EDUARDO DE SOUZA CRUZ
GEOVANDRO CARLOS CREPALDI FIRMINO PEREIRA
RODRIGO RODRIGUES DA SILVA

APLICATIVO SEGURO DE SMS

Texto apresentado à Escola Politécnica da Universidade de São Paulo como requisito para a conclusão do curso de graduação em Engenharia de Computação, junto ao Departamento de Engenharia de Computação e Sistemas Digitais (PCS).

São Paulo
2008

EDUARDO DE SOUZA CRUZ
GEOVANDRO CARLOS CREPALDI FIRMINO PEREIRA
RODRIGO RODRIGUES DA SILVA

APLICATIVO SEGURO DE SMS

Texto apresentado à Escola Politécnica da Universidade de São Paulo como requisito para a conclusão do curso de graduação em Engenharia de Computação, junto ao Departamento de Engenharia de Computação e Sistemas Digitais (PCS).

Área de Concentração:

Engenharia da Computação

Orientador:

Prof. Dr. Paulo Sérgio Licciardi Messeder Barreto

São Paulo
2008

AGRADECIMENTOS

Agradeço a... etc etc etc

RESUMO

Este trabalho consiste na especificação e implementação de um sistema que garanta serviços de segurança na troca de mensagens SMS entre aparelhos de telefonia celular. Ao longo da monografia, apresentaremos aspectos da nossa solução, um esquema criptográfico inovador que viabilizou a implementação, métricas, resultados de testes de desempenho, além de considerações sobre o andamento do trabalho.

ABSTRACT

This work consists...

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

1 INTRODUÇÃO

Atualmente não existem soluções universalmente adotadas para garantir segurança em mensagens SMS. Pelo método tradicional, as mensagens trafegam pela rede celular de forma insegura, passando obrigatoriamente por pelo menos um intermediário não 100% confiável: a operadora do serviço de telefonia. As mensagens podem ficar armazenadas em texto plano no banco de dados da operadora (NG, 2006), de forma que pessoas mal intencionadas infiltradas no sistema podem ser capazes de visualizar, alterar e até enviar mensagens em nome de outra pessoa. Há também outros métodos para interceptar mensagens SMS. (ENCK et al., 2005).

O sistema desenvolvido foi designado por "Aplicativo Seguro de SMS", ou, em uma forma abreviada e internacionalizada mais adequada ao mercado, "Secure-SMS", ou ainda "SSMS". O objetivo do sistema é prover uma camada de segurança a nível de aplicação para mensagens SMS em redes de telefonia móvel. O software é capaz de assinar, cifrar, decifrar e verificar mensagens enviadas por SMS, de forma a garantir a identidade do remetente, e garantir que ela somente poderá ser lida pelo destinatário real, oferencendo assim os serviços de segurança: autenticidade, confidencialidade e integridade.

Apesar de implementado especificamente para ser executado em um telefone celular, garantindo a comunicação pessoal sigilosa, o mesmo esquema pode ser facilmente portado para outras plataformas que têm acesso à comu-

nicação por SMS, como PDAs ou servidores. Desta forma, o esquema pode ser aplicado em áreas que requerem alto nível de segurança da informação que trafega nas redes de telefonia móvel, por exemplo aplicações militares, bancárias, e de comércio eletrônico.

1.1 Cenário

O ambiente em questão não se mostra muito propício para práticas criptográficas. A largura de banda é muito pequena, visto que em cada mensagem SMS podem ser trafegados apenas 140 bytes binários. Além disto, existem limitações de processamento no dispositivo celular, que podem comprometer a usabilidade de um esquema criptográfico tradicional.

Talvez devido a estas dificuldades, o cenário atual não apresenta uma grande variedade de soluções abrangindo objetivos similares aos de nosso sistema, e não há uma solução universalmente adotada.

2 OBJETIVOS

O objetivo do trabalho é quebrar o paradigma atual, desafiar as dificuldades existentes, e produzir um sistema que garante segurança no intercâmbio de mensagens SMS entre dispositivos celulares sem afetar a usabilidade do serviço. Além disto, o nível de segurança obtido deve ser equiparável ao nível de segurança de aplicações de segurança vigentes atualmente, como por exemplo uma transação bancária pela Internet pelo protocolo HTTPS.

2.1 Escopo

O que nosso sistema garante é....

Problemas como... etc etc etc... não são de nossa responsabilidade... etc etc etc...

2.2 Métricas

A seguir, definimos as métricas desejadas.

- Nível de segurança: Desejamos que o nível de segurança de nosso protocolo seja equiparável ao nível de segurança do RSA usando chaves de 1024 bits, que pode ser calculado como 2 a 1024.
- Tempo de espera: Consiste nos tempos para cifrassinar e vericifrar uma

mensagem. Baseando-se em aplicações já existentes e satisfazendo os requisitos de usabilidade de nosso projeto, estimamos que um intervalo de espera para processamento de uma mensagem de no máximo 5 segundos seja tolerável pelo usuário.

- Tamanho máximo de mensagens do protocolo: Consiste da soma dos bytes úteis da mensagem com os bytes de controle do algoritmo. Implementações SMS baseadas em *Sun Wireless Messaging API (WMA)* podem dividir uma única mensagem em, no máximo, 3 segmentos, totalizando 399 bytes binários (ORTIZ, 2002). Porém para evitar problemas devido à segmentação, decidimos que as mensagens do nosso protocolo deverão caber em apenas 1 segmento, o que nos dá um tamanho de 140 bytes binários para cada mensagem do protocolo.

- Tamanho das chaves privada/pública: Devido às limitações de banda, estabeleceu-se que cada o tamanho chave usada não deverá exceder 200 bits. No entanto, essa restrição não deve comprometer o nível de segurança desejado.

- Tamanho máximo de uma mensagem de texto a ser cifrassinada e enviada: Uma mensagem cifrassinada deverá caber em uma única mensagem do protocolo, cujo tamanho máixmo foi defindo em 140 bytes. Porém nem todos os bytes poderão ser usados para a mensagem, pois existirá um overhead do protocolo, devido à cabeçalhos e dados da assinatura. Estabelecemos então um tamanho máximo de 70 bytes para uma mensagem de texto. Desta forma, ficam reservados outros 70 bytes para o overhead e a assinatura.

- Tamanho do certificado: Devido às limitações de banda, estabeleceu-se que o tamanho do certificado de uma chave não deverá exceder 512

bits. Desejamos poder transferir o certificado em um único SMS, sem comprometer o espaço necessário para o overhead do protocolo. [1]

[1]Mais tarde o leitor verá que optamos por um esquema que não usa certificados, ou seja, esta métrica será atendida com 0 bits de tamanho de certificado

3 DISCUSSÃO

Sabendo que um certificado digital típico ocupa entre 2KB e 4KB, nota-se aqui que uma solução baseada em infra-estrutura convencional de chaves públicas inviabilizaria completamente o sistema: antes de se enviar uma mensagem SMS segura para algum usuário, seria necessário receber o certificado desse usuário particionado em 15 a 30 mensagens SMS, além de enviar em resposta outro certificado em mais 15 a 30 mensagens SMS. Esse esforço precisaria ser efetuado novamente para cada novo destinatário a quem determinado usuário desejasse enviar mensagens, ou em cada caso de renovação ou revogação de certificado. Some-se a isto o espaço ocupado por uma única assinatura convencional, tipicamente de 128 bytes por estar baseada no algoritmo RSA com 1024 bits; este *overhead* seria duplicado com o requisito de cifrar e assinar a mensagem, isto é, tomaria 256 bytes do espaço disponível.

Por outro lado, a manutenção de um diretório confiável de chaves públicas, típico de sistemas de criptografia convencionais, seria impraticável em uma rede de telefonia celular. Uma solução tecnológica baseada em alternativas à criptografia convencional é, portanto, imprescindível.

Sendo assim, foi considerado o uso de criptografia em curvas elípticas com assinatura baseada em identidades, de acordo com o conceito proposto inicialmente por Shamir (SHAMIR, 1984). Aprofundando-se na especificação, percebeu-se ainda que a chave pública do usuário poderia ser estabelecida

essencialmente a partir de sua identificação única no sistema, ou seja, seu próprio número de celular. Desse modo, a criptografia em curvas elípticas baseada em identidades com emparelhamentos bilineares parecia ser capaz de atender aos requisitos do nosso aplicativo.

4 PRELIMINARES TEÓRICAS

4.1 Curvas Elítpticas

4.2 Emparelhamento

4.3 Criptografia baseada em identidades

4.4 Alternativas existentes

4.5 Revisão literatura

5 ESCOLHA DE UM ESQUEMA CRIPTOGRÁFICO ADEQUADO

5.1 BLMQ

5.1.1 Definições

5.1.2 Testes de viabilidade

5.2 BDCPS

5.2.1 Definições

5.2.2 Testes de viabilidade

5.3 Conclusão

6 ESPECIFICAÇÃO DO SISTEMA

6.1 Requisitos funcionais

6.2 Requisitos não-funcionais

6.3 Arquitetura

6.4 Classes

6.5 Casos de uso

6.6 Especificação do protocolo

7 IMPLEMENTAÇÃO

7.1 Metodologia

7.2 Ambiente de desenvolvimento

7.3 Bibliotecas usadas

7.4 Problemas encontrados

8 RESULTADOS

8.1 Desempenho

9 CONCLUSÃO

9.1 Análise dos resultados

9.2 Possíveis desenvolvimentos futuros

REFERÊNCIAS

ENCK, W. et al. Exploiting open functionality in sms-capable cellular networks. In: *Proceedings of the 12th ACM conference on Computer and communications security*. New York, NY, USA: ACM Press, 2005. p. 393–404.

NG, Y. L. *Short Message Service (SMS) Security Solution for Mobile Devices*. Monterey, California, USA: [s.n.], 2006. 17–19 p.

ORTIZ, C. *The Wireless Messaging API*. December 2002. Sun Developer Network (SDN) article. `http://developers.sun.com/mobility/midp/articles/wma/index.html`.

SHAMIR, A. Identity based cryptosystems and signature schemes. In: *Advances in Cryptology – Crypto'84*. [S.l.]: Springer, 1984. (Lecture Notes in Computer Science, v. 0196), p. 47–53.

APÊNDICE A – APÊNDICE