Rethinking Digital Signatures

Ueli Maurer

ETH Zurich

Rethinking Digital Signatures
Or: New Approaches to Digital Evidence

Ueli Maurer

ETH Zurich

Overview

- Digital signatures and certificates
- Some provocative claims
- Digital evidence: A systematic treatment
- Fundamental dilemma in DS legislation
- Justification of claims
- The role of conventional signatures
- Digital declarations
Digital signatures

\[ \text{sign} (\mathcal{O}, m) = \text{versig} (\mathcal{O}, m) = \begin{cases} 1 \text{ (yes)} \\ 0 \text{ (no)} \end{cases} \]
Public-key certificates

CA $C$ confirms the **binding** of public key $p_A$ to entity $A$.

\[
\text{sign} \quad \text{Certificate} \quad C \quad \text{when checked with public key } \quad p
\]

Certificate $c$ when checked with public key $p$:

\[
\begin{align*}
\text{id}(p, c) &= \text{identity} \\
\text{pk}(p, c) &= \text{public key} \\
\text{exp}(p, c) &= \text{expiration time} \\
\text{lia}(p, c) &= \text{liability bound}
\end{align*}
\]
Context

Is an entity $A$ liable for a digital document $d$, as a consequence of a digital signature on $d$ with respect to her public key $p_A$?
Is an entity $A$ liable for a digital document $d$, as a consequence of a digital signature on $d$ with respect to her public key $p_A$?

$A$’s possible objections:

1. $p_A$ is not my public key.
Context

Is an entity $A$ liable for a digital document $d$, as a consequence of a digital signature on $d$ with respect to her public key $p_A$?

$A$’s possible objections:

1. $p_A$ is not my public key.
2. I did not sign $d$ (though $p_A$ is my valid public key).
Is an entity $A$ liable for a digital document $d$, as a consequence of a digital signature on $d$ with respect to her public key $p_A$?

$A$’s possible objections:

1. $p_A$ is not my public key.
2. I did not sign $d$ (though $p_A$ is my valid public key).
3. The signature was generated after I revoked $p_A$. 
**Context**

Is an entity $A$ liable for a digital document $d$, as a consequence of a digital signature on $d$ with respect to her public key $p_A$?

$A$’s possible objections:

1. $p_A$ is not my public key.
2. I did not sign $d$ (though $p_A$ is my valid public key).
3. The signature was generated after I revoked $p_A$.
4. I am liable for $p_A$, but only for transaction values below that relevant in document $d$. 
Digital signatures: Promises
Digital signatures: Promises

Automation and digitization of many business and government processes!
Digital signatures: Promises

Automation and digitization of many business and government processes!

- Easy to transmit, archive, search, and verify
- Unambiguous: Verification = math. function
- Higher security than conventional signatures
- Simpler dispute resolution
- Fewer disputes
Digital signatures: Obstacles

- Non-repudiation services: Only isolated use of DS
- Lack of international PKI
- Lack of internationally applicable legislation
- Lack of standardization
- Difficult integration into business processes
- Technological challenges
- Slow user acceptance
- Abstractness and complexity
Digital signatures: Obstacles

- Non-repudiation services: Only isolated use of DS
- Lack of international PKI
- Lack of internationally applicable legislation
- Lack of standardization
- Difficult integration into business processes
- Technological challenges
- Slow user acceptance
- Abstractness and complexity
- **Lack of understanding**
Some provocative claims

- Certificates are (generally) irrelevant as evidence in a dispute.
Some provocative claims

- Certificates are (generally) irrelevant as evidence in a dispute.

- Roles of certification and time-stamping authorities are very different.
Some provocative claims

- Certificates are (generally) irrelevant as evidence in a dispute.
- Roles of certification and time-stamping authorities are very different.
- Public keys and certificates cannot be revoked.
Some provocative claims

- Certificates are (generally) irrelevant as evidence in a dispute.

- Roles of certification and time-stamping authorities are very different.

- Public keys and certificates cannot be revoked.

- Evidence expires, not public keys.
Possible objections

- Judges are humans and cannot be formalized. They make reasonable decisions based on all the available evidence.
Possible objections

- Judges are humans and cannot be formalized. They make reasonable decisions based on all the available evidence.

- This view does not conform with the current laws and is therefore not relevant in practice.
Possible objections

- Judges are humans and cannot be formalized. They make reasonable decisions based on all the available evidence.

- This view does not conform with the current laws and is therefore not relevant in practice.

- These observations are trivialities.
Possible objections

- Judges are humans and cannot be formalized. They make reasonable decisions based on all the available evidence.

- This view does not conform with the current laws and is therefore not relevant in practice.

- These observations are trivialities.

- This is complete nonsense.
Bitstrings as evidence: Example

**Digital check:** A authorizes bank B to pay $100 from her account to anyone (the first) who presents a certain bitstring $c$. 
Bitstrings as evidence: Example

**Digital check:** A authorizes bank B to pay $100 from her account to anyone (the first) who presents a certain bitstring $c$.

How is $c$ specified?
Bitstrings as evidence: Example

**Digital check:** A authorizes bank B to pay $100 from her account to anyone (the first) who presents a certain bitstring $c$.

How is $c$ specified? By a verification predicate

$$v : \{0, 1\}^* \rightarrow \{0, 1\}.$$
Bitstrings as evidence: Example

**Digital check**: A authorizes bank B to pay $100 from her account to anyone (the first) who presents a certain bitstring $c$.

How is $c$ specified? By a **verification predicate**

$$v : \{0, 1\}^* \rightarrow \{0, 1\}.$$  

A is liable if a bitstring $s$ with $v(s) = 1$ is presented.
Bitstrings as evidence: Example

Digital check: A authorizes bank B to pay $100 from her account to anyone (the first) who presents a certain bitstring \(c\).

How is \(c\) specified? By a verification predicate

\[
v : \{0, 1\}^* \rightarrow \{0, 1\}.
\]

A is liable if a bitstring \(s\) with \(v(s) = 1\) is presented.

Realization based on a one-way function \(f\):
Let \(y := f(c)\).

\[
v(s) = 1 \iff f(s) = y
\]
General document space $\mathcal{D}$

A wants to be able to authorize an arbitrary transaction, described by document $d \in \mathcal{D}$.

Verification predicate: $v : \mathcal{D} \times \{0, 1\}^* \rightarrow \{0, 1\}$

String $s$ implies liability for $d$ if and only if $v(d, s) = 1$. 
General document space $\mathcal{D}$

A wants to be able to authorize an arbitrary transaction, described by document $d \in \mathcal{D}$.

Verification predicate: $v : \mathcal{D} \times \{0, 1\}^* \rightarrow \{0, 1\}$

String $s$ implies liability for $d$ if and only if $v(d, s) = 1$.

Requirements:

- **Security:** Infeasible to find $d$ and $s$ with $v(d, s) = 1$.
- **Efficient verifiability:** Check if $v(d, s) = 1$.
- **Feasibility:** For any $d \in \mathcal{D}$, A can efficiently compute $c_d$ with $v(d, c_d) = 1$. 
Digital signatures

\[ \text{sign}(\cdot, \cdot) = \text{versig}(\cdot, \cdot) = \{1 \text{ (yes)}, 0 \text{ (no)} \} \]

\[ v(d, s) = \text{versig}(p_A, d, s). \]
Using certificates

Assume CA $C$’s public key $p_C$ is publicly known.

\[
v(d, s) = s = [\sigma, c] \\
\land \ id(p_C, c) = A \\
\land \ versig(pk(p_C, c), d, \sigma)
\]
Hierarchical certification

Assume root-CA $R$’s public key $p_R$ is publicly known.

$$v(d, s) = s = [\sigma, c, c']$$
$$\land \text{versig}(pk(pk(p_R, c'), c), d, \sigma)$$
$$\land \text{id}(pk(p_R, c'), c) = A$$
Certificate expiration and time-stamping

Assume:

- CA $C$’s public key $p_C$ known.
- Time-stamping authority $T$’s public key $p_T$ known.

\[
v(d, s) = s = [\sigma, c, \tau] \\
\land \text{id}(p_C, c) = A \\
\land \text{versig}(pk(p_C, c), d, \sigma) \\
\land \text{time}(p_T, \tau) \leq \text{exp}(p_C, c) \\
\land \text{string}(p_T, \tau) = \sigma
\]
Certificate revocation

Two mechanisms:

- Certificate revocation list (CRL)
- On-line revalidation: revalidation certificate $r$

Two additional checks:

\[
\text{time}(p_T, \tau) \leq \text{time}(p_C, r) + \Delta
\]

\[
\text{pk}(p_C, c) = \text{pk}(p_C, r)
\]
Dilemma in DS legislation

What implies liability?
Dilemma in DS legislation

What implies liability?

1. Digital evidence?
   - Secret key could have leaked.
   - System vulnerability (e.g. a virus).
   - User interface ambiguous.
   - Cryptographic signature function broken.
   - False certificate.
Dilemma in DS legislation

What implies liability?

2.Willful act?

– Digital signature is only one piece of evidence.
– Which other evidence is considered?
– How can a user prove she did not sign?
– Should the other party present more than digital evidence?
Dilemma in DS legislation

What implies liability?

2. Willful act?
   - Digital signature is only one piece of evidence.
   - Which other evidence is considered?
   - How can a user prove she did not sign?
   - Should the other party present more than digital evidence?

Fundamental dilemma: It cannot be both!
Entering a contract

- Basic act in business and society
- Valid only if entered by both parties
- Requires each parties’ consent, documented by a willful act
- Entities keep some evidence of willful act
- Legal system defines what constitutes valid evidence
Evidence for non-repudiation

- Physical evidence
Evidence for non-repudiation

- Physical evidence
- Statements by witnesses
Evidence for non-repudiation

- Physical evidence
- Statements by witnesses
- Digital evidence
Evidence for non-repudiation

- Physical evidence
- Statements by witnesses
- Digital evidence
  - Digital evidence strings (signatures, certificates, time stamps, revalidation certificates, ...)
Evidence for non-repudiation

- Physical evidence

- Statements by witnesses

- Digital evidence
  - Digital evidence strings (signatures, certificates, time stamps, revalidation certificates, ...)
  - Digital recordings of physical world; have human interpretation
Requirements for contract signing systems and legislation

- Practicality
- Unambiguity
- Security
- Low cost
- Low trust requirements
- Precise and simple legislation
- Smooth integration
- Wide usability and acceptance
Abstraction of the legal system

- Legal system = rules used to make decision
- Includes legislation and juridical practice
- Separation of ambiguous and unambiguous issues
Abstraction of the legal system

- Legal system = rules used to make decision
- Includes legislation and juridical practice
- Separation of ambiguous and unambiguous issues
- Unambiguous description of evidence: $e$
Abstraction of the legal system

- Legal system = rules used to make decision
- Includes legislation and juridical practice
- Separation of ambiguous and unambiguous issues
- Unambiguous description of evidence: $e$
- Ambiguous: - Interpretation of document $d$
  - Does given evidence match $e$?
Abstraction of the legal system

- Legal system = rules used to make decision
- Includes legislation and juridical practice
- Separation of ambiguous and unambiguous issues
- Unambiguous description of evidence: $e$
- Ambiguous:  - Interpretation of document $d$
  - Does given evidence match $e$?
- Legal system can be abstracted as a function
  $$\text{evidence} \rightarrow \{0, 1\}$$
Liability function

$$\lambda : \mathcal{I} \times \mathcal{D} \times \mathcal{E} \times \mathcal{V} \rightarrow \{0, 1\}$$

$\mathcal{I}$ = entity name space
$\mathcal{E}$ = space of evidence descriptions
$\mathcal{V}$ = set of predicates $\mathcal{D} \times \{0, 1\}^* \rightarrow \{0, 1\}$

A is liable for $d$ if:

1. $\lambda(A, d, e, v) = 1$
2. Evidence satisfying description $e$ is presented.
3. A bitstring $s$ satisfying $v(d, s) = 1$ is presented.
Delegation signatures

In order to make forgery of $s$ more difficult, one requires one (or more) additional signature as evidence:

$$v(d, s) = s = [\sigma, \sigma']$$

$$\land \text{versig}(p, d, \sigma)$$
$$\land \text{versig}(p', \sigma, \sigma')$$

$p'$ is controlled by a party trusted (and chosen) by A.
Possible semantics of certificates
Possible semantics of certificates

1. Certificate proves that $p_A$ is $A$’s public key.
Possible semantics of certificates

1. Certificate proves that $p_A$ is A’s public key.

2. Certificates states that the CA holds evidence for the fact that A committed herself to $p_A$. 
Possible semantics of certificates

1. Certificate proves that $p_A$ is A’s public key.

2. Certificates states that the CA holds evidence for the fact that A committed herself to $p_A$.
   - Certificate is irrelevant for the legal system.
   - Role of CA: manage physical evidence and witnesses.
   - Only recipient of signature, not A, must trust the CA.
   - Lower security requirements for CA.
   - New type of trusted entity $\rightarrow$ new business models.
   - Name “certificate”? 
Commitment to verification predicate

- A user declares her commitment to a verification predicate, not a public key.

- The legal system defines which type of (physical) commitment declaration is required for which type of liability.

- Multi-level declarations possible.
Time-stamping makes little sense!

- If $\lambda(A, d, e, \nu) = 1$ and evidence matching description $e$ is presented, then an arbitrary bitstring $s$ with $\nu(d, s) = 1$ proves liability.
Time-stamping makes little sense!

- If $\lambda(A, d, e, v) = 1$ and evidence matching description $e$ is presented, then an arbitrary bitstring $s$ with $\nu(d, s) = 1$ proves liability.

- Irrelevant when, where, how, or by whom $s$ was generated.
Time-stamping makes little sense!

- If $\lambda(A, d, e, v) = 1$ and evidence matching description $e$ is presented, then an arbitrary bitstring $s$ with $\nu(d, s) = 1$ proves liability.

- Irrelevant when, where, how, or by whom $s$ was generated.

- Interpretation: A time stamp is a special type of delegation signature.
Time-stamping makes little sense!

• If $\lambda(A, d, e, v) = 1$ and evidence matching description $e$ is presented, then an arbitrary bitstring $s$ with $v(d, s) = 1$ proves liability.

• Irrelevant when, where, how, or by whom $s$ was generated.

• Interpretation: A time stamp is a special type of delegation signature.

• Interpretation of expiration: Evidence expires, not public keys.
**Time-stamping makes little sense!**

- If \( \lambda(A, d, e, v) = 1 \) and evidence matching description \( e \) is presented, then an arbitrary bitstring \( s \) with \( v(d, s) = 1 \) proves liability.

- Irrelevant when, where, how, or by whom \( s \) was generated.

- Interpretation: A time stamp is a special type of delegation signature.

- Interpretation of expiration: Evidence expires, not public keys.

- Revocation is impossible.
Time-stamping makes little sense!

- If $\lambda(A, d, e, v) = 1$ and evidence matching description $e$ is presented, then an arbitrary bitstring $s$ with $v(d, s) = 1$ proves liability.

- Irrelevant when, where, how, or by whom $s$ was generated.

- Interpretation: A time stamp is a special type of delegation signature.

- Interpretation of expiration: Evidence expires, not public keys.

- Revocation is impossible.

- Revalidation certificate = delegation signature.
The role of conventional signatures

- Conventional hand-written signatures work amazingly well.
- Signatures are quite easy to forge.
- Purpose of signature: Guaranteed user awareness.
- Meaningful to ask user to testify whether she signed.
- In sharp contrast to digital signatures.
- How can we achieve the same (or better) situation with digital evidence?
Digital declarations

- Digital recording of the user’s willful act.

- Examples: voice, image, video, any other technology.

- Human interpretation of recording.

- User can request a digital declaration to be presented.

- Forgery can be denied.
Usefulness of digital declarations

- Guaranteed user awareness.
- Higher deterrence of misbehavior, fewer disputes.
- Improved security compared to conventional signatures.
- Lower cost due to reduced technical security requirements.
- Improved user acceptance of digital signature technology.
- Usability by moderately educated people.