Formal Computational Unlinkability Proofs of RFID Protocols

Hubert Comon, Adrien Koutsos

January 29, 2018

Motivation

Security protocols

Distributed programs which aim at providing some security properties.

RFID protocol

Reader + **Tags** (low computational power and low memory)

The KCL Authentication Protocol

$$R \quad : \quad n_R \xleftarrow{\$}$$
$$T_A \quad : \quad n_T \xleftarrow{\$}$$

$$1 : R \longrightarrow T_A \quad : \quad n_R$$
$$2 : T_A \longrightarrow R \quad : \quad \langle A \oplus n_T \,,\, n_T \oplus H(n_R, k_A) \rangle$$

Security properties

- Security protocols are often very short: few lines of code
- Security properties are very complex: non-secured network + active attacker

$\Rightarrow$ Need to use formal methods to verify security protocols

Our problem

$$\forall \mathcal{A} \in \mathcal{C} \qquad P_{\mathcal{A}} \models \phi$$

Motivations

Dolev-Yao model

- Symbolic model, we work on terms in some algebra.
- Simple model: we specify all that the adversary can do.
- Well-suited for proof automation (ProVerif, Tamarin, APTE . . .).
- Can automatically find attacks.

Problem

This model is not very close to a real-world attacker.

Computational model

- More realistic model: we work on bit-strings.
- The adversaries are any probabilistic polynomial time Turing Machine.

Motivations

Problems of the computational model

- Proofs are long, complicated and errors prone.
- Very little proof automation (EasyCrypt, CryptoVerif . . .).
- Implicit hypothesis that may be wrong.

The Complete Symbolic Attacker model

- All hypothesis appear explicitly in the axioms.
- Proof automation.
- Attackers beyond the computational model.

1. Motivations

2. The Complete Symbolic Attacker Model
   - Syntax
   - Computational semantics

3. Axioms
   - Structural Axioms
   - Pseudo Random Function

4. KCL protocol
   - Security proofs

1 Motivations

2 The Complete Symbolic Attacker Model
- Syntax
- Computational semantics

3 Axioms
- Structural Axioms
- Pseudo Random Function

4 KCL protocol
- Security proofs

Syntax

Term algebra

The terms are build over:

- function symbols $\texttt{if\_then\_else\_}$, $\mathsf{EQ}(;)$, true, false
- a set of function symbols Σ with arities.
  Example: $\Sigma = \{\langle\_,\_\rangle, \pi_1(\_), \pi_2(\_), \mathsf{H}(\_)\}$
- a set of function symbols $\mathcal{G}$ with arities.
- a countable set of names $\mathcal{N}$.
- a countable set of variable symbols $\mathcal{X}$.

Formulas

$\phi ::= \vec{u} \sim \vec{v} \mid \phi \wedge \phi \mid \neg\phi \mid \bot \mid \forall x.\phi$ where $\vec{u}, \vec{v}$ are terms

Example

The KCL protocol

$$\begin{aligned} 1 : R \longrightarrow T_A &\quad : \quad n_R \\ 2 : T_A \longrightarrow R &\quad : \quad \langle A \oplus n_T \,,\, n_T \oplus H(n_R, k_A) \rangle \end{aligned}$$

Example

- **Terms:**

$$m_A = \langle A \oplus n_T \,,\, n_T \oplus H(g(n_R), k_A) \rangle$$

- **Formula:**

$$n_R, m_A \sim n_R, m_B$$

Computational semantics of terms

Computational Model $\mathcal{M}_c$

- $f_{/n} \in \Sigma \cup \mathcal{G}$ interpreted as a polynomial time Turing Machine.
- $\mathsf{n} \in \mathcal{N}$ interpreted as a random sampling
- $\{\texttt{if\_then\_else\_}, \mathsf{EQ}(;), \mathsf{true}, \mathsf{false}\}$ interpretations are the expected ones.

Ground terms

Ground terms are interpreted as *probabilistic* polynomial time Turing Machine.

Computational semantics of formulas

Predicate Interpretation in $\mathcal{M}_c$

$\mathcal{M}_c \models \vec{u} \sim \vec{v}$ iff for any probabilistic polynomial time Turing Machine $\mathcal{A}$

$$\left| Pr\left[\mathcal{A}(\llbracket \vec{u} \rrbracket_{\mathcal{M}_c}(1^\eta)) = 1\right] - Pr\left[\mathcal{A}(\llbracket \vec{v} \rrbracket_{\mathcal{M}_c}(1^\eta)) = 1\right] \right|$$

is negligible in η.

Example

For every computational model $\mathcal{M}_c$ we have:

$$\mathcal{M}_c \models \mathsf{A} \oplus \mathsf{n}_1 \sim \mathsf{B} \oplus \mathsf{n}_2$$

Proof Technique

Goal

Formula $\vec{u} \sim \vec{v}$ expressing the security of the protocol.
(obtained by folding the executions of the protocol)

Axioms $\mathcal{A}$

Computationally valid inferences rules:

- "structural" axioms: always true.
- implementation axioms: cryptographic assumptions ...

Proof Technique

Find axioms allowing us to get a proof derivation of the goal.

1 Motivations

2 The Complete Symbolic Attacker Model
- Syntax
- Computational semantics

3 Axioms
- Structural Axioms
- Pseudo Random Function

4 KCL protocol
- Security proofs

Structural Axioms

Relation Axioms

$$\frac{}{x \sim x}\ Refl \qquad \frac{x \sim y}{y \sim x}\ Sym \qquad \frac{x \sim y \quad y \sim z}{x \sim z}\ Trans$$

$\sim$ is not a congruence!

Counter-Example: $\mathsf{n} \sim \mathsf{n}$ and $\mathsf{n} \sim \mathsf{n}'$, but $\mathsf{n}, \mathsf{n} \not\sim \mathsf{n}, \mathsf{n}'$.

Function Application

If you cannot distinguish the arguments, you cannot distinguish the images.

$$\frac{x_1, \ldots, x_n \sim y_1, \ldots, y_n}{f(x_1, \ldots, x_n) \sim f(y_1, \ldots, y_n)}\ FunApp$$

Structural Axioms

Congruence

If $EQ(u, v) \sim$ true then u and v are (almost always) *equal*
$\Rightarrow$ we have a congruence.

$u = v$ syntactic sugar for $EQ(u, v) \sim$ true

Equational Theory

- `if` y `then` x `else` $x = x$
- $\pi_i(\langle x_1, x_2 \rangle) = x_i$ $\quad i \in \{1, 2\}$ x

Pseudo Random Function

Definition

H is a *Pseudo Random Function* family if for any PPT adversary $\mathcal{A}$:

$$|\mathbf{Pr}(\mathsf{k} : \ \mathcal{A}^{\mathcal{O}_{\mathsf{H}(\cdot,\mathsf{k})}}(1^\eta) = 1) - \mathbf{Pr}(g : \ \mathcal{A}^{\mathcal{O}_{g(\cdot)}}(1^\eta) = 1)|$$

is negligible in η, where:

- k is drawn uniformly in $\{0,1\}^\eta$.
- g is drawn uniformly in the set of all functions from $\{0,1\}^*$ to $\{0,1\}^\eta$.

Translation in the Logic

Bad Axiom

If t and t' are *syntactically* distinct,

$$\mathsf{H}(t, \mathsf{k}), \mathsf{H}(t', \mathsf{k}) \sim \mathsf{H}(t, \mathsf{k}), \mathsf{n}$$

Counter-Example: $t = g(a)$ and $t' = g(a')$, a, a' distinct and g an attacker function.
There exists a model falsifying the formula (e.g. g interpreted as a constant function).

Translation in the Logic

The PRF_2 Axioms

$$\vec{u}, \texttt{if } \mathsf{EQ}(t; t_1) \texttt{ then } \mathbf{0} \texttt{ else } \mathsf{H}(t, \mathsf{k})$$
$$\sim$$
$$\vec{u}, \texttt{if } \mathsf{EQ}(t; t_1) \texttt{ then } \mathbf{0} \texttt{ else } \mathsf{n}$$

where:

- the only occurrences of H (and k) in $\vec{u}, t$ are $\mathsf{H}(t_1, \mathsf{k})$
- n is a name that does not occur in $\vec{u}, t, t_1$

1 Motivations

2 The Complete Symbolic Attacker Model
- Syntax
- Computational semantics

3 Axioms
- Structural Axioms
- Pseudo Random Function

4 KCL protocol
- Security proofs

Attack on KCL [2]

$$\begin{array}{lll|l}
R \to T_{\mathsf{A}} : & \mathsf{n_R} & & \mathsf{n_R} \\
T_{\mathsf{A}} \to R : & \langle \mathsf{A} \oplus \mathsf{n_T}, \mathsf{n_T} \oplus \mathsf{H}(\mathsf{n_R}, \mathsf{k_A})\rangle & & \langle \mathsf{A} \oplus \mathsf{n_T}, \mathsf{n_T} \oplus \mathsf{H}(\mathsf{n_R}, \mathsf{k_A})\rangle \\
E \to (T_{\mathsf{A}} | T_{\mathsf{B}}) : & \mathsf{n_R} & & \mathsf{n_R} \\
(T_{\mathsf{A}} | T_{\mathsf{B}}) \to R : & \langle \mathsf{A} \oplus \mathsf{n'_T}, \mathsf{n'_T} \oplus \mathsf{H}(\mathsf{n_R}, \mathsf{k_A})\rangle & & \langle \mathsf{B} \oplus \mathsf{n'_T}, \mathsf{n'_T} \oplus \mathsf{H}(\mathsf{n_R}, \mathsf{k_B})\rangle
\end{array}$$

Algebraic property

$$\begin{aligned}
\mathsf{A} \oplus \mathsf{n_T} \oplus \mathsf{n_T} \oplus \mathsf{H}(\mathsf{n_R}, \mathsf{k_A}) &= \mathsf{A} \oplus \mathsf{n'_T} \oplus \mathsf{n'_T} \oplus \mathsf{H}(\mathsf{n_R}, \mathsf{k_A}) \\
&= \mathsf{A} \oplus \mathsf{H}(\mathsf{n_R}, \mathsf{k_A})
\end{aligned}$$

Fixing the KCL protocol

We added a hash to break the unwanted algebraic property.

KCL$^+$

$$
\begin{aligned}
R &: \mathsf{n_R} \xleftarrow{\$} \\
T &: \mathsf{n_T} \xleftarrow{\$}
\end{aligned}
$$

$$
\begin{aligned}
1 : R \longrightarrow T_{\mathsf{A}} &: \mathsf{n_R} \\
2 : T_{\mathsf{A}} \longrightarrow R &: \langle \mathsf{A} \oplus \mathsf{H}(\mathsf{n_T}, \mathsf{k_A}), \mathsf{n_T} \oplus \mathsf{H}(\mathsf{n_R}, \mathsf{k_A}) \rangle
\end{aligned}
$$

Security Property

Unlinkability for 2 rounds (A, A vs. A, B)

$$\mathsf{n_R}, m_1, \mathsf{n'_R}, \langle \mathsf{A} \oplus \mathsf{H}(\mathsf{n'_T}, \mathsf{k_A}) \,,\, \mathsf{n'_T} \oplus \mathsf{H}(g_1(\mathsf{n_R}, m_1, \mathsf{n'_R}), \mathsf{k_A})\rangle$$

$$\sim$$

$$\mathsf{n_R}, m_1, \mathsf{n'_R}, \langle \mathsf{B} \oplus \mathsf{H}(\mathsf{n'_T}, \mathsf{k_B}) \,,\, \mathsf{n'_T} \oplus \mathsf{H}(g_1(\mathsf{n_R}, m_1, \mathsf{n'_R}), \mathsf{k_B})\rangle$$

where m_1 is the term:

$$m_1 = \langle \mathsf{A} \oplus \mathsf{H}(\mathsf{n_T}, \mathsf{k_A}) \,,\, \mathsf{n_T} \oplus \mathsf{H}(g(\mathsf{n_R}), \mathsf{k_A})\rangle$$

What Assumption on H?

KCL protocol

$$\begin{array}{lcl} 1: R \longrightarrow T_{\mathsf{A}} & : & \mathsf{n_R} \\ 2: T_{\mathsf{A}} \longrightarrow R & : & \langle \mathsf{A} \oplus \mathsf{H}(\mathsf{n_T}, \mathsf{k_A}), \mathsf{n_T} \oplus \mathsf{H}(\mathsf{n_R}, \mathsf{k_A}) \rangle \end{array}$$

Attack on unlinkability

There is an attack if H is only *OW-CPA* and *CR*:

- First bit of $\mathsf{H}(x, \mathsf{k})$ is the first bit of the key.
- A first bit is 0
- B first bit is 1

With proba $\frac{1}{2}$, $\mathsf{A} \oplus \mathsf{H}(\mathsf{n_T}, \mathsf{k_A})$ and $\mathsf{B} \oplus \mathsf{H}(\mathsf{n_T}, \mathsf{k_B})$ start with different bits.

$$\Rightarrow \text{rounds } \mathsf{A}, \mathsf{A} \not\approx \text{ rounds } \mathsf{A}, \mathsf{B}$$

Security proofs

$$\mathsf{n_R}, m_1, \mathsf{n'_R}, \langle \mathsf{A} \oplus \mathsf{H}(\mathsf{n'_T}, \mathsf{k_A})\, ,\ \mathsf{n'_T} \oplus \mathsf{H}(g_1(\mathsf{n_R}, m_1, \mathsf{n'_R}), \mathsf{k_A})\rangle$$
$$\sim$$
$$\mathsf{n_R}, m_1, \mathsf{n'_R}, \langle \mathsf{B} \oplus \mathsf{H}(\mathsf{n'_T}, \mathsf{k_B})\, ,\ \mathsf{n'_T} \oplus \mathsf{H}(g_1(\mathsf{n_R}, m_1, \mathsf{n'_R}), \mathsf{k_B})\rangle$$

Security proofs

$$\frac{\begin{array}{c}\mathsf{n_R}, m_1, \mathsf{n'_R}, \mathsf{A} \oplus \mathsf{H}(\mathsf{n'_T}, \mathsf{k_A}), \mathsf{n'_T} \oplus \mathsf{H}(g_1(\mathsf{n_R}, m_1, \mathsf{n'_R}), \mathsf{k_A}) \\ \sim \\ \mathsf{n_R}, m_1, \mathsf{n'_R}, \mathsf{B} \oplus \mathsf{H}(\mathsf{n'_T}, \mathsf{k_B}), \mathsf{n'_T} \oplus \mathsf{H}(g_1(\mathsf{n_R}, m_1, \mathsf{n'_R}), \mathsf{k_B})\end{array}}{\begin{array}{c}\mathsf{n_R}, m_1, \mathsf{n'_R}, \langle \mathsf{A} \oplus \mathsf{H}(\mathsf{n'_T}, \mathsf{k_A}) \,,\, \mathsf{n'_T} \oplus \mathsf{H}(g_1(\mathsf{n_R}, m_1, \mathsf{n'_R}), \mathsf{k_A})\rangle \\ \sim \\ \mathsf{n_R}, m_1, \mathsf{n'_R}, \langle \mathsf{B} \oplus \mathsf{H}(\mathsf{n'_T}, \mathsf{k_B}) \,,\, \mathsf{n'_T} \oplus \mathsf{H}(g_1(\mathsf{n_R}, m_1, \mathsf{n'_R}), \mathsf{k_B})\rangle\end{array}} \; FA$$

Security proofs

$$\dfrac{\dfrac{\phi, \mathsf{A} \oplus \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A}) \sim \psi, \mathsf{B} \oplus \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{B})}{\begin{array}{c}\mathsf{n}_\mathsf{R}, m_1, \mathsf{n}'_\mathsf{R}, \mathsf{A} \oplus \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A}), \mathsf{n}'_\mathsf{T} \oplus \mathsf{H}(g_1(\mathsf{n}_\mathsf{R}, m_1, \mathsf{n}'_\mathsf{R}), \mathsf{k}_\mathsf{A}) \\ \sim \\ \mathsf{n}_\mathsf{R}, m_1, \mathsf{n}'_\mathsf{R}, \mathsf{B} \oplus \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{B}), \mathsf{n}'_\mathsf{T} \oplus \mathsf{H}(g_1(\mathsf{n}_\mathsf{R}, m_1, \mathsf{n}'_\mathsf{R}), \mathsf{k}_\mathsf{B})\end{array}} \textit{ Permutation}}{\begin{array}{c}\mathsf{n}_\mathsf{R}, m_1, \mathsf{n}'_\mathsf{R}, \langle \mathsf{A} \oplus \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A}),\ \mathsf{n}'_\mathsf{T} \oplus \mathsf{H}(g_1(\mathsf{n}_\mathsf{R}, m_1, \mathsf{n}'_\mathsf{R}), \mathsf{k}_\mathsf{A})\rangle \\ \sim \\ \mathsf{n}_\mathsf{R}, m_1, \mathsf{n}'_\mathsf{R}, \langle \mathsf{B} \oplus \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{B}),\ \mathsf{n}'_\mathsf{T} \oplus \mathsf{H}(g_1(\mathsf{n}_\mathsf{R}, m_1, \mathsf{n}'_\mathsf{R}), \mathsf{k}_\mathsf{B})\rangle\end{array}} \textit{ FA}$$

Security proofs

$$\frac{\begin{array}{r} \phi, \mathsf{A} \oplus \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A}) \sim \phi, \mathsf{A} \oplus \mathsf{n} \\ \phi, \mathsf{A} \oplus \mathsf{n} \sim \psi, \mathsf{B} \oplus \mathsf{n} \\ \psi, \mathsf{B} \oplus \mathsf{n} \sim \psi, \mathsf{B} \oplus \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{B}) \end{array}}{\psi, \mathsf{A} \oplus \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A}) \sim \psi, \mathsf{B} \oplus \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{B})} \; \textit{Trans}$$

Security Proof

Goal

$$\cfrac{\cfrac{\phi, \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A}) \sim \phi, \mathsf{n}}{\phi, \mathsf{A}, \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A}) \sim \phi, \mathsf{A}, \mathsf{n}} \; FA}{\phi, \mathsf{A} \oplus \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A}) \sim \phi, \mathsf{A} \oplus \mathsf{n}} \; FA$$

We want to apply the *PRF* axioms: we need to introduce distinguishing tests:

$$\cfrac{\begin{array}{l} \phi, \texttt{if } \mathsf{EQ}(\mathsf{n}'_\mathsf{T}; g(\mathsf{n}_\mathsf{R})) \texttt{ then } \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A}) \texttt{ else } \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A}) \\ \sim \\ \phi, \texttt{if } \mathsf{EQ}(\mathsf{n}'_\mathsf{T}; g(\mathsf{n}_\mathsf{R})) \texttt{ then } \mathsf{n} \texttt{ else } \mathsf{n} \end{array}}{\phi, \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A}) \sim \phi, \mathsf{n}} \; R$$

Splitting the proof

CS axiom

$$\frac{b, \texttt{if } b \texttt{ then } \mathbf{0} \texttt{ else } t \sim b', \texttt{if } b' \texttt{ then } \mathbf{0} \texttt{ else } t' \qquad b, \texttt{if } b \texttt{ then } s \texttt{ else } \mathbf{0} \sim b', \texttt{if } b' \texttt{ then } s' \texttt{ else } \mathbf{0}}{\texttt{if } b \texttt{ then } s \texttt{ else } t \sim \texttt{if } b' \texttt{ then } s' \texttt{ else } t'} \; CS$$

Security Proof

Left case

$$\phi, \mathsf{EQ}(\mathsf{n}'_\mathsf{T}; g(\mathsf{n}_\mathsf{R})), \texttt{if}\ \mathsf{EQ}(\mathsf{n}'_\mathsf{T}; g(\mathsf{n}_\mathsf{R}))\ \texttt{then}\ \ \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A})\ \ \texttt{else}\ \mathbf{0}$$

$$\sim$$

$$\phi, \mathsf{EQ}(\mathsf{n}'_\mathsf{T}; g(\mathsf{n}_\mathsf{R})), \texttt{if}\ \mathsf{EQ}(\mathsf{n}'_\mathsf{T}; g(\mathsf{n}_\mathsf{R}))\ \texttt{then}\ \ \mathsf{n}\ \ \texttt{else}\ \mathbf{0}$$

Axiom *EqIndep*

If n is fresh in x then:

$$\mathsf{EQ}(\mathsf{n}; x) = \mathsf{false}$$

Security Proof

$$\dfrac{\dfrac{}{\phi, \mathsf{false}, \mathbf{0} \sim \phi, \mathsf{false}, \mathbf{0}}\ \textit{Refl}}{\begin{array}{c}\phi, \mathsf{false}, \mathtt{if}\ \ \mathsf{false}\ \ \mathtt{then}\ \ \mathsf{H}(\mathsf{n}'_{\mathsf{T}}, \mathsf{k}_{\mathsf{A}})\ \ \mathtt{else}\ \ \mathbf{0} \\ \sim \\ \phi, \mathsf{false}, \mathtt{if}\ \ \mathsf{false}\ \ \mathtt{then}\ \ \mathsf{n}\ \ \mathtt{else}\ \ \mathbf{0}\end{array}}\ R$$

Security Proof

Right case: use the *PRF* axiom!

$$\phi, \mathsf{EQ}(\mathsf{n}'_\mathsf{T}; g(\mathsf{n}_\mathsf{R})), \texttt{if } \mathsf{EQ}(\mathsf{n}'_\mathsf{T}; g(\mathsf{n}_\mathsf{R})) \texttt{ then } \mathbf{0} \texttt{ else } \mathsf{H}(\mathsf{n}'_\mathsf{T}, \mathsf{k}_\mathsf{A})$$

$$\sim$$

$$\phi, \mathsf{EQ}(\mathsf{n}'_\mathsf{T}; g(\mathsf{n}_\mathsf{R})), \texttt{if } \mathsf{EQ}(\mathsf{n}'_\mathsf{T}; g(\mathsf{n}_\mathsf{R})) \texttt{ then } \mathbf{0} \texttt{ else } \mathsf{n}$$

Contributions

- Designed and proved axioms for CR, PRF, XOR and PRNG.
- Formally expressed RFID unlinkability [1] in the CSA model.
- Proved the unlinkability of KCL^+ for arbitrary number of rounds.
- Similar study of the LAK protocol (but only for 2 rounds).

To our knowledge, first formal proof of computational unlinkability of hash based RFID protocol.

Future Works

- More examples, with more primitives (RFID or not).
- Decidability of (a fragment of) the logic.
- Interactive prover.

Thanks for your attention

Ari Juels and Stephen A. Weis.
Defining strong privacy for rfid.
ACM Trans. Inf. Syst. Secur., 13(1):7:1–7:23, November 2009.

Ton Van Deursen and Sasa Radomirovic.
Attacks on rfid protocols.
IACR Cryptology ePrint Archive, 2008:310, 2008.