Segurança Informática em Redes e Sistemas

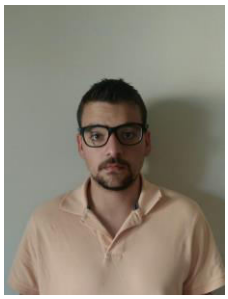
Pagamentos Seguros Via SMS

Mestrado em Engenharia Informática e de Computadores

Grupo T60

João Freitas Nº 81950

Diogo Cardoso Nº 94024



1 - Problema

A aplicação de pagamentos seguros via SMS é uma aplicação que executa uma transação bancária de forma segura usando a tecnologia SMS.

Para o domínio deste problema, a segurança é necessária pois entre o canal cliente-servidor passa informação sensível relativa a dados bancários dos clientes (IBAN do beneficiário e a quantidade de dinheiro a ser transferida) e a operações/transações que este faça. Qualquer fuga de informação poderá causar bastantes prejuízos ao cliente (por exemplo, roubo/transferência de fundos em transações não autorizadas).

A utilização do meio comunicação entre cliente e banco por SMS para fazer a autorização de uma transação bancária poderá ser uma boa ideia pois a tecnologia SMS está presente em qualquer smartphone hoje em dia.

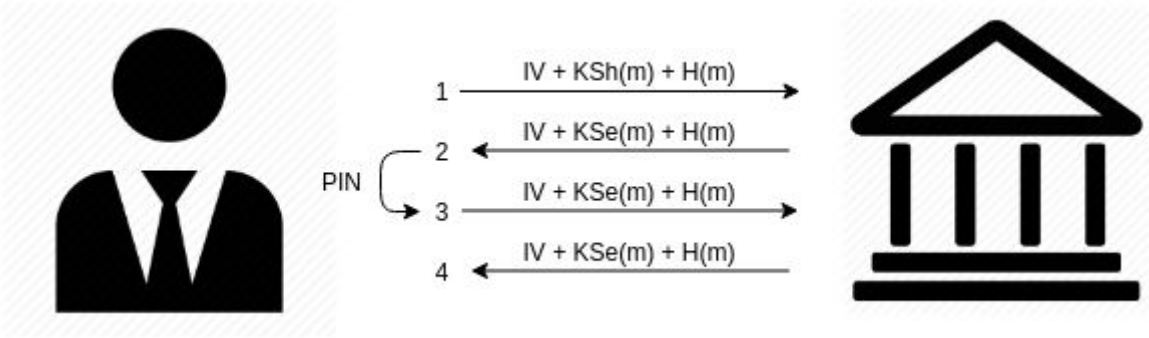
Se os requerimentos de segurança forem bem implementados, esta aplicação é bastante útil em casos onde o utilizador não tem rede 3g/4g, wifi ou acesso a uma ATM, conseguindo mesmo assim fazer uma transferência bancária pelo telemóvel.

2 – Requerimentos

Para que esta aplicação seja viável a segurança é um objetivo crucial. Estes são os requerimentos que são mandatários para conseguir obter uma boa segurança:

- Confidencialidade - garantir que ninguém para além do banco e do cliente saibam a quantia ou o destinatário da transferência.
- Integridade - garantir que o destinatário e/ou a quantia da transação não são alterados.
- Autenticação – Necessitamos para saber se o servidor e o cliente são quem dizem ser e não terceiros a fazerem se passar pelos mesmos.
- Não repúdio - o cliente não poderá afirmar que não foi ele que executou a transação.
- Proteção contra Replay Attacks – a mesma transação não será efetuada duas vezes.

3 - Solução Proposta



IV – Initialization Vector

m – mensagem a enviar

KSh – Encriptação com a chave partilhada

KSe – Encriptação com a chave de sessão

H – Hash da mensagem

Para esta fase é assumido que à priori que o cliente e o servidor acordaram no chave simétrica partilhada.

1 - O cliente irá estabelecer uma comunicação com o servidor, usando a chave secreta partilhada, esta mensagem irá conter um valor aleatório usado para criar uma chave de sessão para as duas partes.

2 - O servidor responde com Ok, para que o cliente saiba que o servidor já tem a chave de sessão.

PIN – Pede ao utilizador para se autenticar com o pin do dispositivo.

3 - O cliente envia os dados da transação que pretende fazer.

4 – O servidor responde com uma mensagem de confirmação do processo.

Todos os passos referidos acima contém uma hash criada com SHA-256, para garantia de integridade.

Para garantia de autenticação e confidencialidade foi usada criptografia simétrica AES com GCM (Counter mode), a encriptação das mensagens para o passo 1 foi feita com recurso há chave inicial partilhada pelo servidor e o cliente, nos passos 2 a 4 é usada a chave de sessão criada posteriormente ao passo 1.

Para não existirem replay attacks foi usado o IV da encriptação simétrica, pois este é criado em Java com recurso há classe SecureRandom e a probabilidade de se encontrar um número repetido é reduzida. No caso deste IV ser alterado por um atacante, para tentar evitar o replay attack, a desencriptação irá falhar.

As versões básica, intermedia e avançada são as seguintes:

Versão Básica: Comunicação entre servidor e cliente ainda sem garantias de segurança, troca de mensagens em claro.

Versão Intermédia: Implementação de alguns requisitos de segurança. Uso da keystore do android para implementação da encriptação das mensagens e criação hash das mensagens.

Versão Avançada: As garantias já das versões anteriores e comunicação entre servidor e cliente utilização dos IV's como nounces.

4 – Resultados

A solução criada sofreu muitas alterações em relação ao inicial proposal nomeadamente a não utilização de chaves públicas e privadas tanto no cliente como no servidor, isto deve-se ao facto de aquando da criação da solução para proposal inicial, não se teve em conta que os SMS's usavam 7-bits para os caracteres e não 8 como usado para web apps, restringindo o tamanho necessário das mensagens.

Tudo o que foi proposto nesta versão final foi cumprido.

5 – Avaliação

O nosso projeto contempla confidencialidade, autenticação, integridade e frescura, mas não contempla não repúdio. Um dos pontos fortes da nossa aplicação é o facto de conseguir garantir a confidencialidade e a integridade das mensagens dentro dos limites de caracteres impostos pelo tamanho dos SMS's. Um dos pontos fracos é o facto de não garantir não repúdio.

6 – Conclusão

O nosso projeto tem os requesitos de segurança propostos com a exceção do não-repúdio e deveria ter também pedir um PIN ao utilizador para efeitos de autenticação.

7 – Referências

Android Studio

Android keystore - https://developer.android.com/training/articles/keystore

Java 1.8

javax.crypto e java.security

android.telephony.SmsManager