de transações pela Internet usando contas bancárias ou cartões de crédito. No fundo, trata-se de um sistema adequado para a transmissão de números de contas pela Internet.

O sistema faz uso de criptografia forte, com cifras simétricas e assinaturas digitais e consiste de três níveis distintos. No primeiro nível, chamado 1KP, apenas as entidades controladoras possuem um certificado vinculando sua identidade a uma chave pública. No segundo nível (2KP), os vendedores já devem ter um tal certificado, e consequentemente, um par de chaves para uso em assinaturas. No terceiro nível (3KP), todos os envolvidos, inclusive os compradores, devem ter cada um seu par de chaves e o certificado correspondente. Esta hierarquia permite a implantação gradual do sistema, com um aumento da segurança à medida que a infra-estrutura se torna disponível.

3.8.1 Tipificação

Modelo de troca: notacional

O iKP é baseado nos sistemas tradicionais de processamento de transações de cartões de crédito ou débito. Foi projetado como uma maneira segura de transferir números de cartões ou de contas bancárias pela Internet. Portanto, usa o modelo notacional.

Envolvimento da entidade emissora: *on-line*

É necessária a participação da entidade que processa as transações de cartões de crédito ou do banco responsável pela conta, já que apenas esta entidade está habilitada a validar a transação. Esta entidade fará a interface entre o sistema iKP e o sistema bancário tradicional, que gerencia as contas usadas nas transações.

Quantias envolvidas: pequenos a grandes pagamentos

O custo das transações com cartões de crédito torna o sistema inadequado para a realização de micropagamentos. O limite estabelecido para cada cartão indica o

teto para o valor das transações. Foi proposta uma variação do iKP para lidar com micropagamentos.

Hardware necessário: uso geral
O sistema não necessita do uso de nenhum tipo de maquinário dedicado ou especial.

Papéis envolvidos: comprador, vendedor e entidade controladora.
A entidade controladora é a entidade habilitada a processar pagamentos realizados com cartões de crédito, entre compradores e vendedores.

Inversibilidade dos papéis: inexistente.
Os papéis desempenhados no sistema são fixos, assim como em transações normais com cartões de crédito. Uma entidade cadastrada como vendedor não pode participar como comprador, a não ser que se cadastre como tal. Um sistema de nível 3, ou 3KP, poderia permitir inversibilidade de papéis para transações com números de contas bancárias.

Privacidade: existente
O sistema protege a identidade do usuário (número do cartão de crédito ou conta) do vendedor e o conteúdo do pedido não é informado à entidade controladora. O uso extensivo de criptografia e certificados garante a segurança e privacidade das mensagens trocadas. O sistema não permite transações totalmente anônimas por fazer uso de cartões de crédito ou contas bancárias.

Divisibilidade: nível 4, o sistema é notacional.

3.8.2 Requisitos

Integridade: O sistema usa assinaturas digitais e certificados para proteger as mensagens e as implementações devem garantir a integridade dos dados críticos que sejam armazenados localmente. A mensagem inicial do protocolo não é cifrada, o que permitiria que intrusos alterassem a proposta de negócio que o vendedor manda ao comprador. Entretanto, nos níveis 2 e 3 do protocolo, o vendedor deve ter um par de chaves e um certificado, o que permitiria que esta primeira mensagem já fosse cifrada.

Robustez: O sistema apresenta as seguintes características:

Atomicidade: monetária. A transação só é realmente processada pela entidade responsável pelo processamento de transações com cartões de crédito para o vendedor, assim, se as informações chegam até ela corretamente a transação

ocorre, senão a transação não acontece. As outras entidades envolvidas serão informadas. As implementações deveriam permitir que os participantes verifiquem mais tarde se a transação foi processada.

Consistência: O protocolo permite garantir que todos os participantes estejam de acordo com os detalhes da transação: quantia e objeto ou serviço adquirido (veja no Capítulo 4).

Isolamento: A interferência entre transações só ocorre quando o usuário ultrapassar o limite de seu cartão ou saldo em conta. Neste caso, transações posteriores serão negadas.

Durabilidade: O sistema tem um mediador, que é a entidade que processa as transações, e este é quem indica o estado do sistema. Assim, todos devem poder consultar o mediador e tomar conhecimento do estado atual. O sistema tradicional de processamento de transações de cartões de crédito provê mecanismos para garantir a durabilidade que devem ser usados quando do processamento através do IKP.

Viabilidade econômica: O sistema é viável economicamente para transações cujos valores estão na faixa daqueles normalmente usados em transações tradicionais com cartões de crédito ou cheques, já que deve usar os canais de processamento já implantados para estes instrumentos.

Escalabilidade: O sistema é tão escalável quanto o sistema tradicional de cartões de crédito. Podem participar vários bancos ou companhias de cartões de crédito, fazendo uso de uma infra-estrutura de certificação comum.

Interoperabilidade: O IKP não prevê integração com outros sistemas de pagamento eletrônico. O sistema poderia ser usado como ponto de partida para alguns sistemas de micropagamentos, que necessitam da realização de macropagamentos para inicializar o sistema.

Auditabilidade: O uso de *logs* e assinaturas digitais permite verificar com bastante eficiência o funcionamento do sistema e detectar problemas, falhas ou uso indevido. O sistema foi projetado de maneira que, em seu nível 3, garanta registros de transações que não permitam disputas.

3.8.3 Funcionamento

As etapas do funcionamento do sistema são apresentadas a seguir e na Figura 3.9.

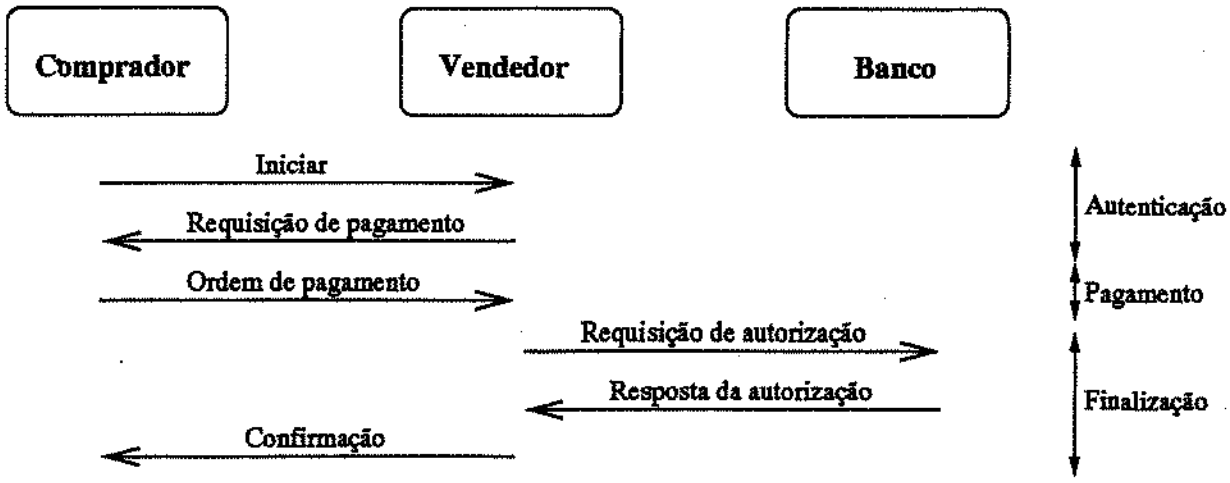


Figura 3.9: Fluxo de mensagens no INTERNET KEYED PROTOCOL

1. **Relacionamento**
   O usuário deve escolher uma das entidades controladoras participantes do sistema e estabelecer com ela um relacionamento comercial. Todos os usuários devem se registrar com as entidade controladoras. Caso o sistema esteja operando nos níveis 2 ou 3, os usuários podem ter de gerar um par de chaves e obter um certificado para sua chave pública. Os usuários recebem uma cópia da chave pública mestra da hierarquia de certificação, que é usada para verificar a validade dos certificados recebidos. Nesta etapa o usuário deve adquirir o *software* necessário ao uso do sistema. O IKP pressupõe que o usuário já fazia uso do sistema de pagamentos tradicional subjacente ao IKP, seja ele o sistema de cartões de crédito ou de contas bancárias.

2. **Retirada**
   Não há.

3. **Transação**

   (a) **Autenticação**
   Nos níveis 2 ou 3, o usuário deve enviar seu certificado ao outro participante da transação, que verificará sua validade. No nível 1, os participantes trocam apenas os dados necessários à inicialização do sistema, como uma identificação de transação e uma fatura, que contém descrição do pedido e valor. Nesta fase, o vendedor informa ao comprador qual entidade controladora foi escolhida para processar a transação, enviando o certificado desta entidade controladora.

   (b) **Pagamento**
   O usuário recebe a fatura, verifica sua validade e gera uma ordem de paga-

mento, que contém o valor, um *hashing* da descrição dos produtos ou serviços comprados, o número de sua conta ou cartão de crédito e, opcionalmente, uma senha. Esta ordem de pagamento é cifrada com a chave pública da entidade controladora escolhida pelo vendedor que recebeu na etapa anterior.

(c) **Recibo**
Não há emissão de recibo.

(d) **Finalização**
Após ter recebido a ordem de pagamento, o vendedor entra em contato con a entidade controladora para verificar a validade da transação. A entidade controladora informa se a transação foi válida ou não e o vendedor passa esta informação ao comprador.

4. **Compensação**
A entidade controladora realiza a compensação automaticamente quando o vendedor valida a transação. Assim sendo, o vendedor não precisa tomar nenhuma atitude para que a transação seja compensada após ter sido informado de que a transação foi validada.

3.8.4 Aspectos de Implementação

Técnicas Criptográficas: são necessários:

Cifra assimétrica: RSA.

Assinatura digital: RSA, usando as mesmas chaves que a cifra assimétrica.

Certificados: do tipo PKCS.

Hashing: MD5.

Conexões de rede: duas conexões.
Uma conexão é necessária entre pagador e recebedor, e uma entre o recebedor e o banco para verificação da validade do pagamento e compensação.

Formato das ordens de pagamento: Cada ordem de pagamento contém:

- valor
- *hashing* de informações de identificação que são do conhecimento do comprador e do vendedor.
- número da conta do pagador
- número aleatório escolhido pelo comprador

- opcionalmente a senha da conta do comprador

Estas informações são cifradas com a chave pública da entidade controladora.

3.8.5 Comentários e Análises

Este sistema não será implementado comercialmente, mas as lições tiradas de seu projeto certamente ajudaram a IBM no esforço para a construção das especificações do SET (Seção 3.3). A sua variante mais completa, o 3KP é bem semelhante ao SET e ao CYBERCASH (seção 3.14).

O iKP seria uma excelente alternativa como o primeiro sistema de pagamento baseado em certificados para a Internet, já que permite uma distribuição gradual de certificados. Depois que infra-estrutura estivesse pronta, outros sistemas poderiam aproveitá-la, diminuindo seus custos de implantação. Este sistema ajudaria a trazer o varejo para a Internet, permitindo a implantação gradual da estrutura de certificação.

3.9 MILLICENT

O MILLICENT [21, 12] é um sistema para a realização de micropagamentos projetado pela DIGITAL EQUIPMENT CORPORATION. É um sistema otimizado para a realização de compras repetidas de pequenos valores. O sistema pressupõe a existência de um ou mais corretores, que vendem cupons em nome das entidades que estão vendendo produtos ou serviços. Estes corretores tem a função de agrupar as diversas compras dos usuários, sejam de um ou mais vendedores, de tal modo a tornar economicamente viável o uso de sistemas para a realização de macropagamentos.

Os cupons usados são específicos para cada vendedor, que deve emitir os cupons e vendê-los ao corretor, ou autorizar o corretor a emitir cupons em seu nome e fazer o acerto mais tarde. Os cupons tem um valor fixo, mas o sistema permite que o vendedor devolva troco caso o valor do cupom seja maior que o valor do produto ou serviço.

3.9.1 Tipificação

Modelo de troca: cupons

Cada vendedor emite cupons que tem um determinado valor monetário. O usuário compra estes cupons de um corretor e pode então realizar transações com este vendedor.

Envolvimento da entidade controladora: *off-line*

Os cupons são emitidos por, ou em nome de, um vendedor, sendo que este é capaz

de identificar e validar seus próprios cupons. Assim, a entidade controladora, ou corretor, não precisa ser contactada durante a transação.

Quantias envolvidas: micropagamentos
O sistema foi projetado para venda de informações na Internet e para minimizar o custo apresentado pelo uso de algoritmos criptográficos. Assim, o nível de segurança foi reduzido e o sistema é adequado apenas para transferências de valores entre US$ 0,001 e US$ 5,00.

Hardware necessário: uso geral
O sistema é todo baseado em *software*, não fazendo uso de *hardware* especial.

Papéis envolvidos: comprador, vendedor e corretor.
O comprador adquire cupons junto ao corretor e pode então realizar transações com o vendedor que emitiu os cupons.

Inversibilidade de papéis: inexistente
O sistema distingue claramente entre compradores e vendedores. O vendedor deve emitir cupons e ter um relacionamento com um corretor que se encarrega de vender os cupons aos compradores.

Privacidade: existente
O sistema prevê um mecanismo para proteger as informações e cupons em trânsito pela rede. Se for usado um sistema anônimo para compra de cupons, a privacidade do comprador é protegida. O vendedor é sempre identificado, já que os cupons lhe são específicos.

Divisibilidade nível 3.
Não há necessidade de trocar cupons por outros de menor valor já que o sistema prevê a devolução de troco: o vendedor deve sempre devolver um cupom ao comprador após receber o pagamento. Este cupom deve ter valor correspondente ao troco.

3.9.2 Requisitos

Integridade: O sistema usa *hashings* para garantir a integridade dos cupons.

Robustez: O sistema não possui um mecanismo que garanta sua consistência. Como as transações envolvem apenas duas entidades e pequenos valores, os requisitos de robustez foram relaxados.

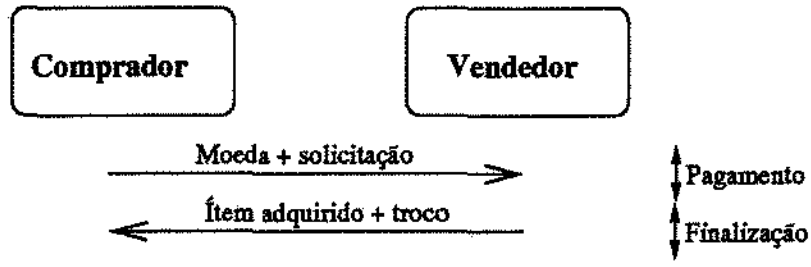


Figura 3.10: Fluxo de mensagens no MILLICENT

Viabilidade econômica Os projetistas tiveram a preocupação em diminuir os custos do sistema, reduzindo o número de operações criptográficas e usando apenas *hashings*. Assim, o protocolo tem custo compatível com a realização de micropagamentos. Testes divulgados com implementações deste sistema confirmam sua viabilidade [12].

Escalabilidade O sistema é bem escalável, sendo que os testes mostram que o principal problema de desempenho pode ser a realização das conexões de rede [12].

Interoperabilidade O sistema não prevê interoperabilidade.

Auditabilidade Cada vendedor controla seus próprios cupons, e é responsável pela manutenção de informação que permita auditar o sistema.

3.9.3 Funcionamento

A Figura 3.10 apresenta o fluxo de mensagens na etapa de transação do protocolo, cujo funcionamento é descrito a seguir:

1. **Relacionamento**
   O usuário deve escolher um corretor de quem comprará os cupons válidos para um determinado vendedor. O corretor deve ser habilitado pelo vendedor em questão. O estabelecimento do relacionamento consiste basicamente em definir que tipo de sistema de macropagamentos será usado para adquirir cupons MILLICENT. O usuário deve então comprar deste corretor um cupom MILLICENT válido para trocas com o próprio corretor. Este cupom será gasto para comprar cupons para uso com os vendedores.

2. **Retirada**
   O comprador envia o cupom do corretor e indica o vendedor e o valor do cupom desejado para este vendedor. O corretor irá obter o cupom do vendedor (ele pode já ter comprado do vendedor ou então estar habilitado a gerar cupons) no valor correto e devolver ao usuário o cupom do vendedor e um cupom com o troco, se houver.

3. **Transação**

   (a) **Autenticação**
   O comprador só se identifica se quiser, e por meios fora do escopo do sistema, enquanto o vendedor é sempre identificado, já que os cupons lhe são específicos.

   (b) **Pagamento**
   O comprador envia ao vendedor o cupom e uma requisição, indicando o produto ou serviço desejado. O sistema prevê o uso de conexões seguras com dois níveis: uso de cifras simétricas, cuja chave é informada ao usuário pelo corretor quando da retirada, ou uso de um esquema simplificado de assinatura, baseado em *hashing* e usando como segredo a chave já citada. Os cupons podem também ser enviados pela rede sem nenhum tipo de proteção.

   (c) **Recibo**
   O sistema não faz uso de recibos.

   (d) **Finalização**
   O vendedor deve devolver o troco ao usuário. O troco é enviado pela rede com o mesmo nível de segurança que o pagamento, consistindo de um cupom comum que pode ser usado em outras compras.

4. **Compensação:**
   Não há necessidade de etapa de compensação, já que o vendedor é responsável pela geração dos seus próprios cupons. O vendedor e o corretor devem concordar num método para realização do acerto referente aos cupons vendidos.

3.9.4 Aspectos de Implementação

Técnicas Criptográficas: o sistema faz uso de:

Hashing: MD5, é o único tipo de algoritmo criptográfico usado, o que permite aumentar a eficiência do sistema.

Conexões de rede: uma conexão.
Apenas o vendedor e o comprador precisam se comunicar durante a transação.

Formato dos cupons: cada cupom consiste dos seguintes campos:

- identificação do vendedor.
- valor
- identificador do cupom

- identificador do comprador, que não precisa ter ligação com a identidade real do comprador, mas é necessário para permitir a transmissão segura dos cupons. Este identificador é usado para gerar as chaves que permitem a transmissão segura dos cupons pela rede.
- data de validade
- informações extras opcionais
- certificado de autenticidade, que é gerado como um *hashing* do cupom concatenado com um segredo escolhido pelo vendedor.

3.9.5 Comentários e Análises

O MILLICENT é o mais conhecido sistema para micropagamentos na Internet, atualmente sendo testado em um projeto piloto. É um sistema simples que evita o uso de cifras simétricas ou assimétricas e baseia sua segurança no uso de funções de espalhamento.

Assim como outros sistemas para micropagamentos, é adequado para venda de informações e substituição de acesso por assinaturas. Apresenta a capacidade de compras em diversos vendedores sem um aumento muito grande dos custos, embora seja otimizado para micropagamentos repetidos, já que, a cada compra num novo vendedor, é necessário entrar em contato com o corretor para adquirir cupons específicos.

Dentre os sistemas apresentados aqui, o MILLICENT rivaliza com o GLOBEID, tendo as vantagens de ser um produto de uma empresa de renome e de apresentar custos bem mais baixos, embora ofereça menos segurança.

3.10 NETBILL

O NETBILL [22] foi desenvolvido na CARNEGIE MELLON UNIVERSITY e é um esquema para venda de informações ou programas pela Internet. É um protocolo completo, que inclui uma fase de negociação de preços e a entrega dos bens é garantida. Neste sistema, um servidor central é responsável pelas contas dos usuários, e pode creditar estas contas a partir de contas correntes em bancos conveniados ou a partir de cartões de crédito. Neste sistema existem os compradores, os vendedores e o servidor central (entidade controladora), cada um com seu papel bem definido.

O sistema faz uso de um mecanismo de autenticação chamado *Public Key Kerberos* (Seção A.5.3), que é uma variação do sistema *Kerberos* (Seção A.5.2) onde não há necessidade de um servidor *Kerberos* para autenticar os usuários. Em compensação, é necessária uma infra-estrutura para emissão de certificados e todos os participantes devem ter um par de chaves para uma cifra assimétrica.

3.10.1 Tipificação

Modelo de troca: notacional

O sistema trabalha com contas especiais gerenciadas por uma entidade central, sendo que as transferências de valor são feitas entre estas contas.

Envolvimento da entidade emissora: *on-line*

Para que as transferências possam acontecer, a entidade que gerencia as contas deve ser acionada. O vendedor repassa uma ordem de pagamento a esta entidade, que valida os dados e realiza a transferência entre as contas.

Quantias envolvidas: micro e pequenos pagamentos

O sistemas foi projetado para venda de informações *on-line*, o que envolve tipicamente quantias na faixa dos micropagamentos. Pelo nível de segurança apresentado, acreditamos que o sistema seja capaz de lidar também com pequenos pagamentos.

Hardware necessário: uso geral

O sistema não faz uso de *hardware* específico e pode ser implementado em qualquer tipo de computador.

Papéis envolvidos: comprador, vendedor e entidade controladora.

A entidade controladora age como um mediador em caso de disputa entre comprador e vendedor e realiza as transferências entre contas.

Inversibilidade de papéis: inexistente

O sistema faz uma clara distinção entre vendedores e compradores e o *software* usado deve prover funções bem distintas.

Privacidade: existente

O uso de cifras em todas as mensagens e o sistema de autenticação dos participantes permitem que as mensagens e informações sejam mantidas secretas quando necessário.

Divisibilidade: nível 4, o sistema é notacional.

3.10.2 Requisitos

Integridade: Assinaturas digitais e *hashings* são usados para garantir a integridade das mensagens.

Robustez: O sistema apresenta as seguintes características:

Atomicidade: entrega certificada. Os bens adquiridos são entregues e é possível verificar se os bens entregues correspondem aos bens negociados.

Consistência: O protocolo garante que todos os participantes estão de acordo com os parâmetros da transação.

Isolamento: Desde que o comprador tenha crédito, as transações são independentes. Se o comprador ultrapassar o seu saldo, novas transações passam a ser negadas em função de transações anteriores.

Durabilidade: A entidade controladora é capaz de informar o estado do sistema aos outros participantes. O servidor desta entidade deve prover mecanismos internos de durabilidade.

Viabilidade econômica: O custo de operação do sistema permite que sejam realizados micro ou pequenos pagamentos. Apesar disto, este é um dos mais caros sistemas de micropagamentos, por usar bastante as cifras assimétricas.

Escalabilidade: O sistema tem como gargalo o servidor da entidade controladora.

Interoperabilidade: O sistema não prevê interoperabilidade.

Auditabilidade: Todas as transações passam pelo servidor central que pode prover mecanismos de *logs*, extratos etc. Além disto todas as transações são autenticadas com base em assinaturas digitais ou *hashings*.

3.10.3 Funcionamento

As etapas do funcionamento do protocolo são descritas a seguir e a Figura 3.11 apresenta o fluxo de mensagens trocadas na etapa de transação.

1. **Relacionamento**
   O usuário que quiser fazer uso do NETBILL deve se cadastrar com o servidor central e obter deste um número de conta. É necessário obter também um par de chaves e um certificado para a chave pública. Neste momento o usuário precisa também obter o *software* necessário ao uso do sistema.

2. **Retirada**: não há.

3. **Transação**

   (a) **Autenticação**
   Antes de poder participar de transações, o usuário deve se autenticar, usando o *Public Key Kerberos* com o vendedor e o servidor central. O vendedor também

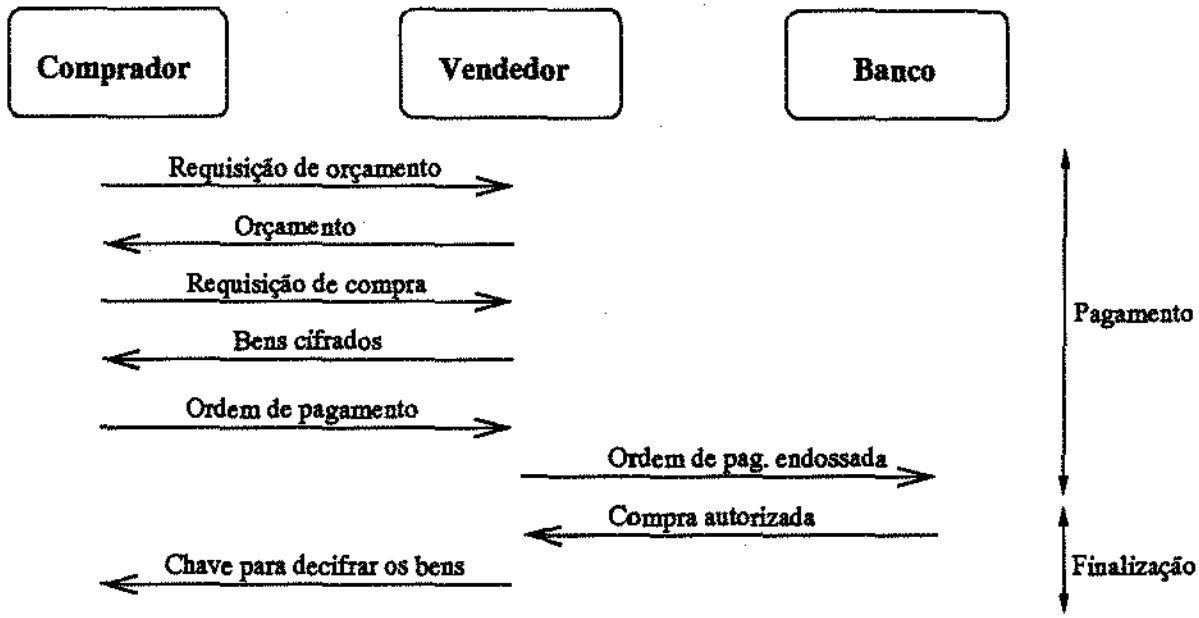


Figura 3.11: Fluxo de mensagens no NETBILL

deve se autenticar com o servidor central. Em outras palavras, para iniciar a transação, o comprador e o vendedor já devem ter obtido seus tíquetes *Kerberos*, que contém chaves de sessão.

(b) **Pagamento**
A transação se inicia quando o comprador requisita um orçamento e o vendedor indica o preço do item escolhido. Tendo recebido o preço, o comprador envia uma requisição de compra e o vendedor devolve os bens negociados cifrados com uma chave de sessão. O comprador verifica se os bens foram recebidos íntegros, e assina e envia uma ordem de pagamento. A partir deste momento o comprador não pode mais cancelar a transação. O vendedor recebe a ordem de pagamento e a envia ao servidor central juntamente com a chave de sessão usada para cifrar os bens enviados ao comprador.

(c) **Recibo**
O recibo é gerado pelo servidor central quando este realiza a transferência do valor da transação entre as contas do comprador e do vendedor. O recibo contém a chave para decifrar os bens que o comprador recebeu e é enviado ao vendedor para ser repassado ao comprador.

(d) **Finalização**
O servidor central informa ao vendedor se a transação foi bem sucedida e, caso afirmativo, envia também o recibo, que o vendedor deve repassar ao comprador. O servidor central mantém um *log* dos recibos para verificação em caso de

disputa.

4. **Compensação**
   A compensação é feita assim que o servidor central recebe a ordem de pagamento, ao fim da etapa de pagamento.

Observações:

1. Este método pode ser adaptado para venda de bens físicos se for retirada a eta de envio dos bens criptografados.

2. Se o vendedor não enviar o recibo ao comprador, este último pode requisitar ao servidor central uma cópia do recibo, bastando indicar a identificação da transação.

3.10.4 Aspectos de Implementação

Técnicas Criptográficas: o sistema necessita dos seguintes itens:

Cifra simétrica: DES.

Cifra assimétrica: RSA.

Certificados: formato não especificado.

Hashing: SHA.

Conexões de rede: duas conexões.
O comprador deve enviar a ordem de pagamento ao vendedor, que irá repassá-la à entidade controladora para compensação.

Formato das ordens de pagamento: As ordens de pagamento consistem de duas partes, a primeira contendo:

1. identificação do comprador,
2. identificação e preço das mercadorias,
3. identidade do vendedor e
4. um código de verificação dos produtos (CRC).

Esta parte pode ser lida pelo vendedor e pela entidade controladora. A Segunda, contendo um tíquete, que autentica o comprador, o número de sua conta e um campo de informações extras, só pode ser lido pela entidade controladora.

3.10.5 Comentários e Análises

Este sistema é um dos mais completos estudados aqui por tratar todas as etapas de uma transação de comércio eletrônico. É um protocolo destinado à venda de bens de informação que trabalha na faixa dos micro a pequenos pagamentos. O custo de operação é um tanto elevado se comparado com outros sistemas para microtransações e o NETBILL deverá ser usado principalmente para a venda de bens de informação de valor acima de US$ 1,00.

Este sistema é o mais adequado para venda de *software* por *download*, por permitir a realização de transações de pequeno valor e garantir a entrega de bens de informação.

É um sistema interessante por garantir a atomicidade das transações, desde o pagamento até a entrega dos bens. O conceito de atomicidade apresentado pelos autores deste sistema poderia ser muito útil aos projetistas de outros sistemas.

Ainda não temos conhecimento de implementações comerciais do NETBILL.

3.11 NETCASH

O grupo de comércio eletrônico da UNIVERSITY OF SOUTHERN CALIFORNIA desenvolveu dois sistemas de pagamento: o NETCASH [19] e o NETCHEQUE (seção 3.12). O primeiro funciona por cupons e o segundo é um sistema notacional. O NETCHEQUE é usado principalmente como método de compensação para o NETCASH.

O NETCASH é um sistema de cupons com anonimato restrito, o que significa que é possível rastrear os pagamentos de um usuário. O principal mecanismo usado para melhorar o nível de privacidade do comprador é a troca de cupons, que pode ser realizada de forma anônima. Assim, um usuário pode tentar trocar seus cupons para tornar mais difícil a identificação de seus gastos. Claramente, este processo não garante anonimato total.

3.11.1 Tipificação

Modelo de troca: cupons

Os cupons deste sistema são chamados de moedas, sendo numerados e contendo uma identificação da entidade emissora, assim como data de validade e valor.

Envolvimento da entidade emissora: *on-line*

O banco que emite cada moeda deve ser contatado para confirmar a validade da moeda. Os bancos devem manter registros dos números das moedas em circulação.

Quantias envolvidas: pequenos pagamentos

Os cupons usados no sistema são assinados pela entidade emissora, o que garante

certa segurança. O uso de assinaturas digitais em cada moeda pode inviabilizar o sua aplicação em microtransações.

Hardware necessário: uso geral

O sistema não faz uso de *hardware* especial, sendo passível de implementação em qualquer tipo de computador.

Papéis envolvidos: pagador, recebedor e servidor de moeda.

Os usuários podem assumir tanto o papel de pagador quanto o papel de recebedor, o que permite que quaisquer dois usuários transfiram valores entre si. Podem existir diversos servidores de moeda, que formam uma rede de compensações. Os usuários podem trabalhar com qualquer servidor de moeda do sistema.

Inversibilidade dos papéis: papéis variáveis.

Os usuários do sistema podem tanto atuar como pagadores quanto como recebedores. O único papel fixo é o papel do servidor de moeda, ou entidade emissora.

Privacidade: existente

O anonimato oferecido pelo sistema é parcial, já que o banco seria capaz de identificar as transações realizadas por determinados usuários. Os projetistas indicam que isto seriam anti-econômico e que a existência de vários bancos faria com que a garantia da privacidade do usuário se tornasse um diferencial de mercado. Uma maneira indicada para melhorar o nível de anonimato do comprador é trocar as moedas em vários bancos antes de usá-las. Assim apenas um acordo entre todos os bancos permitiria a identificação do usuário. O vendedor pode tentar manter sua privacidade trocando as moedas que receber, ao invés de depositá-las.

Divisibilidade: nível 2.

Para obter cupons de menor valor, o usuário deve trocá-los no banco.

3.11.2 Requisitos

Integridade: a integridade das moedas é garantida pelo uso de assinaturas digitais.

Robustez: não é tratada de maneira explicita.

Viabilidade econômica: O servidor de moeda pode gerar cupons *off-line* e precisa apenas verificar a sua assinatura e a validade de um número de série *on-line*. Assim acreditamos que o sistema é viável economicamente. Talvez possa ser implementado de forma a tornar viáveis as microtransações.

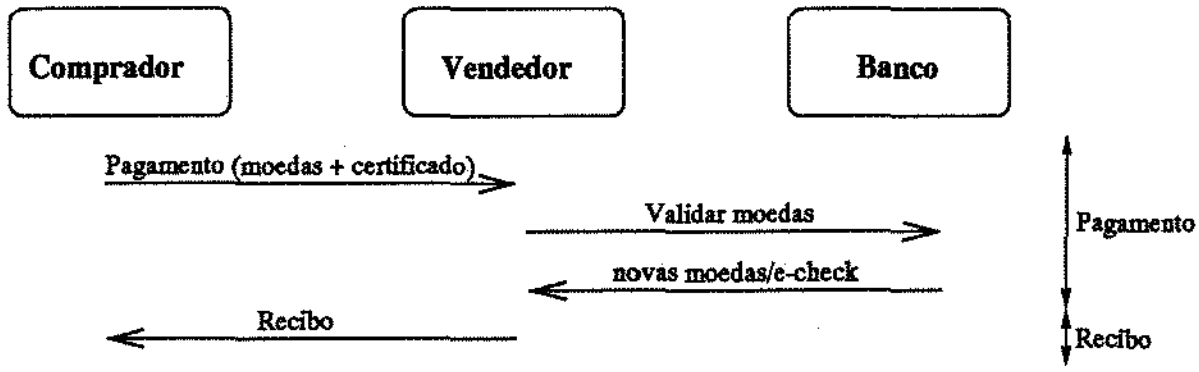


Figura 3.12: Fluxo de mensagens no NETCASH

Escalabilidade: o sistema permite a existência de várias entidades emissoras e a incorporação de novos usuários é bastante fácil. O principal problema quanto à escalabilidade do sistema é a possibilidade do número de moedas válidas de um único banco crescer a ponto de tornar invíavel a manipulação do banco de dados de números de moedas necessário ao funcionamento do sistema.

Interoperabilidade: o sistema prevê a existência de diversas entidades emissoras e a interoperabilidade com o sistema NETCHEQUE. Sendo um sistema *on-line*, é possível a implantação de um serviço de conversão de moedas para outros sistemas.

Auditabilidade: as moedas contém a identificação da entidade emissora e um número de série. Nenhum outro mecanismo de auditoria é especificado.

3.11.3 Funcionamento

As etapas do funcionamento são mostradas abaixo. A Figura 3.12 apresenta os detalhes da etapa de transação deste sistema.

1. **Relacionamento**
   O usuário deve escolher o servidor de moeda com quem deseja trabalhar e negociar qual método de pagamento será usado para a compra de cupons NETCASH. O usuário deve também obter o *software* necessário ao uso do sistema, além do certificado com chave pública do servidor de moeda.

2. **Retirada**
   O usuário deve usar o método escolhido quando do estabelecimento do relacionamento com o servidor de moeda para comprar cupons NETCASH. O usuário envia o pagamento e uma requisição, que indica os valores dos cupons que deseja, cifrados com a chave pública do servidor. Quando o servidor NETCASH recebe o pagamento

e a requisição de cupons, este servidor vai gerar os cupons requisitados, assiná-los e enviá-los ao usuário, cifrados com uma chave de sessão escolhida pelo usuário.

3. **Transação**

   (a) **Autenticação**
   O pagador não precisa se identificar, enquanto que o recebedor deve enviar uma chave pública certificada para o pagador.

   (b) **Pagamento**
   O pagador envia os cupons, uma indicação dos itens adquiridos, e duas chaves de sessão, todos cifrados com a chave pública do vendedor. O vendedor decifra os cupons e os envia ao seu servidor de moeda. O servidor de moeda vai verificar a validade dos cupons com o servidor que os emitiu e retorna ao recebedor um cheque eletrônico ou uma quantidade de cupons correspondente ao valor da transação.

   (c) **Recibo**
   Se o servidor indica uma transação válida, o recebedor emite um recibo assinado e o envia ao pagador. Uma indicação de transação válida ocorre quando o servidor envia ao recebedor os novos cupons ou o cheque no valor correto.

   (d) **Finalização**
   Não existe etapa de finalização neste sistema.

4. **Compensação**
   O depósito dos cupons recebidos ocorre no momento da verificação de sua validade.

3.11.4 Aspectos de Implementação

Técnicas Criptográficas: Os algoritmos não são especificados, sendo que o sistema faz uso de: cifra simétrica, cifra assimétrica, assinatura digital, certificados, *hashing*.

Conexões de rede: duas conexões.
Uma conexão é necessária para que o comprador pague ao vendedor e outra para que o vendedor entre em contato com o servidor de moeda para verificar a validade das moedas que recebeu.

Formato dos cupons: cada cupom consiste de:

- nome do servidor que emitiu a moeda
- endereço deste servidor
- data de validade

- número de série
- valor

Cada moeda é assinada pelo servidor que a emitiu.

3.11.5 Comentários e Análises

O NETCASH é mais um sistema desenvolvido numa universidade, embora tenha pretensões comerciais. Junto com o NETCHEQUE, apresentado abaixo, forma um sistema completo para pagamentos que consiste de um sistema baseado em cupons e um sistema notacional. A idéia dos projetistas é usar o NETCASH para transações de menor valor e aquelas transações em que for necessário o anonimato.

O NETCASH é bem similar ao PAYME(Seção 3.13), embora mais complexo. É um sistema bem escalável que tenta garantir o anonimato dos usuários permitindo a troca de cupons sem a necessidade de autenticação. Embora aumente o grau de anonimato do sistema, este procedimento não consegue garantir a realização de transações anônimas como no E-CASH (ver Seção 3.7).

3.12 NETCHEQUE

O NETCHEQUE [23] é um dos sistemas de pagamento desenvolvidos na USC. É um sistema que imita o funcionamento dos cheques tradicionais e usa a autenticação baseada no KERBEROS, não fazendo uso de cifras assimétricas. Para assinar um cheque, o usuário deve obter um tíquete com o servidor KERBEROS e usar a informação contida neste tíquete para gerar sua assinatura. Todas as entidades devem ser cadastradas nos servidores KERBEROS.

3.12.1 Tipificação

Modelo de troca: notacional

O sistema imita o funcionamento dos cheques bancários, sendo então um sistema notacional.

Envolvimento da entidade emissora: *on-line*

O uso de servidores baseados no *Kerberos* obrigam o sistema a funcionar *on-line*.

Quantias envolvidas: pequenos pagamentos

A segurança do sistema depende da segurança dos servidores e dos algoritmos criptográficos usados. O sistema é adequado para pequenos pagamentos, embora seu custo possa vir a ser pequeno o suficiente para viabilizar micropagamentos.

Hardware necessário: uso geral

O sistema não faz uso de *hardware* especial como *smart cards* ou outros mecanismos deste tipo, sendo passível de implementação em qualquer tipo de computador.

Papéis envolvidos: pagador, recebedor e banco.

O sistema prevê a existência de vários bancos, que mantêm as contas dos usuários do sistema e se comunicam para compensar os cheques emitidos.

Inversibilidade dos papéis: papéis variáveis.

Os usuários do sistema podem tanto atuar como pagadores quanto como recebedores. O único papel fixo é o papel do banco, ou entidade emissora.

Privacidade: inexistente

Não são previstos mecanismos para garantir a privacidade dos usuários ou dos dados trocados.

Divisibilidade: nível 4, o sistema é notacional.

3.12.2 Requisitos

Integridade: a integridade dos cheques é garantida pelo uso de *hashings* criptografados e tíquetes KERBEROS.

Robustez: não é tratada de maneira explicita.

Viabilidade econômica: O sistema só prevê o uso de cifras simétricas e deve ser usado para transações de valores próximos aos valores de transações com cheques. Assim, acreditamos que seja viável economicamente.

Escalabilidade: A necessidade do servidor KERBEROS central pode atrapalhar a escalabilidade do sistema, embora a última versão do KERBEROS já permita o uso de vários servidores.

Interoperabilidade: É prevista a interoperabilidade com o NetCash apenas.

Auditabilidade: Todas as transações são identificadas e devem passar pelo servidor que pode manter *logs*.

3.12.3 Funcionamento

A Figura 3.13 apresenta o fluxo de mensagens na etapa de transação do protocolo, cujo funcionamento é descrito a seguir:

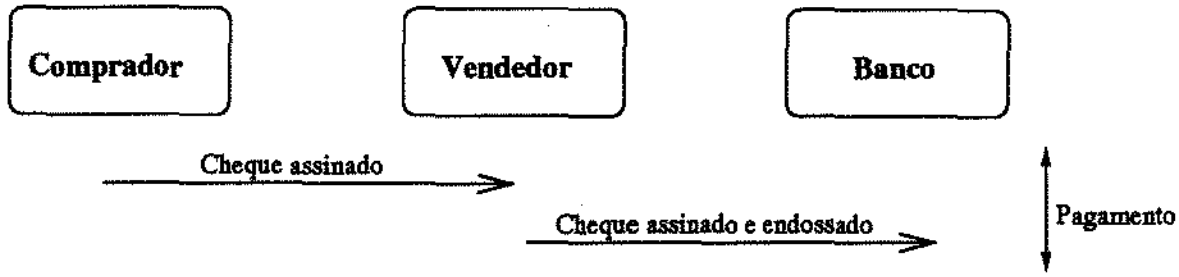


Figura 3.13: Fluxo de mensagens no NETCHEQUE

1. **Relacionamento**
   Os usuários devem abrir contas especiais nos bancos que participam do sistema; estas contas serão usadas nas transferências feitas através do NETCHEQUE. Os usuários devem também obter o *software* necessário ao uso do sistema.

2. **Retirada**
   Não há.

3. **Transação**

   (a) **Autenticação**
   Os usuários devem obter tíquetes junto a um servidor KERBEROS, antes da transação ou à medida em que forem necessários.

   (b) **Pagamento**
   O pagador obtém um tíquete para uso dos serviços do banco e usa a chave de sessão contida no tíquete para cifrar um *hashing* dos dados do cheque. O pagador envia este cheque ao vendedor, que vai obter um tíquete para si, endossar o cheque e enviá-lo ao banco para compensação.

   (c) **Recibo**
   Este sistema não prevê emissão de recibo.

   (d) **Finalização**
   Não há fase de finalização.

4. **Compensação**
   O banco do recebedor deve entrar em contato com o banco do pagador e realizar a compensação do cheque. Quando a compensação for completada, o banco credita o valor na conta do recebedor. O sistema não prevê que o banco avise ao recebedor que o valor foi creditado em sua conta.

Observações:

1. Se os participantes da transação quiserem se comunicar de forma cifrada, o pagador deve obter junto ao servidor KERBEROS um tíquete para compartilhamento de uma chave de sessão com o recebedor.

3.12.4 Aspectos de Implementação

Técnicas Criptográficas: Os algoritmos não são especificados; o sistema faz uso de: cifra simétrica e *hashing*.

Conexões de rede: duas conexões.

Durante a transação, uma conexão é necessária para que o comprador e o vendedor se comuniquem e outra para que o vendedor se comunique com o banco.

Formato das ordens de pagamento: os seguintes campos devem ser preenchidos:

- valor
- unidade monetária
- data
- número da conta
- identificação do recebedor
- assinatura do pagador: *hashing* do cheque com a chave de sessão presente no tíquete.

3.12.5 Comentários e Análises

O NETCHEQUE é o sistema parceiro do NETCASH e funciona como mecanismo de compensação para este sistema. Os projetistas indicam que o NETCHEQUE poderia ser usado pelos servidores NETCASH sempre que um destes servidores tiver que compensar um cupom emitido por outro servidor. O procedimento seria:

1. O servidor A recebe para compensação um cupom NETCASH emitido pelo servidor B.
2. O servidor A envia o cupom ao servidor B.
3. Se o cupom for válido, o servidor B devolve um NETCHEQUE de mesmo valor que o cupom ao servidor A.
4. O servidor A pode realizar a compensação.

O sistema NETCHEQUE é bastante simples mas, por ser baseado no KERBEROS, pode ter problemas de escalabilidade e de responsabilização (ver Capítulo 4).

3.13 PAYME

O PAYME [28] foi desenvolvido no TRINITY COLLEGE, de Dublin, Irlanda e teve como objetivo melhorar os sistemas de pagamentos baseados em cupons, mantendo as qualidades dos sistemas mais conhecidos e eliminando suas deficiências. Os principais sistemas que deram origem ao PAYME foram o E-CASH (seção 3.7) e o NETCASH (seção 3.11). O PAYME tentou manter o alto grau de privacidade oferecido pelo E-CASH e oferecer a escalabilidade do NETCASH, sem com isso diminuir a segurança do sistema. Acreditamos que o sistema tenha níveis de privacidade, escalabilidade e segurança semelhantes ao NETCASH, mesmo sendo mais simples.

O sistema tenta garantir o anonimato de maneira semelhante ao NETCASH, ou seja, permitindo que a troca de moedas seja feita de forma anônima. Assim, um usuário que detém moedas do sistema pode depositá-las em sua conta ou trocá-las por moedas que somem o mesmo valor total. Este mecanismo pode ser usado para obter as moedas necessárias à realização de pagamentos no valor exato ou para tentar mascarar o padrão das compras realizadas por um usuário.

3.13.1 Tipificação

Modelo de troca: cupons

Os cupons deste sistema são chamados de moedas, sendo numerados e contendo uma identificação da entidade emissora, assim como a data de validade.

Envolvimento da entidade emissora: *on-line*

O banco que emite cada moeda deve ser contatado para confirmar sua validade.

Quantias envolvidas: pequenos pagamentos

Os protocolos deste sistema foram projetados para uso na Internet, fazendo uso do TCP/IP como camada inferior, o que implica na possibilidade de exploração dos problemas de segurança do TCP para burlar o sistema. Além disso, a primeira mensagem do protocolo não é criptografada, o que diminui o nível de segurança do sistema. O uso de criptografia garante segurança suficiente à transferência de pequenos valores.

Hardware necessário: uso geral

O sistema não faz uso de *hardware* especial, sendo passível de implementação em qualquer tipo de computador.

Papéis envolvidos: pagadores, recebedores e bancos.

Os usuários (pagadores e recebedores) realizam transações pela rede, e os bancos emitem os cupons, ou moedas, usados no sistema.

Inversibilidade dos papéis: papéis variáveis.

Os usuários do sistema podem tanto atuar como pagadores quanto como recebedores. O único papel fixo é o papel do banco, ou entidade emissora.

Privacidade: existente

O sistema permite aos usuários a realização de trocas de moedas de forma anônima, o que possibilita a realização de transações em que a identidade do pagador é desconhecida. Apenas o banco é identificado por todos os participantes e o pagador sempre sabe a identidade do recebedor.

Divisibilidade: nível 2.

O protocolo prevê que os usuários possam trocar moedas com o banco, tanto para aumentar a privacidade das transações, já que a troca pode ser feita de forma anônima, quanto para obter moedas de menor valor.

3.13.2 Requisitos

Integridade: a integridade das moedas e das mensagens é garantida pelo uso de assinaturas e *hashings*, no modelo de envelope seguro.

Robustez: não é tratada de maneira explicita na especificação. O sistema garante apenas atomicidade monetária.

Viabilidade econômica: Este sistema apresenta um custo de operação razoável por necessitar de conexões *on-line* e fazer uso de cifras assimétricas e certificados, o que o torna inadequado à realização de microtransações.

Escalabilidade: o sistema permite a existência de várias entidades emissoras e a incorporação de novos usuários é bastante fácil. O principal problema quanto à escalabilidade do sistema é a possibilidade do número de moedas válidas de um único banco crescer a ponto de inviabilizar a consulta aos números de série das moedas em circulação.

Interoperabilidade: o sistema prevê a existência de diversas entidades emissoras e a interoperabilidade com o sistema de contas bancárias. Sendo um sistema *on-line*, é possível a implantação de um serviço de conversão de moedas para outros sistemas.

Auditabilidade: as moedas contém a identificação da entidade emissora e um número de série. Nenhum outro mecanismo de auditoria é especificado.

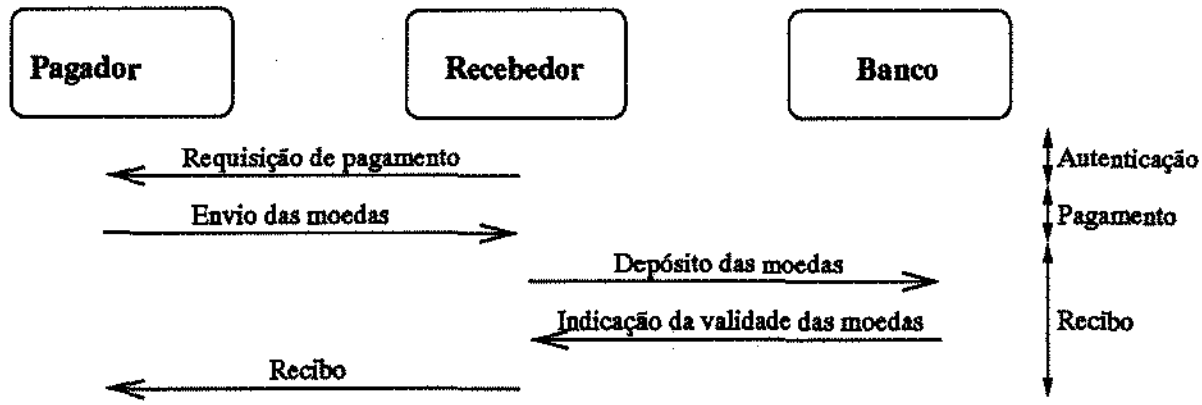


Figura 3.14: Fluxo de mensagens no PAYME

3.13.3 Funcionamento

A Figura 3.14 apresenta as mensagens trocadas na fase de transação do protocolo, cujo funcionamento é descrito a seguir:

1. **Relacionamento**
   O usuário deve manter uma conta corrente num dos bancos que participam do sistema, da qual o usuário poderá realizar retiradas ou depósitos de dinheiro eletrônico. Além disto, o usuário precisa receber e instalar o *software* de carteira eletrônica em seu computador.

2. **Retirada**
   O usuário informa ao banco o número de sua conta, sua senha e o valor da retirada. Esta mensagem é assinada e enviada ao banco. O banco irá responder enviando as moedas ou uma mensagem explicativa em caso de erro. Se o usuário desejar realizar pagamentos anônimos, este deve trocar as moedas com algum banco sem se identificar.

3. **Transação**

   (a) **Autenticação**
   O pagador só se identifica se quiser, e por meios fora do escopo do sistema. O recebedor deve enviar uma mensagem ao pagador requisitando o pagamento. Nesta mensagem vai a chave pública do recebedor.

   (b) **Pagamento**
   Se o pagador aceitar as condições do recebedor, este envia as moedas, cifradas com a chave pública do recebedor.

(c) **Recibo**
O recebedor verifica com o banco a validade das moedas recebidas e devolve ao pagador um recibo. Esta verificação pode ser feita de duas maneiras: depositando as moedas em sua conta junto a um banco ou trocando as moedas por outras junto ao banco que as emitiu. Em qualquer destes casos, o banco indicará se as moedas são válidas. Caso sejam válidas, o recebedor pode optar por depositá-las.

(d) **Finalização:**
Não há.

4. **Compensação**
Os usuários podem depositar ou trocar os cupons a qualquer momento. O recebedor sempre deve realizar uma destas operações para verificar a validade das moedas recebidas.

Observações:

1. O sistema consegue atingir um alto grau de segurança se for usada uma infra-estrutura de certificação de chaves não prevista na especificação.

3.13.4 Aspectos de Implementação

Técnicas Criptográficas: as primitivas criptográficas usadas são:

Cifras simétricas: O protótipo usa o IDEA, embora outros algoritmos possam ser usados. Esta cifra é necessária para compor o modelo de envelope seguro.

Cifras assimétricas: O RSA foi escolhido para o protótipo. Outros sistemas poderiam ser usados se conveniente.

Assinatura digital: O algoritmo RSA é usado, com as mesmas chaves usadas para ciframento.

Hashings: Necessários para compor o modelo de envelope seguro.

Conexões de rede: 2 conexões.
Uma conexão é necessária entre o comprador e o vendedor, e uma entre o vendedor e o banco.

Formato dos cupons: As moedas PAYME têm os seguintes campos:

- valor
- número de série

- identificação e endereço da entidade que emitiu a moeda
- validade

Cada moeda é assinada pelo banco que a emitiu.

3.13.5 Comentários e Análises

O PAYME foi desenvolvido como objeto de uma tese de mestrado no TRINITY COLLEGE de Dublin, aparentemente sem pretensões comerciais. Segundo os projetistas, um dos objetivos do protocolo seria aumentar o anonimato de sistemas como o NETCASH. O resultado foi um sistema com um nível de anonimato ligeiramente superior ao NETCASH, e de compreensão mais fácil. É certamente um sistema que pode ser usado como base para novos desenvolvimentos ou para melhoras em sistemas já existentes.

3.14 CYBERCASH

A CYBERCASH, INC. [10] foi uma das primeiras empresas a oferecer serviços de pagamento na Internet através do sistema que leva o nome da companhia. Este é hoje o mais popular sistema de pagamento seguro na Internet, sendo usado por importantes empresas da área de comércio eletrônico e com mais de meio milhão de cópias do seu programa de carteira eletrônica distribuídas.

O CYBERCASH é um sistema baseado em cartões de crédito que pode ser considerado como uma etapa intermediária entre a coleta de números de cartões pelos comerciantes e os sistemas a serem implantados pelas empresas de cartões de crédito, como o SET. A cybercash vem anunciando que passará a usar o SET, o que permitirá a integração com sistemas de outros fabricantes.

3.14.1 Tipificação

Modelo de troca: notacional

É um sistema baseado no sistema de cartões de crédito, que pode ser visto como uma maneira segura de comunicar números de cartões.

Envolvimento da entidade emissora: *on-line*

O servidor deve ser contactado para que este processe o pagamento junto às administradoras de cartões.

Quantias envolvidas: pequenos a médios pagamentos

A faixa de valores é determinada pelas empresas de cartões de crédito. Em geral, os

custos das transações são altos demais para micropagamentos e os limites de crédito não permitem transações de grandes valores.

Hardware necessário: uso geral
O sistema não faz uso de *hardware* especial como *smart cards* ou outros mecanismos deste tipo, sendo passível de implementação em qualquer tipo de computador. Existem implementações para PCs, *MacIntoshes* e estações UNIX.

Papéis envolvidos: comprador, vendedor e servidor central.
Os papéis são fixos e o sistema prevê a existência de apenas um servidor central, com o qual os compradores e vendedores devem se registrar.

Inversibilidade dos papéis: papéis fixos.
Assim como no sistema de cartões de crédito, os papéis são fixos.

Privacidade: existente.
Embora não permita transações anônimas, o sistema não permite que o vendedor veja o número do cartão do comprador e não permite que o banco tenha conhecimento dos itens adquiridos.

Divisibilidade: nível 4, o sistema é notacional.

3.14.2 Requisitos

Integridade: As mensagens são assinadas e o servidor central garante a integridade do sistema.

Robustez: As transações são realizadas pelo servidor central, que deve se encarregar de garantir a consistência do sistema. As mensagens trocadas garantem que o comprador e o vendedor tem uma visão consistente da transação. O sistema garante atomicidade monetária.

Viabilidade econômica: O sistema está em operação desde 1995 e têm demonstrado ser viável e lucrativo.

Escalabilidade: O sistema é dependente de um servidor central, o que impõe um limite na quantidade de usuários e transações que este é capaz de atender.

Interoperabilidade: Este sistema não prevê interoperabilidade.

Auditabilidade: As mensagens são assinadas e as transações são registradas pelo servidor central.

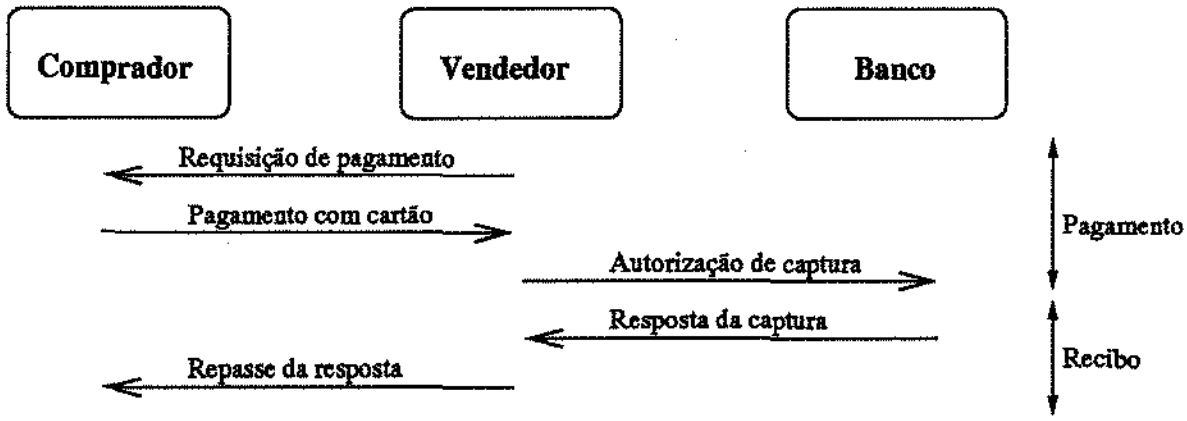


Figura 3.15: Fluxo de mensagens no CYBERCASH

3.14.3 Funcionamento

A Figura 3.15 apresenta o fluxo de mensagens na etapa de transação do protocolo, cujo funcionamento é descrito a seguir:

1. **Relacionamento**
   O usuário deve adquirir o *software* que permite o uso do sistema, e deve também registrar uma identificação junto ao servidor CYBERCASH e escolher uma senha. Após ter se registrado, o usuário deve registrar seus cartões de crédito, o que permite à CYBERCASH realizar uma verificação prévia junto às operadoras quanto à validade destes cartões.

2. **Retirada**
   Não há.

3. **Transação**

   (a) **Autenticação**
   Não há.

   (b) **Pagamento**
   O vendedor assina e envia ao comprador a descrição dos bens e seu preço. O comprador verifica esta descrição e escolhe o cartão a ser usado na transação. O número deste cartão é cifrado com a chave pública do servidor CYBERCASH e é assinado junto com um *hashing* da descrição dos bens e do preço total. Esta mensagem é enviada ao vendedor, que a repassa ao servidor central juntamente com um novo *hashing* da descrição do pedido assinada com sua chave secreta.

   (c) **Recibo**
   O servidor CYBERCASH verifica se os *hashings* da descrição são iguais, verifica

as assinaturas e processa a transação junto às empresas de cartões de crédito. Após a realização da transação, o servidor envia ao vendedor dois recibos: um fica com o vendedor e o outro deve ser repassado ao comprador.

(d) **Finalização**
Não há fase de finalização.

4. **Compensação**
Não há etapa de compensação.

3.14.4 Aspectos de Implementação

Técnicas Criptográficas: são necessários:

Cifra simétrica: DES.

Cifra Assimétrica: RSA com chaves de 1024 bits.

Assinatura Digital: RSA com chaves de 1024 bits.

Hashing: MD5.

Conexões de rede: duas conexões.
O comprador e o vendedor devem se comunicar e o vendedor deve entrar em contato com o servidor CYBERCASH.

Formato das ordens de pagamento: O comprador envia os seguintes dados, assinados com sua chave secreta:

- número do cartão cifrado com a chave pública do servidor central;
- valor;
- *hashing* da descrição das mercadorias.

O vendedor acrescenta um *hashing* assinado da descrição das mercadorias.

3.14.5 Comentários e Análises

O CYBERCASH é atualmente o sistema de pagamentos mais popular na Internet, mas logo terá que competir com os sistemas baseados no SET (Seção 3.3) que entrarão no mercado. Como a CYBERCASH anunciou que tornará seu sistema compatível com o SET, a empresa terá a vantagem de ter uma grande base instalada de seu sistema.

O processo de registro e certificação do sistema ocorre *on-line*, o que pode diminuir a confiança no sistema e torná-lo vulnerável à *engenharia social*[1], embora nenhum caso deste tipo tenha sido anunciado desde que o sistema está em funcionamento comercial.

Em comparação com outros sistemas, a necessidade de um servidor central pode se tornar um problema, já que impede a inclusão de vendedores que não estejam nos Estados Unidos. A necessidade de certificar cada usuário e cartão de crédito pode dificultar a expansão do sistema.

3.15 Outros Sistemas

Existem no mercado outros sistemas de pagamento que não pudemos analisar por diversos motivos. Em geral são sistemas proprietários, cujos detalhes são difíceis de obter, como VISA CASH, VISA ELECTRON, MASTERCARD CASH, NETCHEX, ou mesmo sistemas cujos detalhes de funcionamento são mantidos secretos, como MONDEX, CYBERCOIN, PROTON.

Alguns destes sistemas já estão em uso comercial, outros estão em fase de testes, enquanto outros parecem destinados a não saírem do papel. De qualquer maneira, é difícil analisar ou comparar estes sistemas sem que as companhias que os desenvolveram apresentem os detalhes de seu funcionamento.

Existem também diversos sistemas propostos que consideramos "acadêmicos" porque são de difícil compreensão e implementação, embora apresentem características que os tornam importantes. São, em geral, sistemas que servirão de base para desenvolvimentos futuros. Alguns exemplos podem ser encontrados em [25, 4, 9, 29].

3.16 Resumo das Características dos Sistemas

As Tabelas 3.1, 3.2, 3.3, 3.4 e 3.5 resumem as análises apresentadas nas seções anteriores. Para a coluna de **Papéis** da Tabela 3.2, as letras representam:

B	banco	**U**	usuário
C	comprador	**V**	vendedor

Na próxima seção, apresentamos as similaridades encontradas em diversas classes de sistemas de pagamento durante o processo de estudo e análise. Estas similaridades dentro de algumas classes parecem indicar as direções normalmente seguidas pelos projetistas de sistemas de pagamento e podem ajudar a avaliar a originalidade de novos sistemas.

[1] Engenharia social (*Social Engineering*) é o processo pelo qual hackers conseguem se fazer passar por outrem e obter algum tipo de vantagem

3.16.1 Similaridades em Sistemas de Pagamento Eletrônico

Considere as seguintes classes de sistemas de pagamento eletrônico [26]:

1. Sistemas para transmissão de números de conta (cartão de crédito)
2. Sistemas tipo Cheque Eletrônico
3. Sistemas tipo Dinheiro Eletrônico
4. Sistemas para micropagamentos

Em cada classe, é possível encontrar tendências de projeto que ocorrem na maioria dos sistemas mais conhecidos. Estas tendências são:

Sistemas para transmissão de números de conta:

1. Os sistemas são baseados em cifras assimétricas com certificação de chaves.
2. O número de conta é tratado como segredo compartilhado.
3. As partes devem concordar explicitamente com os detalhes da transação
4. Para evitar vazamentos, o número da conta é transferido em dois *hops* (do comprador para o vendedor e depois do vendedor para o banco), mas é cifrado.

Cheque eletrônico: Existem dois tipos, dependendo do método de autenticação:

Autenticação com chaves simétricas:

1. Autenticação dos participantes ocorre no sistema de autenticação subjacente (eg. KERBEROS).

Autenticação usando chaves assimétricas:

1. As chaves públicas são autenticadas ou certificadas.

Um ponto comum a estes sistemas é a exigência de que todos os participantes assinem as ordens de pagamento. O banco apenas verifica estas assinaturas.

Dinheiro eletrônico: Usam cupons contendo número de série, prazo de validade e assinados pelo banco emissor

Micropagamentos: Geralmente,

1. evitam usar cifras;
2. fazem uso maciço de funções de espalhamento;
3. os cupons são simples e sem autenticação forte;

	Mod. de troca	Env. Banco	Quantias	Hardware
Fisrt Virtual	notacional	*on-line*	micro	geral
Globe ID	notacional	*on-line*	pequenas	geral
Payme	cupons	*on-line*	pequenas	geral
SET	notacional	*on-line*	pequ. a grandes	geral
PayWord	cupons	*off-line*	micro	geral
MicroMint	cupons	*off-line*	micro	geral
CAFE	híbrido	*off-line*	pequenas	específico
E-cash	cupons	*on-line*	pequenas	geral
iKP	notacional	*on-line*	pequ. a grandes	geral
Millicent	cupons	*off-line*	micro	geral
NetBill	notacional	*on-line*	pequenas	geral
NetCash	cupons	*on-line*	pequenas	geral
NetCheque	notacional	*on-line*	pequenas	geral
CyberCash	notacional	*on-line*	pequ. a médias	geral

Tabela 3.1: Primeira parte da Tabela-resumo de Tipificação

4. parte da segurança é baseada no baixo valor das transações, o que torna a fraude antieconômica.

Dentre os sistemas para micropagamentos, aqueles que objetivam a realização de pagamentos repetidos, em geral, realizam apenas uma autenticação no primeiro pagamento e as operações seguintes confiam nesta autenticação.

3.17 Conclusões

Este capítulo apresenta as análises de diversos sistemas de pagamento com base no esquema apresentado no capítulo anterior. Esta análise permite uma comparação dos sistemas e facilita a escolha do sistema mais adequado a cada situação.

Apresentamos também tabelas comparativas e uma descrição das similaridades encontradas em sistemas de pagamento durante o trabalho de análise cujo resultado é apresentado neste capítulo. Esta similaridades indicam tendências gerais no projeto de sistemas de pagamento e permitem evidenciar quais as possibilidades de inovação no projeto destes sistemas.

	Papéis	Invers.	Privacidade	Nível de divisib.
Fisrt Virtual	CVB	não	não	4
Globe ID	CVB	não	sim	4
Payme	CVB	sim	sim	2
SET	CVB	não	sim	4
PayWord	CVB	não	não	4
MicroMint	UB	sim	não	4
CAFE	CVB	sim	sim	1
E-cash	UB	sim	sim	2
iKP	CVB	não	sim	4
Millicent	CVB	não	sim	3
NetBill	CVB	não	sim	4
NetCash	CVB	sim	sim	2
NetCheque	CVB	sim	não	4
CyberCash	CVB	não	sim	4

Tabela 3.2: Segunda parte da Tabela-resumo de Tipificação

	Integridade	Robustez	Viabilidade	Escalab.	Interoper.	Audit.
Fist Virtual		√	√	√		√
Globe ID	√	√	√			√
Payme	√		√	√		
SET	√	√	√	√		√
PayWord			√	√		√
MicroMint			√	√		√
CAFE	√	√	√	√	√	√
E-cash	√		√			√
iKP	√	√	√	√		√
Millicent	√		√	√		√
NetBill	√	√	√			√
NetCash	√		√	√	√	
NetCheque	√		√		√	√
CyberCash	√	√	√			√

Tabela 3.3: Tabela-resumo das Caracteríticas Desejáveis

	Primitivas Criptográficas	Conexões de Rede	Cont. Ordens de Pagamento
Fist Virtual		3	número de conta
Globe ID	Cifra assimétrica Assinatura *hashing*	2	não especificado
Payme	Cifra simétrica (IDEA) Cifra assimétrica (RSA) Assinatura digital (RSA)	2	valor do cupom número de série identificação do banco data de validade assinatura do banco
SET	Cifra Simétrica (DES) Cifra assimétrica (RSA) Assinatura digital (RSA) Certificado (X.509) *Hashing*	2	dados do cartão de crédito ID da transação valor da transação *hashing* da descrição do produtos
PayWord	Assinatura digital Certificados *Hashings*	1	valor de cadeia de *hashings*
MicroMint	*Hashing*	1	colisão de *hashing*
CAFE	Assinatura digital (Schnorr)	1	chaves públicas ID do vendedor data da transação valor da tarnsaçao
E-cash	Cifra simétrica (Triplo-DES) Cifra assimétrica (RSA) Assinatura digital *Hashing* (SHA)	2	número de série do cupom data de validade assinatura do banco
iKP	Cifra assimétrica (RSA) Assinatura digital (RSA) Certificados (PKCS) *Hashing* (MD5)	2	*hashing* de informações gerais número da conta número aleatório
Millicent	*Hashing* (MD5)	1	ID do vendedor valor do cupom número de série ID do comprador data de validade certificado de autenticidade

Tabela 3.4: Primeira parte da Tabela de Aspectos de Implementação

	Primitivas Criptográficas	Conexões de Rede	Cont. Ordens de Pagamento
NetBill	Cifra simétrica (DES) Cifra assimétrica (RSA) Certificados *Hashing* (SHA)	2	tíquete *Kerberos* número da conta
NetCash	Cifra simétrica Cifra assimétrica Assinatura digital Certificado *Hashing*	2	ID do emissor do cupom Endereço do emissor data de validade número de série valor do cupom assinatura do emissor
NetCheque	Cifra simétrica *Hashing*	1	valor da transação unidade monetária data da transação número da conta do comprador ID do vendedor assinatura do comprador
CyberCash	Cifra simétrica (DES) Cifra assimétrica (RSA) Assinatura digital (RSA) *Hashing*	2	número do cartão valor *hashing* da desc. das mercadorias

Tabela 3.5: Segunda parte da Tabela de Aspectos de Implementação

Capítulo 4

Análise Usando Métodos Formais

Os sistemas de pagamento apresentam certas características comuns em sistemas criptográficos e é difícil determinar empiricamente se realmente cumprem seus objetivos em relação estas características. Por exemplo, é difícil determinar empiricamente se um sistema apresenta determinado nível de autenticação ou se o método de autenticação usado é adequado.

Para estudar estas características, pode-se lançar mão do uso de técnicas de análise formal, que permitem provar que certos objetivos são atingidos. Diversas técnicas de análise formal de protocolos criptográficos são apresentadas na literatura [18], sendo adequadas a diversos tipos de análise, desde o projeto até a implementação dos sistemas.

Neste capítulo, apresentamos dois métodos formais adequados ao estudo de sistemas criptográficos e, em particular, sistemas de pagamento eletrônico. Os métodos são apresentados na próxima seção e as análises e conclusões obtidas com o uso destes métodos são apresentadas a seguir. Apresentamos também um modelo que generaliza o modo de funcionamento dos sistemas de pagamento baseados em cartões de crédito na seção 4.3.1.

4.1 Métodos Formais Escolhidos

Para realizar a análise formal, foram escolhidos dois métodos adequados ao estudo das propriedades de sistemas criptográficos. O primeiro método é normalmente denominado **lógica BAN**[5], devido às iniciais de seus autores. Este método é bem conhecido e é considerado adequado para a análise dos aspectos de autenticação dos protocolos num nível de abstração mais elevado[18].

Para a análise do modelo geral de sistemas baseados em cartões de crédito apresentado na seção 4.3.1, é usado também um framework desenvolvido por Kailar [16], que permite o estudo das características de reponsabilização em protocolos de comércio eletrônico.

4.1.1 Lógica BAN

A lógica BAN foi definida e utilizada por seus autores para descrever e analisar diversos sistemas de autenticação. O artigo original [5] apresenta a lógica e sua simbologia, além da análise de diversos sistemas de autenticação.

Esta lógica está baseada nas crenças que cada uma das entidades envolvidas têm e nas possíveis inferências que pode fazer quando recebe alguma mensagem. Alguns exemplos dos princípios nos quais a lógica BAN está baseada são:

1. Se Gisele enviou a Pedro um número que nunca havia usado antes para este fim e depois recebe de Pedro uma mensagem contendo este número, então Gisele pode acreditar que esta mensagem foi originada recentemente - na verdade depois da mensagem que enviou.

2. Se Pedro acredita que somente ele e Gisele conhecem a chave K, então qualquer mensagem que Pedro receber, cifrada com K, deve ter sido enviada por Gisele.

3. Se Gisele acredita que K é a chave privada de Pedro e recebe uma mensagem M cifrada com esta chave, então Gisele deve acreditar que M foi enviada por Pedro.

4. Se Pedro acredita que somente ele e Gisele sabem X e recebe uma mensagem contendo X, então a mensagem deve ter sido enviada por Gisele.

Um protocolo de autenticação é geralmente descrito como uma série de mensagens trocadas pelos participantes, sendo especificado o conteúdo de cada uma das mensagens mas este formato não é adequado para tratamento pela lógica BAN. Assim, transformaremos cada mensagem numa fórmula lógica, que consiste de uma versão idealizada desta mensagem e indica não o conteúdo mas o significado de cada mensagem. O protocolo idealizado é então anotado com asserções, que indicam as crenças de cada entidade participante naquele ponto da execução do protocolo. Descrevemos abaixo a notação usada nesta lógica mas antes trataremos com mais detalhes a questão da autenticação.

Autenticação

Autenticar uma entidade é um dos problemas básicos da segurança de informações, consistindo de um procedimento para garantir a identificação correta de entidades num sistema de computação, especialmente em sistemas distribuídos. Quando duas entidades se comunicam numa rede, estas devem ter alguma garantia de que estão em contato com a entidade correta.

Uma das maneiras mais comuns de garantir a autenticação é o compartilhamento de segredos. Assim, durante o protocolo de autenticação, a revelação do segredo por

parte de uma entidade dá às demais uma garantia de sua identidade. Estes segredos são, muitas vezes, chaves criptográficas. Neste caso, os protocolos se preocupam em permitir o compartilhamento de chaves secretas ou em distribuir chaves públicas.

O aparecimento de diversos protocolos de autenticação, cada um adequado a um determinado ambiente, levou à necessidade de uma avaliação formal da efetividade de cada um deles. Desta necessidade surgem os métodos formais para avaliação deste tipo de protocolo, entre eles a lógica BAN.

Para maiores detalhes relativos a autenticação, recomendamos consultar [20] ou [31].

Notação

Os principais tipos de objetos envolvidos são: entidades, chaves criptográficas e fórmulas. Os símbolos usados são: letra maiúsculas como A, B, S denotam entidades ou afirmações (em geral denotadas por X ou Y); K_{ab}, K_{as} e K_{bs} denotam chaves compartilhadas, neste caso por A e B, A e S, B e S respectivamente; K_a, K_b e K_s denotam chaves públicas e K_a^{-1}, K_b^{-1} e K_s^{-1} denotam as chaves privadas. A conjunção lógica é denotada pela vírgula, que é usada também para separar os diversos elementos de uma fórmula, desde que estejam entre parênteses. As seguintes construções são possíveis:

$P \equiv X$: P acredita em X. A entidade P pode agir como se X fosse verdadeiro.

$P \triangleleft X$: P viu X. Tendo recebido uma mensagem contendo X, P pode ler e repetir X.

$P \mid\sim X$: P disse X. A entidade P enviou alguma mensagem contendo X, embora não se saiba se esta mensagem é recente, ou foi enviada durante outra execução do protocolo.

$P \Rightarrow X$: P tem autoridade sobre X. A entidade P tem autoridade sobre X e o respeito dos demais com relação a esta autoridade. Essa construção é usada quando uma entidade delega autoridade sobre alguma fórmula, por exemplo, em protocolos em que um servidor central gera chaves criptográficas para outras entidades, tendo assim autoridade sobre afirmações quanto à qualidade das chaves.

$\sharp(X)$: X é recente, isto é, X não foi enviado em nenhum momento antes da execução corrente do protocolo.

$P \overset{K}{\leftrightarrow} Q$: P e Q podem usar a chave compartilhada K para se comunicarem. A chave K é *boa* no sentido de que nunca vai ser descoberta por uma entidade diferente de P ou Q ou uma entidade em que P ou Q confiam (servidor).

$\overset{K}{\mapsto} P$: K é uma chave pública de P. A chave privada K^{-1} correspondente nunca será descoberta por outra entidade que não P ou uma entidade em que P confia.

$P \overset{X}{=} Q$: A fórmula X é um segredo conhecido apenas por P e Q, e possivelmente alguma entidade em que confiam. P e Q podem usar X para provar suas identidades um ao outro. Um exemplo de segredo compartilhado é uma senha.

$\{X\}_K$: Representa a fórmula X cifrada com a chave K.

$\langle X \rangle_Y$: Representa X combinado com Y, que deve ser um segredo compartilhado e cuja presença prova a identidade do remetente de $\langle X \rangle_Y$. Em muitas implementações, X é simplesmente concatenado com a senha Y, mas esta notação é capaz de expressar que Y desempenha um papel especial como prova da origem de X.

$\frac{X}{Y}$: Indica que se X for verdadeiro, é possível concluir que Y também é válido.

Postulados Lógicos

A lógica BAN distingue duas épocas: passado e presente. O presente começa no início da execução corrente do protocolo. Qualquer outra mensagem é considerada como fazendo parte do passado, e o protocolo deve evitar que estas sejam consideradas como parte do presente.

Os principais postulados usados são:

- *Significado das mensagens*: considera a interpretação de mensagens cifradas ou com segredos. Para chaves compartilhadas é postulado:

$$\frac{P \models Q \overset{K}{\leftrightarrow} P,\ P \triangleleft \{X\}_K}{P \models Q \mid\sim X} \tag{4.1}$$

  Isto é, se P acredita que K é uma chave compartilhada com Q e vê uma mensagem X cifrada com esta chave, então P acredita que Q já disse X. Regras similares podem ser deduzidas para chaves públicas e segredos compartilhados.

- *Verificação de nonces*: permite verificar se uma mensagem é recente.

$$\frac{P \models \sharp(X),\ P \models Q \mid\sim X}{P \models Q \models X} \tag{4.2}$$

  Isto é, se P acredita que X é recente e P acredita que Q falou X, então P acredita que Q acredita em X.

- *Jurisdição*: Se P acredita que Q tem autoridade sobre algo, então P confia nas decisões de Q a respeito deste assunto:

$$\frac{P \models Q \Rightarrow X,\ P \models Q \models X}{P \models X} \tag{4.3}$$

- *Conjunção*: P acredita em um conjunto de fórmulas se e somente se acredita em cada uma delas separadamente.

$$\frac{P \models X, P \models Y}{P \models (X,Y)} \quad \frac{P \models (X,Y)}{P \models X} \quad \frac{P \models Q \models (X,Y)}{P \models Q \models X} \quad \frac{P \models Q \mid\sim (X,Y)}{P \models Q \mid\sim X} \tag{4.4}$$

Note-se que $P \models Q \mid\sim X$ e $P \models Q \mid\sim Y$, não implica em $P \models Q \mid\sim (X,Y)$, já que isto implicaria em Q ter enviado uma mesma mensagem contendo X e Y.

- *Decifração*: Se uma entidade vê uma fórmula, vê também seus componentes, desde que conheça as chaves necessárias:

$$\frac{P \triangleleft (X,Y)}{P \triangleleft X} \quad \frac{P \triangleleft \langle X \rangle_Y}{P \triangleleft X} \quad \frac{P \models Q \overset{K}{\leftrightarrow} P,\ P \triangleleft \{X\}_K}{P \triangleleft X}$$
$$\frac{P \models \overset{K}{\mapsto} P,\ P \triangleleft \{X\}_K}{P \triangleleft X} \quad \frac{P \models \overset{K}{\mapsto} Q,\ P \triangleleft \{X\}_{K^{-1}}}{P \triangleleft X} \tag{4.5}$$

- *Novidade*: se parte de uma fórmula é recente, então toda a fórmula é recente.

$$\frac{P \models \sharp(X)}{P \models \sharp(X,Y)} \tag{4.6}$$

Similarmente, se X é novo, então $\{X\}_K$ é novo.

Protocolos Idealizados

Para que se possa usar a lógica é necessário especificar os passos de maneira particular: enquanto normalmente as descrições de protocolos ficam bem próximas das implementações e descrevem o conteúdo de cada mensagem, a análise é baseada no significado das mensagens.

Desta maneira é necessário transcrever cada passo do protocolo da notação convencional para uma notação idealizada. Um exemplo seria a mensagem:

$$A \rightarrow B : \{A, K_{ab}\}_{K_{bs}}$$

Esta mensagem diria a B, que conhece a chave K_{bs}, que K_{ab} é uma chave a ser usada na comunicação com A. Este passo seria idealizado como:

$$A \rightarrow B : \{A \overset{K_{ab}}{\leftrightarrow} B\}_{K_{bs}}$$

Como se pode notar, a forma idealizada omite certas partes da mensagem que não contribuem para a formação de crenças da parte do recebedor. Em alguns casos mensagens inteiras pode ser eliminadas, por exemplos mensagens usadas para iniciar a execução de um protocolo.

Metodologia de análise de protocolos

Os passos da análise de um protocolo são:

1. Reescrita do protocolo idealizado a partir do original.
2. Colocação das suposições acerca do estado inicial.
3. Associação de fórmulas lógicas (asserções) descrevendo cada passo o estado do protocolo.
4. Aplicação dos postulados às suposições e asserções de maneira a descobrir as crenças dos participantes do protocolo.

Assim, o protocolo anotado toma a forma:

(suposições) S_1 (asserção 1) . . . (asserção $n-1$) S_n (conclusões)

onde cada S_i é um dos passos do protocolo.

Formalização dos Objetivos da Autenticação

Suposições iniciais são necessárias a qualquer protocolo. Estas, em geral, afirmam que algumas chaves são compartilhadas entre os participantes, quais dos participantes geraram novos *nonces* e quais participantes são confiáveis (detêm autoridade) em relação a alguma afirmação. Depois de escrever as suposições, a verificação de um protocolo consiste em provar que determinadas fórmulas são conclusões válidas.

É improvável que se chegue a um acordo de quais são os objetivos da autenticação, mas já que muitos protocolos de autenticação são o ponto de partida para outros protocolos de comunicação, podemos supor que as conclusões desejadas impliquem em fornecer aos participantes chaves compartilhadas:

$$A \models A \overset{K}{\leftrightarrow} B \text{ e } B \models A \overset{K}{\leftrightarrow} B$$

Alguns protocolos, no entanto, obtêm mais que isso. Em geral, muitos conseguem:

$$A \models B \models A \overset{K}{\leftrightarrow} B \text{ e } B \models A \models A \overset{K}{\leftrightarrow} B$$

Para os protocolos que não se baseiam em chaves compartilhadas, os objetivos podem ser outros, como obter uma chave pública ou compartilhar um segredo N:

$$A \models \overset{K}{\mapsto} B \text{ ou } A \models A \overset{N}{\rightleftharpoons} B$$

4.1.2 O Framework de Kailar

A garantia de não-repúdio é de suma importância em sistemas comerciais pois permite que, após o fim da transação, os participantes continuem responsáveis por atos e afirmações de que tenham sido responsáveis durante a transação. Assim, um vendedor pode ser responsabilizado pela má qualidade de um produto ou o comprador é impedido de negar sua participação na transação para ser ressarcido do valor de uma mercadoria que tenha comprado.

A principal preocupação deste framework é estabelecer o nível de habilidade dos participantes em provar a origem de mensagens (ou afirmações). Estes participantes são representados por letras maiúsculas. Uma mensagem assinada serve como uma afirmação feita (sem possibilidade de repúdio) por um participante. As afirmações são denotadas por letras minúsculas.

As construções Necessárias ao Framework

No framework, a prova de uma afirmação x é algo que convence outra entidade que a afirmação é verdadeira. O framework trabalha com dois tipos básicos de prova:

Prova forte: "A CanProve x"
: A entidade A é capaz de provar a afirmação x para qualquer entidade Y, ou seja, A é capaz de executar uma seqüência de operações que convencem Y da veracidade de x.

Prova fraca: "A CanProve x to B"
: Neste tipo de prova, A tem capacidade apenas para provar a validade de x à entidade B.

Outras construções usadas são:

Autenticação: "K Authenticates A"
: Indica que a chave K pode ser usada para autenticar a entidade A, ou seja, associa a identidade de A a qualquer afirmação cifrada com K^{-1}. Assim, K e K^{-1} são as chaves pública e privada de um par de chaves assimétricas.

Interpretação: "x in m"
: x é a interpretação de um campo ou conjunto de campos da mensagem m.

Afirmação: "A Says x"
: A entidade A é responsável pela afirmação x, e por qualquer afirmação que possa ser derivada de x. O postulado

$$A \,\mathsf{Says}\, (x, y) \Rightarrow A \,\mathsf{Says}\, x$$

é usado implicitamente durante a análise.

Recebimento de Mensagens: "A Receives m SignedWith K^{-1}"

A entidade A recebe a mensagem m, que foi assinada com a chave K^{-1}. O postulado

$$\frac{A \text{ Receives } m \text{ SignedWith } K^{-1};\ x \text{ in } m}{A \text{ Receives } x \text{ SignedWith } K^{-1}}$$

é usado implicitamente durante a análise.

Autoridade: "A IsTrustedOn x"

A entidade A tem autoridade sobre a afirmação x, ou seja, as entidades confiam na palavra de A em relação a x. A autoridade pode ser global ou não, sendo que todas as entidade confiam nas afirmações de uma autoridade global. Se a autoridade não é global, as entidades que aceitam esta autoridade devem ser especificadas.

Todas estas configurações podem ser qualificadas também quanto a domínios, configurações ou tempo, como, por exemplo, em A IsTrustedOn x Until T.

Postulados

A notação introduzida é agora usada para articular algumas propriedades de responsabilidade. A apresentação destes postulados é similar àquela da seção 4.1.1, qual seja:

$$\frac{P\,;\,Q}{R}$$

significando que se as afirmações P e Q são verdadeiras simultaneamente, então a afirmação R também é verdadeira.

- Conjunção:

$$\textbf{Conj} : \frac{A \text{ CanProve } x;\ A \text{ CanProve } y}{A \text{ CanProve } (x \wedge y)}$$

Isto é, se A é capaz de provar que x é verdadeiro e que y é verdadeiro, então A é capaz de provar que $(x \wedge y)$ também é verdadeiro.

- Inferência:

$$\textbf{Inf} : \frac{A \text{ CanProve } x;\ x \Rightarrow y}{A \text{ CanProve } y}$$

Isto é, se A é capaz de provar x e x implica em y, então A é capaz de provar y.

- Crença:
  O seguinte postulado descreve a relação entre as noções de *provabilidade* e *crença*.

$$(A \,\mathsf{Believes}\, x) \Leftrightarrow (A \,\mathsf{CanProve}\, x \,\mathsf{to}\, A)$$

  Isto significa que A acredita em uma afirmação x se e somente se A é capaz de provar esta afirmação para si mesmo.

- Assinaturas Digitais:
  Uma mensagem assinada serve para ligar as afirmações que contém à identidade de um dos participantes e torná-lo responsável por estas afirmações. Esta responsabilidade pode ser provada por qualquer entidade que puder provar a associação entre a assinatura e a entidade que a emitiu.

$$\textbf{Sign} : \frac{A \,\mathsf{Receives}\, (m \,\mathsf{SignedWith}\, K^{-1}); x \,\mathsf{in}\, m; A \,\mathsf{CanProve}\, (K \,\mathsf{Authenticates}\, B)}{A \,\mathsf{CanProve}\, (B \,\mathsf{Says}\, x)}$$

  Isto é, se A recebe uma mensagem m, contendo uma afirmação x, assinada com a chave K^{-1} e A é capaz de provar que K é a chave que autentica B, então A é capaz de provar que B emitiu a afirmação x.

- Afirmações de Autoridades:
  Em alguns sistemas, provar uma afirmação consiste em provar que esta afirmação foi endossada por uma autoridade sobre o assunto:

$$\textbf{Trust} : \frac{A \,\mathsf{CanProve}\, (B \,\mathsf{Says}\, x); A \,\mathsf{CanProve}\, (B \,\mathsf{IsTrustedOn}\, x)}{A \,\mathsf{CanProve}\, x}$$

  Isto é, se A é capaz de provar que uma autoridade B afirmou x, então x é verdadeiro.

Premissas do Framework

O framework pressupõe algumas propriedades dos sistemas analisados. São elas:

Algoritmos de assinatura digital Os algoritmos são baseados no paradigma de chave pública apresentado por Diffie e Hellman. Supõe-se que os algoritmos sejam robustos o suficiente para que as assinaturas sejam associadas a usuários individuais sem possibilidade de disputas e que são computacionalmente inquebráveis por um longo tempo. As assinaturas devem prover autenticação de origem, integridade de conteúdo e não-repúdio de origem.

Confiança Os participantes não devem compartilhar suas chaves privadas com nenhuma entidade pela qual não se responsabilizem.

No caso da não existência de uma entidade cuja autoridade sobre uma afirmação seja reconhecida pela audiência de uma prova, é impossível provar esta afirmação. Isto significa que, na ausência de uma entidade com autoridade global, é necessária uma hierarquia de autoridade.

Integridade das Mensagens Como em outros métodos de análise de alto nível, assume-se que a integridade das mensagens é preservada; ou seja, é impossível forjar uma mensagem assinada ou calcular a chave privada e usá-la para assinar uma mensagem falsa.

Disponibilidade de Serviços A afirmação A CanProve x implica na capacidade de A em enviar as mensagens requeridas para provar x. Assim, assume-se que, em caso de falhas que impeçam que A envie as mensagens, estas falhas serão corrigidas no futuro, permitindo o envio da prova.

Revogação de Certificados Supõe-se que a revogação de certificados é feita de tal maneira que impeça o uso de certificados que não sejam mais válidos. Assim, o uso de certificados inválidos ou revogados não permite a uma entidade a realização de provas que envolvam este certificado.

Formalização dos Objetivos das transações

Antes de iniciar a análise de um protocolo, é necessário definir seus objetivos. Em protocolos de pagamento eletrônico, Kailar [16] indica os seguintes objetivos para a obtenção de responsabilidade total:

$$\begin{aligned} \textit{pagador} \quad & \mathsf{CanProve} \quad (\textit{recebedor recebeu o pagamento}) \\ \textit{recebedor} \quad & \mathsf{CanProve} \quad (\textit{pagador realizou pagamento}) \\ \textit{recebedor} \quad & \mathsf{CanProve} \quad (\textit{pagamento é válido}) \end{aligned}$$

As afirmações *(recebedor recebeu o pagamento), (pagador realizou o pagamento), (pagamento é válido)* são interpretações de mensagens assinadas.

Os objetivos de cada protocolo devem ser especificados de acordo com o protocolo em questão, juntamente com o significado das mensagens do protocolo. A análise deve mostrar que o protocolo atinge todos os seus objetivos.

4.2 Usando a lógica BAN

Os dois métodos formais apresentados nos permitem estudar mais objetivamente os sistemas de pagamento. Entretanto, nem todos os sistemas fazem uso de técnicas de au-

tenticação ou assinaturas digitais, o que inviabiliza a aplicação respectivamente da lógica BAN e do *framework* de Kailar.

Os sistemas que provêem mecanismos de autenticação e portanto permitem a aplicação da lógica BAN são: NETCHEQUE, NETBILL, IKP, SET e CYBERCASH. Durante a análise, percebemos que as diferentes variantes do IKP e o SET levam praticamente ao mesmo protocolo idealizado e, portanto, apenas a análise do 1KP será detalhada.

4.2.1 NetCheque

O NETCHEQUE [23], descrito na seção 3.12, é um dos sistemas de pagamento desenvolvidos na USC. É um sistema que imita o funcionamento dos cheques tradicionais e usa a autenticação baseada no KERBEROS[33], não fazendo uso de cifras assimétricas. Para assinar um cheque, o usuário deve obter um tíquete com o servidor KERBEROS e usar a informação contida neste tíquete para gerar sua assinatura. Todas as entidades devem ser cadastradas nos servidores KERBEROS.

Para realizar um pagamento usando o NETCHEQUE, o comprador deve obter um tíquete KERBEROS para se comunicar com seu banco. Este tíquete contém uma chave de sessão que permite a comunicação segura entre o comprador e o banco. Depois, o comprador preenche um cheque eletrônico com os campos valor, unidade monetária, data, número da conta, recebedor. Este cheque é assinado da seguinte maneira: o comprador gera um *hash* do cheque e o inclui num autenticador KERBEROS. Este autenticador é cifrado com a chave de sessão comprador-banco, anexado ao tíquete obtido e enviado ao vendedor. O vendedor endossa o cheque de maneira semelhante e o envia ao banco.

Quanto ao NETCHEQUE, vamos analisar a transação de pagamento, ou seja, a transferência de um cheque eletrônico. Neste caso, gostaríamos de demonstrar que o banco, ao receber o cheque, é capaz de reconhecer que o *hash* foi enviado pelo comprador. O banco poderá então verificar que este *hash* corresponde ao cheque e certificar-se de que o cheque é válido.

O protocolo idealizado

O participantes são o comprador (A), o banco (B) e o servidor KERBEROS (S). Não consideramos a participação do vendedor, já que este apenas repassa os dados que recebeu do comprador para o banco. As mensagens trocadas no protocolo são:

$$\begin{aligned}
&Mensagem\ 1.\ A \rightarrow S:\ A, B\\
&Mensagem\ 2.\ S \rightarrow A:\ \{T_s, L, K_{ab}, B, \{T_s, L, K_{ab}, A\}_{K_{bs}}\}_{K_{as}}\\
&Mensagem\ 3.\ A \rightarrow B:\ \{T_s, L, K_{ab}, A\}_{K_{bs}}, \{A, T_a, M\}_{K_{ab}}, Cheque.
\end{aligned}$$

Onde T_s e T_a são *timestamps*, L é um tempo de validade e M é o *hashing* do cheque. A mensagem 3 somente passa pelo vendedor, que atua como intermediário e é incapaz de alterá-la.

O protocolo idealizado é o seguinte:

$$Mensagem\ 2.\ S \rightarrow A:\ \{T_s, (A \overset{K_{ab}}{\leftrightarrow} B), \{T_s, A \overset{K_{ab}}{\leftrightarrow} B\}_{K_{bs}}\}_{K_{as}}$$
$$Mensagem\ 3.\ A \rightarrow B:\ \{T_s, A \overset{K_{ab}}{\leftrightarrow} B\}_{K_{bs}}, \{T_a, M, A \overset{K_{ab}}{\leftrightarrow} B\}_{K_{ab}}$$

Note-se que a primeira mensagem foi eliminada, já que não contribui para as propriedades lógicas do protocolo. Para simplificar, o tempo de validade L foi combinado com o *timestamp* T_s e é tratado como um *nonce*[1].

A análise do protocolo

As suposições são:

$$A \models A \overset{K_{as}}{\leftrightarrow} S \quad B \models B \overset{K_{bs}}{\leftrightarrow} S$$
$$S \models A \overset{K_{as}}{\leftrightarrow} S \quad S \models B \overset{K_{bs}}{\leftrightarrow} S$$
$$S \models A \overset{K_{ab}}{\leftrightarrow} B$$

Que indicam que os participantes conhecem e confiam nas chaves para comunicação com o servidor *Kerberos*, que gerou uma chave adequada para a comunicação entre A e B.

$$A \models (S \Rightarrow A \overset{K}{\leftrightarrow} B) \quad B \models (S \Rightarrow A \overset{K}{\leftrightarrow} B)$$

Indicam que o servidor tem jurisdição para gerar chaves para comunicação entre A e B.

$$A \models \sharp(T_s) \quad B \models \sharp(T_s) \quad B \models \sharp(T_a)$$

Estas três últimas suposições indicam que o KERBEROS depende de relógios sincronizados, já que os participantes acreditam que *timestamps* gerados por outros são recentes.

Quando A recebe a mensagem 2, usando as regras de verificação de *nonces* e de significado das mensagens, obtemos:

$$A \models A \overset{K_{ab}}{\leftrightarrow} B \quad A \triangleleft \{T_s, A \overset{K_{ab}}{\leftrightarrow} B\}_{K_{bs}}$$

[1] Um *nonce* é um valor aleatório usado apenas uma vez que permite determinar se uma mensagem faz parte da execução corrente do protocolo.

A envia a terceira mensagem para B, que consegue decifrar a primeira parte, já que conhece a chave compartilhada com S. Desta forma B obtém a chave a ser usada com A e pode decifrar a segunda parte da mensagem. Após estes passos, obtemos:

$$B \models A \overset{K_{ab}}{\leftrightarrow} B \quad B \models A \models A \overset{K_{ab}}{\leftrightarrow} B$$
$$B \models A \models M$$

Assim, B já se convenceu que A enviou M, e que A também acredita em M. Em suma, se M corresponde a um cheque, B está convencido de que A realmente pagou alguém usando este cheque. O recebedor deste cheque é identificado num dos campos que compõem o cheque.

4.2.2 NetBill

O NETBILL [22] (seção 3.10) foi desenvolvido na CARNEGIE MELLON UNIVERSITY e é um esquema para venda de informações pela Internet. É um protocolo completo, que inclui uma fase de negociação de preços e a entrega dos bens é garantida. Neste sistema, um servidor central é responsável pelas contas dos usuários, e pode creditar estas contas a partir de contas correntes em bancos conveniados ou a partir de cartões de crédito. Neste sistema existem os compradores, os vendedores e o servidor central (entidade controladora), cada um com seu papel bem definido.

O sistema faz uso de um mecanismo de autenticação chamado PUBLIC KEY KERBEROS, que é uma variação do sistema KERBEROS tradicional onde não há necessidade de um servidor central para autenticar os usuários. Em compensação, é necessária uma infra-estrutura para emissão de certificados e todos os participantes devem ter um par de chaves para uma cifra assimétrica.

No PUBLIC KEY KERBEROS, o cliente não obtém o tíquete de um servidor central, mas do próprio servidor do qual vai requisitar os serviços. O cliente deve construir uma requisição, que vai ser cifrada com uma chave de sessão que por sua vez é cifrada com a chave pública do servidor. A mensagem completa é então assinada e enviada. Ao receber a requisição, o servidor irá gerar a chave de sessão a ser usada nas próximas comunicações e um tíquete semelhante àquele usado no Kerberos tradicional; irá também enviar este tíquete e a nova chave de sessão, cifrados com a chave de sessão escolhida pelo cliente. Após este processo, o PUBLIC KEY KERBEROS funciona como o KERBEROS tradicional.

O objetivo é demonstrar que este protocolo permite que o cliente receba uma chave de sessão e um tíquete, o que o habilita a continuar o protocolo, como no KERBEROS tradicional.

O Protocolo idealizado

O participantes são o comprador (A) e o vendedor (B) e as mensagens trocadas no protocolo são:

$$\textit{Mensagem } 1.\ A \rightarrow B:\ \{A, B, T_a, K_c\}_{K_o}, \{K_o\}_{K_c}, \{H\}_{K_a^{-1}}$$
$$\textit{Mensagem } 2.\ B \rightarrow A:\ \{Tiq, K_{ab}\}_{K_c}$$

Onde T_a é um *timestamp*, K_c e K_o são chaves temporárias, Tiq é o tíquete requisitado e K_{ab} é a chave de sessão a ser usada entre A e B, e H é um *hashing* do resto da mensagem. O protocolo idealizado é o seguinte:

$$\textit{Mensagem } 1.\ A \rightarrow B:\ \{T_a, A \overset{K_s}{\leftrightarrow} B\}_{K_o}, \{A \overset{K_s}{\leftrightarrow} B\}_{K_c}, \{H\}_{K_a^{-1}}$$
$$\textit{Mensagem } 2.\ B \rightarrow A:\ \{Tiq, A \overset{K_{ab}}{\leftrightarrow} B\}_{K_c}$$

A análise do protocolo

As suposições são:

$$A \models \sharp K_c \quad A \models \sharp T_a \quad A \models \sharp K_o$$
$$B \models \sharp T_a$$

Que indicam que as chaves e *timestamps* foram geradas pelos participantes para esta rodada do protocolo.

$$B \models A \Rightarrow A \overset{K}{\leftrightarrow} B \quad A \models B \Rightarrow A \overset{K}{\leftrightarrow} B$$

Indicam que, neste protocolo, cada participante confia no outro em relação à geração de chaves adequadas.

$$B \models \overset{K_s}{\leftrightarrow} A \quad A \models \overset{K_b}{\leftrightarrow} B$$

Os participantes devem conhecer as chaves dos parceiros a priori ou então obtê-las em certificados.

$$A \models A \overset{K_s}{\leftrightarrow} B \quad A \models A \overset{K_s}{\leftrightarrow} B \quad A \models A \overset{K_{ab}}{\leftrightarrow} B$$

Estas últimas suposições refletem o fato de que os participantes confiam nas chaves que geraram.

Quando B recebe a primeira mensagem, este vê H e pode concluir, pela verificação do *hash*:

$$B \models A \mid\sim H \quad B \models A \mid\sim (\{T_a, A \overset{K_s}{\leftrightarrow} B\}_{K_a}, \{A \overset{K_s}{\leftrightarrow} B\}_{K_b})$$

B então obtém K_a e as informações cifradas com K_a:

$$B \triangleleft A \overset{K_s}{\leftrightarrow} B \quad B \triangleleft (T_a, A \overset{K_s}{\leftrightarrow} B)$$

Assim, B pode concluir que A enviou T_a e K_c:

$$A \mid\sim (T_a, A \overset{K_s}{\leftrightarrow} B)$$

Agora B tem condições de verificar o *timestamp* e assim concluir que K_c é recente e, sabendo que K_c foi enviada por A, concluir:

$$B \models A \models A \overset{K_s}{\leftrightarrow} B$$

Assim, usando o postulado da jurisdição, B conclui:

$$B \models A \overset{K_s}{\leftrightarrow} B$$

Quando A recebe a mensagem 2, este conclui que a mensagem veio de B, já que está cifrada com K_c, e conclui que a mensagem é recente. A então pode concluir:

$$A \models A \overset{K_{ab}}{\leftrightarrow} B \quad A \triangleleft Tiq$$

A obteve então o tíquete para se autenticar com B seguindo agora o protocolo KERBEROS tradicional. O restante do protocolo KERBEROS foi estudado no artigo original da lógica BAN [5] e não será tratado aqui.

4.2.3 1KP

A divisão de pesquisas da IBM desenvolveu o IKP [14] (seção 3.8) para ser um sistema adequado à realização de transações pela Internet usando contas bancárias ou cartões de crédito. No fundo, trata-se de um sistema adequado para a transmissão de números de contas pela Internet.

O sistema faz uso de criptografia forte, com cifras simétricas e assinaturas digitais e consiste de três níveis distintos. No primeiro nível, apenas as entidades controladoras possuem um certificado vinculando sua identidade a uma chave pública. No segundo nível, o vendedores já devem ter um certificado e, conseqüentemente, um par de chaves

para uso em assinaturas. No terceiro nível, todos os envolvidos, inclusive os compradores, devem ter cada um seu par de chaves e o certificado correspondente. Esta hierarquia permite a implantação gradual do sistema, com um aumento da segurança à medida que a infra-estrutura se torna disponível.

Vamos aqui analisar o nível 1 do sistema, batizado de 1KP. Neste, o comprador envia ao vendedor um *nonce* e sua identificação, e o vendedor responde com informações não confidenciais, iniciando a transação. O comprador então envia as instruções de pagamento ao banco através do vendedor. O vendedor adiciona alguns dados às instruções de pagamento para indicar ao banco que concorda com a transação. Ao receber os dados, o banco verifica sua validade e realiza a transferência.

O objetivo é provar que o banco recebe o preço e a identificação do comprador (possivelmente um número de conta corrente ou cartão de crédito), assim como uma identificação da transação e do vendedor. O banco irá validar a transação apenas se os todos os dados estiverem corretos. A data da transação pode ser usada para diminuir o tamanho do banco de dados contendo os números de transação.

O protocolo idealizado

Os participantes são o comprador (C), o vendedor (M) e a instituição que processa transações com cartões de crédito ou banco (A). As mensagens trocadas são:

$$\begin{aligned}
&\textit{Mensagem } 1.\ C \rightarrow M:\ N_a, ID_C \\
&\textit{Mensagem } 2.\ M \rightarrow C:\ ID_M, TID, Data, N_m, H \\
&\textit{Mensagem } 3.\ C \rightarrow M:\ \{P, H, CAN, R_c\}_{K_a} \\
&\textit{Mensagem } 4.\ M \rightarrow A:\ ID_M, TID_M, Data, N_m, H, H(desc), \{P, H, CAN, R_c\}_{K_a} \\
&\textit{Mensagem } 5.\ A \rightarrow M:\ Resp_{K_a^{-1}} \\
&\textit{Mensagem } 6.\ M \rightarrow C:\ Resp_{K_a^{-1}}
\end{aligned}$$

Onde N_x é um *nonce* gerado por X; ID_X é a identificação de X; TID é o identificador da transação; *desc* é a descrição dos produtos; H é um *hashing* de P (preço), ID_M, TID, $Data$, N_m, ID_C, $H(desc)$; CAN o número da conta (cartão de crédito) do cliente e R_c um número aleatório.

O protocolo idealizado fica:

$$\textit{Mensagem } 4.\ M \rightarrow A:\ (ID_M, N_m, H), \{\langle P, H, N_c \rangle_{CAN}\}_{K_a}$$

Podemos notar que apenas a mensagem 4 afeta as propriedades lógicas do protocolo. As outras mensagens pode ser omitidas. No protocolo idealizado, N_m incorpora os outros dados transmitidos ($Data$, TID, etc).

A análise do protocolo

As suposições são:

$$M \models \overset{K_S}{\mapsto} A \qquad C \models \overset{K_S}{\mapsto} A \qquad A \models \overset{K_S}{\mapsto} A$$
$$A \models \sharp Data \quad A \models A \overset{CAN}{\rightleftharpoons} C$$

A primeira linha das suposições indica que todos os participantes conhecem a chave pública de A, ou têm acesso a esta chave num certificado. A segunda linha traz as suposições de que o banco consegue verificar que a data é recente e que o número de conta (cartão de crédito) é um segredo compartilhado entre o banco e o comprador.

Quando A recebe a mensagem 4, este vê $(P, H, R_c)_{CAN}$ e pode deduzir:

$$A \models C \mid\sim (P, H, R_c)$$

A também vê $(P, ID_M, TID, Data, N_M, H(desc))$, que representamos por N_M no protocolo idealizado, o que lhe permite calcular H e, sabendo que a data é recente (suposições), lhe permite concluir que H é recente. Assim, A pode concluir (pela regra da verificação de *nonces*):

$$A \models C \models (P, H, R_c)$$

Desta forma, descobrimos que o protocolo permite ao banco concluir que o comprador enviou uma ordem de pagamento de valor P, a ser paga a ID_M. O número da transação (TID) permite ao banco descobrir se o vendedor não está apresentando a mesma ordem de pagamento mais de uma vez. Vale lembrar que ID_M e TID foram usados para gerar H.

O problema detectado neste protocolo é que um adversário poderia interceptar a mensagem 4 e, alterando os dados não criptografados, impedir o que o pagamento aconteça, já que o banco receberia dados inconsistentes. Este fato configura a possibilidade de ataques de negação de serviço.

3KP, SET e outros sistemas

O IKP descreve mais dois níveis de segurança do protocolo, à medida que os participantes passam a possuir pares de chaves certificadas. No nível 3, o último, todos devem ter seus pares de chaves. Assim, todas as mensagens podem ser assinadas, o que permite evitar alteração do conteúdo das mensagens como descrito acima. Também é possível garantir o não-repúdio da participação nas transações. Em seu nível 3, o IKP é muito semelhante ao SET e ao CYBERCASH, dois outros protocolos com os mesmos objetivos do IKP.

Em geral, os sistemas que objetivam a transmissão segura de dados de cartões de crédito e usam sistemas de chaves públicas tem funcionamento semelhante ao IKP. As

principais variações dizem respeito à quantidade e tipo das informações auxiliares usadas e ao uso ou não de assinaturas ou cifras em cada mensagem ou em cada parte das mensagens. A idéia geral corresponde ao 1KP.

4.2.4 CyberCash

A transação de pagamento no CYBERCASH consiste, basicamente, do acordo do comprador e do vendedor acerca do preço e descrição de mercadoria, e do envio ao servidor CYBERCASH de informações que indiquem que ambos concordam e permitam identificar os participantes. O comprador é identificado por um cartão de crédito e o vendedor por uma conta com a CYBERCASH.

Queremos mostrar que o servidor recebe a identificação do cartão de crédito e (da conta) do vendedor, o valor da transação e consegue inferir que estes dados são recentes, e não uma repetição de mensagens de uma transação anterior.

O protocolo idealizado

Abaixo apresentamos uma versão simplificada do protocolo, que mantém as mesmas características do protocolo original mas reduz a quantidade de informação presente em cada mensagem. Sempre que possível, valores foram combinados ou omitidos.

O participantes são o comprador (C), o vendedor (M) e o servidor CYBERCASH (S). As mensagens trocadas no protocolo são:

$$
\begin{aligned}
&\textit{Mensagem } 1.\ M \rightarrow C:\ Desc, T_M, \{h(Desc, T_M)\}_{K_M^{-1}}\\
&\textit{Mensagem } 2.\ C \rightarrow M:\ \{CardData, amount, h(COrderID)\}_{K_S},\\
&\quad COrderID, \{h(\{h(Desc)\}_{K_M^{-1}}, CardData, amount, h(Desc), COrderID)\}_{K_C^{-1}}\\
&\textit{Mensagem } 3.\ M \rightarrow S:\ \{h(X, \{amount, sig_M(Desc), T_M\}_{PK_S})\}_{K_M^{-1}}\\
&\textit{Mensagem } 4.\ S \rightarrow M:\ Rec_M, Rec_C\\
&\textit{Mensagem } 5.\ M \rightarrow C:\ Rec_C
\end{aligned}
$$

Onde $Desc$ é a descrição do pedido; T_M corresponde à identificação da transação junto com a data para M; $CardData$ é a indicação do cartão de crédito a ser usado; $amount$ é o valor da transação; $COrderID$ é a identificação da transação e a data para o comprador; e Rec_A é um recibo emitido para A. X corresponde à mensagem 2, que é repassada para o servidor sem alterações.

O protocolo idealizado é o seguinte:

$$\textit{Mensagem } 1.\ M \rightarrow C:\ \{Desc, T_M\}_{K_M^{-1}}$$
$$\textit{Mensagem } 2.\ C \rightarrow M:\ \{\{(amount)_{CardData}\}_{K_C^{-1}}\}_{K_S}, \{\{h(Desc)\}_{K_M^{-1}}, h(Desc)\}_{K_C^{-1}}$$
$$\textit{Mensagem } 3.\ M \rightarrow S:\ \{X, \{amount, h(Desc)\}_{K_S}\}_{K_M^{-1}}$$

É importante notar que as duas últimas mensagens foram eliminadas, já que não contribuem para a autenticação dos participantes, e servem apenas para distribuir recibos após a transação ter se completado. Na conversão para o protocolo idealizado, convertemos as assinaturas que usavam *hashings* em simples aplicações da cifra com a chave privada. Para mais detalhes consulte a seção 12 de [5].

A análise do protocolo

As suposições são:

$$\begin{array}{lll}
C \models \overset{PK_C}{\mapsto} C & C \models \overset{PK_M}{\mapsto} M & C \models \overset{PK_S}{\mapsto} S \\
M \models \overset{PK_M}{\mapsto} M & M \models \overset{PK_S}{\mapsto} S & \\
S \models \overset{PK_C}{\mapsto} S & S \models \overset{PK_M}{\mapsto} M & S \models \overset{PK_S}{\mapsto} S \\
C \models \sharp COrderID & C \models \sharp T_M & M \models \sharp T_M \\
S \models \sharp COrderID & S \models \sharp T_M & \\
A \models A \overset{CardData}{\rightleftharpoons} S & S \models A \overset{CardData}{\rightleftharpoons} S &
\end{array}$$

As três primeiras linhas indicam quais chaves públicas os participantes precisam conhecer ou receber sob a forma de certificados. O certificado permite identificar M e sua conta. Supomos sempre que os participantes conhecem suas respectivas chaves privadas.

As duas linhas seguintes indicam que C e M geram *nonces* que são reconhecidos como recentes por S. O *nonce* gerado por M é reconhecido também por C. Vale lembrar que os *nonces* considerados incluem um valor de data e um número de série, permitindo que sejam facilmente reconhecidos como recentes. A última linha indica que A e S consideram $CardData$ como sendo um segredo compartilhado.

Quando C recebe a primeira mensagem, este passa a conhecer $Desc$ e, pelo postulado do significado das mensagens, pode concluir:

$$C \models M \mid\sim Desc \quad C \models M \mid\sim T_M$$

Depois, pelo postulado de verificação de *nonces*, C pode concluir:

$$C \models M \models Desc$$

Assim, C pode mandar a mensagem 2 para M, que não tem como identificar seu conteúdo. M apenas repassa a mensagem 2 a S dentro da terceira mensagem. S pode concluir, a partir da mensagem 3:

$$S \models M \models (amount, h(Desc))$$
$$S \models \sharp h(Desc)$$
$$S \models C \models (amount)_{CardData}$$
$$S \models C \models h(Desc)$$

A primeira linha, que pode ser obtida pela aplicação do postulado de conteúdo das mensagens e do postulado de verificação de *nonces*, indica que S pode concluir que M está de acordo com $h(Desc)$ e *amount*. Na segunda linha, está a indicação de que S conclui que $h(Desc)$ é recente, por verificação do *nonce* T_M. Na linha seguinte indicamos que S descobre que C está de acordo com *amount* e vincula este valor à conta indicada por $CardData$. Na última linha está a indicação de que C concorda com $h(Desc)$.

Assim, S é capaz de concluir que C e M concordam quanto a *amount* (valor da transação) e *Desc*, mesmo sem conhecer *Desc*. Com estas informações, S pode realizar a retirada de *amount* da conta vinculada a $CardData$ e transferir este valor para a conta de M.

Após esta análise é possível perceber que existem dois valores usados para identificar o comprador: sua chave pública e $CardData$. O uso de dois valores se dá pelo fato de que o sistema permite que um único usuário participe do sistema com mais de um cartão de crédito. Assim, é necessário identificar cada um dos cartões. O uso de uma única chave por usuário facilita a gerência e emissão de certificados. Como a identificação do cartão pode ser independente da chave do usuário, o sistema poderia permitir que dois ou mais usuários usassem o mesmo cartão, o que poderia ser útil para famílias ou empresas. Neste caso, seria possível identificar qual usuário realizou a transação por meio da chave usada.

Este protocolo é bem similar na sua concepção lógica ao SET e 3KP, já que consiste basicamente de uma maneira de indicar ao servidor um valor e um número de cartão de crédito. Também há a preocupação em garantir que as partes envolvidas concordam com o valor e o conteúdo da transação, sem contudo informar a descrição da transação ao servidor. Outra preocupação é impedir que o vendedor tenha acesso a informações relativas ao cartão de crédito do comprador.

4.3 Usando o Framework de Kailar

Os sistemas de pagamento que usam assinaturas digitais de maneira a permitir a aplicação do *framework* de Kailar são: IKP, SET e CYBERCASH. As interpretações destes protocolos para análise pelo *framework* são também muito semelhantes, o que nos levou à conclusão de que é possível construir um modelo geral para os sistemas de pagamento baseados em cartões de crédito.

Este modelo geral tem como objetivo capturar as principais características lógicas destes protocolos, simplificando o conteúdo das mensagens trocadas pelos participantes. Estudando estes sistemas, pudemos verificar que existe um conteúdo básico para cada mensagem, sendo que as variações consistem normalmente da maneira como são gerados *nonces* ou *timestamps*, do tipo das informações relativas ao cartão de crédito e de como estas informações são cifradas e também do tipo e quantidade de campos auxiliares nas mensagens. Para construir este modelo geral, trabalhamos com os sistemas CYBERCASH, SET, 3KP [10, 37, 14], sendo que o modelo é adequado também para generalizar o sistema SEPP [26].

4.3.1 Apresentação do modelo geral

O modelo consiste das seguintes mensagens, trocadas entre o vendedor (M ou *merchant*), o comprador (C) e o banco (A ou *acquirer*):

Msg 1. $M \rightarrow C : Cert_M,\ Desc,\ \{TID, Date, H(Desc)\}_{K_m^{-1}}$
Msg 2. $C \rightarrow M : Cert_C,\ \{\{TID, Date, H(Desc), CardData, amount\}_{K_c^{-1}}\}_{K_a},$
$\{H(Desc), TID, Data\}_{K_c^{-1}}$
Msg 3. $M \rightarrow A : \{\{TID, Date, H(Desc), CardData, amount\}_{K_c^{-1}}\}_{K_a},$
$\{\{H(Desc), amount, ID_M, TID, Date\}_{K_m^{-1}}\}_{K_a}$
Msg 4. $A \rightarrow M : \{Rec\}_{K_a^{-1}}$
Msg 5. $M \rightarrow C : \{Rec\}_{K_a^{-1}}$

Onde $Cert_X$ é o certificado de X; TID é um identificador da transação (nos protocolos originais TID pode ser gerado por C ou M, ou ambos, podendo corresponder a um ou mais campos nas mensagens); $Date$ corresponde à data e hora correntes ou um *timestamp*; $Desc$ é a descrição dos produtos ou serviços, incluindo o preço; H é uma função de *hashing*; K_X e K_X^{-1} são as chaves pública e privada de X; $CardData$ corresponde aos dados do cartão de crédito; $amount$ corresponde ao valor da transação e Rec é um recibo correspondente à transação.

4.3.2 Estudando o modelo

Objetivos do modelo geral

Tendo em vista o modelo geral, o tipo de transação ao qual este tipo de protocolo se destina e o objetivo do método de análise utilizado, listamos os seguintes objetivos para as transações:

1. M CanProve (C Says $H(Desc)$) // C concorda com $Desc$
2. M CanProve (A Says Rec) // Valor transferido de C para M
3. C CanProve (M Says $H(Desc)$) // M concordou com $Desc$
4. C CanProve (A Says Rec) // Valor transferido de C para M
5. A CanProve (M Says $H(Desc) \wedge M$ Says $amount \wedge C$ Says $H(Desc) \wedge C$ Says $amount$) // C e M concordam com $H(Desc)$ e $amount$
6. A CanProve (M Says $TID \wedge C$ Says TID) // C e M estão falando da mesma transação
7. A CanProve (C Says $CardData \wedge C$ Says $amount$) //C autorizou transferência

Assim, dentre os objetivos do protocolo, os que estão relacionados com a responsabilidade de cada um dos participantes são:

- O vendedor (M) deve ser capaz de provar que o comprador (C) concorda com $Desc$ e que o pagamento foi efetuado de maneira correta, ou deveria ter sido efetuado.
- O comprador (C) deve ser capaz de provar que o vendedor (M) concorda com $Desc$ e que o pagamento foi, ou deveria ter sido, efetuado de maneira correta.
- O banco (A) deve ser capaz de provar que o comprador e o vendedor concordam quanto à transação, ou seja, o mesmo TID, o mesmo valor ($amount$) e a mesma descrição de produtos ou serviços ($Desc$), e que o comprador autorizou a transferência informando o número de seu cartão de crédito.

Suposições do modelo

O modelo desenvolvido envolve algumas suposições para que possa funcionar corretamente. Entre as suposições está a necessidade de uso de pares de chaves assimétricas para autenticar os participantes das transações. Isto implica a necessidade de uma estrutura de certificação ou distribuição de chaves com segurança. É interessante notar que o

aparecimento desta necessidade de um sistema de distribuição de chaves no modelo geral ocorre porque os sistemas nos quais é baseado fazem uso de uma estrutura de certificação de chaves.

As suposições são:

1. C, M CanProve (K_a Authenticates A)
2. C, A CanProve (K_m Authenticates M)
3. M, A CanProve (K_c Authenticates C)
4. (X Says $H(Desc)$) $\Rightarrow$ (X concorda com $Desc$)
5. (C e M concordam com $H(Desc)$ e $amount$) $\wedge$ (C Says $CardData$) $\Rightarrow$ (C, M autorizaram a transferência)
6. (C Says $H(x)$) $\wedge$ (M Says $H(x)$) $\Rightarrow$ (C e M concordam com x)
7. A Says Rec $\Rightarrow$ (transferência efetuada)

A três primeiras linhas dizem respeito à estrutura de certificação e indicam que os participantes têm acesso aos certificados dos parceiros da transação. As outras linhas dão as interpretações das mensagens trocadas. Sempre que um dos participantes faz uma afirmação, há um significado implícito indicado nestas suposições.

Interpretação do protocolo

Para que possamos realizar a análise, devemos reescrever o protocolo de maneira a usar as construções apresentadas acima. No caso do modelo geral, o protocolo interpretado é apresentado a seguir. Cada um dos itens abaixo corresponde a uma das mensagens do protocolo original.

1. C Receives ($H(Desc)$) SignedWith K_m^{-1}
2. M Receives ($H(Desc)$) SignedWith K_c^{-1}
3. A Receives ($H(Desc), CardData, Amount$) SignedWith K_c^{-1}
   A Receives ($H(Desc), Amount$) SignedWith K_m^{-1}
4. M Receives Rec SignedWith K_a^{-1}
5. C Receives Rec SignedWith K_a^{-1}

Análise Vamos agora usar o framework apresentado para analisar a interpretação do protocolo generalizado. A cada mensagem indicamos as conclusões a que podemos chegar.

Mensagem 1 usando a suposição 2 e o postulado sign:
C CanProve (M Says $H(Desc)$) $\Rightarrow$ C CanProve (M concorda com $Desc$)

Mensagem 2 usando a suposição 3 e o postulado sign:
M CanProve (C Says $H(Desc)$) $\Rightarrow$ M CanProve (C concorda com $Desc$)

Mensagem 3 usando a suposição 3 e o postulado sign:
A CanProve (C Says ($H(Desc), CardData, Amount$))
Usando a suposição 8 e o postulado sign:
A CanProve (M Says ($H(Desc), Amount$))
Usando a suposição 9 e depois a suposição 6:
A CanProve (C e M concordam com $H(Desc)$ e $Amount$) $\Rightarrow$ A CanProve (C e M autorizam a transferência)

Mensagem 4 usando a suposição 7 e o postulado sign:
M CanProve (A Says Rec) $\Rightarrow$ M CanProve (transferência efetuada)

Mensagem 5 usando a suposição 7 e o postulado sign:
C CanProve (A Says Rec) $\Rightarrow$ C CanProve (transferência efetuada)

4.3.3 Outros sistemas

O uso do framework em outros sistemas, especialmente o NETCHEQUE[23] e o NETBILL[22], que são baseados no KERBEROS[33], não é possível. Embora o NETBILL faça uso de chaves públicas certificadas, as assinaturas usadas no protocolo são baseadas em chaves KERBEROS compartilhadas e não permitem que o emissor da mensagem seja responsabilizado. O framework apresentado é adequado apenas para a análise de protocolos que usem sistemas de assinatura digital baseados em chaves assimétricas.

Os sistemas baseados em cupons não usam, em geral, mensagens assinadas, já que tentam manter o anonimato das transações. Nas transações anônimas, não existe a identificação de todos os participantes, o que impede que estes sejam responsabilizados.

4.4 Conclusões

As análises acima permitem concluir que o NETCHEQUE realmente garante a realização de transações seguras. O NETBILL permite a autenticação dos participantes, o que garante a segurança das assinaturas usadas durante as transações. No 1KP, pudemos concluir

que as ordens de pagamento são recebidas e reconhecidas pelo banco, embora este sistema esteja sujeito a ataques de negação de serviço.

Mostramos também a generalização dos principais protocolos de pagamento eletrônico que se baseiam em cartões de crédito. O modelo obtido desta generalização pode ser usado para melhor compreender este tipo de sistema. O modelo é útil também para a realização de análises de alto nível, já que apresenta as características dos sistemas e omite muitos dos detalhes ligados à implementação destes.

Em seguida, apresentamos a análise formal do protocolo generalizado usando o framework de Kailar. Desta análise pudemos concluir que estes sistemas, em geral, necessitam de uma estrutura segura de distribuição de chaves, seja ela um diretório ou um mecanismo de certificação de chaves. Se considerarmos a existência desta estrutura, podemos concluir que o uso extensivo de assinaturas digitais pelos sistemas de pagamento baseados em cartões de crédito permite que todos os participantes sejam responsabilizados por seus atos.

Esta conclusão já não é possível nos sistemas baseados no KERBEROS (NETCHEQUE, NETBILL), já que usam autenticação baseada em chaves compartilhadas e servidores confiáveis. A premissa de que o servidor KERBEROS é confiável pode se tornar um problema em caso de disputa legal acerca de uma transação, ou se a segurança do servidor for comprometida. Os sistemas que tentam manter o anonimato das transações não permitem responsabilizar os participantes.

Capítulo 5

Conclusões

Neste trabalho apresentamos um esquema que permite estudar os sistemas de pagamento eletrônico com base em suas características, requisitos, funcionamento e aspectos de implementação. Este esquema foi aplicado em diversos sistemas encontrados na literatura, o que nos permitiu obter uma visão geral dos sistemas de pagamento existentes.

Pudemos comprovar que o modelo de funcionamento proposto no esquema é interessante para descrever uma variada gama de sistemas de pagamento e que o esquema pode ajudar no projeto de novos sistemas de pagamento. Isto se deu pela implementação do protótipo descrito na Seção 2.2.

Apresentamos também dois métodos formais usados para estudar sistemas criptográficos e os resultados da aplicação destes métodos formais nos sistemas de pagamento aos quais se adequavam. A utilização deste métodos formais nos permitiram montar o modelo de sistemas de pagamento com cartões de crédito apresentado na Seção 4.3.1 e ressaltar algumas características importantes deste tipo de sistema.

Como contribuições deste trabalho, ressaltamos:

1. o esquema de classificação de sistemas de pagamento eletrônico (Capítulo 2), que consideramos um avanço na área;

2. o modelo geral dos sistemas de pagamento baseados em cartões de crédito (Capítulo 4), que facilita a compreensão de muitos dos sistemas em uso comercial atualmente;

3. o protótipo do gerador automático de sistemas de pagamento (Seção 2.2), que consideramos de muito útil para o ensino deste tipo de sistema;

4. e o conjunto de tendências no desenvolvimento de sistemas de pagamento, apresentadas sob a forma de similaridades encontradas em sistemas de determinadas classes (Seção 3.16.1).

Apêndice A

Conceitos Básicos de Criptografia

Este apêndice apresenta os principais conceitos criptográficos usados em outras partes deste trabalho. Esta pequena introdução à criptografia não pretende ser completa ou exaustiva. Para maiores detalhes, consulte os livros de Menezes et al. [20], Schneier [31], ou Stinson [35].

A.1 Funções Criptográficas Elementares

Algumas classes de funções são básicas na implementação de técnicas criptográficas:

Função unidirecional: é uma função de um conjunto X em um conjunto Y, para a qual é *fácil* calcular $y = f(x)$, mas é *difícil* calcular $x = f^{-1}(y)$, $y \in Y$, $x \in X$, para a grande maioria dos elementos em Y.

Função unidirecional com porta de escape: é uma função unidirecional $f : X \rightarrow Y$ com a propriedade adicional de que, dada certa informação extra, torna-se *fácil* achar um $x \in X$ para um $y \in Y$, tal que $f(x) = y$.

Permutação: Seja S um conjunto finito. Uma permutação π em S é qualquer bijeção de S em si mesmo.

A.2 Cifras

Uma cifra é uma maneira de garantir a confidencialidade dos dados, ou seja, de codificar uma informação de forma a impedir que esta chegue ao conhecimento de pessoas não autorizadas. Nas seções abaixo apresentamos os dois tipos de cifras existentes: as cifras simétricas ou de chave secreta e as assimétricas ou de chave pública.

Uma cifra consiste de duas transformações: uma é usada para cifrar e outra para decifrar. A segurança das cifras está baseada no fato de um adversário ser incapaz de descobrir qual foi a transformação usada.

Geralmente, estas transformações são funções que requerem o uso de chaves, ou seja, de parâmetros adicionais. Assim, apenas as chaves precisam ser tratadas como segredo. A Figura A.1 mostra como funcionam as cifras de maneira geral.

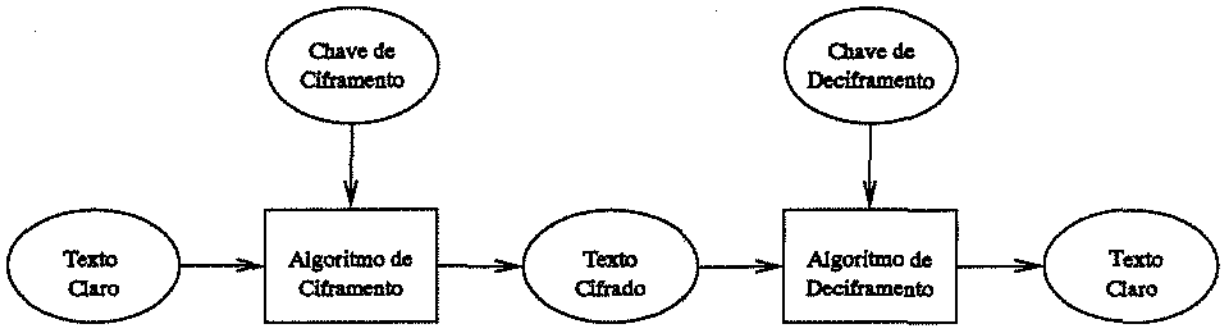


Figura A.1: Funcionamento de uma cifra

A.2.1 Cifras Simétricas

Uma cifra de *chave secreta* ou *cifra simétrica* é uma cifra para a qual é "fácil" determinar a chave de decifração a partir da chave de ciframento; em muitos casos as duas chaves são iguais. Portanto, as chaves devem ser mantidas em segredo. As cifras simétricas são classificadas em dois grupos principais: as cifras de blocos (*block ciphers*) e as cifras de fluxo (*stream ciphers*).

As cifras de blocos são aquelas que quebram o texto de entrada em blocos de tamanho fixo e cifram cada bloco separadamente com a mesma chave. A transformação aplicada para cifrar os blocos é a mesma em todos eles. As principais cifras simétricas usadas atualmente são cifras de blocos.

As cifras de fluxo são cifras que usam blocos de tamanho unitário e que variam a transformação aplicada a cada bloco. Este tipo de cifra é útil porque permite que os dados sejam processados um símbolo de cada vez mantendo a segurança, podendo ser úteis para cifrar as comunicações oriundas de dispositivos como teclados.

Uma consideração importante para avaliar a segurança de cifras simétricas é o tamanho do espaço de chaves, ou seja, quantas chaves diferentes podem ser usadas com uma determinada cifra. Uma condição necessária, mas não suficiente, para que uma cifra seja segura é que seu espaço de chaves seja grande o suficiente para dificultar uma busca exaustiva.

A maioria das cifras simétricas são permutações do espaço de textos claros (não cifrados), principalmente as cifras de blocos. Nestas, o bloco de saída pertence ao mesmo universo que o bloco de entrada, o que na prática quer dizer que os blocos de entrada e saída têm o mesmo número de bits.

Alguns exemplos de cifras simétricas são: DES, IDEA, LUCIFER, BLOWFISH.

A.2.2 Cifras Assimétricas

Uma cifra de chave pública ou cifra assimétrica é aquela na qual é "difícil" determin a mensagem original se soubermos apenas a mensagem cifrada e a chave de ciframento. Isto implica que dada a chave de ciframento é *difícil* obter a chave de decifração. Assim, a função de ciframento de uma cifra assimétrica é uma função unidirecional com porta de escape (a chave de decifração).

Quando cifras assimétricas são usadas, cada entidade tem duas chaves: uma é secreta e deve ser guardada com cuidado e a outra é pública e deve ser divulgada. As mensagens destinadas a uma entidade devem ser cifradas com sua chave pública e, no recebimento, a entidade usa a chave secreta para decifrar a mensagem. Como todos os usuários precisam ter acesso às chaves públicas existentes, deve haver uma maneira de autenticar a origem destas chaves.

As cifras assimétricas também são suscetíveis a ataques por busca exaustiva em seu espaço de chaves, embora, normalmente, existam ataques mais eficientes que obriguem ao uso de um espaço de chaves maior. Por exemplo, o esquema RSA é uma cifra assimétrica baseado na dificuldade de fatoração de números grandes. Neste esquema, ataques que usem algoritmos de fatoração são mais eficientes que busca exaustiva de chaves e já obrigam o espaço de chaves a ser bastante grande, da ordem de 1000 bits.

As cifras assimétricas mais populares são: RSA E ELGAMAL.

Funcionamento da cifra RSA

Uma das cifras assimétricas mais conhecidas e usadas foi desenvolvida por Rivest, Shamir e Adleman e é conhecida pela sigla RSA. Esta cifra está baseada em aritmética modular e sua segurança depende da dificuldade notória do problema da fatoração de números compostos muitos grandes. Seu princípio de funcionamento é resumido a seguir:

Inicialização do sistema

1. Escolha um módulo $n = pq$, onde p e q são números primos grandes (≥ 300 dígitos decimais).

2. Escolha uma chave pública e tal que $mdc(e, (p-1)(q-1)) = 1$.

3. Calcule uma chave privada d, tal que

$$ed \equiv 1 \bmod ((p-1)(q-1)).$$

4. Divulgue o par (e, n) como sendo a sua chave pública e guarde d como sua chave privada.

Ciframento

Para cifrar uma mensagem m, use a seguinte fórmula:

$$c \leftarrow m^e \bmod n$$

Decifração

Para decifrar uma mensagem c, calcule:

$$m \leftarrow c^d \bmod n$$

Como

$$c^d = (m^e)^d = m^{ed} = m^{k(p-1)(q-1)+1} = mm^{k(p-1)(q-1)} \equiv m.1 = m \pmod{n}$$

o algoritmo funciona corretamente. A expressão acima é válida apenas quando $mdc(m, n) = 1$, mas a demonstração para o caso geral será omitida.

A.3 Assinaturas Digitais

As assinaturas digitais são importantes para que se possa realizar autenticação, autorização e não-repúdio. O objetivo de uma assinatura digital é prover um mecanismo para que uma entidade associe sua identidade a alguma informação.

Um esquema de assinatura digital deve prover um mecanismo para geração de assinaturas e um mecanismo para verificação desta assinatura. Qualquer outra entidade, de posse da assinatura e da informação original, deve conseguir verificar sua validade, certificando-se assim que a informação não foi alterada e que a entidade que assinou tem conhecimento desta informação. Assim, dado um esquema de assinatura digital, devemos tem uma função de assinatura $assina : M \rightarrow A$ e uma função de verificação $verifica : M \times A \rightarrow \{Verdadeiro, Falso\}$, onde M é a mensagem e A é a assinatura. A aplicação da função *assina* gera uma assinatura que, em conjunto com a mensagem original, leva a função de verificação a emitir um resultado *Verdadeiro*, se e somente se a assinatura corresponde à mensagem original.

Uma maneira simples de se conseguir uma assinatura digital é com o uso de cifras assimétricas comutativa, ou seja, aquelas em que $D_A(C_A(X)) = C_A(D_A(X)) = X, \forall X$, onde C_A é a aplicação da cifra usando a chave pública de A e D_A é a aplicação da cifra usando a chave secreta de A. Assim, se A "cifra" a mensagem com sua chave secreta, qualquer entidade que tenha acesso a sua chave pública pode verificar que a mensagem foi enviada por A e não foi alterada, já que somente A conhece sua chave secreta. Se a mensagem tivesse sido alterada, o resultado da tentativa de verificação usando a chave pública de A produziria um texto ininteligível. O resultado de uma assinatura digital, ao contrário da assinatura manual, depende da informação sendo assinada.

Uma assinatura digital pode levar a informação original consigo ou não. No primeiro caso, a verificação da assinatura pode ser feita de forma automática. No segundo, é necessário obter a informação original antes de realizar a verificação.

A.3.1 Assinatura às Cegas

Algumas vezes, é necessário um mecanismo que permita a uma entidade assinar uma mensagem sem conhecer seu conteúdo. Um destes mecanismos chama-se *assinatura às cegas* (*blind signature*) [8]. Consiste em aplicar uma transformação que "esconde" o conteúdo da mensagem, mantendo-a passível de ser assinada. Este conceito tem propriedades análogas a um envelope munido internamente de papel carbono: uma assinatura no exterior do envelope é transferida para o seu conteúdo sem revelá-lo. Desta maneira, a entidade que assina o valor não é capaz de identificar o que assinou. A transformação aplicada é chamada *blindagem*.

Para a cifra RSA, se Pedro tem d como chave privada, (e, n) como pública e Gisele deseja que Pedro assine a mensagem m sem conhecê-la, Gisele executa os seguintes passos:

1. Gisele escolhe um número aleatório k, entre 1 e n e calcula

$$t \leftarrow mk^e \bmod n$$

2. Pedro assina t:

$$t^d = (mk^e) \bmod n$$

3. Gisele recupera a mensagem assinada calculando

$$s \leftarrow t^d/k \equiv m^d k/k \equiv m^d \bmod n$$

4. Gisele então está de posse da mensagem assinada

$$s = m^d \bmod n$$

A.4 Funções de Espalhamento

Uma *função de espalhamento* é uma função h computacionalmente eficiente que mapeia cadeias de tamanho arbitrário em cadeias de um tamanho fixo predeterminado, normalmente menor que o tamanho das cadeias originais. Dado um valor x, o valor $h(x)$ é conhecido como *valor de hash* de x, ou *hash de* x. Em suma, uma função de espalhamento é uma função que, dada uma cadeia de um universo possível, apresenta uma saída de um contra-domínio de menor cardinalidade.

Uma cadeia de *hashings* é uma seqüência de valores $n_0, n_1, ..., n_k$, onde

$$n_i = h(n_{i-1}),\ i \geq 1,$$

h é uma função de espalhamento criptograficamente forte e n_0 é um valor aleatório. Como h deve ser difícil de inverter, é computacionalmente impossível obter n_i a partir de n_{i+1}.

Uma colisão de uma função de *hashing* é um conjunto de valores $x_1, x_2, ..., x_i, i \geq 2$, tais que $h(x_i) = h(x_j), i \neq j$.

Uma propriedade importante para que uma função de espalhamento seja usada em criptografia é ser computacionalmente difícil gerar duas cadeias $x \neq y$ tais que $h(x) = h(y)$. Deve também ser difícil encontrar uma cadeia x que corresponda a um dado valor de hash y. Isto é, h deve ser unidirecional e resistente a colisões. As funções de espalhamento são usadas para:

- Gerar assinaturas digitais compactas: normalmente, para assinar um documento, gera-se um *hash* deste documento que é então assinado. Para verificar a assinatura, gera-se um novo *hash* do documento original que é comparado com o *hash* assinado. Se a função de espalhamento usada for adequada não será possível encontrar outro documento que gere o mesmo *hash* e a assinatura será segura.

- Garantir a integridade de dados: um *hash* dos dados é armazenado ou enviado juntamente com os dados originais. Pode-se, depois, verificar se os dados levam ao mesmo valor de *hash*, o que significa que não foram alterados.

- Protocolos com acordos a priori: a verificação de informação trocada a priori pode ser feita sem que a informação tenha que transitar numa rede. Para isso cada participante envia um *hash* da informação aos outros.

Alguns exemplos de funções de espalhamento são MD5, MD4, SHA.

A.4.1 Assinaturas Duais

Uma das aplicações de funções de espalhamento é a geração de *assinaturas duais*. Assinaturas duais ou *dual signatures* são um tipo de assinatura que permite apresentar uma

ligação entre duas partes de uma informação sem a necessidade de revelar uma das partes. Para gerar uma assinatura dual, é necessário calcular um *hashing* de cada parte da informação. Depois basta assinar a concatenação dos dois *hashings*.

Assim, o assinante pode enviar uma parte da informação e o *hashing* da outra parte e o recebedor será capaz de verificar a validade da informação. Se o assinante resolver provar que as duas partes da informação estão ligadas, basta revelar a outra parte. Assim o recebedor refaz a assinatura e verifica sua validade.

A.5 Protocolos Criptográficos

Um protocolo criptográfico é um algoritmo distribuído que descreve as ações necessárias para que duas entidades atinjam um determinado objetivo de segurança. Os principais protocolos criptográficos são aqueles projetados para permitir:

Estabelecimento de chaves: permitir que uma chave secreta chegue ao conhecimento de duas ou mais entidades, sem contudo torná-la acessível a entidades não autorizadas.

Gerência de chaves: processos que permitem o estabelecimento de chaves e a manutenção das relações entre as entidades envolvidas no transporte e armazenamento de chaves.

Certificação de chaves: garantia de que a chave recebida provém da origem correta e de que a chave não foi alterada sem autorização.

Autenticação: garantia da autenticidade da identidade de uma entidade ou da origem de mensagens.

Este protocolos podem ser combinados, produzidos outros capazes de realizar tarefas mais complexas ou específicas.

A.5.1 Protocolo Challenge-Response

O protocolo de *challenge-response* é um dos protocolos para verificação de acordo a priori mais conhecidos. Em geral é usado para verificar se outra entidade está de posse de um segredo compartilhado.

Uma das maneiras de se autenticar um usuário é pelo uso de senhas secretas. No protocolo básico de autenticação por senhas, o usuário envia a senha pela rede e o servidor irá verificar se a senha está correta e então permitir ao usuário acesso a seus serviços. Este protocolo é falho no sentido de que um adversário poderia escutar as transmissões que

circulam na rede e interceptar a senha. A partir daí, o adversário poderia personificar o usuário, tendo acesso não-autorizado aos serviços do servidor.

Para corrigir esta falha, usa-se o protocolo challenge-response, que consiste em enviar um pedido de inicio de conexão ao servidor, que irá enviar um desafio ao usuário. Este desafio é, normalmente, um número aleatório. O usuário irá usar um programa para cifrar este desafio usando sua senha como chave (alguns sistemas invertem e usam o desafio como chave para cifrar a senha) e devolvê-lo ao servidor. O servidor então verifica se a resposta ao desafio está correta e informa ao usuário que a autenticação foi bem sucedida.

A.5.2 O Sistema Kerberos

O KERBEROS [33] é um dos mais famosos sistemas de segurança da atualidade. O Kerberos foi baseado num sistema de estabelecimento de chaves com servidor confiável e permite realizar autenticação e estabelecimento de chaves para comunicação segura entres duas entidades de uma rede.

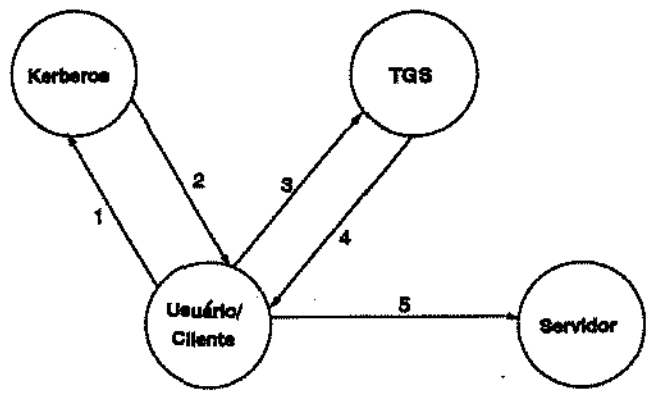


Figura A.2: Etapas do protocolo KERBEROS

No KERBEROS, participam entidades, que podem ser programas ou pessoas, e um servidor central. Sempre que alguma entidade (cliente) deseja solicitar um serviço a outra entidade (servidor), este cliente deve enviar uma solicitação de autenticação ao servidor KERBEROS, que irá verificar a identidade deste cliente através de *challenge-response* e lhe enviará um tíquete e uma chave de sessão. Com este tíquete, o cliente pode estabelecer comunicação com o servidor e se autenticar. O servidor aceitará esta autenticação e poderá usar a chave de seção para estabelecer um canal seguro com o cliente. Para que este processo possa acontecer, o servidor KERBEROS deve conhecer a senha de cada um dos participantes. O funcionamento do KERBEROS é mostrado na Figura A.2, cuja legenda é:

1. Requisição pelo *Ticket-Granting ticket*, que permite ao usuário requisitar ao TGS

(*Ticket-Granting server* ou servidor de tíquetes) tíquetes específicos para uso com os diversos serviços da rede.

2. O servidor KERBEROS envia o *Ticket-Granting ticket.*

3. Requisição de um tíquete de serviço, que permitirá o uso de serviços da rede.

4. Envio do tíquete de serviço.

5. Requisição do serviço, que inclui o tíquete correspondente a este serviço.

O sistema KERBEROS faz uso de cifras simétricas e senhas, que nunca trafegam na rede, necessitando de um protocolo *off-line* para seu estabelecimento. Algumas variações já foram propostas, inclusive para permitir a uso de cifras assimétricas e eliminar a necessidade de um servidor central.

A.5.3 Public Key Kerberos

Para eliminar a necessidade de um servidor KERBEROS central, foi desenvolvida uma variação do KERBEROS que faz uso também de cifras assimétricas. Cada usuário deve ter um par de chaves pública/privada.

Neste protocolo, as etapas 1 a 4 da Figura A.2 são condensadas em duas:

1. Tendo obtido a chave pública do servidor ao qual deseja requisitar um serviço, o cliente envia a este servidor uma requisição por um tíquete de serviço.

2. O servidor envia ao cliente o tíquete requisitado.

Após estas etapas, o protocolo funciona como o KERBEROS tradicional.

Bibliografia Comentada

[1] N. Asokan, Michael Steiner, and Michael Waidner. The state of the art in electronic payment systems. *IEEE Computer*, pages 28–35, September 1997.
Apresenta uma classificação dos sistemas de pagamento eletrônico e descreve alguns sistemas para micropagamentos.

[2] Anish Bhimani. Securing the commercial internet. *Communications of the ACM*, 39(6):29–35, June 1996.
Este artigo apresenta diversos requisitos necessários à realização de comércio na Internet.

[3] Jean Paul Boly, Antoon Bosselaers, Ronald Cramer, Rolf Michelsen, Stig Mjolsnes, Frank Muller, Torben Pedersen, Birgit Pfitzmann, Peter de Rooij, Berry Schoenmakers, Matthias Schunter, Luc Vallee, and Michael Waidner. The esprit project CAFE: High security digital payment systems. In *ESORICS 94, LNCS 875*, pages 217–230. Springer-Verlag, 1994.
Apresenta o sistema CAFE.

[4] Stefan Brands. Untraceable off-line cash in wallets with observers. In *Proceedings of Crypto 93*. Springer-Verlag, 1994.
Apresenta um interessante sistema de pagamento que faz uso de carteiras eletrônicas como obervadores e garante a privacidade dos usuários.

[5] Michael Burrows, Martín Abadi, and Roger Needham. A logic of authentication. *Proccedings of the Royal Society of London A*, 426, 1989.
Apresenta a lógica BAN e sua aplicação em diversos protocolos de autenticação.

[6] L. Jean Camp and Marvin Sirbu. Critical issues in internet commerce. *IEEE Communications Magazine*, pages 58–62, May 1997.
Apresenta os aspectos (confiabilidade, privacidade e segurança) do comércio eletrônico que os autores consideram críticos.

[7] L. Jean Camp, Marvin Sirbu, and J. D. Tygar. Token and notational money in electronic commerce. In *Proceedings of the first USENIX Workshop of Electronic*

Commerce: July 11–12, 1995, New York, New York, USA, pages 1–12, July 1995.
Apresenta as propriedades consideradas importantes em sistemas de pagamento eletrônico. Introduz a notação usada neste texto: sistemas notacionais ou baseados em cupons.

[8] David Chaum. Achieving electronic privacy. *Scientific American*, pages 96–101, August 1992.
Apresenta as ideias de blind signatures e de carteiras eletronicas com observadores.

[9] David Chaum, A. Fiat, and M. Naor. Untraceable electronic cash. In S. Goldwasser, editor, *Proceedings of Crypto 88*, LNCS 403, pages 319–327, Santa Barbara, CA, USA, August 1990. Springer-Verlag.
Apresenta um sistema off-line com privacidade total para os usuários que nao tentarem burlar o sistema.

[10] Cybercash web site.
URL: http://www.cybercash.com, 1998.

[11] Digicash web site.
URL: http://www.digicash.com, 1997.

[12] Steve Glassman, Mark Manasse, Martín Abadi, Paul Gauthier, and Patrick Sobalvarro. The millicent protocol for inexepensive electronic commerce. In *Proceedings of the 4th International World Wide Web Conference*, December 1995.
Apresenta o sistema Millicent.

[13] Iaik web site.
URL: http://www.iaik.tu-graz.at, 1998.

[14] P. Janson.and M. Waidner. Electronic payment systems. Activity Paper 211ZR018, Semper/IBM Zurich Research Lab, May 1996.
Este artigo apresenta uma classificação dos esquemas de pagamento eletrônico via Internet e dá detalhes do *iKP*, que foi desenvolvido pela IBM, além de uma comparação de diversos métodos de pagamento eletrônico. Este artigo está disponível eletronicamente na URL: http://semper.zurich.ibm.com/info/211ZR018.ps.

[15] Java web site.
URL: http://www.javasoft.com, 1998.

[16] Rajashekar Kailar. Accountability in electronic commerce protocols. *IEEE Transactions on software engineering*, 22(5), May 1996.
Apresenta o framework de Kailar que permite verificar as características de responsabilização em sistemas de comércio eletrônico.

[17] Larry Loeb. *Secure Electronic Transactions: Introduction and Technical Reference.* Artech House, 1998.
Um livro que facilita o entendimento do protocolo SET.

[18] Catherine A. Meadows. Formal verification of cryptographic protocols: A survey. In *Advances in Cryptology – Asiacrypt '94*, number 917 in LNCS, pages 133–150. Springer Verlag, 1995.
Este survey apresenta e discute os principais métodos formais para verificação de protocolos criptográficos.

[19] Gennady Medvinsky and B. Clifford Neuman. Netcash: A design for practical electronic currency on the internet. In *Proceedings of the First ACM Conference on Computer and Communications Security.* ACM, November 1993.
Apresenta os detalhes do sistema NETCASH.

[20] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography.* CRC Press, October 1996.
Um livro bem completo sobre criptografia, técnicas criptográficas e aplicações.

[21] Millicent web site.
URL: http://www.millicent.digital.com, 1997.

[22] Netbill web site.
URL: http://www.ini.cmu.edu/netbill/, 1997.

[23] Netcheque web site.
URL: http://nii.isi.edu/info/netcheque/documentation.html, 1997.

[24] B. Clifford Neuman. Security, payment and privacy for network commerce. *IEEE Journal on Selected Areas in Communications*, 13(8):1523–1531, october 1995.
Apresenta os requisitos para sistemas de pagamento eletrônico e algumas técnicas para atingí-los.

[25] T. Okamoto and K. Ohta. Universal electronic cash. In J. Feigenbaum, editor, *Proceedings of Crypto 91, LNCS 576*, pages 324–337. Springer-Verlag, 1992.
Apresenta um sistema de pagamento off-line com transferibilidade e indica 6 características que os sistemas de pagamento por cupons devem ter para serem considerados universais.

[26] Donal O'Mahony, Michael Peirce, and Hitesh Tewari. *Electronic Payment Systems.* Artech House, 1997.
Um livro bem completo sobre sistemas de pagamento. Apresenta os principais sistemas em detalhe, embora não tenha um tratamento uniforme.

[27] Paul-André Pays and Fabrice de Comarmond. An intermediation and payment system technology. In *Fifth International World Wide Web Conference*. GCTech, May 1996.
Apresenta o sistema Globe ID para comércio eletrônico.

[28] Michael Peirce and Donal O'Mahony. Scaleable, secure cash payment for WWW resources with the PayMe protocol set. In *Fourth International Conference on the World-Wide Web*, MIT, Boston, December 1995.
Apresenta o protocolo Payme, para pagamentos na Internet.

[29] Birgit Pfirtzmann and Michael Waidner. Strong loss tolerance of electronic coin systems. *ACM Transactions on Computer Systems*, 15(2):194–213, May 1997.
Trata de sistemas de pagamento com tolerância à perda de moedas.

[30] Ronald L. Rivest and Adi Shamir. Payword and micromint: Two simple micropayment schemes.
Disponivel em http://theory.lcs.mit.edu/ rivest/RivestShamir-mpay.ps, 1995.

[31] Bruce Schneier. *Applied Cryptography: Protocols, Algorithms and Source Code in C*. John Wiley, 2 edition, 1995.
Um dos livros mais populares sobre criptografia e suas aplicações.

[32] Andreas Schoter and Rachel Willmer. Digital money online: A review of some existing technologies. Technical report, Intertrader Ltd., February 1997.
Apresenta resumidamente os sistemas de pagamento eletronico mais comuns e os classifica de acordo com o trabalho do grupo que desenvolveu o NetBill.

[33] Jennifer G. Stein, Clifford Neuman, and Jeffrey L. Schiller. Kerberos: An authentication service for open network systems. In *USENIX Conference Proceedings*, pages 203–211, Winter 1988.
Este artigo apresenta o famoso sistema Kerberos, do M.I.T. O Kerberos permite autenticação e troca de mensagens cifradas numa rede.

[34] Lee H. Stein, Einar A. Stefferud, Nathaniel S. Borenstein, and Marshall T. Rose. The green commerce model. Internet Draft Internet Draft, First Virtual Holdings Inc., May 1995.
Especificação do Green Commerce Model com todos os detalhes.

[35] Douglas R. Stinson. *Cryptography: Theory and Practice*. The CRC Press Series on Discrete Mathematics and Its Applications. CRC Press, 1995.
Este livro apresenta uma ótima introdução aos aspectos teóricos de criptografia.

[36] Paul Timmers. Electronic commerce: An introduction. Published electronically at http://www.cordis.lu/esprit/src/ecomint.htm, May 1996.
Este artigo apresenta as principais formas de comércio eletrônico já existentes. Por fim, apresenta os principais problemas para o desenvolvimento do comércio na Internet.

[37] Visa and Mastercard. *Secure Electronic Transactions (SET) Specification - Book 1: Business Description*, June 1996.
Especificação do SET. Este primeiro volume apresenta uma visão de alto nível do protocolo, que foi projetado para permitir transações com cartões de crédito via Internet.