Eleven Principles and Why Cryptosystems Fail

Sape J. Mullender
Huygens Systems Research Laboratory
Universiteit Twente
Enschede
Abadi and Needham’s Eleven Principles

See SRC Report 125: *Prudent Engineering Practice for Cryptographic Protocols*, by Martín Abadi and Roger Needham
Principle 1 – Explicit Communication

*Every message should say what it means*

The interpretation should only depend on its content.
Principle 2 – Appropriate Action

If you believe that only a trusted server can choose good keys, you should not be using the Wide-Mouthed Frog protocol.

The condition for a message to be acted upon should be clearly set out so that someone reviewing a design may see whether they are acceptable or not.
Bob wants to verify it is in Alice’s presence:

1. $A$
2. $N_B$
3. $\{N_B\}^{}_{K_{AT}}$
4. $\{A, \{N_B\}^{}_{K_{AT}}\}^{}_{K_{BT}}$
5. $\{N_B\}^{}_{K_{BT}}$
Breaking Woo and Lam

1. 😞 ➔ 🎖 A
1’. 😞 ➔ 🎖 M
2. 😞 ➔ 🎖 NB
2’. 😞 ➔ 🎖 NB’
3. 😞 ➔ 🎖 \{NB\}_{KMT}
3’. 😞 ➔ 🎖 \{NB\}_{KMT}
4. 😞 ➔ 🎖 \{A, \{NB\}_{KMT}\}_{KBT}
4’. 😞 ➔ 🎖 \{M, \{NB\}_{KMT}\}_{KBT}
5. 😞 ➔ 🎖 \{NB’’\}_{KBT}
5’. 😞 ➔ 🎖 \{NB\}_{KBT}
If the identity of a principal is essential to the meaning of a message, it is prudent to mention the principal’s name explicitly in the message.
1. \[ \begin{array}{c|c}
   & A, B \\
   \end{array} \]

2. \[ \begin{array}{c|c}
   & \{T_T, L, K_{AB}, B, \{T_T, L, K_{AB}, A\}_{K_{BT}}\}_{K_{AT}} \\
   \end{array} \]

3. \[ \begin{array}{c|c}
   & \{T_T, L, K_{AB}, A\}_{K_{BT}}, \{A, T_A\}_{K_{AB}} \\
   \end{array} \]

4. \[ \begin{array}{c|c}
   & \{T_A + 1\}_{K_{AB}} \\
   \end{array} \]

---

Kerberos

---

8
Principle 4 – Encryption

Be clear about why encryption is done

Encryption is not cheap and it is not synonymous with security.
1. $A, \{T_A, N_A, B, X_A, \{Y_A\}_{K_B}\}_{K_A^{-1}}$

$X_A$ and $Y_A$ are data. The protocol is intended to ensure the integrity $X_A$ and $Y_A$, assuring the recipient of their origin, and to guarantee the privacy of $Y_A$. 
1. $\{X\}_{K_B}, \{H(X)\}_{K_A^{-1}}$

The idea is that $A$ supports $X$ by signing its hash. But there is no evidence that $A$ knows $X$. It could have received $H(X)$ from a friend.
Principle 5 – Signing encrypted data

When a principal signs material that has already been encrypted it should not be inferred that the principal knows the contents of the message. On the other hand, it is proper to infer that the principal that signs a message and then encrypts it for privacy knows the contents of that message.
1. $M, A, B, \{N_A, M, A, B\}_{K_{AT}}$

2. $M, A, B, \{N_A, M, A, B\}_{K_{AT}}, \{N_B, M, A, B\}_{K_{BT}}$

3. $M, \{N_A, K_{AB}\}_{K_{AT}}, \{N_B, K_{AB}\}_{K_{BT}}$

4. $M, \{N_A, K_{AB}\}_{K_{AT}}$

In the BAN-Logic analysis, the authors claimed that encryption of $N_B$ in message 2 was unnecessary.
Mao and Boyd since then showed that it is essential: they bind Messages 1 and 2 to Messages 3 and 4. They *name* $A$ and $B$ in Trent’s reply. Here’s an alternative way to do this:

1. $\begin{array}{c} \includegraphics[width=0.1\textwidth]{icon1} \\ \rightarrow \\ \includegraphics[width=0.1\textwidth]{icon2} \end{array}$ $A, B, N_A$

2. $\begin{array}{c} \includegraphics[width=0.1\textwidth]{icon3} \\ \rightarrow \\ \includegraphics[width=0.1\textwidth]{icon4} \end{array}$ $A, B, N_A, N_B$

3. $\begin{array}{c} \includegraphics[width=0.1\textwidth]{icon5} \\ \rightarrow \\ \includegraphics[width=0.1\textwidth]{icon6} \end{array}$ $\{N_A, A, B, K_{AB}\}_{K_{AT}}, \{N_B, A, B, K_{AB}\}_{K_{BT}}$

4. $\begin{array}{c} \includegraphics[width=0.1\textwidth]{icon7} \\ \rightarrow \\ \includegraphics[width=0.1\textwidth]{icon8} \end{array}$ $\{N_A, A, B, K_{AB}\}_{K_{AT}}$
Be clear what properties you are assuming about nonces. What may do for ensuring temporal succession may not do for ensuring association.

Perhaps association is best achieved by other means.
Principle 7 – Nonces

The use of a predictable quantity (such as the value of a counter) can serve in guaranteeing newness through a challenge-response exchange. But if a predictable quantity is to be effective, it should be protected so that an intruder cannot simulate a challenge and later replay a response.
If timestamps are used as freshness guarantees by reference to absolute time, then the difference between local clocks on various machines must be less than the allowable age of a message deemed to be valid. Furthermore, the time maintenance mechanism everywhere becomes part of the trusted computing base.
Needham-Schroeder

1. $\rightarrow A, B, N_A$
2. $\rightarrow \{N_A, B, K_{AB}, \{A, K_{AB}\}_K_{BT}\}_K_{AT}$
3. $\rightarrow \{A, K_{AB}\}_K_{BT}$
4. $\rightarrow \{N_B\}_K_{AB}$
5. $\rightarrow \{N_B - 1\}_K_{AB}$
The Needham-Schroeder Protocol is Flawed

When Bob receives message 3, he has no way of telling how fresh it is.

After a year of cryptanalysis, Mallory breaks one of Alice and Bob’s old keys. He can then do this:

3. 😛 ➔ 😻 🍭
   \( \{A, K_{AB}\}_{K_{BT}} \) (replay)

4. 😢 ➔ 🙆‍♀️ 😁
   \( \{N'_B\}_{K_{AB}} \)

5. 😞 ➔ 😻
   \( \{N'_B - 1\}_{K_{AB}} \)
Principle 9 – Use vs. generation

A key may have been used recently, for example to encrypt a nonce, yet be quite old, and possibly compromised. Recent use does not make the key look better than it would otherwise.
4. \begin{align*}
\text{\includegraphics[height=0.5cm]{expression1.png}} & \rightarrow \begin{tikzpicture}
\node [circle,fill,inner sep=0.5cm] at (0cm,0cm) {}; \end{tikzpicture} \\
\{N_B\}_K_{AB}
\end{align*}

5. \begin{align*}
\begin{tikzpicture}
\node [circle,fill,inner sep=0.5cm] at (0cm,0cm) {}; \end{tikzpicture} & \rightarrow \begin{tikzpicture}
\node [circle,fill,inner sep=0.5cm] at (0cm,0cm) {}; \end{tikzpicture} \\
\{N_B - 1\}_K_{AB}
\end{align*}

The purpose of decrementing $N_B$ is to distinguish it from the previous message.
3. \[ \{T_T, L, K_{AB}, A\}_{K_{BT}}, \{A, T_A\}_{K_{AB}} \]

4. \[ \{T_A + 1\}_{K_{AB}} \]

Here, incrementing $T_A$ serves no purpose.
A better alternative in Needham-Schroeder would have been:

4. 😒 🧐 → 👨‍🦰 👨‍🦰 \(\{NS4 : N_B\}_{K_{AB}}\)

5. 👨‍🦰 👨‍🦰 → 😒 😎 \(\{NS5 : N_B\}_{K_{AB}}\)
If an encoding is to be used to present the meaning of a message, then it should be possible to tell which encoding is being used. In the common case where the encoding is protocol independent, it should be possible to deduce that the message belongs to this protocol, and in fact to a particular run of the protocol, and to know its number in the protocol.
1. Alice $\rightarrow$ Bob: $A, \{T_A, B, K_{AB}\}_{K_{AT}}$

2. Bob $\rightarrow$ Eve: $\{T_S, A, K_{AB}\}_{K_{BT}}$

Alice is trusted to choose the session key.
Everyone must trust their own clock as well as Trent’s.
The protocol designer should always know which trust relations his protocol depends on, and why the dependence is necessary. The reason for particular trust relations being acceptable should be explicit though they will be founded on judgment and policy rather than logic.
The recent case of R v Munden describes one of Cambridge’s local police constables coming home from holiday to find his bank account empty; he asked for a statement, found six withdrawals for a total of £460 which he did not recall making, and complained to his bank. It responded by having him prosecuted for attempting to obtain money by deception.
• The bank had no security management or quality assurance function. The software was developed by ‘code & fix’; production code changed as often as twice a week.

• There had never been external audits of the system; the manager who gave technical evidence was the man who built the system 20 years earlier and still ran it. He claimed that system programmers could not access encryption keys in source code.

• The disputed transactions were not properly investigated; the manager looked at the mainframe logs and found nothing wrong with them. Another 150–200 transactions under dispute had not been investigated either.
Nevertheless, police constable Munden was convicted.
An appeal was brought about, in part thanks to the security research community which rallied round in support of Munden’s defense.

The case returned to court and was dismissed after the bank refused to disclose its logs and allow an audit of their system.

The lessons ... ?
Mathematical proof and Legal proof

They are not the same. At best they are unrelated, at worst they are contradictory.

Secure systems, especially financial ones, require proper embedding in a legal framework.
How could PC Munden lose his money?

The cryptographic algorithms used in electronic banking are strong enough — not a single case is known where a cryptanalytic attack was successful; it is moot whether a serious cryptanalytic attack has ever been attempted.

PC Munden lost his money as a consequence of bad key management, corrupt bank employees, or by letting somebody see his PIN.
Things Banks Have Done

- Put full account numbers on ATM tickets
- Fail to encrypt authorization responses
- Make postal theft really convenient
- Put the 14-digit ATM test sequence in the branch manual
- Write the encrypted PIN on the card
- Keep encrypted PINS on line
- Reduce the PIN space...
• A bank used a scheme for off-line checking: \( d_1 + d_3 = d_2 + d_4 \)

• Banks have handed out memory cards (PIN = 2256):

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This nicely reduces the probability of discovery from 1 in 3000 to 1 in 8.

• One bank issued the same PIN to all customers

• A programmer made sure the bank only issued three different PINs
Things Bank Employees Have Done

• Issued extra cards to themselves
• Fit ATMs with organizers
• The equivalent has been done by a fuel station attendant
• Set up test equipment to use the master keys and charge the underworld for the PINS they calculated
• Real managers don’t type

Employees often rely on the fact that most banks will ignore customer complaints
Things Crooks Have Done

- Extort PINS
- Exploit naive customers to get PIN
- Simultaneous plundering of ATMs that are off line (Italy!)
- Build fake ATMs
The Common Thread?

Most successful attacks were not sophisticated at all.
None of them attacked the cryptosystems
Clumsy management and faulty procedures are often at the root of attacks.
Secure Systems Design

Expect the real problems to come from blunders in the application design and in the way the system is operated
Every ATM or bank branch that can autonomously verify PINs needs cryptographic equipment and a secret key. The number of copies of a secret key is therefore large.

Banks have used *security modules* to guard keys. These are tamperproof cryptographic boxes that guard the key. But many banks have ‘software implementations’ of such boxes. System programmers are quite good at finding the keys.
Banks don’t like to expose their dirty laundry. They keep their mistakes and stupidities secret. As a consequence, many banks have fallen for the same trap.

Banks could learn from the airline industry. Every accident is investigated. The result is always made public. Often, the outcome of an investigation leads to immediate ‘airworthiness directives’

Cases in point: engine mounting pins on 747s after the El-Al 1862 crash in Amsterdam; insulation of wires that go through the tanks on many Boeing models after the TWA 800 crash over Long Island; reverse thuster deployment in the Lauda Air disaster.
In *R v Hendy*, Norma Hendy was accused of stealing money from a colleague (she knew the colleague’s PIN). The bank claimed it was not possible to withdraw money without the PIN. The defense proved this to be not true by revealing the bank’s record of complaints. The defense demanded that the bank’s records that were used as proof against the defendant would be subject to independent audit. The bank refused and Norma was acquitted.
When accused of electronic bank fraud, one should ask the prosecution to produce a full set of the bank’s security and quality documentation, including security policies and standards, crypto key management procedures and logs, audit and insurance inspectors’ reports, test and bug reports, ATM balancing records and logs, and details of all customer complaints of the last seven years. Banks often cannot or will not comply.
Security systems which are to provide evidence must be designed and certified on the assumption that they will be examined in detail by a hostile expert.
British banks claim that their systems are infallible: it is not possible for an ATM debit to appear on someone’s account unless the card and PIN issued to him had been used in that ATM.

People who complain are routinely told that they must be lying, mistaken, or the victim of fraud by a friend or relative (which makes them negligent).
In the case of Judd vs. Citibank, Dorothy Judd claimed that she had not made a number of ATM withdrawals that appeared on her account; Citibank claimed that she must have done.

The judge ruled that Citibank was wrong in law to claim that its systems were infallible, as this placed an unmeetable burden of proof on the plaintiff.

Since then, upon a dispute by a client, US banks must refund within 30 days unless they can prove that the claim is attempted fraud.
Burden of Proof

The Bank says: ‘You made that withdrawal!’ The Customer says: ‘I did no such thing!’

If the customer has to prove where the bank went wrong, he hasn’t got a chance. If the bank must establish the security of its system in the face of hostile expert witnesses, it hasn’t got a chance.
Maybe the purpose isn’t security, but merely due diligence: limit the damage crooks can cause to acceptable levels.
Before setting out to build a computer security system, make sure you understand what its real purpose is (especially if this differs from its advertised purpose)
One should never use encryption without understanding what it is for. (Abadí & Needham, 1994)