SSMS - A Secure SMS Messaging Protocol for the M-Payment Systems

Presented at the 13\textsuperscript{th} IEEE Symposium on Computers and Communications (ISCC’08)

by:

Mohsen Toorani

(ResearcherID: A-9528-2009)
Security Flaws of the GSM

- Unilateral authentication, and vulnerability to the man-in-the-middle attack
- Flaws in implementation of A3/A8 algorithms
- SIM card cloning
- Over-the-air cracking Ki
- Flaws in cryptographic algorithms
- Short range of protection
- Lack of user visibility
- Leaking the user anonymity
- Vulnerability to the DoS attack
- Absence of integrity protection
- Vulnerability to replay attacks
- Increased redundancy due to the coding preference
Security Flaws of the SMS

- All the GSM security vulnerabilities are inherited to SMS.
- SMS also suffers from some additional vulnerabilities due to:
  - Its store-and-forward feature,
  - The problem of fake SMS that can be conducted via the Internet,
  - Vulnerability to disclosure of previous exchanged messages by an unauthorized access to the phone.
End-to-end Security in the GSM

End-to-end security in the cellular systems can be provided using the processing capabilities of one or some of the following items:

- The *Mobile Equipment* (ME) using programming languages
- SIM card using SAT
- An additional smart card, e.g. JavaCard
- A crypto-processor that is embedded in the ME
- A portable PC (laptop) connected to the ME.

Our approach (SSMS) is based on the first solution and suggests J2ME (Java 2 Mobile Edition) as the programming platform.
The Proposed Protocol (SSMS)

SSMS consists of three phases:

- Initialization
- Message exchange
- Judge verification
SSMS: Initialization Phase

The initialization phase includes:

- Selecting the domain parameters,
- Registering the user details into the system, generating the public/private keys, and issuing a certificate for the public key of each user.
- Installing the application software on the user’s mobile phone.
SSMS: Initialization Phase (Cont.)

Selecting the domain parameters

Domain parameters of the SSMS:

- Weierstrass equation of the selected EC: \( y^2 = x^3 + ax + b \)
- Defined over finite field \( F_q \) where \( q \) is a large prime number
  \[
  a, b \in F_q
  \]
- To assure the non-singularity: \( 4a^3 + 27b^2 \neq 0 (\text{mod } q) \)
- \( G \in E(F_q) \) is base-point of EC and is of order \( n \)
- To thwart small subgroup attacks:
  \[
  \begin{cases}
  nG = O \\
  n > 4\sqrt{q}
  \end{cases}
  \]
- To thwart other known attacks on EC:
  \[
  \begin{cases}
  n \neq q \\
  n \nmid q' - 1, \quad 1 \leq i \leq 20
  \end{cases}
  \]
- To guarantee the intractability of ECDLP:
  \[
  n > 2^{160}
  \]

(for ordinary applications)
SSMS: Initialization Phase (Cont.)
Public/private key generation

Public/private key generation in SSMS:

Key Generation in a Key Generating Server (KGS) (First Approach)
- SIM Card
- 1. ID_U
- 2. PK_U, SK_U, Cert_U, PK_CA
- Database Server (LDAP)
- 3. ID_U, PK_U, Cert_U

Key Generation in the ME (Second Approach)
- CA Server
- 1. ID_U, PK_U
- Mobile User
- 2. Cert_U
- Gateway
- 3. ID_U, PK_U, Cert_U
- Database Server (LDAP)

The private key of user U: \( SK_U \in_R [1, n-1] \)
The public key of user U: \( PK_U = SK_U G \)
SSMS: Initialization Phase (Cont.)

Application Installation

When application installation is accomplished via an OTA server, and the key generation is taken place in the ME.
SSMS: Message Exchange

Basic configuration for the SSMS

Generating and extracting SSMS

SSMS - A Secure SMS Messaging Protocol for the M-Payment Systems
(Copyright © 2008 IEEE) Presented at ISCC'08 by Mohsen Toorani
SSMS: Message Exchange (Cont.)

Optimized configuration for the SSMS:
**SSMS: Judge Verification (in disputes)**

*Bob* claims that he has received an SSMS from *Alice* containing \((R, C, s)\).

The trusted third party (judge) wants him to provide \((R, C, M, k, s)\).

*Bob* simply extracts \(M\) and \(k\) from the previously saved \((R, C, s)\).

The judge follows the following steps to decide on what *Bob* claims:

- Checks the validity of \(Cert_A\) and uses it for verifying \(PK_A\)
- Verifies whether \(M = D_k(C)\)
- Computes \(t = H(M \parallel x_R \parallel ID_A \parallel y_R \parallel ID_B \parallel k)\)
- Verifies the *Alice's* signature by checking the \(sG + R = tPK_A\) condition.
On the Security of SSMS

SSMS provides the following security attributes:

- Confidentiality
- Authentication
- Integrity
- Unforgeability
- Non-repudiation
- Forward secrecy of message confidentiality

It also provides the Public verifiability as a facility.
Conclusions

- SSMS is a new application layer protocol that provides the confidentiality, integrity, authentication, non-repudiation, public verification, and forward secrecy of message confidentiality.
- SSMS efficiently combines encryption and digital signature and uses public keys for a secure key establishment to be used for encrypting short messages via a symmetric encryption.
- SSMS has great computational advantages over previously proposed public key solutions while simultaneously providing the most feasible security services.
- It has great advantages to be used in real m-payment applications and whenever the secure SMS messaging is important.
- The solution is suitable for other store-and-forward technologies.
Thanks

Thank you for your attention!