Segurança em Internet Banking

Márcio A. S. Correia[1], **André L. M. dos Santos**[1], **Pablo R. Ximenes Ponte**[1], **Joaquim Celestino Jr.**[1], **Sérgio Luis O. B. Correia**[1]

[1]Departamento de Ciência da Computação - Universidade Estadual do Ceará (UECE)

marcioandre@gmail.com, andre@dossantos.org, pablo@ximenes.info, {celestino,sergio}@larces.uece.br

Abstract. *This article presents the security mechanism used by Brazilian banks that provide internet banking, developing a quick analysis of their benefits and problems. It also presents a complete set of tests used in the Bank of Brazil's e-banking safety measures as well as the vulnerabilities that were found.*

Resumo. *Este artigo apresenta os mecanismos de segurança utilizados pelos internet bankings brasileiros, desenvolvendo uma rápida análise dos seus benefícios e problemas. Em seguida, é apresentado um completo conjunto de testes realizados nos mecanismos utilizados pelo Banco do Brasil no seu e-banking e as vulnerabilidades encontradas.*

1. Introdução

O acesso aos serviços bancários por meio de internet bankings (*e-bankings*) ganhou destaque nos últimos anos. O fato do número de transações pela internet já superar as transações realizadas nos meios tradicionais comprova a popularidade do serviço [Febraban 2008] . Isto se justifica pelo fato do serviço aliar a conveniência para o usuário e o baixo custo para as instituições.

A segurança é um desafio enfrentado pelas instituições que oferecem o serviço de *e-banking*, uma vez que ele é alvo freqüente de fraudes [dos Santos et al. 2001]. Na tentativa de superar esse problema, as instituições bancárias brasileiras investiram, apenas no último ano, mais de seis milhões de reais em tecnologia [Febraban 2008].

O objetivo desse trabalho é fazer um levantamento dos mecanismos de segurança utilizados nos sistemas de *e-banking* das maiores instituições bancárias brasileiras. Além disso, este artigo apresenta um estudo detalhado da implementação de alguns desses mecanismos, constatando vulnerabilidades decorrentes da utilização de práticas não recomendadas nas suas elaborações.

Este trabalho faz parte do projeto final de curso de Márcio A. S. Correia, concludente do curso de graduação em Ciência da Computação da Universidade Estadual do Ceará (UECE), sob a orientação de André L. M. dos Santos, Joaquim Celestino Jr. e Pablo R. Ximenes Ponte, e contou com a ajuda de Sérgio Luis O. B. Correia.

O artigo está organizado da seguinte forma: a seção 2 apresenta um levantamento dos mecanismos de segurança utilizados nos *e-bankings* brasileiros, a seção 3 apresenta um exame de caso, a seção 4 discute o impacto dos resultados encontrados no exame de caso e finalmente, na seção 5 é construída a conclusão.

2. Mecanismos de segurança utilizados

Os sistemas de *e-banking* brasileiros utilizam vários mecanismos de segurança para proteger usuários de seus sistemas. Estes mecanismos de segurança podem ser divididos em duas categorias: mecanismos contra ataques remotos e mecanismos contra ataques locais.

Os mecanismos contra ataques remotos visam a proteger usuários contra ataques nos quais indivíduos agindo de má fé possam capturar dados sensíveis sem que tenham penetrado no computador do usuário do sistema. Estes mecanismos são altamente eficientes contra alguns tipos de ataques, como tentativas de *sniffing*[1], no entanto, falham totalmente contra outros ataques, como *phishing*[2], que devem ser tratados por mecanismos contra ataques locais.

A seguir são descritos os mecanismos de segurança encontrados nos serviços pesquisados.

2.1. Transport Layer Security

O Transport Layer Security (TLS) [Dierks and Allen 1999], é um protocolo criptográfico que provê comunicação na Internet para diversos serviços, inclusive o HTTP [Rescorla 2000], que é utilizado pelos *e-bankings*. O protocolo TLS provê a autenticidade, privacidade e a integridade dos dados transmitidos entre duas aplicações que estejam se comunicando pela Internet. Isto ocorre através da troca de credenciais, da criptografia e checagem dos dados. Esse protocolo ajuda a prevenir que atacantes tenham acesso ou falsifiquem os dados transmitidos. Uma vez estabelecida conexão entre o navegador e o servidor HTTP, é iniciada a geração de números randômicos por cada lado e em seguida esses números são trocados e encriptados, juntamente com as respectivas chaves públicas. Estes números randômicos são utilizados numa fase seguinte para geração de chaves criptográficas usadas para troca de dados criptografados entre os dois computadores.

O protocolo TLS é bastante flexível, permitindo sua utilização com diferentes algoritmos de criptografia, tamanhos de chaves, tempo para renegociação das chaves e etc. A utilização do protocolo TLS nos *e-bankings* brasileiros se mostrou uniforme, usando chaves de 1024 bits e o algoritmo RC4 para a criptografia dos dados em todos os serviços verificados. O maior problema de segurança do TLS não é devido a como ele é utilizado em aplicações de *e-banking*, mas sua necessidade de uma infra-estrutura de chaves públicas que seja de fácil utilização por usuários. Ataques como *phishing* exploram este problema, enganando usuários para que revelem seus dados sensíveis para entidades maliciosas.

2.2. Encerramento da Sessão

Uma sessão é representada por informações que são mantidas nos servidores a respeito de cada conexão recebida por ele. Quando o cliente efetua a autenticação, são mantidas informações nos servidores que permitem que ele seja identificado como um usuário

[1] *sniffing* é o procedimento realizado por uma ferramenta conhecida como *Sniffer*, capaz de interceptar e registrar o tráfego de dados em uma rede de computadores.

[2] *phishing* é uma forma de fraude eletrônica, caracterizada por tentativas de adquirir informações sensíveis, tais como senhas e números de cartão de crédito, ao se fazer passar como uma pessoa confiável ou uma empresa enviando uma comunicação eletrônica oficial.

autenticado e garantem acesso ao serviço. A sessão é mantida de acordo com regras impostas pelo serviço. O não atendimento a alguma dessas regras acarreta no encerramento da sessão, obrigando o usuário a autenticar-se novamente.

As regras de encerramento da sessão nos sistemas de *e-banking* que foram verificados são baseadas no tempo de inatividade e no endereço IP do cliente. Alguns *e-bankings* oferecem a flexibilidade de configuração do tempo de inatividade máximo, permitindo que o usuário configure a tolerância para até uma hora. Esta possibilidade de configuração da tolerância pode ser usada de maneira ingênua pelo usuário fazendo com que ele possa sofrer ataques, como roubo de cookie de sessão [Stallings 2007].

2.3. Chave temporal

A chave temporal é um mecanismo baseado no conceito do algoritmo criptográfico one-time pad [Litterio 1995]. Apesar das características de segurança comprovadas, as implementações de chaves temporais, por várias restrições fora do escopo deste artigo, estão longe de ter as mesmas características de segurança do one-time pad [Schneier 2002a].

Chaves temporais são utilizadas para tentar contornar o risco do roubo de senhas. O seu conceito se baseia na utilização de chaves que são válidas por um curto período de tempo. A segurança de uma chave temporal é baseada no fato de que uma chave comprometida num tempo t será inválida num tempo $t + \delta$, por ter sido trocada por outra chave durante o período de tempo δ. Portanto, tal chave comprometida só pode ser utilizada no período de tempo entre a troca de chaves, que idealmente é pequeno. Para que não haja possibilidade de uso de uma chave temporal fora da condição notada acima, elas não podem se repetir. Apesar de algumas implementações assegurarem a não repetição das chaves por um longo período, outras não dão esta garantia, o que compromete a segurança do mecanismo.

Figura 1. Chave temporal eletrônica (esquerda) e em cartão (direita)

A chave temporal foi encontrada em duas abordagens nos sistemas de *e-banking* brasileiros verificados (Figura 1). Uma abordagem implementa chaves temporais na forma de *token*, que é um dispositivo eletrônico especializado na geração das chaves temporais e garante a não repetição por um longo período. A outra abordagem implementa chaves temporais na forma de um cartão com uma seqüência de chaves, que serão solicitadas uma a cada tentativa de autenticação no serviço. A segunda abordagem reutiliza extensivamente as chaves, diminuindo consideravelmente a segurança do mecanismo. A atratividade desta solução é o custo, bem menor do que o dos *tokens*.

2.4. Teclado virtual

O teclado virtual é o principal mecanismo de segurança utilizado para prevenir ataques locais gerados por softwares maliciosamente instalados nos computadores dos usuários. A idéia de teclados virtuais se baseia no conceito de *Completely Automated Public Turing test to tell Computers and Humans Apart* (CAPTCHA) [von Ahn et al. 2004]. CAPTCHA's utilizam problemas de fácil solução para humanos e difícil solução para computadores para diferenciar entre um computador e um ser humano. Um dos grandes desafios no projeto de mecanismos de segurança baseados em CAPTCHA é garantir elevado nível de segurança mantendo a usabilidade do sistema.

O teclado virtual desenha números e/ou letras na tela de um computador para que humanos selecionem aqueles de seu interesse. Os princípios do CAPTCHA garantem que, embora seja fácil para humanos selecionar uma opção, seja muito difícil para um software, como um vírus instalado no computador do usuário, reconhecer esta seleção. Esse mecanismo de defesa foi bastante popularizado no Brasil e foi encontrado em todos os *e-bankings* verificados.

Figura 2. Teclado virtual com representações direta (esquerda) e indireta (direita)

Alguns teclados virtuais encontrados nos sistemas de *e-banking* brasileiros fazem uso de problemas que não são tão difíceis de serem resolvidos por computadores, o que limita a proteção proporcionada por este mecanismo. Além disso, teclados virtuais podem ser capturados como imagem que pode ser enviada para um sistema malicioso remoto junto com informações sobre a seleção do usuário. Um humano pode então, de posse da imagem e informações de seleção, resolver o CAPTCHA e identificar a seleção. A Figura 2 apresenta dois teclados virtuais que utilizam diferentes abordagens na representação dos dados. O da esquerda representa diretamente os dados informados pelo cliente, facilitando a reconstrução dos dados. O da direita faz uma representação indireta, gerando diversas possibilidades sobre os dados informados.

Outra preocupação é a de impedir a reconstrução dos dados informados pelo usuário através das informações que o teclado virtual passa para o navegador submeter para o servidor. A possibilidade de reconstrução dos dados a partir dessas informações

comprometeria completamente a segurança do mecanismo, tornando inútil a utilização do teclado virtual.

2.5. Identificação do Computador

O mecanismo de identificação do computador é um software que realiza a coleta de dados com o objetivo de caracterizar de forma única o equipamento de onde deve ser permitido o acesso ao serviço. As informações coletadas por esse tipo de software normalmente envolvem identificação de alguns dispositivos de hardware e o software. Como exemplos temos processador, memória, disco rígido, interface de rede e sistema operacional.

Esse mecanismo pode agregar bastante segurança ao acesso do serviço, porém são necessários cuidados para que outro computador não possa se passar por um equipamento autorizado. Outra dificuldade é que a coleta dessas informações no sistema requer comunicação com o sistema operacional. Esta característica torna necessário o desenvolvimento de software específico para cada plataforma.

O que ocorre na pratica é que é o mecanismo atende apenas as plataformas mais utilizadas. Dependendo do tratamento dado a essas plataformas que não são atendidas, pode haver o risco de um fraudador, de posse dos dados secretos de uma vítima, poder acessar o serviço a partir de uma plataforma não atendida pela solução sem a necessidade de identificação do computador.

2.6. Complemento de segurança para o navegador

Quase todos os *e-bankings* brasileiros incluem em seu pacote de segurança complementos para o navegador. A utilização deles é obrigatória para alguns dos serviços verificados, porém estão disponíveis apenas para o Internet Explorer e Mozila Firefox. Como exemplo dos mais utilizados temos a solução da Gás Tecnologia[3] (G-Buster) e a solução da Scopus Tecnologia[4] (SCPSEG).

Numa tentativa de coletar informações precisas sobre a finalidade e o funcionamento destes softwares, não foi encontrada nenhuma fonte segura. As informações contidas nos sites dos bancos a respeito desses mecanismos são superficiais, quando existem. Os fornecedores das soluções também não disponibilizam material relevante para uma análise mais aprofundada.

3. Exame de Caso

O Banco do Brasil possui um dos mais populares portais de acesso bancário via internet do país, com seis milhões de usuários [RSAConference 2008], dos quase trinta milhões no país [Febraban 2008]. Esta popularidade o torna alvo de um grande número de tentativas de fraude. Tal informação pode ser confirmada com a velocidade com que as pragas virtuais se adaptam às mudanças na forma como o seu serviço é disponibilizado [McAfeeSAGE 2008]. Estas informações ajudaram na escolha do Banco do Brasil como instituição para uma análise mais detalhada de alguns mecanismos de segurança adotados para combater fraudes no ambiente de internet.

[3]http://www.gastecnologia.com.br
[4]http://http://www.scopus.com.br

O objetivo da análise é identificar possíveis falhas nos mecanismos que autorizam o efetivo uso do serviço. O uso é considerado efetivo quando o usuário obtiver acesso completo ao portal e puder realizar transações.

Seis mecanismos de segurança foram identificados no serviço de acesso bancário via internet do Banco do Brasil, sendo os cinco a seguir de uso obrigatório: TLS; encerramento da sessão; senha de internet; identificação de computadores; senha de auto-atendimento.

Foram encontrados problemas que possibilitam a reconstrução da senha de internet, da senha de auto-atendimento e da identificação do computador. De posse dessas informações, o fraudador pode se passar pelo usuário fraudado, simular a utilização do computador cadastrado e realizar transações por meio do portal do banco.

Para interceptar os dados, é necessário apenas que o software malicioso tenha acesso a algumas informações ofuscadas que são submetidas nos formulários HTML do portal. O desenvolvimento do software malicioso não faz parte do escopo desse trabalho; para simular o seu funcionamento, foi utilizado o Firebug[5] complemento do navegador Mozila Firefox (Firefox) amplamente utilizado em desenvolvimento e teste de software para web. Vale ressaltar que essas vulnerabilidades não estão limitadas a serem exploradas no Firefox, pode ser desenvolvido um complemento malicioso para qualquer navegador e fazer com que o usuário seja levado a instalá-lo em seu computador, tornando-o vulnerável.

As falhas foram identificadas no teclado virtual e no identificador do computador, conforme descrição a seguir.

3.1. Teclado virtual

Figura 3. Teclado virtual do Banco do Brasil

O teclado virtual é utilizado no portal do Banco do Brasil no fornecimento da senha de internet, para acesso ao sistema de *e-banking*, e da senha de auto-atendimento, para confirmação de cada transação realizada. Ele consiste em um *applet*[6] onde o cliente entra com sua senha através de cliques nas imagens que representam números de 0 a 9 (Figura 3). Em seguida, o usuário clica no botão "entrar" para concluir a operação. Ao clicar no botão "entrar", a senha é ofuscada e o resultado é injetado no campo apropriado da página para que seja submetida para o servidor.

O método de ofuscação da senha é composto de três elementos:

[5]http://getfirebug.com/

[6]No contexto de Java, applets são aplicativos que se servem da JVM (*Java Virtual Machine*) existente na máquina cliente ou embutida no próprio navegador do cliente para interpretar o seu bytecode.

- Índice i que pode variar de 0 a 49;
- Lista L de 50 strings, com 8 caracteres cada, que podem variar de A a J;
- Conjunto C de 10 funções básicas.

As funções do conjunto C são compostas de uma combinação de operações de soma, subtração, multiplicação e divisão.

O algoritmo de ofuscação consiste em, dado um índice i definido pelo servidor e informado ao teclado virtual, i se refere a uma posição da lista L que consiste numa combinação de 8 caracteres de A a J. Essa combinação define a ordem com que as funções do conjunto C devem ser aplicadas sobre a senha. No caso da senha de internet, que é formada por 8 caracteres, 8 funções serão chamadas, uma para cada caractere. Já no caso da senha de auto-a tendimento que é formada por 6 caracteres, 6 funções serão chamadas, também uma para cada caractere;

O método de ofuscação não garante segurança à informação, pois é passivo de ser revertido com as informações disponíveis. Para isso, é necessário o índice i e o resultado do método de ofuscação, que podem ser obtidos no navegador.

Os dados necessários foram interceptados no navegador e revertidos à senha original. Foram implementadas as 10 funções inversas do conjunto C. O processo foi realizado tanto no acesso ao portal, com a senha de internet, como na confirmação das transações, com a senha de auto-atendimento.

O principal motivo para a utilização de teclados virtuais é tentar isolar o fornecimento da senha dos demais aplicativos operando na estação. A forma como a informação foi tratada antes de ser fornecida para o ambiente do cliente é completamente incoerente com esse objetivo, tornando ineficiente a utilização do teclado virtual.

O processo falha em dois pontos principais. Primeiro, porque tenta evitar sua reconstrução por obscuridade, o que não é uma boa prática [Schneier 2002b]. Segundo, porque é utilizada uma espécie de criptografia simétrica e a chave está disponível no navegador, possibilitando o roubo.

3.2. Identificador do computador

O identificador do computador é utilizado no portal do Banco do Brasil todas as vezes que se precisa identificar o computador do qual está sendo realizado o acesso. Essa identificação é necessariamente feita no acesso ao portal e também no cadastro de novos computadores. O software de identificação do computador consiste em um *applet* que tenta coletar informações da máquina operando de duas formas diferentes, de acordo com a arquitetura do processador e o sistema operacional.

No primeiro caso, onde o computador tem arquitetura $x86$[7] e é Windows ou Linux, o *applet* carrega dinamicamente uma biblioteca nativa específica para o sistema operacional. Com essa biblioteca, ele obtém as assinaturas do processador, da memória principal, do disco rígido e da placa de rede. É gerado um hash MD5 [Rivest 1992] com uma combinação das informações coletadas que identifica o computador. Antes de copiar essa assinatura para o navegador, ela passa por um processo de criptografia simétrica (3DES-EDE-CBC) [Standard 1977] utilizando uma derivação da chave informada pelo servidor.

[7] $x86$ ou $80x86$ é o nome genérico dado à família de processadores baseados no Intel 8086, da Intel Corporation.

Essa derivação inicia com o hash MD5 de 16 bytes da chave original, que resultará em três chaves para as três chamadas do DES. A primeira chave são os primeiros 8 bytes do MD5; a segunda chave da criptografia são os últimos 8 bytes do MD5 e a terceira chave da criptografia são os bytes pares do MD5. Já que a chave e a assinatura criptografada estão disponíveis no navegador, descriptografar a assinatura e recuperar a identificação do computador é trivial.

No segundo caso, onde existe uma mudança na arquitetura do computador ou nos sistemas operacionais previstos no primeiro caso, o processo é bem mais simples. Como o *applet* não conta com opções de código nativo para esses casos, ele simplesmente faz um *XOR*[8] da chave informada pelo servidor com uma *string* padrão. Como estão disponíveis tanto a chave como o resultado do *XOR* no navegador, a aplicação do mesmo método sobre o resultado da primeira chamada e a chave resulta na *string* padrão que identifica o computador.

Em ambos os casos, os dados reais foram interceptados no navegador e revertidos aos dados originais, obtendo sucesso na recuperação das assinaturas dos computadores.

O processo falha nos mesmos dois pontos que a implementação do teclado virtual, segurança por obscuridade e criptografia simétrica com a chave disponível no navegador.

4. Impacto das vulnerabilidades

As vulnerabilidades encontradas no serviço de *e-banking* do Banco do Brasil acarretam no total comprometimento da segurança. O impacto aumenta pelo fato da senha de auto-atendimento ser revelada, o que compromete outros canais de atendimento.

Pela natureza das falhas encontradas, também pode ser levado em consideração o impacto à imagem da instituição perante seus clientes, que depositam confiança nas soluções de segurança oferecidas pela instituição.

5. Conclusão

Este trabalho apresentou os mecanismos de segurança atualmente empregados pelos *e-bankings* brasileiros e vulnerabilidades encontradas no serviço oferecido pelo Banco do Brasil.

Ficou evidente a necessidade de um estudo mais aprofundado sobre as ferramentas disponíveis e a elaboração de mecanismos de segurança mais robustos que atendam aos requisitos impostos pela natureza crítica dos sistemas de *e-bankings*.

Ações mais amplas podem ser pensadas, como por exemplo, a elaboração de um conselho que avalie e dê consultoria sobre os mecanismos de segurança adotados pelas instituições financeiras para as transações via internet.

Agradecimentos

O desenvolvimento deste trabalho contou com ajuda de algumas pessoas, onde destaco família, orientador, co-orientador e amigos. Também dedico sincero agradecimento a Cláudio Matsuoka [Matsuoka 2008] que se mostrou bastante disponível no repasse do seu

[8] *XOR* é uma operação lógica de dois operandos que resulta em um valor lógico verdadeiro se, e somente se, exatamente um dos operandos tem valor verdadeiro.

estudo realizado sobre o processo de identificação de computadores adotado pelo Banco do Brasil, de fundamental importância para a conclusão desse trabalho.

Referências

Dierks, T. and Allen, C. (1999). The tls protocol version 1.0. RFC 2246 (Proposed Standard). Obsoleted by RFC 4346, updated by RFC 3546.

dos Santos, A., Vigna, G., and Kemmerer, R. (2001). Security Testing of an Online Banking Service. *E-Commerce Security and Privacy, Advances in Information Security*, pages 3–15.

Febraban (2008). Setor bancário em números. *Em http://www.febraban.org.br*.

Litterio, F. (1995). Why are one-time pads perfectly secure? *Em http://world.std.com/ franl/crypto/one-time-pad.html*.

Matsuoka, C. (2008). Análise: Autenticação linux. *Em http://helllabs.org/blog/20080226/analise-autenticacao-linux/*.

McAfeeSAGE (2008). Sage - uma internet muitos mundos. *Em http://www.mcafee.com/us/local_content/reports/sage_pt_br_2008.pdf*.

Rescorla, E. (2000). Http over tls. RFC 2818 (Informational).

Rivest, R. (1992). The md5 message-digest algorithm. RFC 1321 (Informational).

RSAConference (2008). Internet banking case study: Banco do brasil. *Em http://www.bankinfosecurity.com/articles.php?art_id=814*.

Schneier, B. (2002a). One-time pads. *Cryptogram, Oct*.

Schneier, B. (2002b). Secrecy, security, and obscurity. *Cryptogram, May*.

Stallings, W. (2007). *Network Security Essentials: Applications and Standards*. Prentice Hall.

Standard, D. (1977). Federal Information Processing Standards Publication 46. *National Bureau of Standards, US Department of Commerce*.

von Ahn, L., Blum, M., and Langford, J. (2004). Telling humans and computers apart automatically. *Communications of the ACM*, 47(2):56–60.