Non-Repudiation In Practice

Chii-Ren Tsai

Citigroup
Citigroup Information Security Office
12401 Prosperity Drive
Silver Spring, MD 20904, USA
chiiren.tsai@citigroup.com

ABSTRACT

In this paper, fundamental properties of non-repudiation are derived. These properties can be used to verify whether a non-repudiation architecture or design is sufficient to satisfy the non-repudiation requirement. Two potential non-repudiation architectures for financial transactions using challenge-response one-time password tokens and digital signatures are described and compared. Finally, this paper concludes that supporting non-repudiation for high-value transactions even without legal binding is a sound practice.

KEY WORDS

Certification Authority, Certificate Revocation List, Non-Repudiation, Digital Signature, One-Time Password, Secure Socket Layer, Transport Layer Security, Trusted Third Party.

1. Introduction

The notion of non-repudiation for an electronic transaction is to prevent both parties, namely customers and financial institutions, from repudiating the transaction after it is committed. This simple notion is well understood by information security technologists, but has never been formalized as a core business requirement for financial transactions in most countries for the following reasons:

- Repudiation disputes for financial transactions are often resolved by contractual agreements or legal proceedings without too much technical evidence.
- To implement a non-repudiation solution for low-value transactions may not be cost-effective in mitigating the risk. Non-repudiation happens to be more cost-justifiable for high-value transactions.
- No legislation has ever been done globally to make it as a legal requirement or tool for preventing repudiation or resolving disputes.
- No technical standards and guidelines have ever been formally defined to promote its implementation with or without a legal binding.

Even if non-repudiation has not yet emerged as a core business requirement, some countries (e.g., Taiwan) already require that non-repudiation be supported for high-value transactions as a critical security measure. Some people strongly believe that non-repudiation can be supported only by digital signature or that digital signature by itself is sufficient to address non-repudiation without taking into account the complexity of creating a trusted third party [1] and other security requirements. Note that a trusted third party for certification and digital time-stamping is to prove:

- authenticity of digitally signed information,
- information is not tampered with, and
- digitally signed information existed at a precise time

For example, a financial institution or an accredited Certification Authority (CA) can become a trusted third party.

In this paper, we would list fundamental properties of non-repudiation in Section 2. Non-repudiation can be implemented in different ways as long as all its properties are satisfied. In Section 3, we would describe two potential non-repudiation architectures, one with challenge-response one-time password (OTP) or dynamic tokens and the other with digital signatures, and compare their pros and cons. Finally, we would conclude the paper.

2. Properties of Non-Repudiation

Non-repudiation means a service that provides proof of the origin and integrity of data, both in an unforgeable relationship, which can be verified by any third party at any time [2]. Consequently, the following properties can be derived and must be satisfied to support non-repudiation for financial transactions.

1. Transactions and customers must be tightly bound
2. Transactions must be difficult to forge
3. Transactions must be unalterable
4. Transactions must be verifiable

The first two properties would guarantee non-repudiation of submission, while the last two properties could achieve non-repudiation of receipt. First of all, each transaction must be bound to a customer via an acceptable authentication mechanism and the customer must get authenticated prior to submitting transactions to a financial institution. The authentication mechanism should be strong enough to uniquely hold the customer accountable for initiating the transactions as a result of authentication. Additional secure measures and mechanisms should be tightly coupled with the authentication mechanism to prevent transactions from being forged. Both the customer and the financial institution should be able to mutually authenticate each other to ensure that transactions be done by and with the right parties.

After a transactions is initiated, its contents including user ID, date and time, and transaction details, cannot be altered while in transit to maintain transaction integrity and allow future verifications if and whenever necessary. It must ensure that the transaction is unaltered and logged after it is committed and confirmed. Logs must be archived and properly protected to prevent unauthorized alteration. Whenever there is a repudiation dispute, transaction logs along with other logs or data can be retrieved to verify the initiator, date and time, transaction history, and so on.

3. Potential Examples for Non-Repudiation

Due to lack of standards and legal binding for non-repudiation, it is not straightforward to judge which solution is acceptable. To be objective, any design satisfying the above non-repudiation properties should be sufficient enough to address non-repudiation as a security measure. We will elaborate two potential non-repudiation architectures: one is based on challenge-response OTP tokens, and the other takes advantage of digital signatures. Both challenge-response OTP and digital signature provide strong authentication and unique binding between individuals and their OTP token and digital signature, respectively. Therefore, they can be used for satisfying part of the non-repudiation requirement.

A typical Internet-based application shown in Figure 1 would be used to describe detailed interactions among the components of any application based on one of these two architectures. In general, customers would use a web browser to connect to a web server of a financial institution via Secure Socket Layer (SSL) or Transport Layer Security (TLS) protected sessions for server authentication and conducting transactions. The web server would communicate with a generic authentication server for user authentication and with a backend for mediating customers' transaction requests. The authentication server supports OTP or static/dynamic passwords for both architectures, respectively. The backend in many cases could represent both a logical combination of an application server and a backend server and/or databases.

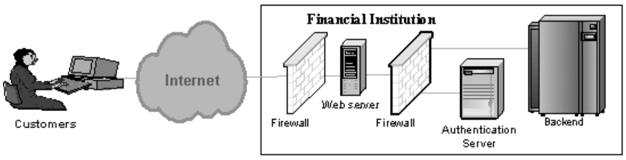

Figure 1. Network Architecture for an Internet Application.

3.1 Non-Repudiation With Challenge-Response OTP Tokens

In this scenario, non-repudiation is achieved through the legal document binding and the binding of the following security mechanisms and trusted processes for server management: SSL, a challenge-response OTP token, secure hashing, and audit logs. The detailed design is described as follows.

First of all, a customer must present a positive ID for authentication and sign legal documents to open an account at a financial institution. After that, a challenge-response OTP token (e.g., Secure Computing's SafeWord) is given to the customer. Each token that has a unique serial number is tamper-proof and cannot be duplicated or forged. Therefore, there is a strong binding between an OTP token and the customer. With the token, the customer is ready to conduct transactions. Information flows between a client and servers for transactions with a challenge-response OTP token are shown in Figure 2.

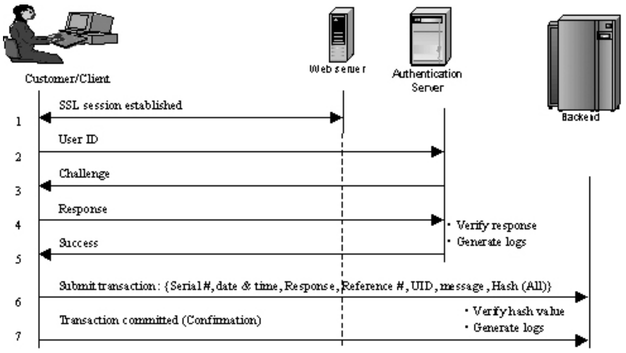

Figure 2. Information Flows for Transactions with a Challenge-Response OTP Token.

Transactions and Customers are Tightly Bound

When a customer or client is ready to request a transaction via a web browser, an SSL session first of all is established between the browser and a web server to authenticate the web server and protect data transported in between (1). SSL allows the customer to authenticate the web server by checking its SSL digital certificate. After that, the customer would start to engage in the authentication or login process. The customer would send his/her user ID over SSL to the web server en route to the authentication server (2), which immediately returns a challenge (3). The customer has to enter the challenge into his/her OTP token to generate a response for authentication. The response is then being put on the browser and transported to the authentication server for validation (4). If successful, the customer and the server mutually authenticate each other and an audit record for this login is created on the audit logs on the authentication server (5). The customer is authorized to submit transactions. The authentication server must be managed with a well-defined administration procedure and a trusted process by which the audit logs cannot be tampered with.

Transactions are Difficult to Forge and Unalterable

Each customer must possess an OTP token to get authenticated and authorized to perform transactions. Challenge-response OTP tokens provide real-time two-factor authentication. It would be very difficult to randomly guess the response of a challenge or compromise a smart token. Besides, the account would be locked after several unsuccessful attempts.

For each transaction, a secure hash (e.g., SHA-1) against authorized transaction based on the following data is being calculated at the customer's side:

- the serial number of the OTP token,
- date & time,
- response from the smart token,
- transaction reference number,
- user ID, and
- transaction message.

Including the response of the OTP token in the calculation of the secure hash of each transaction would make forging a legitimate transaction extremely unlikely even if the authorized SSL session were hijacked. The transaction along with its hash value transported via SSL would also ensure that data integrity is maintained and the transaction is unaltered in transit (6). When the request is received by the backend, a new hash value is being calculated against all the above data to compare with the hash value from the customer. If they are the same, the transaction request is being executed and an audit record with the above data and the hash value is being generated on transaction logs. After that, a digitally signed confirmation signed with the server's signing private key is returned to the client to indicate the transaction is committed (7). The backend is managed by another trusted process so that transaction logs on the backend cannot be tampered with.

Transactions are Verifiable

In the event of a non-repudiation dispute over a transaction being raised, the customer can present the confirmation, which was digitally signed by the server, and both archived authentication logs and transaction logs can be retrieved jointly to verify and resolve the dispute.

Pre-requisites

Non-repudiation in this design is partly achieved by the trusted processes for server management with separation of roles. By separation of roles, we mean the authentication and backend servers being managed by different administrative groups. Consequently, it would not be possible to inject fraudulent transactions and corresponding audit logs on the authentication server and the backend without being detected. In addition, the financial institution presumably must have published and enforced its information security policy and standards, and server management procedures against which the trusted processes are being audited, and audit logs are retained to comply with the regulatory requirements.

3.2 Non-Repudiation with Digital Signatures

A common practice for implementing non-repudiation is to take advantage of digital signatures, which could be considered as one of the best alternatives for replacing traditional signatures in electronic data processing. To enable digital signatures, a trusted third party (TTP) or public key infrastructure (PKI) should be available. The TTP or PKI may support at least a Certification Authority (CA) for issuing digital certificates and Certificate Revocation Lists (CRLs) for checking against revoked certificates. A non-repudiation design with digital signature is described in the following.

As mentioned before, account opening requires that each new customer present a positive ID and sign legal documents. The customer would receive specific code and instructions on how to generate public/private key pairs and then request digital certificates, and on how to configure and launch a financial application. After being generated, the private key(s) must be kept on a smart card if the underlying operating system is not secure enough to protect against unauthorized access to ensure that only the customer can access the signing private key. With public key technology and proper security measures to protect the signing private key(s), there is a strong binding between the customer and private keys with digital certificates. Information flows between a client and servers for transactions with this scenario are shown in Figure 3.

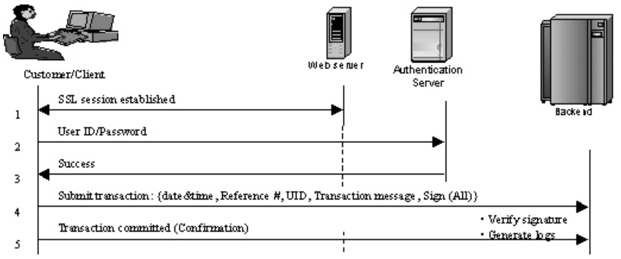

Figure 3. Information Flows for Transactions with Digital Signatures.

Transactions and Customers are Tightly Bound

When the customer launches the financial application, an SSL session is established between the client browser and a web server (1). The customer may submit ID/password to the web server en route to the authentication server for authentication (2). If successful, the customer is authorized to conduct transactions (3). A transaction request may consist of the following attributes: date & time, transaction reference number, user ID, transaction messages, and a signature of all previous attributes signed with the customer's signing private key (4). When the backend receives the request, it would verify the validity of the signature. If successful, the transaction is being executed, an audit record that contains transaction attributes and their digital signature is generated on transaction logs and then a digitally signed confirmation signed with the server's signing private key is returned to the customer to indicate that the transaction is committed (5).

Transactions are Difficult to Forge

If each customer's signing private key is long enough and well protected, it would be very difficult to forge a customer's signature unless the CA were compromised. If the CA is managed by a TTP, it is very unlikely to get compromised. Technically, it would take a very long period of time to conduct a brute-force attack on CA's root keys, which length is normally 2048 bits or above.

Transactions are Unalterable

For each transaction, a secure hash (e.g., SHA-1) against the following transaction attributes is being calculated and then digitally signed:

- date & time,
- transaction reference number,
- user ID, and
- transaction message.

The transaction attributes and its digital signature over SSL would ensure that the transaction is unaltered as long as the signature is valid.

Transactions are Verifiable

Whenever a repudiation dispute over a transaction is being raised, the customer can present the confirmation, which was digitally signed by the server, and archived transaction logs and customer's digital certificates could be retrieved jointly to verify and resolve the dispute.

Pre-requisites

Digital signatures should be implemented with certain guidelines [3][4][5]. Non-repudiation with digital signatures may require an accredited CA for trustiness. The CA must retain digital certificates long enough to satisfy local and regulatory requirements, especially when it would issue certificates to customers across several countries or geographical regions.

It is paramount that a smart card be required for each customer to generate and store private keys when untrusted PCs are being used to conduct transactions. Without a smart card, private keys would be encrypted and stored on PCs, which may not provide sufficient protection to safeguard the keys and are subject to unauthorized access. To equip each customer with a smart card, a smart card reader and digital certificates, the aggregated per-user costs would increase. More technical and customer support efforts may be necessary.

3.3 Comparison

The ad hoc OTP token solution is fairly simple, but cannot be standardized because all OTP mechanisms are proprietary and the strength of OTP mechanisms may not be as strong as that of public key technology. Besides, it requires too much trust on system administrative processes.

The digital signature solution is commonly adopted even if it may encounter challenges in creating a trusted third party for financial transactions [6]. Legislation and standardization on digital signature are gaining momentums and may gradually become available to unify its implementations and reduce costs. A brief comparison of the two solutions is listed in Table 1.

Table 1. Comparison of Two Non-Repudiation Solutions.

Features Solutions	OTP Tokens	Digital Signatures
Costs	Low	High
Technical and Customer Support Efforts	Low	High
Trusted Processes	More	Less
Network Messages and Latency Time	More	Less
Standardization	Unlikely	Likely

4. Conclusion

The two examples presented above could address non-repudiation as a security measure without legal binding. Even if the digital signature solution would be considered more strategic, it may take a while to get adopted as an acceptable legal solution because it solely depends on digital signature legislation and standardization. At present, digital signature is not fully legislated in most countries nor standardized in countries where electronic signatures can be legalized. Under the circumstances, we strongly believe it is a sound practice to enforce non-repudiation for high-value transactions. It can be used to quickly resolve repudiation disputes and reduce liabilities for both parties under certain contractual agreement. In this mean time, we should continue to promote non-repudiation as a major security requirement for high-value transactions and drive digital signature legislation and accreditation of trusted third parties globally.

Acknowledgment

We are grateful to Trevor Dudley and David Solo for discussing the OTP architecture. We also thank Tim Voss and Steve Russell for supporting this effort.

References

[1] Michael Froomkin, "The Essential Role of Trusted Third Parties in Electronic Commerce," October 1996, http://www.law.miami.edu/~froomkin/articles/trusted.htm.

[2] Adrian McCullagh and William Caelli, "Non-Repudiation in the Digital Environment," First Monday, volume 5, number 8, August 2000, http://firstmonday.org/issues/issue5_8/mccullagh/index.html.

[3] American Bar Association, “Digital Signatures Guidelines,” August 1996, http://www.abanet.org/scitech/ec/isc/dsgfree.html.

[4] Santosh Chokhani and Warwick Ford, “Certificate Policy and Certification Practice Statement Framework,” National Institutes of Standards and Technology, November 1996, http://csrc.nist.gov/pki/docs/fmk03nov.doc.

[5] Russell Housley, Tim Polk, Warwick Ford and David Solo, “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile,” Internet RFC 3280, April 2002, http://www.ietf.org/rfc/rfc3280.txt?number=3280.

[6] Chii-Ren Tsai, “Challenges for Building A Financial Public Key Infrastructure,” 7th Information Security Conference (INFOSEC'97), Taipei, Taiwan, May 1997.