Trusted timestamping

From Wikipedia, the free encyclopedia

Trusted timestamping is the process of securely keeping track of the creation and modification time of a document. Security here means that no one — not even the owner of the document — should be able to change it once it has been recorded provided that the timestamper's integrity is never compromised.

The administrative aspect involves setting up a publicly available, trusted timestamp management infrastructure to collect, process and renew timestamps.

Contents

History

The idea of timestamping information is actually centuries old. For example, when Robert Hooke discovered Hooke's law in 1660, he did not want to publish it yet, but wanted to be able to claim priority. So he published the anagram *ceiiinosssttuv* and later published the translation *ut tensio sic vis* (Latin for "as is the extension, so is the force"). Similarly, Galileo first published his discovery of the phases of Venus in the anagram form.

Sir Isaac Newton, in responding to questions from Leibnitz in a letter in 1677, concealed the details of his "fluxional technique" with an anagram:

The foundations of these operations is evident enough, in fact; but because I cannot proceed with the explanation of it now, I have preferred to conceal it thus: 6accdae13eff7i3l9n4o4qrr4s8t12ux. On this foundation I have also tried to simplify the theories which concern the squaring of curves, and I have arrived at certain general Theorems.

Classification

There are many timestamping schemes with different security goals:

- PKI-based - Timestamp token is protected using PKI digital signature.
- Linking-based schemes - timestamp is generated such a way that it is related to other timestamps.
- Distributed schemes - timestamp is generated in cooperation of multiple parties.
- Transient key scheme - variant of PKI with short-living signing keys.
- MAC - simple secret key based scheme, found in ANSI ASC X9.95 Standard.
- Database - Document hashes are stored in trusted archive; there is online lookup service for verification.
- Hybrid schemes - the linked and signed method is prevailing, see X9.95.

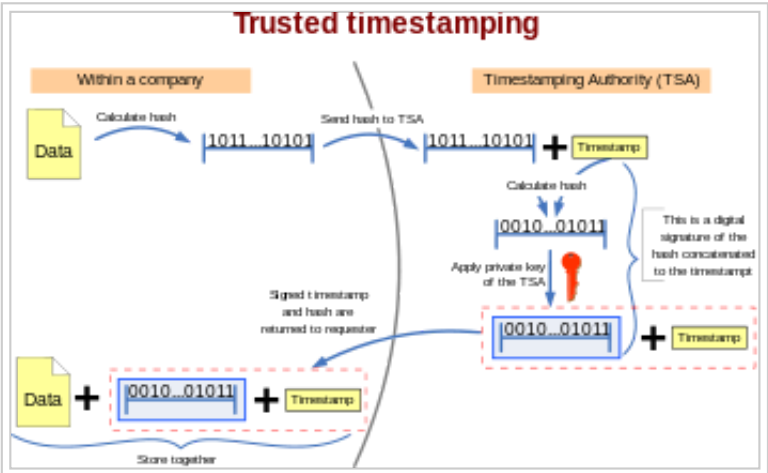
Coverage in standards:

Scheme	RFC 3161	X9.95	ISO/IEC 18014
PKI	Yes	Yes	Yes
Linked		Yes	Yes
MAC		Yes	
Database			Yes
Transient key		Yes	
Linked and signed		Yes	

For systematic classification and evaluation of timestamping schemes see works by Masashi Une.[1]

Trusted (digital) timestamping

According to the RFC 3161 standard (https://tools.ietf.org/html/rfc3161), a trusted timestamp is a timestamp issued by a trusted third party (TTP) acting as a **Time Stamping Authority** (**TSA**). It is used to prove the existence of certain data before a certain point (e.g. contracts, research data, medical records,...) without the possibility that the owner can backdate the timestamps. Multiple TSAs can be used to increase reliability and reduce vulnerability.



Getting a timestamp from a trusted third party.

The newer ANSI ASC X9.95 Standard for trusted timestamps augments the RFC 3161 standard with data-level security requirements to ensure data integrity against a reliable time source that is provable to any third party. This standard has been applied to authenticating digitally signed data for regulatory compliance, financial transactions, and legal evidence.

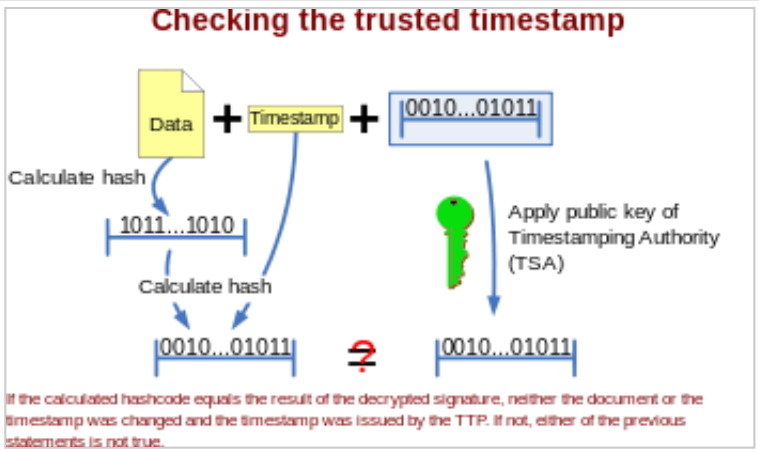
Creating a timestamp

The technique is based on digital signatures and hash functions. First a hash is calculated from the data. A hash is a sort of digital fingerprint of the original data: a string of bits that is practically impossible to duplicate with any other set of data. If the original data is changed then this will result in a completely different hash. This hash is sent to the TSA. The TSA concatenates a timestamp to the hash and calculates the hash of this concatenation. This hash is in turn digitally signed with the private key of the TSA. This signed hash + the timestamp is sent back to the requester of the timestamp who stores these with the original data (see diagram).

Since the original data cannot be calculated from the hash (because the hash function is a one way function), the TSA never gets to see the original data, which allows the use of this method for confidential data.

Checking the timestamp

Anyone trusting the timestamper can then verify that the document was *not* created *after* the date that the timestamper vouches. It can also no longer be repudiated that the requester of the timestamp was in possession of the original data at the time given by the timestamp. To prove this (see diagram) the hash of the original data is calculated, the timestamp given by the TSA is appended to it and the hash of the result of this concatenation is calculated, call this hash A.



Checking correctness of a timestamp generated by a time stamping authority (TSA).

Then the digital signature of the TSA needs to be validated. This can be done by checking that the signed hash provided by the TSA was indeed signed with their private key by digital signature verification. The hash A is compared with the hash B inside the signed TSA message to confirm they are equal, proving that the timestamp and message is unaltered and was issued by the TSA. If not, then either the timestamp was altered or the timestamp was not issued by the TSA.

Decentralized timestamps

With the advent of crypto currencies like Bitcoin it has become possible to securely timestamp information in a decentralized fashion. Data can be hashed and placed in the Block chain which serves as a proof of the time that data existed. The proof is due to a tremendous amount of computation performed after the hash was submitted to the block chain. Breaking the timestamp would also lead to breaking the entire integrity of the digital currency. A free implementation is available at Originstamp.org (http://www.originstamp.org) and described in this publication (http://www.gipp.com/wp-content/papercite-data/pdf/gipp15a.pdf).

See also

- Timestamp
- Timestamping (computing)
- Cryptography

- Computer security
- Digital signature
- Digital postmark
- Smart contract
- CAdES - CMS Advanced Electronic Signature
- PAdES - PDF Advanced Electronic Signature
- XAdES - XML Advanced Electronic Signature

References

1. ^ Une, Masashi (2001). "The Security Evaluation of Time Stamping Schemes: The Present Situation and Studies" (http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.23.7486). IMES Discussion Papers Series 2001-E-18.

External links

- RFC 3161 Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP)
- RFC 3628 Policy Requirements for Time-Stamping Authorities (TSAs)
- Decentralized Trusted Timestamping using the Crypto Currency Bitcoin (http://www.gipp.com/wp-content/papercite-data/pdf/gipp15a.pdf)
- ANSI ASC X9.95 Standard for Trusted Time Stamps (http://www.techstreet.com/cgi-bin/detail?product_id=1327239)
- ETSI TS 101 861 V1.4.1 (http://www.etsi.org/deliver/etsi_ts/101800_101899/101861/01.04.01_60/ts_101861v010401p.pdf) Electronic Signatures and Infrastructures (ESI); Time stamping profile
- ETSI TS 102 023 V1.2.2 (http://www.etsi.org/deliver/etsi_ts/102000_102099/102023/01.02.02_60/ts_102023v010202p.pdf) Electronic Signatures and Infrastructures (ESI); Policy requirements for time-stamping authorities
- CEN CWA 14167-1 (http://www.dnielectronico.es/seccion_integradores/cwa14167-01-2003-Jun.pdf) Security Requirements for Trustworthy Systems Managing Certificates for Electronic Signatures - Part 1: System Security Requirements
- Analysis of a Secure Time Stamp Device (https://www.sans.org/reading-room/whitepapers/vpns/analysis-secure-time-stamp-device-746) (2001) SANS Institute
- Implementation of TSP Protocol (http://www.csee.umbc.edu/~yzou1/681report.pdf) CMSC 681 Project Report, Youyong Zou
- XML Security Time Stamping Protocol (http://xml.coverpages.org/ApvrilleTStamp.pdf)
- Free TSA Service (http://www.freetsa.org) Free TSA Service
- TrueTimeStamp.org, a free timestamping (TSA) service (http://truetimestamp.org) Free time stamp

service that stores SHA-2 fingerprints of files, and provides signed certificates.

Retrieved from "http://en.wikipedia.org/w/index.php?title=Trusted_timestamping&oldid=650815610"

Categories: Computer security | Time | Authentication methods

---