Um sistema de pagamento eletrônico com garantia de privacidade baseado no algoritmo criptográfico RSA

Gustavo Gattino[1], **Marcelo Danesi**[1], **Luciano Ignaczak**[1]

[1]Instituto de Informática – Universidade do Vale do Rio dos Sinos (UNISINOS)
Caixa Postal 275 – 93.022-000 – São Leopoldo – RS – Brasil

gustavo.gattino@gmail.com,{mdanesi, lignaczak}@unisinos.br

Abstract. *The exposure of customer's data can lead to profile mapping and unpleasant situations depending on the nature of the goods purchased. To solve this problem, a system to execute e-commerce transactions assuring the privacy of the customer is implemented. Through it, the customer is assured that the privacy of your information is secured toward the other entities involved in the process. The system uses a model with three entities — customer, shop and payment processor — which none of then can associate the purchase to the buyer.*

Resumo. *A exposição dos dados do cliente a diferentes entidades durante o processo de compra pode levar a mapeamento de perfil e situações constrangedoras, dependendo da natureza dos produtos adquiridos. Para solucionar esse problema, um sistema para realização de transações de comércio eletrônico garantindo a privacidade da parte compradora é implementado. Através dele, o usuário tem a garantia de que a privacidade de suas informações estão seguras para com as demais entidades envolvidas no processo. Esse sistema utiliza um modelo com três entidades — cliente, loja virtual e processador de pagamento — as quais não conseguem associar a compra realizada ao cliente comprador.*

1. Introdução

Recentemente as pessoas tornaram-se mais sensíveis em relação a sua privacidade *on-line* — o direito de controlar ou influenciar quais informações sobre elas podem ser coletadas, armazenadas e divulgadas [ISO 7498-2 1989] — percebendo que deixam todos os tipos de rastros ao navegar [Souza]. Casos como o monitoramento de pagamentos internacionais, bancos e transações de cartão de crédito tornaram usuários de serviços de pagamento *on-line* mais preocupados com quem acessa seus dados [Der Spiegel 2013].

A criptografia fornece meios de proteção à privacidade com relação a terceiros, de modo a dificultar a espionagem. No entanto, existem situações em que a proteção da privacidade deve ir ainda mais longe e é neste momento em que a identidade do usuário passa a ser o ponto mais importante nas trocas de informações. A demanda pela omissão da identidade *on-line* é completamente justificável, uma vez que muitas situações da vida *off-line* são anônimas: lojas físicas oferecem um certo grau de anonimato para seus clientes se esses pagarem com dinheiro.

O principal problema com pagamentos digitais é que o cliente é obrigado a passar as suas informações de identificação e pagamento ao vendedor, sendo essa falta de privacidade uma das principais razões que impedem o crescimento do comércio eletrônico, uma vez que limita a confiança de potenciais clientes [Grudzinski 2013] [Thomasson 2013].

Diante do problema levantado, o objetivo deste trabalho é elaborar um sistema de pagamento que garanta a privacidade das informações de pagamento do comprador e a não vinculabilidade entre as compras realizadas. Através desse sistema, o usuário terá a garantia total de que a privacidade de suas informações de transação estarão seguras para com as demais entidades envolvidas no processo de pagamento. O sistema define um modelo com três entidades: cliente, loja virtual e processador de pagamento. Entre as três entidades, apenas o processador de pagamento conhece a identidade do cliente, no entanto nenhuma entidade consegue associar a compra realizada ao cliente comprador.

A principal contribuição gerada por este trabalho é a construção do sistema de pagamento digital que faz uso da função criptográfica RSA, amplamente utilizada para garantir autenticidade e confidencialidade, como forma de garantir a não vinculabilidade entre uma compra realizada e o seu comprador. Essa garantia é possível através da utilização das propriedades de injeção e não sobrejeção que a função possui.

A segunda seção do artigo trata de trabalhos relacionados ao assunto de pagamentos digitais e privacidade de usuários na Internet. A terceira aborda o funcionamento do modelo elaborado. A quarta seção relata sobre a implementação do sistema criado, enquanto a quinta e a sexta seção apresentam os experimentos realizados e seus resultados. Por fim, a seção oito apresenta as considerações finais.

2. Trabalhos relacionados

Trabalhos anteriores mostram a dificuldade de manter a privacidade de um usuário no contexto de dados em rede e serviços *on-line* que expõem informações parciais do seu comportamento. [Backstrom et al. 2007] consideraram ataques à privacidade dos usuários, identificando-os através da estrutura de rede que os cercam e discutiram a dificuldade de garantir o anonimato do usuário quando há presença de dados na rede que os identificam. [Crandall et al. 2010] correlacionam laços sociais entre usuários, em que nenhuma correlação foi explicitamente declarada, apenas ao identificar padrões *off-line*: tendo em conta que duas pessoas estão aproximadamente na mesma localidade geográfica, aproximadamente ao mesmo tempo e em diversas ocasiões, estas, provavelmente, estão relacionadas. [Narayanan and Shmatikov 2008] quebraram o anonimato do *dataset* do algoritmo *Netflix Prize* usando informações do IMDB2, a qual tinha conteúdos de usuário similar, mostrando que a correlação estatística entre diferentes, mas relacionados, conjuntos de dados podem ser usados para atacar a privacidade.

As dificuldades na manutenção da privacidade motivaram a elaboração de novos modelos e sistemas. [Rennhard et al. 2004] apresentam novos componentes que permitem um comércio eletrônico seguro baseado em pseudônimos. De um lado, estes componentes permitem que clientes possam navegar através de um loja virtual, selecionar seus bens e pagá-los com seu cartão de crédito, de tal forma que nem a loja, nem o emissor do cartão de crédito, nem um intruso serão capazes de obter qualquer informações sobre a identidade do cliente. Por outro lado, é garantido que nenhuma das partes envolvidas é capaz de atuar desonestamente durante o pagamento. [Konidala et al. 2012] propõe um modelo de pagamento *mobile* pré-pago baseado em HTTPS, no qual o cliente obtém informações sobre a conta bancária do comerciante e instrui seu banco a transferir dinheiro a conta no momento do pagamento. O modelo faz uso do esquema de assinatura parcialmente cega para esconder a identidade do cliente do banco e co-

merciante. Como resultado, o modelo provê ao cliente um maior controle sobre seus pagamentos e proteção de sua privacidade, tanto para com o banco quanto ao comerciante. [Nakanishi and Sugiyama 2005] propõe um sistema de moeda eletrônica *on-line* anônima. No modelo, a moeda eletrônica é uma assinatura digital do banco, assinada de forma cega através do algoritmo Camenisch-Lysyanskaya e oculta através de criptografia e técnicas *zero-knowledge* durante o pagamento, impossibilitando a ligação entre transações. O modelo pode revogar o anonimato impedindo sua utilização para fins ilícitos.

3. Sistema de pagamento eletrônico

Este artigo define um sistema para realização de transações de comércio eletrônico garantindo a privacidade da parte compradora. Através dele, o usuário tem a sua privacidade assegurada durante o processo de compra. Essa segurança se dá através do uso de pseudônimos na compra com a loja virtual e da ofuscação do *ticket* escolhido pelo usuário com o processador de pagamento. Através desses dois elementos, o anonimato da compra é alcançado, garantindo a não vinculabilidade entre uma ou mais compras realizadas através do sistema e as identidades utilizadas para a compra. O sistema utiliza um modelo com três entidades — cliente, loja virtual e processador de pagamento.

3.1. Funcionamento do sistema

O sistema utiliza oito etapas para a realização da compra, conforme mostrado na Figura 1, as quais são descritas a seguir.

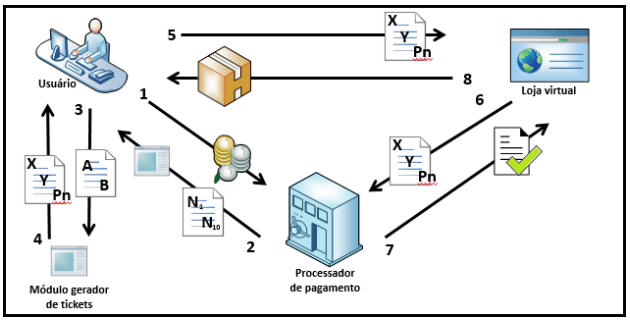

Figura 1. Processo de compra do sistema de pagamento proposto

No início do processo de compra, o cliente entra em contato com o processador de pagamento solicitando a aquisição de um *ticket* de pagamento. Nesse momento, para adquirir o *ticket*, o cliente realiza um pagamento digital amplamente adotado (1) — este pagamento pode ocorrer através de cartão de crédito, boleto bancário, Paypal, etc.

Após concluída a transação, o processador de pagamento envia ao cliente o módulo de geração de *tickets* e um conjunto de n valores (2). O módulo e os valores enviados são utilizados para a geração do *ticket* do usuário. A partir do momento em que esses dados são recebidos, o usuário tem um espaço de tempo t para geração e gasto do seu *ticket*.

Ao receber os dois componentes, o usuário escolhe um par de valores dentre os n recebidos — os dois valores escolhidos serão representados por A e B — e submete-os ao módulo de geração (3). O módulo de geração deriva matematicamente os valores A

e B em dois novos valores distintos, representados por X e Y, usando o algoritmo criptográfico RSA. Esses dois novos valores não podem ser vinculados aos valores originais, porém o processador de pagamento consegue verificar a sua autenticidade. A relação matemática entre A e B e X e Y é usada como autenticação.

Após a derivação, o módulo de geração utiliza os valores resultantes em uma função *Proof-of-Work*. Sua execução exige $(t/2) + 1$ minutos para conclusão (visando impedir a utilização de mais do que dois dos valores enviados — *double spending*). Após concluído o PoW, o módulo de geração retorna ao usuário o *ticket*: uma *string* que concatena os valores X e Y e o valor utilizado pelo PoW como *nounce*(4). Após a execução do módulo gerador, o usuário tem em torno $(t/2) - 1$ minutos para realizar a compra.

De posse do *ticket*, o usuário já pode realizar a sua compra virtual. Para isso, o usuário envia-o à loja virtual (5) que, por sua vez, valida sua autenticidade junto ao processador de pagamento (6). O processador de pagamento comprova a validade do ticket através de um verificador de PoW e confere se o valor recebido já não consta em seu banco de dados de tickets recebidos (7).

Validado o *ticket*, a loja virtual libera o produto ao comprador (8). A compra realizada não pode ser vinculada ao usuário, pois o processador de pagamento não sabe qual A e B o usuário escolheu para gerar X e Y e o mesmo não enviou seus dados reais.

O funcionamento correto do sistema depende do atendimento de alguns requisitos obrigatórios, os quais definem limites no escopo do seu uso.

1. Limite de instâncias do módulo gerador de *tickets*: O usuário deve ser incapaz de executar duas ou mais instâncias do módulo de geração de *tickets* de forma paralela.
2. Definição do processador de pagamento como entidade de confiança: O processador de pagamento deve ser uma entidade confiável e não vincula os números enviados a clientes com seus valores derivados finais através da submissão desses ao módulo gerador de *tickets*.
3. Definição do usuário como entidade anônima na rede: O usuário faz uso de tecnologias que o qualificam como anônimo na rede ao comunicar-se com as outras entidades envolvidas no processo de compra.

4. Implementação

Para a realização de experimentos foi necessário o desenvolvimento de um protótipo para realizar testes de segurança das funcionalidades do sistema proposto. A implementação contemplou o módulo de geração de *tickets*, utilizado pelo comprador, e o módulo de validação dos *tickets*, presente no sistema de processamento de pagamentos. Os dois módulos foram desenvolvidos usando a linguagem de programação Python 3.3.

O módulo de geração de *tickets* recebe o par de valores selecionados pelo usuário e aplica-os na função m^e mod N do algoritmo RSA. Os valores de e e N são fixados pelo processador de pagamento de forma que sejam coprimos para que o resultado da função seja sempre único. Os valores resultantes da função são utilizados no PoW baseado em Hashcash onde, concatenados a um nounce, são submetidos a uma função *hash* SHA-1. O resultado desse hash é convertido para a base hexadecimal e tem seus dígitos iniciais analisados a procura de uma quantidade de zeros. Para que o *ticket* seja informado ao usuário,

a quantidade de zeros presentes nos dígitos iniciais do resultado do *hash* deverá ser maior ou igual a z. Caso a quantidade não seja suficiente, o processo é reiniciado informando um novo *nounce* até que esse requisito seja atendido. O número de zeros hexadecimais a serem gerados pelo PoW são escolhidos com base na velocidade de criação de *tickets* que se deseja alcançar. Ao fim do processamento, o módulo apresenta ao usuário o *ticket* de pagamento, formado pela concatenação do último nounce utilizado e os dois números resultantes da função m^e mod N.

O módulo validador recebe o *ticket* enviado pela loja virtual, identificando se ele foi gerado através do módulo gerador e se o par de valores usados são válidos para a compra. O *ticket* recebido é aplicado a uma função *hash* SHA-1 e tem seu valor resultante convertido para a base hexadecimal. O valor convertido é analisado verificando se a quantidade de zeros aceitos configurados no PoW está correta. Se a quantidade mínima de zeros iniciais for identificada, então o módulo verifica se o par de valores passados está presente no banco de dados de *tickets* válidos. Caso os critérios sejam atendidos, o módulo classifica o *ticket* recebido como válido.

5. Experimentos

Para analisar a segurança do sistema a respeito de colisões na escolha dos valores enviados aos compradores e a resistência do sistema a ataques de força bruta de fabricação de *tickets* foi necessário implementar novas aplicações.

Para a realização dos testes de colisão e ataques de fabricação através do método de força bruta, foram utilizados três conjuntos de anonimato contendo uma quantidade de 100, 250 e 500 participantes. Além disso, os testes consideraram três quantidades de valores para a geração do *ticket*, sendo eles 1.000, 5.000 e 10.000. Esses valores foram escolhidos pelo autor com o objetivo de validar diferentes proporções na relação usuários e valores, uma vez que não foram encontradas referências abordando o assunto.

Para a análise das colisões na seleção de valores foi implementado um aplicativo capaz de gerar o par de valores pseudoaleatórios para cada usuário do teste, comparando-os à procura de colisões. O experimento fez uso de nove cenários de teste — 2 fatores com 3 níveis cada — no qual 100, 250 e 500 usuários escolhem um par de valores contidos em uma listagem de 1.000, 5.000 e 10.000 valores disponíveis. O número total de colisões geradas para cada uma das replicações foi utilizado como variável de resposta.

Para a avaliação da resistência a um ataque de força bruta foi implementado um aplicativo capaz de gerar aleatoriamente um par de valores pseudoaleatórios, submetê-los no módulo gerador de *tickets* e comparar os valores gerados com uma listagem de valores aceitos pelo processador de pagamento. A avaliação fez uso de três cenários de teste, nos quais um par de valores é escolhido de forma pseudoaleatória, submetido ao módulo gerador de *tickets* e é comparado a uma listagem de 1.000, 5.000 e 10.000 valores válidos. O número total de *tickets* válidos aceitos pelo processador de pagamento após a ocorrência de 100.000 tentativas foi utilizado como variável de resposta. Um total de 30 replicações foi utilizado para cada nível do fator primário.

6. Resultados

Nesta seção são apresentados e analisados os resultados provenientes dos experimentos identificados na Seção 5, os quais computaram 62 milhões de tentativas de colisão na

escolha de valores para variação e 9 milhões de tentativas de fabricação de *tickets* através do modulo gerador de *tickets*. Visando uma melhor compreensão, a análise dos resultados foi dividida em subseções distintas para cada teste realizado.

6.1. Colisão na seleção dos valores a serem variados

Inicialmente foram coletados o número total de colisões geradas para cada um dos três grupos de usuários e suas diferentes quantidades de valores disponíveis. Na Tabela 1, são identificados os resultados do teste para cada fator e seus níveis. A coluna "colisões" representa o total de ocorrências em que um par de valores foi escolhido duas ou mais vezes em uma mesma execução do teste.

Usuários	Valores	Repetições	Colisões	% de colisão
100	1.000	499.500	106	0,021 %
250	1.000	499.500	30.245	6,055 %
500	1.000	499.500	110.948	22,212 %
100	5.000	12.497.500	2.092	0,017 %
250	5.000	12.497.500	31.163	0,249 %
500	5.000	12.497.500	124.099	0,993 %
100	10.000	49.995.000	4.918	0,010 %
250	10.000	49.995.000	30.941	0,062 %
500	10.000	49.995.000	124.346	0,249 %

Tabela 1. Resultados do teste de colisão na escolha dos pares

Através da observação da Tabela 1 é possível perceber o aumento no número de colisões à medida que a quantidade de usuários cresce. Essa característica fica evidente, por exemplo, nos resultados dos níveis de 10 mil valores, os quais a duplicação da quantidade de usuários praticamente quadruplica a taxa de colisões — à medida que 250 usuários geram 30 mil colisões, 500 usuários aumentam a ocorrência para 124 mil.

Para identificar a relação entre o número de valores disponíveis para a escolha e taxa de colisões, um cálculo de probabilidade foi realizado, conforme apresentado na coluna "% de colisão". O resultado desse cálculo representa a chance de ocorrência de colisão para cada vez que os usuários devem escolher um par de números.

É possível observar que a probabilidade de ocorrência de colisão diminui a medida que mais valores para escolha são adicionados. Essa característica fica clara ao comparar as chances de colisão para 500 usuários, ao qual dispondo de mil valores para a escolha resulta em uma chance de 22% de colisão por rodada, porém o valor é reduzido assim que 5 mil valores são disponibilizados, diminuindo as chances para menos de 1%.

Após a análise dos dados, fica claro que nos dois diferentes fatores testados há um aumento na ocorrência de colisões, conforme a quantidade de usuários cresce. Essa condição independe da quantidade de valores disponíveis para escolha. A análise permite concluir que a proporção mínima adequada deve ser de 1 usuário para 100 valores, para que as chances de colisão fiquem abaixo de 0,01%.

6.2. Fabricação de *tickets* através de força bruta

Inicialmente, para a realização do experimento de fabricação através de força bruta, foi coletado o número total de *tickets* válidos gerados em cada um dos três níveis de valo-

res. Na Tabela 2, são apresentados os resultados dos testes para cada um dos níveis ao fator primário. A coluna "sucessos" identifica a quantidade total de *tickets* de pagamento válidos gerados para cada um dos níveis de teste.

Valores válidos	Repetições	Replicações	Sucessos
1.000	100.000	30	76
5.000	100.000	30	1.859
10.000	100.000	30	7.422

Tabela 2. Resultados do teste de fabricação de *tickets* através de força bruta

Após a coleta dos dados, os resultados do experimento foram analisados relacionando a quantidade de sucessos na geração dos *tickets* com a quantidade de valores aceitos pelo processador de pagamento. É possível perceber um crescimento da taxa de sucesso de criação de *tickets* válidos a medida que mais valores aceitos são disponibilizados para validação. Esse ponto apresenta uma relação inversa com a quantidade de valores disponíveis comparado aos dados apresentados pelo teste de colisão, no qual há uma maior dificuldade do usuário legítimo gerar um *ticket* válido quando a quantidade de valores disponíveis para escolha é menor.

No pior cenário, em que 10 mil valores estão disponíveis para validação, a taxa de fabricação de *tickets* é de 404 tentativas por sucesso. Ao considerar uma implementação do sistema configurado para que a aceitação do PoW exija 6 zeros de validação, um atacante gastará um tempo de 40 horas para obter sucesso na criação de um *ticket*. Todavia, apesar do aumento na taxa de sucesso na fabricação dos *tickets*, em virtude da funcionalidade de expiração dos valores enviados aos usuários, esse indicador não apresenta risco para o sistema caso uma grande quantidade de valores para escolha seja utilizada.

7. Considerações finais

Neste trabalho, foi elaborado um sistema para realização de transações de comércio eletrônico que assegura a privacidade do comprador durante o processo de pagamento da compra. Ela é garantida através do uso de pseudônimos no momento da compra com a loja virtual e da quebra de relação do *ticket* final utilizado para compra com os itens fornecidos pelo processador de pagamento para a sua geração. Através desses dois elementos, o anonimato da compra é alcançado, garantindo a não vinculabilidade entre uma ou mais compras realizadas através do sistema e as identidades utilizadas para a compra.

Dentre os itens enviados pelo processador de pagamento, o mais sensível é o conjunto de valores para a escolha do comprador, pois ele deve ser encaminhado a diferentes compradores para que o conjunto de anonimato do pagamento não permita ao processador de pagamento identificar qual usuário escolheu qual par de valores. Contudo, essa definição pode acarretar em colisão na escolha dos números — dois usuários diferentes podem escolher o mesmo par de números. Para tratar essa possibilidade, o sistema faz uso da mesma técnica utilizada pelos algoritmos criptográficos de chave pública: o conjunto de valores disponíveis para escolha é grande o suficiente para que a probabilidade de colisão seja mínima. No sistema, testes foram realizados identificando a necessidade de proporção de, pelo menos, 1 usuário para cada 100 valores disponíveis, fazendo com que as chances de colisão fiquem abaixo de 0,01%.

Além disso, o sistema é resistente a cenários nos quais um possível usuário mal-intencionado realize tentativas de fabricação de *tickets* válidos através de força bruta. Sua resistência a este tipo de ação ocorre através do uso do *proof-of-work*, o qual, além de exigir um *nounce* válido para o processamento do *ticket*, torna o processo de fabricação desinteressante a medida que seu processamento fica muito custoso. Mesmo assim, testes foram realizados simulando um cenário de ataque, os quais identificaram que, mesmo no pior cenário, um atacante não poderia fabricar um *ticket* rápido o suficiente para vencer a expiração dos valores disponibilizados para escolha.

As contribuições apresentadas neste trabalho representam um primeiro esforço para o uso de funções injetoras não-sobrejetoras como forma de alcançar a não vinculabilidade entre transações de comércio eletrônico. Apesar da segurança apresentada por meio dos estudos experimentais, o modelo pode ser aprimorado de forma que um limite de instâncias do módulo gerador de *tickets* seja assegurada, fazendo com que o usuário não possa gerar mais de um *ticket* de forma paralela. Em um segundo momento, também é possível que o processador de pagamento não seja mais considerado uma entidade de confiança, identificando uma forma mais segura para envio dos valores a serem derivados aos usuários.

Referências

Backstrom, L., Dwork, C., and Kleinberg, J. (2007). Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In *Proceedings of the 16th international conference on World Wide Web*, pages 181–190. ACM.

Crandall, D. J., Backstrom, L., Cosley, D., Suri, S., Huttenlocher, D., and Kleinberg, J. (2010). Inferring social ties from geographic coincidences. *Proceedings of the National Academy of Sciences*, 107(52):22436–22441.

Der Spiegel (2013). Follow the money: Nsa spies on international payments. http://is.gd/SpiegelNSA.

Grudzinski, G. (2013). Do online shoppers care about privacy? http://is.gd/IRPriv.

ISO 7498-2 (1989). *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 2: Security Architecture*. IEC.

Konidala, D. M., Dwijaksara, M. H., Kim, K., Lee, D., Lee, B., Kim, D., and Kim, S. (2012). Resuscitating privacy-preserving mobile payment with customer in complete control. *Personal and Ubiquitous Computing*, 16(6):643–654.

Nakanishi, T. and Sugiyama, Y. (2005). An efficient on-line electronic cash with unlinkable exact payments. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 88(10):2769–2777.

Narayanan, A. and Shmatikov, V. (2008). Robust de-anonymization of large sparse datasets. In *Security and Privacy, 2008. IEEE Symposium*, pages 111–125. IEEE.

Rennhard, M., Rafaeli, S., Mathy, L., Plattner, B., and Hutchison, D. (2004). Towards pseudonymous e-commerce. *Electronic Commerce Research*, 4(1-2):83–111.

Souza, E. A privacidade como diferencial. goo.gl/IbtzLr.

Thomasson, E. (2013). Big retailer is watching you: stores seek to match online savvy. http://is.gd/ITWatchYou.