CPR Verification Analysis Based on TSA

Zhi-qiang LIU

School of Information Science and Engineering, Linyi University, Shuangling Road, Linyi City, Shandong Province, China 276002
Corresponding author

Keywords: Time Stamp Protocol, CPR, Verification.

Abstract. Time stamp protocol is a message encryption protocol based on Hash function. This paper presents a scheme of electronic medical record authentication system based on timestamp protocol. This scheme encrypts the digital digest of the electronic medical record by means of digital signature, which can reduce the dependence on the communication bandwidth while encrypting. Through the analysis of the anti-attack ability, we can see that based on the context correlation of timestamp chain, this scheme can effectively prevent the modification of the content of electronic medical records and authentication time.

Introduction

As an important data security technology, digital signature plays a very important role in ensuring the security, confidentiality and non-repudiation of data transmitted on the network. With the development and popularization of network technology, digital signature will play a more important role. Digital certificates are encrypted asymmetrically, i.e. public key system. Users have a key pair that can be encrypted and decrypted. The private key user of the key pair keeps it by himself and others can't know it, and decrypts and signs it with the sub-private key, while the public key mountain of the key pair is publicly distributed in the public database or e-mail box. If a confidential message needs to be sent, the sender first encrypts the message using the public key of the receiver, and the receiver decrypts the message using its own private key after receiving it, so that the information can safely reach the receiver. As long as the digital certificate is used, the user's information security can be guaranteed as follows: the information is transmitted between the sender and the receiver, and other people can't get it; other people can't tamper with the information content in the process of information transmission; from the digital certificate, we can know the identity of both sides of communication: whether the information sent by the sender has or not. Repudiation: message information is complete and authentic in the whole process of sending.

Time stamp is an important part of the improved digital signature scheme in this paper. This paper makes full use of the advantages of time stamp to improve the existing digital signature scheme.

The Concept of Time-Stamping Protocol

Time stamp is a time mark to mark documents. Water is used to ensure the reliable existence of any electronic document at any time. The timestamp in this paper is used to mark the specific time when the document is issued, and to compare and verify with the confirmation information.

Time is a very important digital information in network transmission. In paper contracts, the time of signing documents is as important as signature, which can effectively prevent forgery and tampering of documents.

In the network data transmission, the same time and time of data transmission are equally important, and security measures need to be taken. This is achieved by digital time-stamp service DTS (Digital Time-Stamp Service), which can provide security protection Yang for electronic documents when they are published. This service is a security service on Internet, which is provided by special network service organizations.

Time Stamp is a certificate given by an authoritative authority that a message exists at a specific time I and J. It consists of three parts:
(1) Document summaries requiring time stamps;
(2) The date and time of receipt of documents by DTS;
(3) Digital signature of DTS.

**Working Principle of Time Stamp**

The schematic diagram of the time stamp server is shown in Figure 1. Its working process can be described as follows:

1. Users first hash the message data:
2. The user sends the hash result to the timestamp server to make a timestamp request:
3. The time stamp server signs the open list results and the date and time, and generates the digital time stamp:
4. The timestamp server sends the digital timestamp and message data back to the user.

![Figure 1. Working Principle of Time Stamp.](image)

**Hash Function**

Methods to verify data completeness and authenticity include message encryption, message authentication code (MAC), and hash function. Each of the above methods follows a different authentication code—the ciphertext of the whole message, password verification and MAC, and digital digest of the original text, respectively. The digital digest, also known as the “fingerprint”, uses the hash function to map the original text of any length to the Hash value of the fixed length. The digital digest, though not an encryption mechanism, can use the verification code to judge the completeness of the data information. Take the message, X, of any length for example. The hash function, H (X), can generate a Hash value, h, of a fixed length, namely h=H (X).

- H can be applied to the message of any length and its output is of a fixed length;
- H is one-dimensional and irreversible;
- H is unique, which is sometimes known as “weak collision resistance”;
- When the same messages are input, H can always output the same Hash value through the same processing;
- It is unfeasible for any (x, y) to achieve H (X) = H (Y). In other words, any minor change forgiven by the message will exert a huge influence on the output Hash value, which is called “strong collision resistance”. The commonly-used hash algorithms include MD5 message abstract algorithm, SHA-1 compression function algorithm, RIPEMD-160 algorithm, etc.

**Time Stamp Protocol (TSP)**

Time Stamp Protocol (TSP) [3] includes Time Stamp Authority (TSA), Subscriber (S), and Relying Party (R). The Subscriber sends the file to be stamped (full length or digital digest of the computer-based patient record, CPR) to TSA, and obtain the digital signatory bearing the time-stamp from TSA. If verification is required, the Subscriber can send the signed file to Relying Party for authentication of the time stamp.

**Construction of TSP**

The S sends the file y to TSA, and TSA sends back Z=SIG (v,t), where t denotes the time taken to receive X. In order to ensure the speed and safety of system data exchange, y is usually the digital
digest obtained by the original text X through Hash coding, namely y=Hash(X). R just needs to verify whether SIG(y, t) is the signatory of TSA, and whether y is the digital digest of X. If both are true, the time-stamping is deemed as valid. The TSP is simple, which cannot prevent TSA from being jointly attacked by the Subscriber, so TSA should have adequate credibility. Besides, once the TSA encryption key (generally referring to the secret key in the conventional encryption system) becomes invalid, the whole time-stamping system will become completely invalid [4].

**Principle of CPR Certification System Adopting TSP**

The computer-based patient record (CPR) system is designed in this paper to authenticate CPRs submitted by the medical institution as the evidence in case of the doctor-patient conflict. The system consists of one independent and credible third-party CPR Authentication Server (CAS), and one or multiple time-stamping institutions, TSAs, as the mediation departments (or the Relying Party) to certify user conflicts. The CAS does not participate in the stamping process but just serves as the intermediary, which submits the verification results to the conflict mediation department.

The system time-stamping process adopts the protocol, which combines the binary tree simplified scheme and the link list scheme to facilitate storage of signatories and verification. The time-stamp occurrence process: The S submits the digital digest, \( Y_n = H(X_n) \), of the full text, \( X_n \), of the CPR to the TSA, and according to the protocol, the signatory time-stamp can be generated by the following recursive algorithm:

\[
Z_n = \text{SIG}_{\text{TSA}}(n, t, \text{ID}_n, Y_n, Z_{n-1})
\]

The TSA sends back the time-stamp \( Z_n \) to the S, and also submits the \( (Z_n, \text{ID}_n) \) to the database of the CAS. The CAS is responsible for storing all time-stamp records and forming a complete time-stamp record link list.

![TSP realization process.](image)

**Verification process:** S submits the full text, \( X'_n \), and the signatory, \( Z_n \), of the CPR to be verified to the CAS for verification, and the CAS judges whether the following equation is substantiated:

\[
Z_n = \text{SIG}_{\text{TSA}}(n, t, \text{ID}_n, Y'_n, Z_{n-1})
\]

If the above equation is substantiated, \( X'_n = X_n \), and \( X'_n \leftrightarrow X_n \), suggesting that the original text of the CPR is modified.

**Anti-attack Analysis**

The potential types of attack to be considered by the system include:

*Falsification of the original text: Under the prerequisite of maintaining the original time-stamp, Zn, the medical institution, S, modifies the plain text, \( X'_n \), to replace the original text, \( X_n \). Attack analysis of r: Since \( Z_n = \text{SIG}_{\text{TSA}}(n, t_n, \text{ID}_n, H(X_n), Z_{n-1}) \), the Hash function is a one-way function, so the medical institution cannot alter the time-stamp.*
function, and the signatory of the TSA is based on the secret key encryption or Hash function, the calculation results cannot ensure \( Z_n = \text{SIG}_{\text{TSA}} (n, t_n, ID_n, H (X'_n), Z_{n-1}) \) to be substantiated.

Modification of time: As to \( X'_n \) submitted by \( S \) at the time larger than \( t_n \) and \( t'_n \), the time-stamp, \( Z'_n \), is constructed together by the \( S \) and TSA at the time of \( t_n \), and through decoding of \( Z'_n \), the time, \( t'_n \), can be obtained.

Anti-attack analysis: Obviously, if a simple TSP is adopted, the falsification can use the legitimate signatory of the time-stamp, which cannot be verified. As to the linear link list scheme, the CAS stores all time-stamps, \( Z_n \), and the time, \( t_{n-1} \), of \( Z_{n-1} \), must be smaller than the time \( t_n \) of \( Z_n \). In practical applications, several time-stamps, \( Z_{n-1} \), have already been stored and published at different \( t_{n-1} \) periods of time from \( t_n \) to \( t'_n \). As a result, at the time \( t'_n \) to generate \( Z'_n \), the system cannot insert it into the link list.

Besides, since the time-stamp stored by the CAS is also published to several users, \( S \), the CAS cannot arbitrarily modify the records in the time-stamp link during the verification process, which can in turn restrict the behaviors of the CAS.

Conclusions

The time spent in each step of encryption or decryption in RSA arithmetic by common digital signature scheme and timestamp signature scheme. By analyzing the time spent in RSA arithmetic towel by two digital signature schemes, we can get the data in Table 1.

<table>
<thead>
<tr>
<th>Units(ms)</th>
<th>Receiver’s Non-repudiation Digital Signature</th>
<th>Digital signature of timestamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>Action</td>
<td>Time</td>
</tr>
<tr>
<td>Sender</td>
<td>Symmetric encrypted plaintext</td>
<td>910</td>
</tr>
<tr>
<td>Sender</td>
<td>Private key signature</td>
<td>9011</td>
</tr>
<tr>
<td>Sender</td>
<td>Receiver Public Key Encryption</td>
<td>9023</td>
</tr>
<tr>
<td>Receiver</td>
<td>Private key decryption</td>
<td>9027</td>
</tr>
<tr>
<td>Receiver</td>
<td>Sender Public Key Decryption</td>
<td>9025</td>
</tr>
<tr>
<td>Receiver</td>
<td>Private key signature</td>
<td>9014</td>
</tr>
<tr>
<td>Receiver</td>
<td>Sender Public Key Encryption</td>
<td>9020</td>
</tr>
<tr>
<td>Sender</td>
<td>Private key decryption</td>
<td>9028</td>
</tr>
<tr>
<td>Sender</td>
<td>Receiver Public Key Encryption</td>
<td>9021</td>
</tr>
<tr>
<td>Sender</td>
<td>Private Key Encryption K</td>
<td>51</td>
</tr>
<tr>
<td>Sender</td>
<td>Receiver Public Key Encryption K</td>
<td>56</td>
</tr>
<tr>
<td>Receiver</td>
<td>Private key decryption K</td>
<td>58</td>
</tr>
<tr>
<td>Receiver</td>
<td>Sender Public Key Decryption K</td>
<td>55</td>
</tr>
<tr>
<td>Receiver</td>
<td>Decrypt plaintext with K</td>
<td>990</td>
</tr>
<tr>
<td>Total time</td>
<td>-</td>
<td>74289</td>
</tr>
</tbody>
</table>

From Table 1, we can see that in the non-repudiation digital signature of the receiver, the receiver receives the information sent by the sender, decrypts it, then signs and encrypts it and sends it to the sender. The sender decrypts it again to know whether the receiver has received the information sent by itself. In this way, the acknowledgement information sent by the receiver to the sender is the whole encrypted information of the transmitted data.

References


