Proxy signature scheme based on McEliece public key cryptosystem

Zhao Cheng-cheng,

Li Zi-chen , Yang Ya-tao

Beijing Electronic Science and Technology Institute

Context

Context

1. Introduction
2. Main idea
3. Detail of scheme
4. Analysis
5. Conclusion

1. Introduction

- Computer forensics is the technology of applying computer technology to access, investigate and analyze the evidence of computer crimes.

  It mainly includes the processes of determining and obtaining digital evidence, analyzing and taking data, filing and submitting result. Hence, digital signature is very useful for computer forensics.

1. Introduction

- As we all know, the security of digital signature base on difficult problem, eg. RSA-PSS(R) base on Factori zation Problem, DSA and ECDSA base on Discrete L ogarithm Problem. However, Peter Shor proposed a Quantum Algorithm, which can solve Factorization P roblem and Discrete Logarithm Problem within polyn omial time.

1. Introduction

- With quantum computer, Peter Shor algorithm can break all digital signature schemes that based on Factorization Problem or Discrete Logarithm Problem. Therefore, the security of digital signature is faced with serious threat. The so-called post-quantum public key cryptosystem has became the focus of research. McEliece public key cryptosystem is one of it.

Context

1. Introduction
2. Main idea
3. Detail of scheme
4. Analysis
5. Conclusion

2. Main idea

- **2.1 McEliece public key cryptosystem**
- **Key generation:**
- **The Public Keys**

The public key is given by the public $n \times k$ generator matrix $G_p = SG_sP$ over binary field F_2 , where G_s is a generator ma trix of the secret code Γ.

- **The Private Keys**

The McEliece secret key consists of the Goppa polynomial $g(Y)$ of degree t defining the secret code Γ , an $n \times n$ perm utation matrix P and a non-singular $k \times k$ matrix S over bi nary field F_2 .

2. Main idea

- **The Encryption Process**

To encrypt a message $m \in F_2$, where F_2 is binary field, the us er choose a random vector $e \in F_2$ with hamming weight $w_H(e) = t$, and compute that $c = mG_p + e$, where e is a ran dom error vector, then obtain the ciphertext c.

- **The Decryption Process**

First, we calculate that

$$c' = cP^T H^T = mSGPP^T H^T + eP^T H^T ,$$

then we use the rapid Goppa code decoding algorithm to the $eP^T H^T$. Since the hamming weight of eP^T and e are eq ual that is $W_H(eP^T) = W_H(e) = t$, we can get mS by decoding.

Finally, the plaintext m can be recovered from calculating mSS^{-1}.

Context

3. Detail of scheme

Parameter Selection

- Original signer A choose a error-correcting binary Goppa codes C_A . As for C_A , there exists a $k \times n$ generator matrix G_A and a $(n-k) \times n$ parity check matrix H_A . Then choose an $n \times n$ permutation matrix *P* and a non-singular $k \times k$ matrix S over F_2 . Our main task is looking for the matrix G_A^* to make $G_A G_A^* = I_k$ be established, where I_k is a unit matrix.
- Let $J_A = P_A^{-1} G_A^* S_A^{-1}$, $W_A = G_A^* S_A^{-1}$ and $T_A = P_A^{-1} H_A^T$.
- Suppose original signer A is honest, choose another corresponding $k \times n$ generator matrix G_B for code C_A and generate a non-singular $k \times k$ matrix S_B to make $S_B G_B = S_A G_A$ satisfied. We keep S_B and G_B secret as private key and give it to proxy signer B.

3. Detail of scheme

List 1. Parameter List of Proxy Signature

	Public key	Private key
Original Signer A	J_A, W_A, T_A, H_A, t_A and t' （where t' are integers less than t_A ）	S_A, G_A, P_A
Proxy Signer B	The same as A	S_B, G_B, P_A

3. Detail of scheme

- 3.2 Signature Process

Proxy signer B sign message m_j as follows:

1) Randomly select a binary vector e_j with the length of *n*, and hamming weight is $W(e_j)=t'$;

2) Signature c_j calculate by $c_j=(e_j+m_jS_BG_B)P_A$

- 3.3 Verification Process

Because the whole signature process may be disturbed by noise, thus signature may make a mistake. Therefore let received signature be c_j' , then the verification process is as follows:

First, we compute

$$
\begin{aligned}
D_1(c_j') &= c_j'T_A \\
&= [(e_j+m_jS_BG_B)P_A]'P_A^{-1}H_A^T \\
&= e_j'H_A^T+m_jS_BG_BH_A^T
\end{aligned}
$$

3. Detail of scheme

- From the above, we will get e_j' through Berlekamp-Massey algorithm. Compare the hamming weight of e_j' and e_j, if $W(e_j') \neq t'$ or generate decoding error, the receiver will request retransmit the signature.

  If $W(e_j') = W(e_j) = t'$, then go on the next step.
- Let $D_2(c_j') = D_2(c_j) = c_j J_A$, then receiver calculate $D_3(c_j') = D_3(c_j) = D_2(c_j) + e_j W_A = c_j J_A + e_j W_A$ and verify whether the value of $D_3(c_j')$ is equal to m_j. The signature is effective if the answer is yes, or it is invalid.

Context

4. Analysis

- **4.1 Correctness Analysis**
- **Let** $D_2(c_j^{'}) = D_2(c_j) = c_j J_A$ **, substitute** $e_j + m_j S_B G_B$ **and** $P_A^{-1} G_A^{*} S_A^{-1}$ **for** c_j **and** J_A **respectively, we get**

$$\begin{aligned} D_2(c_j^{'}) &= D_2(c_j) \\ &= c_j J_A \\ &= [(e_j + m_j S_B G_B) P_A] P_A^{-1} G_A^{*} S_A^{-1} \\ &= e_j G_A^{*} S_A^{-1} + m_j S_B G_B G_A^{*} S_A^{-1} \end{aligned}$$

And then we compute that

$$\begin{aligned} D_3(c_j^{'}) &= D_3(c_j) \\ &= D_2(c_j) + e_j W_A \\ &= e_j G_A^{*} S_A^{-1} + m_j S_B G_B G_A^{*} S_A^{-1} + e_j G_A^{*} S_A^{-1} \\ &= e_j G_A^{*} S_A^{-1} + m_j S_A G_A G_A^{*} S_A^{-1} + e_j G_A^{*} S_A^{-1} \\ &= m_j \end{aligned}$$

4. Analysis

Receiver verify $D_3(c'_j)$ by public key to see whether it is equal to m_j. The sign is effective if it is, otherwise the sign is invalid.

- **4.2 Security Analysis**
- 1) Verifiability

All the needed parameters for verification are open.

Such as identity authentication, message *m*, public ke ys, etc. Therefore any verifier can verify the effective ness of proxy signature.

- 2) Distinguishability

Since the private keys of original signer and proxy sig ner are different, verifier can verify the validity of sig nature easily.

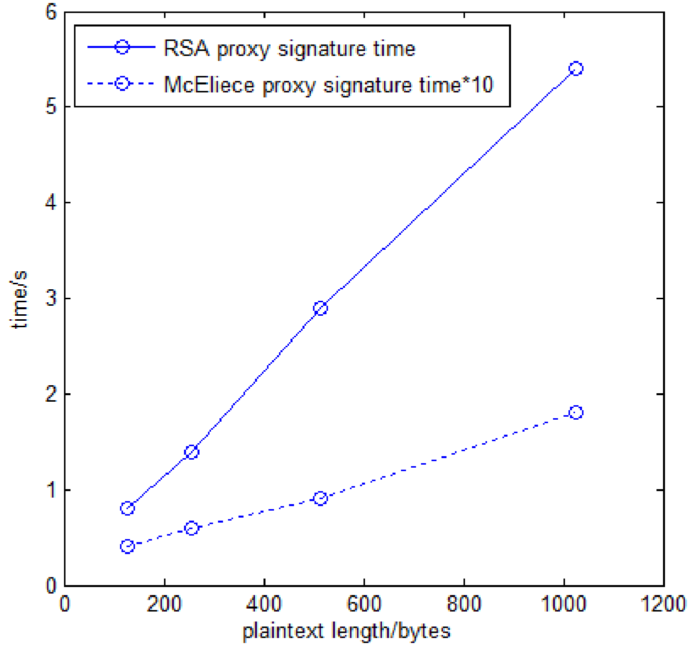
4. Analysis

- 3) Non-repudiation
- Once there is a dispute, verifier could judge by equation $D_3(c'_j) = e_j G_A^* S_A^{-1} + m_j S_B G_B G_A^* S_A^{-1} + e_j G_A^* S_A^{-1}$. If $D_3(c'_j) = m_j$, it is proxy signature, or it is original signature.
- 4) Non-forgeability
- It is equivalent to the matrix decomposition NPC problem. Att acker can’t obtain private key, neither can he forge proxy signa ture. At the beginning, we suppose the original signer is honest , so he couldn’t forge proxy signature, either.
- 5) Prevent the abuse of signature
- Every time, original signer select private key and give it to pro xy signer secretly, i.e., original signer authorize to proxy signe r. Therefore, proxy signer not allowed signing unauthorized do cument. Of course, the original signer not permit to transfer sig nature right illegally.

4. Analysis

❖4.2 Efficiency Analysis

❖ We choose different length of plaintexts and sign them respectively. Plaintexts are 128bytes, 256bytes, 512bytes and 1024bits.



Graph 1. Comparison signature time of RSA and McEliece

Context

5.Conclusion

- From the graph1 above we can find that McEliece proxy signature is much faster than RSA proxy signature. So McEliece proxy signature is superior to RSA proxy signature in efficiency.
- According to security analysis, to solve private keys is equivalent to matrix decomposition NPC problem. Therefore, it is impossible to decipher private keys. Neither can he decipher ciphertext.

Acknowledgment

- This work is supported by the National Natural Science Foundation of China under Grants No. 61070219.

Thank you！