Some Timestamping Protocol Failures

March 12, 1998

Mike Just
School of Computer Science
Carleton University
Ottawa, Canada

just@scs.carleton.ca
http://www.scs.carleton.ca/~just/

Outline

- Results
- Model
- Temporal Authenticity
- Benaloh/de Mare scheme
- Haber/Stornetta scheme
- Concluding Remarks

Results

- define two **classes** of timestamping schemes and appropriate **measures** of their temporal authenticity
- show how confusion between the classes (through lack of proper measurement) leads to a protocol failure
- show how overly ambitious assumptions and incomplete protocol descriptions lead to a protocol failure

Model

Goal is to

- authentically associate a **time** with data
- so the time and its authenticity can be respectively **measured** and **verified** at some later time.

Stamping Protocol

- On input y, produces a timestamp s.

Verification Protocol

- The authenticity of s is verified. If successful, the measure of time associated to y through s is accepted.

Applications

- patent submissions
- digital signatures
- intellectual property (e.g., lab books, academic papers)
- electronic commerce

Temporal Authenticity

Message (data-origin) authentication: assurance of the source of a message y.

Temporal authentication: message authentication + uniqueness + timelineness of a message y

- **Absolute:** assurance of the particular time at which a message was timestamped
- **Relative:** assurance of the temporal ordering (induced by the timestamp construction) of two messages
- **Hybrid:** assurance of the provision of both absolute and relative temporal authentication

Verifying Temporal Authenticity

1. verify the message authenticity of the timestamp
2. measure the time associated with the data by the timestamp

Absolute Measure: determines a particular time of stamping

Relative Measure: determines the ordering of two stamped messages

A message y has been **backdated** if a temporal measurement infers that y' was stamped before y when in fact, y' was stamped after y.

Benaloh/de Mare Timestamping (Eurocrypt '93)

Each round produces one stamp for m messages (**bulk authentication**). Let s_r be the stamp for round r. Results computed in a group of unknown order, e.g. $\mathbb{Z}_n$ where $n = pq$. Let y_i be submitted by user u_i.

$$s_r = x^{y_1 \cdots y_m}$$

Authenticity of s_r is maintained (irrelevant here) and u_i keeps $\{z_i, y_i\}$.

(**Verification**) Given y_i and $z_i = x^{y_1 \cdots y_{i-1} y_{i+1} \cdots y_m}$, u_i demonstrates that

$$z_i^{\,y_i} \equiv s_r$$

To provide timeliness, it is suggested to use

$$x = f(\text{current time})$$

Protocol Failure

Absolute measurement?

- Given y_i, z_i and s_r, how is x either recovered, or verified for its correctness? (Solution: it isn't.)

Providing a recoverable measurement

- **Absolute:** authentically store the current time along with s_r

- **Relative:** (chaining) authentically store

$$s'_r = h(s_r, s_{r-1})$$

Haber/Stornetta Timestamping (Journal of Cryptology '91)

Let s_r the stamp for round r. Let T be a timestamping service that

- is unable to backdate
- requires no record keeping

1. u sends y_r and $ID_r = ID_u$ where ID_u is the unique identification for user u, to T.
2. T computes the timestamp $s_r = sig_T(C_r)$, where

$$\begin{aligned} C_r &= (r, t_r, ID_r, y_r; L_r) \\ L_r &= (t_{r-1}, ID_{r-1}, y_{r-1}, H(L_{r-1})) \end{aligned}$$

3. For next request from user v, T sends $(s_r, ID_{r+1} = ID_v)$ to u.

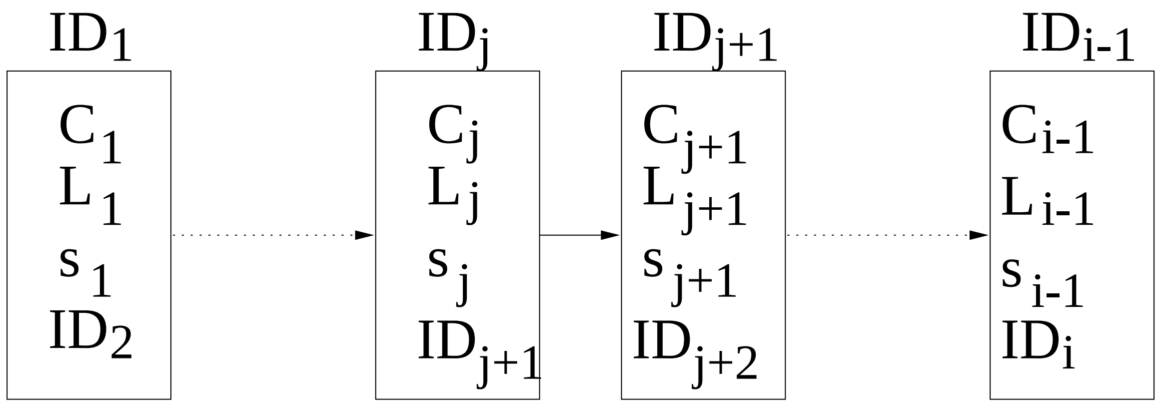
Haber/Stornetta (cont'd)

Absolute timestamp is provided by the inclusion of the time t_r.

Relative timestamp is provided by the inclusion of the linking information L_r.

Therefore a **hybrid timestamp** is provided.

Verification



1. ID_j produces (s_j, ID_{j+1}) for a challenger

2. signature on s_j is verified

3. (collusion protection) contact ID_{j+1} and obtain (s_{j+1}, ID_{j+2}) where

$$s_{j+1} = sig_T(j+1, t_{j+1}, ID_{j+1}, y_{j+1}; L_{j+1})$$

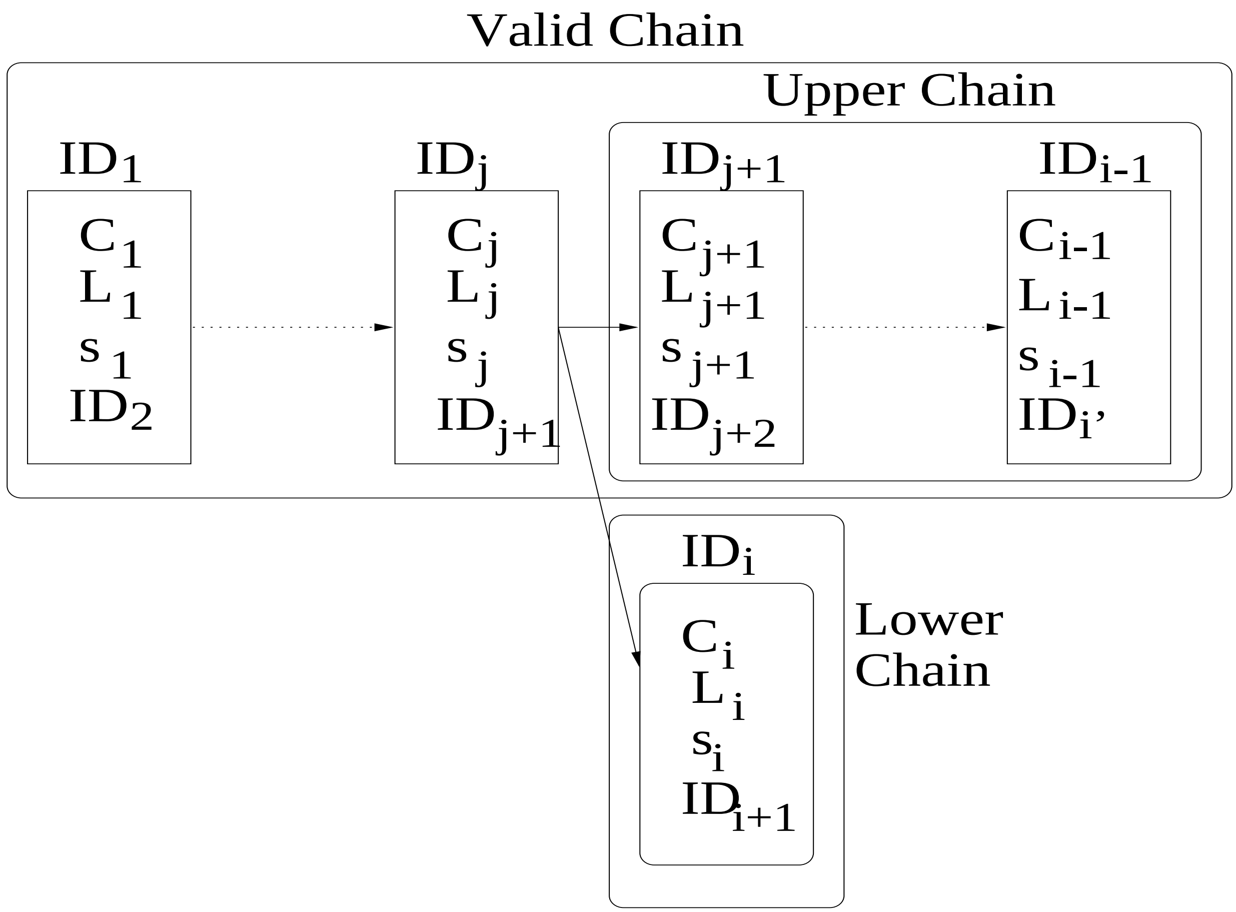
4. check that L_{j+1} contains both y_j and $H(L_j)$

5. can also check with ID_{j+2} or ID_{j-1}, etc.

Attack

- fake-chain attack (Haber/Stornetta)

- partial insertion attack



backdated (when measured absolutely) if $t_i < t_{j+1}$

Attack Detection?

- Verifying backwards from i to j.

  ($ID_i = ID_{j+1}$ or another collusion.)

- Repeated round numbers.

  (extra checks are required)

- Lags in time (because of backdating).

  (depends on the frequency of attacks and specifics of verification)

Attack Prevention

1. proper message authentication, e.g., storage (widespread or otherwise)

2. relative measurements (stamps are measured in pairs; combined with periodic authenticated storage)

Are **not straightforward** preventions since

- Item 1 alters the original stamping and verification procedures

- Item 2 alters the verification procedure.

- Item 2 can be used without Item 1

Concluding Remarks

- stamping and verification protocols must be fully explained
  - verification of authenticity
  - absolute timestamps require an absolute measurement
  - relative timestamps require a relative measurement
- important to indicate what level of trust is required for each entity
- evidence (e.g., storage of stamps) is important for **dispute resolution** as well as for verification