

MODELS OF SECURITY PROTOCOLS

Hubert Comon-Lundh

ENS Cachan, INRIA and AIST, Tokyo

`h.comon-lundh@aist.go.jp`

THIS IS NOT THE END OF THE TALK

Thanks !

ANNOUNCEMENT



Organizes a spring school/workshop on Computational and Symbolic proofs of security: [CoSyProofs09](#)
April 6-9, 2009

Invited speakers: Martìn Abadi, Michael Backes, Bruno Blanchet, Ralf Küsters, John Mitchell, Kazuo Ohta, Olivier Pereira, David Pointcheval, Roberto Segala, Bogdan Warinschi.

Visit <http://www.rcis.aist.go.jp/events/csps2009/>

ANNOUNCEMENT 2

Master/PhD research topics (together with **S. Delaune** or **S. Kremer**):

- starting point: deducibility constraints.
- Ongoing work with **S. Bursuc** and **S. Delaune**
- A clean and beautiful theory is emerging ?
- A prototype implementation ?
- Research is dynamic; so is a PhD research program.

CAN WE TRUST SECURITY PROOFS ?

We must increase our confidence in security protocols. We need **Security proofs**.

CAN WE TRUST SECURITY PROOFS ?

We must increase our confidence in security protocols. We need [Security proofs](#).

There are proved protocols on which attacks were later found. [How can we explain this paradox ?](#)

CAN WE TRUST SECURITY PROOFS ?

We must increase our confidence in security protocols. We need **Security proofs**.

There are proved protocols on which attacks were later found. **How can we explain this paradox ?**

- Proofs were too informal or actually contain mistakes. **Example:**
OAEP

CAN WE TRUST SECURITY PROOFS ?

We must increase our confidence in security protocols. We need **Security proofs**.

There are proved protocols on which attacks were later found. **How can we explain this paradox ?**

- Proofs were too informal or actually contain mistakes. **Example:**
OAEP
- There are implicit hypotheses in the proofs, that might not be matched in the use of the result. **Examples:** many

CAN WE TRUST SECURITY PROOFS ?

We must increase our confidence in security protocols. We need **Security proofs**.

There are proved protocols on which attacks were later found. **How can we explain this paradox ?**

- Proofs were too informal or actually contain mistakes. **Example: OAEP**
- There are implicit hypotheses in the proofs, that might not be matched in the use of the result. **Examples: many**
- There are formally proved (with mechanically checked proofs) protocols on which attacks were later found (**Example: Bull & Otway authentication protocol**).

CAN WE TRUST SECURITY PROOFS ?

We must increase our confidence in security protocols. We need **Security proofs**.

There are proved protocols on which attacks were later found. **How can we explain this paradox ?**

- Proofs were too informal or actually contain mistakes. **Example: OAEP**
- There are implicit hypotheses in the proofs, that might not be matched in the use of the result. **Examples: many**
- There are formally proved (with mechanically checked proofs) protocols on which attacks were later found (**Example: Bull & Otway authentication protocol**).
- The proof and the attack were in different models.

REMARKS AND QUESTIONS

- There are often implicit assumptions, simply because of the chosen model. **How can we ensure that these assumptions are explicit ?**
- The Dolev-Yao model: can we trust the proofs performed in this model ? Under which assumptions ?
- Under which hypotheses a security proof in one model implies a security proof in another model ?
- Is there any chance to find results that are independent of the chosen model ?

THE PIONEERING WORK OF ABADI AND ROGAWAY (1)

Symbolic	Computational
Simplified attacker	More realistic Attacker
Idealized messages	Bitstrings
Formal Detailed Proofs	Informal or sketchy proofs
Automatizable	Hand written

THE PIONEERING WORK OF ABADI AND ROGAWAY (2)

The symbolic setting:

- Formal expressions built on $\text{enc}(x, k, r)$, $\text{dec}(x, k)$, $\langle x, y \rangle$, $\pi_1(x)$, $\pi_2(x)$, $k, n, r, 0$
- Equations $\text{dec}(k, \text{enc}(x, k, r)) = x$, $\pi_1(\langle x, y \rangle) = x$, $\pi_2(\langle x, y \rangle) = y$
- s_1, \dots, s_n and t_1, \dots, t_n are indistinguishable if, for any contexts ζ_1, ζ_2 , not using the private names,

$$\zeta_1[s_1, \dots, s_n] =_E \zeta_2[s_1, \dots, s_n] \Leftrightarrow \zeta_1[t_1, \dots, t_n] =_E \zeta_2[t_1, \dots, t_n]$$

Examples:

- $k \sim k'$,
- $\text{enc}(k, n, r) \sim \text{enc}(k, 0, r)$ a
- $k, \text{enc}(k, n, r) \not\sim k, \text{enc}(k, 0, r)$: choose $\zeta_1 = \text{dec}(x_2, x_1)$ and $\zeta_2 = 0$.

$$\zeta_1(k, \text{enc}(k, n, r)) \neq_E 0,$$

$$\zeta_1(k, \text{enc}(k, 0, r)) =_E 0$$

THE PIONEERING WORK OF ABADI AND ROGAWAY (3)

The computational setting:

- Given a security parameter $\eta \in \mathbb{N}$, the keys are drawn according to a distribution μ_η (the *key generation algorithm*)
- Each symbol is associated with a polynomial time (randomized) algorithm.
- Each sampling τ yields an interpretation $\llbracket s \rrbracket_\tau$ of symbolic terms/list of terms s .
- s_1, \dots, s_n and t_1, \dots, t_n are computationally indistinguishable if, for any attacker \mathcal{A} ,

$$\Pr\{\tau : \mathcal{A}(\llbracket s_1, \dots, s_n \rrbracket_\tau) = 1\} - \Pr\{\tau : \mathcal{A}(\llbracket t_1, \dots, t_n \rrbracket_\tau) = 1\}$$

is negligible.

Example: $\text{enc}(k, 1, r) \approx \text{enc}(k, 0, r)$.

THE PIONEERING WORK OF ABADI AND ROGAWAY (4)

Encryption is assumed to be IND-CPA: for every attacker A

$$\Pr\{k \xleftarrow{R} \mathcal{K}_\eta, \bar{r} \xleftarrow{R} U : A^{O_k^l}(0^\eta) = 1\} - \Pr\{k \xleftarrow{R} \mathcal{K}_\eta, \bar{r} \xleftarrow{R} U : A^{O_k^r}(0^\eta) = 1\}$$

is negligible.

The main result:

Theorem [Abadi and Rogaway, 2000]:

If s_1, \dots, s_n and t_1, \dots, t_n are symbolically indistinguishable, then they are computationally indistinguishable.

Further related works by [Backes, Pfitzmann 2003-2008; Micciancio, Warinschi 2003-2004; Cortier, Warinschi 2005-2008,...]

Models: various kinds of interactive Turing machines, process algebras, pro-

WHAT IS THIS ?

2 | 3

WHAT IS THIS ?

2 | 3

0.666667, the rational number two thirds, the rational number 3 halves, 0,
the enumerated type consisting of two elements 2,3...

GOALS

- Clean syntax/semantics: design a model theory.
- Parametrize the notions of protocols, traces, indistinguishability,... by the model (e.g. symbolic, computational, programs,...)
- Active attackers, unbounded network, arbitrary protocols
- No commitment on a particular attacker: it could be a (worst case) polynomial time randomized Turing machine, but we can switch to any class of attackers (that has basic properties).
- Preservation of security properties from one model to another corresponds to existence of some morphism between the structures.

Acknowledgments: **David Nowak**

WHAT CAN WE SAY AT THIS ABSTRACT LEVEL?

Besides saving space and improving consistency,

WHAT CAN WE SAY AT THIS ABSTRACT LEVEL?

Besides saving space and improving consistency,

- Characterization of the preservation of indistinguishability properties:

Tree soundness + Trace mapping \Rightarrow soundness of indistinguishability

Proved in [CL, Cortier 2008] in a particular setting, gives an explanation and a converse implication of [Backes et al. 2007].

This is independent of the model and the cryptographic primitives

WHAT CAN WE STATE AT THIS ABSTRACT LEVEL?

Besides saving space and improving consistency,

- Characterization of the preservation of indistinguishability properties:
Tree soundness + Trace mapping \Rightarrow soundness of indistinguishability
Proved in [CL, Cortier 2008] in a particular setting, gives an explanation and a converse implication of [Backes et al. 2007].
This is independent of the model and the cryptographic primitives
- Study modularity: composing protocols and primitives.

FIRST-ORDER MODELS

- \mathcal{F} : function symbols, (typically: encryption, signatures,...)
- $\mathcal{N} = \bigcup \mathcal{N}_i$: names (intended to be randomly generated),
- \mathcal{X} : variables (intended to be intruder's inputs)
- \mathcal{P} : predicate symbols (observational abilities)
- \mathcal{E} : equational theory (relations between the primitives)
- formulas: a subset of Boolean combinations of atomic formulas

Examples:

- **Symbolic model**: a Herbrand structure whose domain is $T(\mathcal{F}, \mathcal{N}) / \equiv_{\mathcal{E}}$
- **Computational model**: $\{0, 1\}^*$, \mathcal{F}, \mathcal{P} : **deterministic** (Boolean) functions.
Names: bitstrings.

Any first order structure is a possible model.

THREADS

$$?n, \nu \bar{z}, ?\bar{y}. P$$

- n is a thread identifier. We need it to distinguish different copies of the same thread. It is forged by the attacker (we could also draw it).
- \bar{z} : local names
- \bar{y} : local variables
- P : any program

There are possibly free names and free variables.

Configurations:

- a pid p (assigned to n)
- a state q
- a partial binding σ of local variables \bar{y}

TRANSITION RELATIONS

For every pair of states q_1, q_2 , two formulas Φ_{q_1, q_2} and Ψ_{q_1, q_2}

Given a first-order structure \mathcal{M} ,

$$q_1, \sigma \xrightarrow{\bar{c}(s)} q_2, \sigma \uplus \{y \mapsto s\}$$

if $s \in \mathcal{M}$, y is the (only) free variable of Φ_{q_1, q_2} , that is not bound by σ and $\mathcal{M}, (\sigma \uplus \{y \mapsto s\})\theta \models \Phi_{q_1, q_2}$

$$q_1, \sigma \xrightarrow{c(s)} q_2, \sigma$$

if $s \in \mathcal{M}$, x is the (only) free variable of Ψ_{q_1, q_2} that is not bound by σ , and $\mathcal{M}, (\sigma \uplus \{x \mapsto s\})\theta \models \Psi_{q_1, q_2}$

EXAMPLE

A waits for a key encrypted with her shared key k_{AB} with B and sends back an acknowledgment:

```
 $A =$  ? $n$   $\nu$   $r$  ? $y$ .  
    get( $y$ );  
    if  $\pi_1(y) = n$  and dec( $y, k_{ab}$ )  
    then let  $x =$  dec( $k_{ab}, y$ ) in  
        send(enc( $x, \text{OK}, r$ ))
```

States:

$\Phi_{q_0, q_1}(y) \stackrel{\text{def}}{=} \pi_1(y) = n \wedge \text{dec}(y, k_{ab})$

$\Psi_{q_1, q_2} \stackrel{\text{def}}{=} \text{true}$

PROTOCOLS

$$P = \text{Threads} \cup P \parallel P \cup (P)! \cup (\nu z.P) \cup (?y.P)$$

Example:

$$(?a, b, \nu k.(Q(a, b, k)!))!$$

$Q(a, b, k)$ is a thread with free variables a, b .

STRUCTURAL EQUIVALENCE

$$P_1 \parallel P_2 \equiv P_2 \parallel P_1 \quad \text{and} \quad P_1 \parallel (P_2 \parallel P_3) \equiv (P_1 \parallel P_2) \parallel P_3$$

$$\nu z. P_1 \equiv \nu z'. P_1 \{z \mapsto z'\} \quad \text{if } z' \text{ does not occur in } P_1$$

$$?y. P_1 \equiv ?y'. P_1 \{y \mapsto y'\} \quad \text{if } y' \text{ does not occur in } P_1$$

$$(\nu z. P_1) \parallel P_2 \equiv \nu z. (P_1 \parallel P_2) \quad \text{if } z \text{ does not occur free in } P_2$$

$$?y. \nu z. P \equiv \nu z. ?y. P$$

$$P! \equiv P! \parallel P$$

TRANSITION RELATIONS

Configurations: P_0, L_C, θ

- G : generated names
- P_0 : the inactive part of the original process.
- L_C : a finite multiset of thread configurations
- σ : a memory, mapping variables to values in the interpretation domain of \mathcal{M}

Transition relations:

- $G, P_0 \parallel ?y.P, L_C, \theta \xrightarrow{\bar{c}(\text{start}_{?y.P, v})} G, P_0 \parallel P, L_C, \theta \uplus \{y \mapsto v\}$
($y \notin \text{dom}(\theta)$)

- $G, P_0 \parallel A, L_C, \theta \xrightarrow{\bar{c}(\text{start}_A, pid)} G \cup \bar{z}, P_0, L_C \uplus \{(q_0, pid, \emptyset), \theta \uplus \{n \mapsto pid\}$
if $A = ?n, \nu \bar{z}, ?\bar{y}.P_A, \bar{z} \cap G = \emptyset$ and $n \notin \text{dom}(\theta)$

- $G, P_0, L_C \uplus \{c_1\}, \theta \xrightarrow{\tilde{c}(w)} G, P_0, L_C \uplus \{c_2\}, \theta$ if $c_1 \xrightarrow{\tilde{c}(w)} c_2$

ATTACKERS

Again parametrized by \mathcal{M} :

\mathcal{A} is a function from $\mathcal{Q} \times \mathcal{M}^* (\times \mathcal{R})$ to $\mathcal{Q} \times \mathcal{M}^* \times \mathcal{M} \times \mathcal{E}$:

in a state q , with a local memory Σ , the attacker returns a new state, a new local memory, a message m and an event e (either send, require a new copy of a thread, or nothing).

Examples:

- The symbolic attacker: $m =_{\mathcal{E}} C_q[s_1, \dots, s_n]$. $\Sigma = (s_1, \dots, s_n)$.
- A computational attacker: PPT \mathcal{A} which, in state q with tape content Σ and random tape r , computes a message m .

TRACES

Traces in \mathcal{M} :

$$G_0, P_0, L_0, \theta_0 \xrightarrow{\tilde{c}(w_1)} G_1, P_1, L_1, \theta_1 \dots \xrightarrow{\tilde{c}(w_n)} G_n, P_n, L_n, \theta_n \dots$$

$\tilde{c}(w_i)$ is either $c(w_i)$ or $\bar{c}(w_i)$ and P_i is closed

Each time it is a received message, the attacker must be able to provide with w_i from its current knowledge.

Morphisms between first-order structures \mathcal{M}_1 and \mathcal{M}_2 are extended to configurations and traces.

TRACE MAPPING

Definition: There is a **Trace mapping** from a model \mathcal{M}_1 to a model \mathcal{M}_2 if there is a morphism \mathbf{h} from \mathcal{M}_1 to \mathcal{M}_2 such that for any attacker \mathcal{A}_1 in the model \mathcal{M}_1 ,

$$\Pr\{\tau \stackrel{R}{\leftarrow} \mu^\eta : \exists \mathcal{A}_2, \mathbf{h}(\text{trace}_\tau^{\mathcal{A}_1}) = \text{trace}_{\mathbf{h} \circ \tau}^{\mathcal{A}_2}\}$$

is overwhelming.

TRACE MAPPING

Definition: There is a **Trace mapping** from a model \mathcal{M}_1 to a model \mathcal{M}_2 if there is a morphism \mathbf{h} from \mathcal{M}_1 to \mathcal{M}_2 such that for any attacker \mathcal{A}_1 in the model \mathcal{M}_1 ,

$$\Pr\{\tau \stackrel{R}{\leftarrow} \mu^\eta : \exists \mathcal{A}_2, \mathbf{h}(\text{trace}_\tau^{\mathcal{A}_1}) = \text{trace}_{\mathbf{h} \circ \tau}^{\mathcal{A}_2}\}$$

is overwhelming.

Lemma: For any model \mathcal{M} , there is a trace mapping from the symbolic model \mathcal{M}_s to \mathcal{M} .

TRACE MAPPING

Definition: There is a **Trace mapping** from a model \mathcal{M}_1 to a model \mathcal{M}_2 if there is a morphism \mathbf{h} from \mathcal{M}_1 to \mathcal{M}_2 such that for any attacker \mathcal{A}_1 in the model \mathcal{M}_1 ,

$$\Pr\{\tau \stackrel{R}{\leftarrow} \mu^\eta : \exists \mathcal{A}_2, \mathbf{h}(\text{trace}_\tau^{\mathcal{A}_1}) = \text{trace}_{\mathbf{h} \circ \tau}^{\mathcal{A}_2}\}$$

is overwhelming.

Lemma: For any model \mathcal{M} , there is a trace mapping from the symbolic model \mathcal{M}_s to \mathcal{M} .

Theorem [Cortier & Warinschi 2005, Janvier, Lakhnech & Mazaré 2005]:
Assuming IND-CCA for a public-key encryption scheme and existential unforgeability for a signing scheme, there is a trace mapping from a computational model to the symbolic model.

There are many more trace mapping properties...

OBSERVATIONAL EQUIVALENCE

Definition The protocols P, Q are **observationally equivalent with respect to the model \mathcal{M}** , which we write $P \approx_{\mathcal{M}} Q$, if, for any attacker \mathcal{A} ,

$$|\Pr\{\tau \stackrel{R}{\leftarrow} \mu^\eta : \mathcal{A}(P, \tau) = 1\} - \Pr\{\tau \stackrel{R}{\leftarrow} \mu^\eta : \mathcal{A}(Q, \tau) = 1\}|$$

is negligible.

Examples:

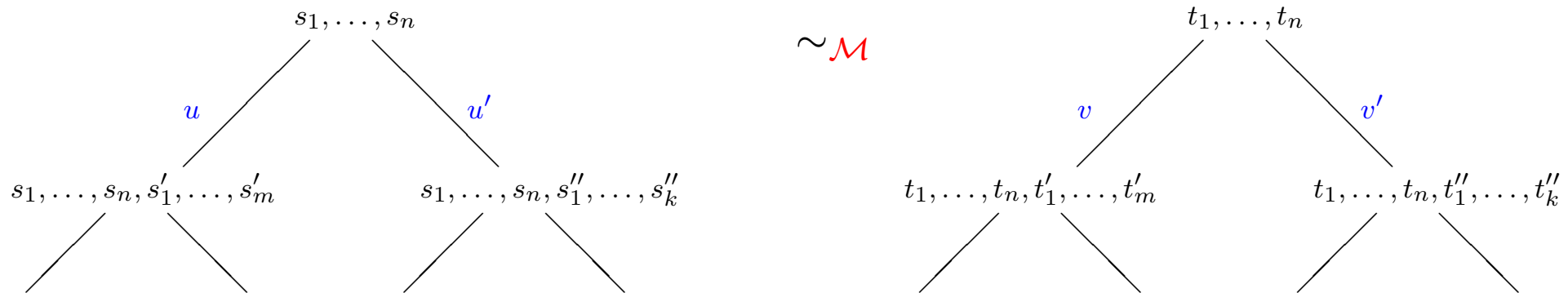
- In the symbolic model: the usual observational equivalence between processes
- In a computational model, universal composability:

$$P \approx_{\mathcal{M}_c} P_i \Rightarrow P! \approx_{\mathcal{M}_c} P_i!$$

joined state universal composability:

$$\nu \bar{z} P \approx_{\mathcal{M}_c} \bar{z} : P_i \Rightarrow \bar{z} . (P!) \approx_{\mathcal{M}_c} \bar{z} . (P_i!)$$

TREE EQUIVALENCE



If any attacker in \mathcal{M} , cannot make any significant difference between interacting with the left tree and interacting with the right tree.

The **Tree soundness** (from \mathcal{M}_1 to \mathcal{M}_2) states that $\sim_{\mathcal{M}_1}$ implies $\sim_{\mathcal{M}_2}$.

Examples:

- IND-CPA implies the tree soundness from \mathcal{M}_s to \mathcal{M}_c (the standard computational model). Note: Abadi-Rogaway need only a weaker security property
- Symmetric encryption case: [CL,Cortier 2008],
- Ring signatures: [CL,Kawamoto, Sakurada 2009].

MAIN LEMMA

Lemma:

Tree soundness from \mathcal{M}_s to \mathcal{M} + trace mapping from \mathcal{M} to \mathcal{M}_s

$$\begin{array}{c} \Rightarrow \\ P \approx_{\mathcal{M}_s} Q \Rightarrow P \approx_{\mathcal{M}} Q \end{array}$$

Application: reduce the problem of proving indistinguishability in \mathcal{M} to the symbolic observational equivalence of processes.

FURTHER WORK

Splitting further the tasks:

Modularity of trace mapping and tree soundness

Application: mixing protocols, use of several encryption schemes: currently there is no proof of protocol security considering several encryption schemes that may share some data.

ANNOUNCEMENT



Organizes a spring school/workshop on Computational and Symbolic proofs of security: [CoSyProofs09](#).

Invited speakers: Martìn Abadi, Michael Backes, Bruno Blanchet, Ralf Küsters, John Mitchell, Kazuo Ohta, Olivier Pereira, David Pointcheval, Roberto Segala, Bogdan Warinschi.

Visit <http://www.rcis.aist.go.jp/events/csps2009/>