Uma análise dos certificados digitais utilizados nas conexões TLS dos aplicativos de Mobile Banking na plataforma Android

Diego Baierle Sebastiany[1], **Mirelle Daiara Vieira Freitas**[1], **Luciano Ignaczak**[1]

[1] Universidade do Vale do Rio dos Sinos (UNISINOS)
CEP 93.022-000 – São Leopoldo – RS – Brasil

diego.sebastiany@hotmail.com, mdfreitass@outlook.com, lignaczak@unisinos.br

Abstract. *Currently, it is increasingly common to use mobile banking applications on smartphones. Such applications must implement TLS to use encryption to secure communication between the client and the bank. However, often the applications have developmental problems that compromise their safety. This article analyzes whether the digital certificates used by banks from three countries in TLS connections on m-banking applications are recognized as trusted by Android. Furthermore, the size of keys and the validity of these digital certificates is discussed.*

Resumo. *Atualmente, é cada vez mais comum a utilização de aplicativos de mobile banking em smartphones. Tais aplicativos devem implementar o protocolo TLS para empregar criptografia para proteger a comunicação entre o cliente e o seu banco. No entanto, muitas vezes os aplicativos apresentam problemas de desenvolvimento que comprometem a sua segurança. Este artigo analisa se os certificados digitais usados por bancos de três países nas conexões TLS com as aplicações de m-banking são reconhecidos como confiáveis pelo Android. Além disso, é discutido o tamanho das chaves e o período de validade desses certificados digitais.*

1. Introdução

A crescente popularização da *internet* tem levado a um aumento expressivo da quantidade de dispositivos conectados. Com isso, cresceu também a quantidade de usuários que utilizam aplicativos para gerenciamento e movimentação financeira de suas contas bancárias. Uma pesquisa mostra que 52% das transações bancárias feitas no Brasil em 2014 foram realizadas via *internet* e *mobile banking* (m-banking). Entre as contas ativas no país em 2014, 24% dos clientes (25 milhões) realizaram transações utilizando m-banking em seus *smartphones* [Febraban 2015].

Os aplicativos de m-banking precisam implementar mecanismos de segurança para garantir que os dados do usuário não fiquem vulneráveis a roubo e interceptação na Internet. O protocolo TLS (*Transport Layer Security*) é utilizado como padrão para fornecer a segurança nesse ambiente [Elkhodr et al. 2012]. Além de garantir o sigilo e a integridade na comunicação, o protocolo TLS autentica o servidor do banco no qual o aplicativo está se conectando. A autenticação é importante e necessária

para confirmar que o computador que está respondendo é realmente a entidade que afirma ser [Stallings 2008] [Adams and Lloyd 2003].

O estabelecimento de uma comunicação segura exige que a autenticação do banco seja feita de forma correta. Ao fazer o *handshake* do protocolo TLS, o banco envia o seu certificado digital para que o cliente (o aplicativo) verifique a sua identidade. Para isso, o aplicativo deve utilizar um dos certificados raiz instalados no sistema Android para fazer a validação da confiança do certificado do banco. A tentativa de conexão deveria falhar se o certificado raiz usado pelo banco não for considerado confiável pelo Android. No entanto, muitas vezes os aplicativos falham ou simplesmente não executam a validação do certificado digital [Six 2012].

Além da validação da confiança, o tamanho da chave criptográfica e o período de validade de um certificado digital são muito importantes e devem ser considerados no momento de sua emissão para conferir-lhe resistência contra ataques de força bruta. As recomendações mais recentes sugerem um tamanho de chave mínimo de 2.048 bits [Barker and Roginsky 2011]. Além disso, a Microsoft recomenda que certificados digitais com tamanho da chave de 1.024 bits devem possuir até 1 ano de validade; certificados digitais com tamanho de chave de 2.048 bits devem possuir no máximo 2 anos de validade; e certificados digitais com tamanho de chave de 4.096 bits podem possuir validade de até 16 anos [OMeally 2009].

O objetivo deste trabalho é analisar a confiança dos certificados digitais utilizados por bancos nas conexões TLS com os aplicativos de m-banking na plataforma Android, além do período de validade e o tamanho da chave criptográfica desses certificados. A análise foi realizada a partir de uma amostra de 60 aplicativos de m-banking disponibilizados por bancos de três países: Brasil, Estados Unidos e Reino Unido. Para cada aplicativo foi realizada uma simulação de acesso à conta bancária e, a partir do tráfego capturado, foi verificada a utilização do TLS e obtidos os certificados digitais utilizados na conexão.

O restante deste trabalho segue com a seção 2 que apresenta alguns trabalhos relacionados. A seção 3 descreve a metodologia utilizada na realização desta análise, a seção 4 mostra os resultados obtidos da análise dos aplicativos de m-banking e a seção 5 expõe as considerações finais deste trabalho.

2. Trabalhos Relacionados

Muitos aplicativos vêm apresentando problemas de implementação, os quais têm motivado muitos trabalhos que discutem suas causas e possíveis soluções. O trabalho de [Georgiev et al. 2012] mostra que a segurança oferecida pelo TLS depende da correta validação do certificado digital fornecido quando a conexão é estabelecida. Esse trabalho analisa como alguns *softwares* e aplicativos implementam as funções do TLS para validação dos certificados digitais e mostra que mesmo os aplicativos desenvolvidos por grandes empresas possuem falhas graves. Muitas vezes, segundo os autores, as falhas na validação de um certificado é causada pela falta de entendimento e interpretação das APIs (*Application Programming Interface*) utilizadas pelos desenvolvedores. A falta de conhecimento e informação sobre essas APIs conduz o desenvolvedor ao erro e deixa o aplicativo vulnerável a ataques do homem do meio. O trabalho de [Hubbard et al. 2014] realizou uma pesquisa com o objetivo de

identificar falhas na validação dos certificados digitais. Com uma pequena amostra de 41 aplicativos para a plataforma Android, 11 falharam em estabelecer a relação de confiança necessária, pois aceitaram um certificado digital falsificado que, portanto, não pertencia à base de confiança do Android. O artigo também destaca que a falha dos aplicativos pode estar relacionada às APIs utilizadas. Por serem pouco restritivas, permitem que os desenvolvedores cometam erros de implementação do código, permitindo que qualquer certificado digital seja aceito pelo aplicativo ou, até mesmo, que nenhuma validação seja realizada.

A inconsistência da base de certificados raiz da plataforma Android também já foi alvo de estudo. [Vallina-Rodriguez et al. 2014] examinou os certificados raiz instalados nas diversas versões do Android em vários dispositivos. O trabalho analisou a composição dessas bases de confiança e como elas variam de acordo com a versão e marca do dispositivo. Como resultado, foi verificado que a base oficial de certificados digitais confiados pelo Android é modificada ou ampliada. Em alguns casos, o próprio fabricante do dispositivo e/ou a operadora de telefonia adicionam certificados digitais aos dispositivos para estabelecer a relação de confiança para aplicativos embarcados ou prestação de serviços. O trabalho também alerta para o fato que, em dispositivos que rodam com usuário *root*, aplicativos maliciosos podem instalar certificados digitais no Android sem o conhecimento do usuário, quebrando o modelo de confiança de certificados digitais supervisionados e auditados como confiáveis.

[Fahl et al. 2012] investigou o uso inadequado do TLS em 13.500 aplicativos de diversas categorias, obtidos da Google Play Market. Dos aplicativos analisados, 1.074 (17,28% dos que utilizam TLS) continham erros de código do TLS que permitiam a validação de qualquer certificado digital ou confiavam em qualquer certificado raiz. O autor mostra também que as falhas na implementação do TLS ocorrem porque o Android permite que os desenvolvedores criem códigos personalizados para seus aplicativos. Ele destaca que esse recurso devia ser desativado e que as APIs para Android deviam forçar a utilização das implementações padrão do TLS. Em outro trabalho, [Fahl et al. 2013] continua investigando as possíveis causas da má implementação do TLS em aplicativos. Os resultados da pesquisa mostram que as causas não são simplesmente a falta de cuidado por parte dos desenvolvedores, mas também questões e limitações envolvendo o atual paradigma de desenvolvimento do TLS. O trabalho sugere mudanças no atual paradigma em direção a uma maior abstração do código fornecido pelas APIs, permitindo que desenvolvedores utilizem corretamente o TLS com menos esforço e prevenindo falhas na validação dos certificados digitais.

Os trabalhos relacionados reforçam a necessidade de um maior cuidado na implementação do TLS em aplicativos que transmitem dados confidenciais. As falhas de implementação em aplicativos de m-banking podem acarretar muitos prejuízos para o cliente e para o banco. Os artigos citados nesta seção realizaram análises dos certificados digitais de diversos aplicativos, sem abordar um segmento específico. Já este artigo, analisou especificamente como aplicativos de m-banking estão validando os certificados digitais dos bancos.

3. Metodologia

Para a realização desta análise foi selecionada uma amostra com 60 aplicativos de m-banking divididos igualmente em três países: Brasil, Estados Unidos (EUA) e o Reino Unido (UK). A seleção dos aplicativos foi realizada utilizando *rankings* do Banco Central do Brasil[1], do *Federal Reserve System*[2] para os EUA, e do Relbanks[3] para o UK, que classificam os bancos com maiores ativos em cada país. Baseados nestes *rankings*, os autores selecionaram os 20 primeiros bancos de varejo que possuem aplicativos de m-banking. A lista da Relbanks possui apenas 11 bancos e foi utilizada porque não foi encontrado um *ranking* oficial do Banco Central do Reino Unido. A amostra de aplicativos do país foi incrementada com mais 10 bancos conhecidos do Reino Unido, retirados do site do seu Banco Central [4]. A análise consistiu na avaliação das seguintes características de cada aplicativo:

- se o aplicativo utiliza o protocolo TLS para comunicação segura;
- a verificação da confiança no certificado raiz do aplicativo;
- o período de validade do certificado digital do aplicativo;
- o tamanho da chave do certificado digital do aplicativo;

O *software* Genymotion[5] foi usado para emular um dispositivo rodando a versão 4.4 do sistema Android, que está instalada atualmente em 39,3% dos dispositivos dessa plataforma [Android 2015]. Desse dispositivo foram extraído todos os certificados digitais armazenados em `/system/etc/security/cacerts/`. Esses são os certificados digitais das autoridades de certificação confiadas por esta versão do Android. Os certificados digitais extraídos foram armazenados para, posteriormente, analisar a confiança dos certificados raiz utilizados nas conexões TLS pelos aplicativos de m-banking. Para possibilitar a análise, os aplicativos de m-banking selecionados foram instalados no dispositivo virtual. Além disso, para que fosse possível a captura do tráfego TLS gerado pelo aplicativo de m-banking foi utilizado o tcpdump, disponível no emulador.

Após a instalação de cada aplicativo de m-banking, foram realizadas tentativas de acesso à conta bancária. O acesso foi simulado pela inserção dos dados necessários (como número da conta e senha) aceitos pelo aplicativo, para que ele iniciasse a comunicação com o banco, e assim, estabelecesse a conexão segura (TLS). Com o tráfego gerado pala simulação do acesso foi avaliado o primeiro critério desta análise: se o aplicativo utiliza o TLS.

No caso dos aplicativos de m-banking que possibilitaram a verificação da implementação do protocolo TLS com a captura do tráfego foram extraídos os certificados digitais utilizados pelo *handshake*: o certificado do banco e o certificado raiz, que é utilizado para avaliar a relação de confiança entre o banco e o sistema Android. A partir do certificado digital do banco foi avaliado o tamanho da chave criptográfica bem como o seu período de validade. Para auxiliar na consolidação dos resultados dessa análise foi utilizado um *software* desenvolvido pelos autores,

[1]Disponível em: `http://www4.bcb.gov.br/top50/port/top50.asp`
[2]Disponível em: `http://www.federalreserve.gov/Releases/Lbr/current/default.htm`
[3]Disponível em: `http://www.relbanks.com/europe/uk`
[4]Disponível em: `http://www.bankofengland.co.uk`
[5]Disponível em: `https://www.genymotion.com`

na linguagem C#, que coleta os dados dos certificados e exporta os resultados no formato XML. O arquivo exportado foi utilizado como fonte para a construção de uma planilha.

Um segundo *software* na linguagem C# também necessitou ser desenvolvido pelos autores para analisar a confiança dos certificados raiz capturados. Esse *software* realizou o cruzamento entre os certificados raiz capturados e a base de certificados raiz considerada confiável pela versão avaliada do Android. O cruzamento desses certificados digitais consistiu em comparar os campos *Subject Key Identifier*, ou na ausência deste, a própria chave pública contida no campo *Subject Public Key Info*. A saída desse programa foi salva e adicionada à planilha anterior, usada como base para a avaliação dos resultados.

Não foi possível avaliar alguns aplicativos de m-banking pois a captura do tráfego desses aplicativos no momento da autenticação não apresenta a utilização do TLS, tampouco revela os dados do usuário em texto claro. Isso pode acontecer quando o aplicativo implementa os requisitos de segurança na camada de aplicação. Por isso, não é possível afirmar que o aplicativo falha em oferecer segurança para o usuário. Os aplicativos com essas características foram classificados como indefinidos.

A última etapa desse trabalho consistiu na realização da análise dos resultados obtidos. Nesta etapa foram efetuados cálculos de porcentagem, cruzamento de informações e médias, a fim de comparar as definições dos bancos dos três países em relação aos dados dos certificados digitais que são alvo deste artigo.

4. Resultados

A análise dos 60 aplicativos selecionados resultou em 2 aplicativos, ambos do Brasil, classificados como indefinidos, e 58 aplicativos que utilizaram o TLS para estabelecer a conexão segura.

Não foi possível verificar a utilização do TLS ao analisar a captura do tráfego gerado pelos 2 aplicativos que foram classificados como indefinidos. Embora não seja possível afirmar, o mecanismo de segurança utilizado por esses aplicativos pode ser o próprio TLS, mas implementado de forma personalizada pelos desenvolvedores. Isso é possível porque as APIs utilizadas para o Android permitem esse nível de personalização do código.

O resultado mostrou que os outros 58 aplicativos analisados utilizam o TLS, realizando o *handshake* e apresentando o certificado digital do banco como é padrão do protocolo. Porém, 18 (31%) desses aplicativos não poderiam ser considerados confiáveis, pois esses utilizam certificados digitais emitidos por autoridades de certificação que não são confiadas pelo Android. O resultado dessa verificação é apresentado na Tabela 1.

A análise da relação de confiança mostrou que o cenário mais preocupante é o brasileiro, onde 44% dos aplicativos analisados não podem ser considerados confiáveis pela versão da plataforma Android analisada. O Reino Unido apresentou o menor número de certificados digitais não confiáveis (15%), porém, ainda é preocupante considerando que o segmento analisado é o bancário, que deveria possuir um cuidado adicional no uso de certificados digitais.

Tabela 1. Certificados raiz sem relação de confiança com o Android.

Origem	Total de certificados raiz não confiáveis	Percentual de certificados raiz não confiáveis
Brasil	8	44,44%
Estados Unidos	7	35,00%
Reino Unido	3	15,00%
TOTAL	18	31,03%

Como foi mostrado pelos trabalhos relacionados, as falhas de validação da confiança expõem o cliente a diversos riscos, e são resultado da forma de implementação do código do aplicativo. Semelhante às análises nesses trabalhos, esta análise dos aplicativos de m-banking revelou um cenário inquietante, pois nenhum dos aplicativos que utilizam certificados não confiados pela plataforma Android apresentou qualquer mensagem de alerta durante o *handshake* do TLS.

A segunda parte desta análise, avaliou o período de validade e o tamanho da chave criptográfica do certificado digital do banco. Todos os 58 certificados digitais possuem o tamanho da chave igual a 2.048 bits com períodos de validade distintos. Os períodos de validade dos certificados digitais usados pelos aplicativos de m-banking são apresentados na Tabela 2.

Tabela 2. Período de validade dos certificados digitais dos bancos.

Origem	Total de certificados	Período de validade			
		1 ano	2 anos	3 anos	4 anos
Brasil	18	10	8	0	0
Estados Unidos	20	14	3	1	2
Reino Unido	20	12	8	0	0

Embora todos os certificados digitais dos bancos analisados atendam à recomendação do NIST no que diz respeito ao tamanho da chave [Barker and Roginsky 2011], 3 deles, todos dos EUA, possuem o período de validade superior a 2 anos. Conforme a recomendação da Microsoft, o período máximo de validade deve ser de 2 anos para certificados com tamanhos de chave de 2.048 bits. Um período de validade muito grande diminui a resistência da chave associada ao certificado digital, pois os avanços da tecnologia de computação podem comprometer um certificado digital que, para os padrões atuais, é considerado forte.

5. Considerações Finais

Quando o usuário instala e utiliza um aplicativo em seu *smartphone*, ele o faz confiando que a comunicação e seus dados estarão seguros. Quando se trata do segmento de m-banking, espera-se que todos os aplicativos implementem o TLS para atender os requisitos de segurança e proteger o usuário. Ao usuário resta apenas confiar no aplicativo, pois o sistema Android não oferece nenhuma indicação de que a comunicação é estabelecida de forma segura.

Existem normas e recomendações que os desenvolvedores de aplicativos devem seguir para atender requisitos no desenvolvimento de seus aplicativos e evitar erros comuns ao utilizar códigos personalizados. O segmento de m-banking deve observar especialmente as recomendações de segurança como a do NIST [Barker and Roginsky 2011] que especifica o tamanho mínimo da chave do certificado digital em 2.048 bits. Como foi mostrado nos resultados deste trabalho, todos os aplicativos seguiram essa recomendação pois todos possuem tamanho da chave igual a 2.048 bits. No entanto, 3 desses certificados digitais possuem o período de validade maior que 2 anos, em desacordo com a recomendação da Microsoft [OMeally 2009]. Isso pode resultar em uma falha, pois os avanços da tecnologia de computação poderão permitir a quebra de chaves com tamanho de 2.048 bits durante o período de validade do certificado digital.

Ademais, uma parcela significativa dos aplicativos, considerando-se sistemas de m-banking, falham na implementação da validação do certificado digital porque a relação de confiança que deveria existir entre o certificado raiz do banco e o sistema Android não é estabelecida. Sem a validação da confiança um certificado digital é aceito sem qualquer restrição, quebrando completamente o sistema de certificação digital, supervisionado e auditado como confiável. Esse problema mostra-se ainda mais grave quando considerado que isso ocorre de forma transparente para o usuário. Embora o protocolo TLS forneça o recurso para avisar o usuário que a relação de confiança não foi estabelecida, muitas vezes esse recurso é desativado ou mau implementado pelo desenvolvedor. Neste trabalho, dos 18 aplicativos que falharam ao estabelecer a relação de confiança, nenhum mostrou qualquer mensagem de aviso sobre essa falha, e todos prosseguiram funcionando como se nenhum erro tivesse ocorrido.

Este trabalho analisou uma amostra reduzida de certificados digitais utilizados por aplicativos de m-banking durante a conexão TLS. Como trabalho futuro é sugerido a análise de uma amostra mais ampla que reflita com mais precisão a realidade no segmento dos aplicativos de m-banking. Além disso, trabalhos futuros podem comparar as características de certificados digitais usados no TLS por aplicativos de outros segmentos.

Referências

[Adams and Lloyd 2003] Adams, C. and Lloyd, S. (2003). *Understanding PKI: Concepts, Standards, and Deployment Considerations*. Addison-Wesley Pearson Education, Boston, MA, second edition.

[Android 2015] Android, D. (2015). Dashboards, platform versions. Disponível em: `https://developer.android.com/about/dashboards/index.html`.

[Barker and Roginsky 2011] Barker, E. and Roginsky, A. (2011). Transitions: Recommendation for transitioning the use of cryptographic algorithms and key lengths. NIST Special Publication 800-131A. Disponível em: `http://csrc.nist.gov/publications/nistpubs/800-131A/sp800-131A.pdf`.

[Elkhodr et al. 2012] Elkhodr, M., Shahrestani, S., and Kourouche, K. (2012). A proposal to improve the security of mobile banking applications. In *ICT and*

Knowledge Engineering (ICT Knowledge Engineering), 2012 10th International Conference on, pages 260–265.

[Fahl et al. 2012] Fahl, S., Harbach, M., Muders, T., Baumgärtner, L., Freisleben, B., and Smith, M. (2012). Why eve and mallory love android: An analysis of android ssl (in)security. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, CCS '12, pages 50–61, New York, NY, USA. ACM.

[Fahl et al. 2013] Fahl, S., Harbach, M., Perl, H., Koetter, M., and Smith, M. (2013). Rethinking ssl development in an appified world. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, CCS '13, pages 49–60, New York, NY, USA. ACM.

[Febraban 2015] Febraban (2015). Pesquisa febraban de tecnologia bancária 2014. Disponível em: `https://www.febraban.org.br/Noticias1.asp?id_texto=2626`.

[Georgiev et al. 2012] Georgiev, M., Iyengar, S., Jana, S., Anubhai, R., Boneh, D., and Shmatikov, V. (2012). The most dangerous code in the world: Validating ssl certificates in non-browser software. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, CCS '12, pages 38–49, New York, NY, USA. ACM.

[Hubbard et al. 2014] Hubbard, J., Weimer, K., and Chen, Y. (2014). A study of ssl proxy attacks on android and ios mobile applications. In *Consumer Communications and Networking Conference (CCNC), 2014 IEEE 11th*, pages 86–91.

[OMeally 2009] OMeally, Y. (2009). Recommendations for pki key lengths and validity periods with configuration manager. Disponível em: `http://blogs.technet.com/b/configmgrteam/archive/2009/06/12/recommendations-for-pki-key-lengths-and-validity-periods-with-configuration-manager.aspx`.

[Six 2012] Six, J. (2012). *Segurança de aplicativos Android*. Novatec Editora Ltda., São Paulo, SP.

[Stallings 2008] Stallings, W. (2008). *Criptografia e segurança de redes*. Pearson Education do Brasil Ltda., São Paulo, SP, fourth edition.

[Vallina-Rodriguez et al. 2014] Vallina-Rodriguez, N., Amann, J., Kreibich, C., Weaver, N., and Paxson, V. (2014). A tangled mass: The android root certificate stores. In *Proceedings of the 10th ACM International on Conference on Emerging Networking Experiments and Technologies*, CoNEXT '14, pages 141–148, New York, NY, USA. ACM.