Verification of Cryptographic Protocols in Presence of Algebraic Properties

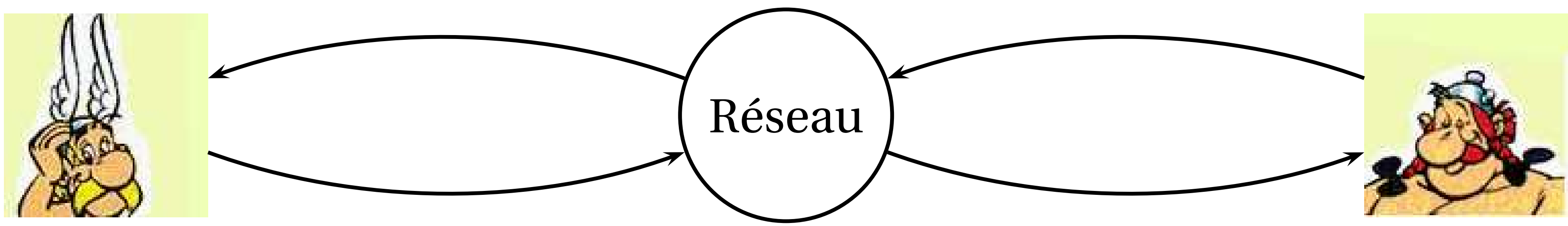
Stéphanie Delaune

delaune@lsv.ens-cachan.fr

France Telecom R&D

LSV, CNRS UMR 8643, ENS de Cachan

Cryptographic Protocols



Cryptographic Protocols

Protocol

↪ rules of message exchanges

Goal

↪ secure communications: *secrecy, authentication, anonymity* ...

Applications

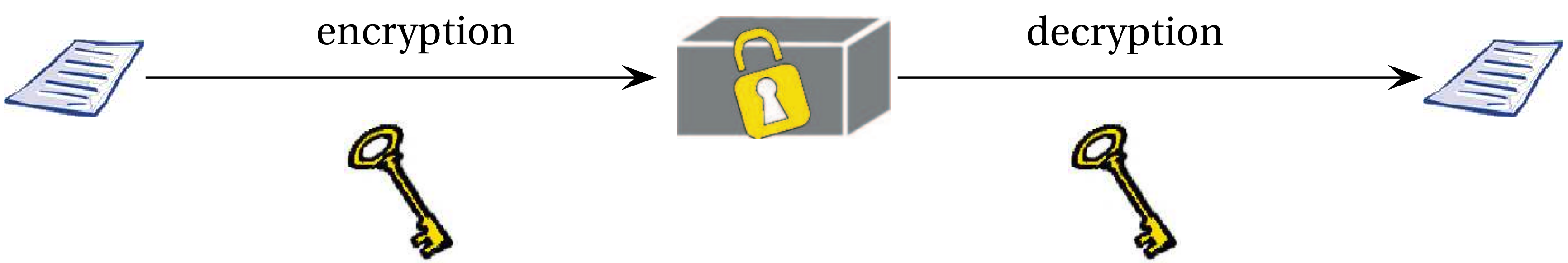
↪ mobile phone, e-voting, e-commerce, ...

Goals

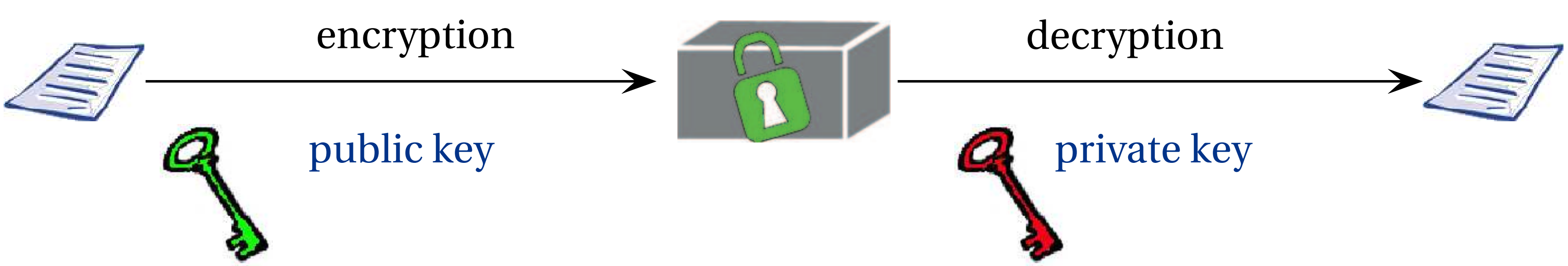
- **Secrecy**: May a dishonest participant learn some secret message between two honest participants ?
- **Authentication**: Is the agent Alice really talking to Bob ?
- **Anonymity**: Who is talking to Alice ?
- **Privacy**: Alice participate to an election. May a participant learn something about the vote of Alice ?
- **Fairness**: Alice and Bob want to sign a contract. Alice initiates the protocol. May Bob obtain some advantage ?

Encryption

- **Symmetric** Encryption



- **Asymmetric** Encryption



Modelisation of Messages

Messages = ground terms built over an alphabet $\mathcal{F}$

containing:

- constant a, i, ...
- pairing $\langle m_1, m_2 \rangle$,
- encryption $\{m\}_k^s$, $\{m\}^a{}_k$ (symmetric /asymmetric),
- public and private key $\mathsf{pub}(a)$, $\mathsf{priv}(a)$,
- inverse k^{-1},
- hash function $h(m)$,
- ...
- exclusive or $\oplus$, modular exponentiation exp, ...
- ...

The standard Dolev-Yao Intruder Model

Let be u, v terms and T a finite set of terms *(intruder's knowledge)*

Axiom (A)	$\dfrac{u \in T}{T \vdash u}$	Pairing (P)	$\dfrac{T \vdash u \quad T \vdash v}{T \vdash \langle u, v \rangle}$
Unpairing (UL)	$\dfrac{T \vdash \langle u, v \rangle}{T \vdash u}$	Unpairing (UR)	$\dfrac{T \vdash \langle u, v \rangle}{T \vdash v}$
Encryption (E)	$\dfrac{T \vdash u \quad T \vdash v}{T \vdash \{u\}_v}$	Decryption (D)	$\dfrac{T \vdash \{u\}_v \quad T \vdash v^{-1}}{T \vdash u}$

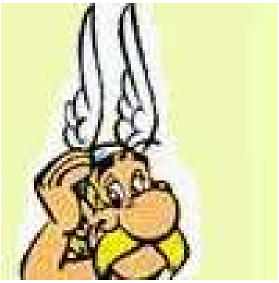
$\hookrightarrow$ **Perfect Cryptography Assumption**

no way to obtain knowledge about u from $\{u\}_v$ without knowing v^{-1}

Needham-Schroeder's Protocol (1978)

- $$\begin{aligned} A &\rightarrow B: & \{A, N_a\}_{\mathsf{pub}(B)} \\ B &\rightarrow A: & \{N_a, N_b\}_{\mathsf{pub}(A)} \\ A &\rightarrow B: & \{N_b\}_{\mathsf{pub}(B)} \end{aligned}$$

Needham-Schroeder's Protocol (1978)



$$A \rightarrow B : \{A, N_a\}_{\mathsf{pub}(B)}$$

- $$B \rightarrow A : \{N_a, N_b\}_{\mathsf{pub}(A)}$$

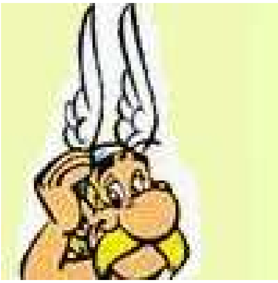
$$A \rightarrow B : \{N_b\}_{\mathsf{pub}(B)}$$

Needham-Schroeder's Protocol (1978)

$$A \rightarrow B: \quad \{A, N_a\}_{\mathsf{pub}(B)}$$
$$B \rightarrow A: \quad \{N_a, N_b\}_{\mathsf{pub}(A)}$$

- $A \rightarrow B: \quad \{N_b\}_{\mathsf{pub}(B)}$

Needham-Schroeder's Protocol (1978)

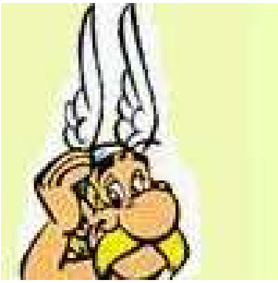


$$\begin{array}{rcrr} A & \rightarrow & B: & \{A, N_a\}_{\mathsf{pub}(B)} \\ B & \rightarrow & A: & \{N_a, N_b\}_{\mathsf{pub}(A)} \\ A & \rightarrow & B: & \{N_b\}_{\mathsf{pub}(B)} \end{array}$$

Questions

- Is N_b secret between A and B ?
- When B receives $\{N_b\}_{\mathsf{pub}(B)}$, does this message really comes from A ?

Needham-Schroeder's Protocol (1978)



$$\begin{array}{rcll} A & \rightarrow & B : & \{A, N_a\}_{\mathsf{pub}(B)} \\ B & \rightarrow & A : & \{N_a, N_b\}_{\mathsf{pub}(A)} \\ A & \rightarrow & B : & \{N_b\}_{\mathsf{pub}(B)} \end{array}$$

Questions

- Is N_b secret between A and B ?
- When B receives $\{N_b\}_{\mathsf{pub}(B)}$, does this message really comes from A ?

An attack was found 17 years after its publication ! [Lowe 96]

Attack on Needham-Schroeder's Protocol

Agent A

Intrus I

Agent B

$$\begin{array}{lcll} A & \longrightarrow & B & : \{N_a, A\}_{\mathsf{pub}(B)} \\ B & \longrightarrow & A & : \{N_a, N_b\}_{\mathsf{pub}(A)} \\ A & \longrightarrow & B & : \{N_b\}_{\mathsf{pub}(B)} \end{array}$$

Attack on Needham-Schroeder's Protocol

 $\xrightarrow{\{N_a, A\}_{\mathsf{pub}(I)}}$  $\xrightarrow{\{N_a, A\}_{\mathsf{pub}(B)}}$ 

Agent A Intrus I Agent B

- $$\begin{array}{lcll} A & \longrightarrow & B & : \{N_a, A\}_{\mathsf{pub}(B)} \\ B & \longrightarrow & A & : \{N_a, N_b\}_{\mathsf{pub}(A)} \\ A & \longrightarrow & B & : \{N_b\}_{\mathsf{pub}(B)} \end{array}$$

Attack on Needham-Schroeder's Protocol

Agent A

$\xrightarrow{\{N_a, A\}_{\mathsf{pub}(I)}}$

$\xleftarrow{\{N_a, N_b\}_{\mathsf{pub}(A)}}$

Intrus I

$\xrightarrow{\{N_a, A\}_{\mathsf{pub}(B)}}$

$\xleftarrow{\{N_a, N_b\}_{\mathsf{pub}(A)}}$

Agent B

$$
\begin{array}{lcll}
A & \longrightarrow & B & : \{N_a, A\}_{\mathsf{pub}(B)} \\
\bullet\ B & \longrightarrow & A & : \{N_a, N_b\}_{\mathsf{pub}(A)} \\
A & \longrightarrow & B & : \{N_b\}_{\mathsf{pub}(B)}
\end{array}
$$

Attack on Needham-Schroeder's Protocol

$\{N_a, A\}_{\mathsf{pub}(I)} \longrightarrow$

$\longleftarrow \{N_a, N_b\}_{\mathsf{pub}(A)}$

$\{N_b\}_{\mathsf{pub}(I)} \longrightarrow$

$\{N_a, A\}_{\mathsf{pub}(B)} \longrightarrow$

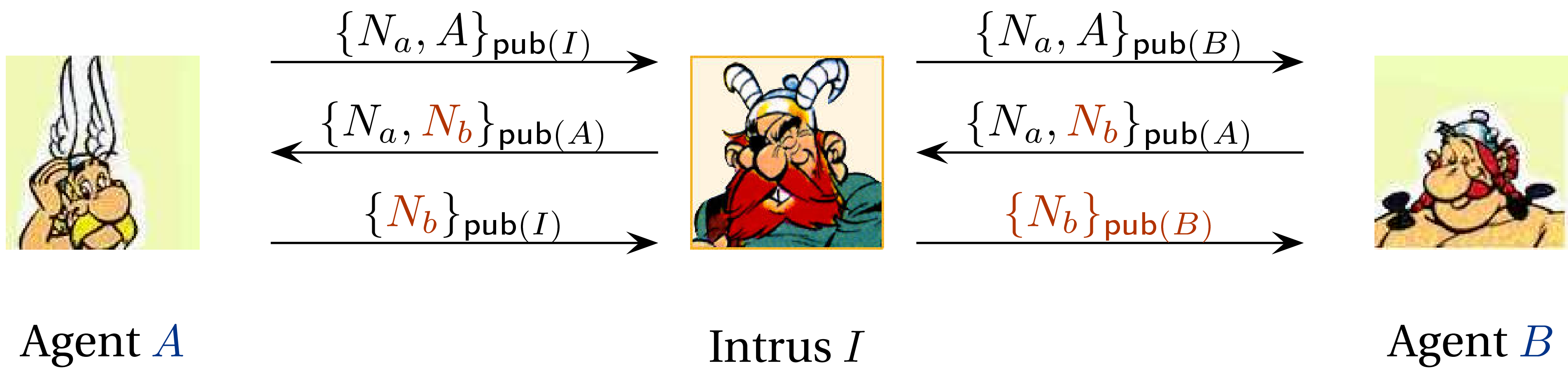
$\longleftarrow \{N_a, N_b\}_{\mathsf{pub}(A)}$

$\{N_b\}_{\mathsf{pub}(B)} \longrightarrow$

Agent A Intrus I Agent B

$$
\begin{array}{llll}
 & A & \longrightarrow & B : \{N_a, A\}_{\mathsf{pub}(B)} \\
 & B & \longrightarrow & A : \{N_a, N_b\}_{\mathsf{pub}(A)} \\
\bullet & A & \longrightarrow & B : \{N_b\}_{\mathsf{pub}(B)}
\end{array}
$$

Attack on Needham-Schroeder's Protocol



Responses

- Is N_b secret between A and B ?
  $\hookrightarrow$ No
- When B receives $\{N_b\}_{\mathsf{pub}(B)}$, does this message really comes from A ?
  $\hookrightarrow$ No

Difficulties of the Verification

Semantic

- large number of models with implicit hypotheses,
- security properties are not clearly defined.

The intruder

- intercepts all messages,
- analyzes and synthesizes new messages,
- sends fake messages,
- has an unbounded memory.

Protocol

- an unbounded number of agents,
- an unbounded number of sessions,
- an unbounded depth of messages.

Roadmap of the Talk

I) Existing Results

- Results in the standard Dolev-Yao intruder model
- Constraints solving approach
- Relaxing the perfect cryptographic assumption

II) My Contributions

- Motivations
- Results
- Decision procedure

III) Works in Progress

- To develop a general framework
- To deal with non equivalence static constraints

Roadmap of the Talk

I) Existing Results

- Results in the standard Dolev-Yao intruder model
- Constraints solving approach
- Relaxing the perfect cryptographic assumption

II) My Contributions

- Motivations
- Results
- Decision procedure

III) Works in Progress

- To develop a general framework
- To deal with non equivalence static constraints

Undecidability of the Secrecy Problem

Post Correspondance Problem (Undecidable)

INPUT: $(u_i, v_i)_{1 \leq i \leq n}$ such that $u_i, v_i \in \{a, b\}*$

OUTPUT: Yes iff there exists a sequence $i_0, \ldots, i_k$ such that

$$u_{i_0} u_{i_1} \ldots u_{i_k} = v_{i_0} v_{i_1} \ldots v_{i_k}$$

Undecidability of the Secrecy Problem

Post Correspondance Problem (Undecidable)

INPUT: $(u_i, v_i)_{1\leq i\leq n}$ such that $u_i, v_i \in \{a, b\}*$

OUTPUT: Yes iff there exists a sequence $i_0, \ldots, i_k$ such that

$$u_{i_0} u_{i_1} \ldots u_{i_k} = v_{i_0} v_{i_1} \ldots v_{i_k}$$

Protocol

$$\begin{array}{rcll} A & \rightarrow & B: & \{\langle u_i, v_i\rangle\}_{K_{ab}} & 1 \leq i \leq n \\ B: \{\langle x, y\rangle\}_{K_{ab}} & \rightarrow & A: & \{\langle x u_i, y v_i\rangle\}_{K_{ab}} & 1 \leq i \leq n \\ A: \{\langle x, x\rangle\}_{K_{ab}} & \rightarrow & B: & secret & \end{array}$$

$secret$ is secret $\iff$ PCP has no solution

Undecidability of the Secrecy Problem

Post Correspondance Problem (Undecidable)

INPUT: $(u_i, v_i)_{1 \leq i \leq n}$ such that $u_i, v_i \in \{a, b\}*$

OUTPUT: Yes iff there exists a sequence $i_0, \ldots, i_k$ such that

$$u_{i_0} u_{i_1} \ldots u_{i_k} = v_{i_0} v_{i_1} \ldots v_{i_k}$$

Protocol

$$\begin{array}{rcll} A & \rightarrow & B: \{\langle u_i, v_i \rangle\}_{K_{ab}} & 1 \leq i \leq n \\ B: \{\langle x, y \rangle\}_{K_{ab}} & \rightarrow & A: \{\langle x u_i, y v_i \rangle\}_{K_{ab}} & 1 \leq i \leq n \\ A: \{\langle x, x \rangle\}_{K_{ab}} & \rightarrow & B: secret & \end{array}$$

$secret$ is secret $\Longleftrightarrow$ PCP has no solution

$\Longrightarrow$ The secrecy problem is undecidable.

Decidability /Undecidabilty Results

Unbounded number of sessions

<table>
<tr><th>Without nonces</th><th>With nonces</th></tr>
<tr><td>Bounded message length:
DEXPTIME-complete
[Durgin et al., 99]
[Chevalier et al., 03]</td><td>Bounded message length:
Undecidable
[Durgin et al., 99]
[Amadio & Charatonik, 02]</td></tr>
<tr><td rowspan="2">Tagged protocols:
EXPTIME
[Blanchet & Podelski, 03]</td><td>Strongly typed protocols:
Decidable [Lowe, 98]</td></tr>
<tr><td>Tagged protocols:
Decidable [Ramanujam & Suresh, 03]</td></tr>
<tr><td>One copy: 3-EXPTIME
[Comon-Lundh & Cortier, 03]</td><td>Ping-pong protocols:
PTIME [Dolev & Even, 83]</td></tr>
</table>

Decidability /Undecidabilty Results

Unbounded number of sessions

Without nonces	With nonces
Bounded message length: *DEXPTIME-complete* [Durgin et al., 99] [Chevalier et al., 03]	**Bounded message length:** *Undecidable* [Durgin et al., 99] [Amadio & Charatonik, 02]
Tagged protocols: *EXPTIME* [Blanchet & Podelski, 03]	**Strongly typed protocols:** *Decidable* [Lowe, 98] **Tagged protocols:** *Decidable* [Ramanujam & Suresh, 03]
One copy: *3-EXPTIME* [Comon-Lundh & Cortier, 03]	**Ping-pong protocols:** *PTIME* [Dolev & Even, 83]

Bounded number of sessions

co-NP-complete [Rusinowitch & Turuani, 01]

Roadmap of the Talk

I) Existing Results
- Results in the standard Dolev-Yao intruder model
- Constraints solving approach
- Relaxing the perfect cryptographic assumption

II) My Contributions
- Motivations
- Results
- Decision procedure

III) Works in Progress
- To develop a general framework
- To deal with non equivalence static constraints

Intruder Deduction Problem

Secrecy Problem in Presence of a Passive Attacker:

INPUT: A finite set of messages T and a message m

OUTPUT: Yes iff the intruder is able to deduce m from T

Intruder Deduction Problem

Secrecy Problem in Presence of a Passive Attacker:

INPUT: A finite set of messages T and a message m

OUTPUT: Yes iff the intruder is able to deduce m from T

Example:

$T = \{\{s_1\}_k,\ \{s_2\}_k,\ k\}$ and $m = \langle s_1, s_2\rangle$

$$\dfrac{\dfrac{\dfrac{\{s_1\}_k \in T}{T \vdash \{s_1\}_k}\text{(A)} \quad \dfrac{k \in T}{T \vdash k}\text{(A)}}{T \vdash s_1}\text{(D)} \quad \dfrac{\dfrac{\{s_2\}_k \in T}{T \vdash \{s_2\}_k}\text{(A)} \quad \dfrac{k \in T}{T \vdash k}\text{(A)}}{T \vdash s_2}\text{(D)}}{T \vdash \langle s_1, s_2\rangle}\text{(P)}$$

Intruder Deduction Problem

Secrecy Problem in Presence of a Passive Attacker:

INPUT: A finite set of messages T and a message m

OUTPUT: Yes iff the intruder is able to deduce m from T

Example:

$T = \{\{s_1\}_k,\ \{s_2\}_k,\ k\}$ and $m = \langle s_1, s_2 \rangle$

$$\dfrac{\dfrac{\dfrac{\{s_1\}_k \in T}{T \vdash \{s_1\}_k}\,(\mathsf{A}) \qquad \dfrac{k \in T}{T \vdash k}\,(\mathsf{A})}{T \vdash s_1}\,(\mathsf{D}) \qquad \dfrac{\dfrac{\{s_2\}_k \in T}{T \vdash \{s_2\}_k}\,(\mathsf{A}) \qquad \dfrac{k \in T}{T \vdash k}\,(\mathsf{A})}{T \vdash s_2}\,(\mathsf{D})}{T \vdash \langle s_1, s_2 \rangle}\,(\mathsf{P})$$

Result:
The intruder deduction problem is decidable in PTIME in the standard Dolev-Yao model.

Secrecy Problem – Finite Number of Sessions

Secrecy Problem in Presence of an Active Attacker:

Given $\mathcal{P}$, s, $\mathcal{S}_0$ and T_0, does there exist a running sequence of protocol rules such that:

- at the end, the intruder's knowledge is T,
- there exists a proof of $T \vdash s$?

Result:

The secrecy problem for a finite number of sessions is co-NP-complete in the standard Dolev-Yao model. [Rusinowitch&Turuani, 01]

Example - Needham Schroeder's Protocol (1)

Protocol specification – common syntax

$$
\begin{array}{lll}
A & \rightarrow B: & \{A, N_a\}_{\mathsf{pub}(B)} \\
B & \rightarrow A: & \{N_a, N_b\}_{\mathsf{pub}(A)} \\
A & \rightarrow B: & \{N_b\}_{\mathsf{pub}(B)}
\end{array}
$$

Example - Needham Schroeder's Protocol (1)

Protocol specification – common syntax

$$\begin{array}{lll} A & \rightarrow B : & \{A, N_a\}_{\mathsf{pub}(B)} \\ B & \rightarrow A : & \{N_a, N_b\}_{\mathsf{pub}(A)} \\ A & \rightarrow B : & \{N_b\}_{\mathsf{pub}(B)} \end{array}$$

Protocol specification – role syntax

$$\begin{array}{llll} roleA : & & \rightsquigarrow & \{A, N_a\}_{\mathsf{pub}(B)} \\ & \{N_a, x_{N_b}\}_{\mathsf{pub}(A)} & \rightsquigarrow & \{x_{N_b}\}_{\mathsf{pub}(B)} \\ & & & \\ roleB : & \{x_A, x_{N_a}\}_{\mathsf{pub}(B)} & \rightsquigarrow & \{x_{N_a}, N_b\}_{\mathsf{pub}(x_A)} \end{array}$$

Example - Needham Schroeder's Protocol (2)

Session 1: a / I			Session 2: a / b		
A		$\rightsquigarrow \{a, n_a\}_{\mathsf{pub}(I)}$	A		$\rightsquigarrow \{a, n'_a\}_{\mathsf{pub}(b)}$
	$\{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x_{n_b}\}_{\mathsf{pub}(I)}$		$\{n'_a, x'_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x'_{n_b}\}_{\mathsf{pub}(b)}$
B	$\{x_a, x_{n_a}\}_{\mathsf{pub}(I)}$	$\rightsquigarrow \{x_{n_a}, n_b\}_{\mathsf{pub}(x_a)}$	B	$\{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$	$\rightsquigarrow \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}$

Example - Needham Schroeder's Protocol (2)

Session 1: a / I			Session 2: a / b		
1		$\rightsquigarrow \{a, n_a\}_{\mathsf{pub}(I)}$			$\rightsquigarrow \{a, n'_a\}_{\mathsf{pub}(b)}$
3	$\{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x_{n_b}\}_{\mathsf{pub}(I)}$		$\{n'_a, x'_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x'_{n_b}\}_{\mathsf{pub}(b)}$
	$\{x_a, x_{n_a}\}_{\mathsf{pub}(I)}$	$\rightsquigarrow \{x_{n_a}, n_b\}_{\mathsf{pub}(x_a)}$	2	$\{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$	$\rightsquigarrow \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}$

Example - Needham Schroeder's Protocol (2)

Session 1: a / I			Session 2: a / b		
1		$\rightsquigarrow \{a, n_a\}_{\mathsf{pub}(I)}$			$\rightsquigarrow \{a, n'_a\}_{\mathsf{pub}(b)}$
3	$\{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x_{n_b}\}_{\mathsf{pub}(I)}$		$\{n'_a, x'_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x'_{n_b}\}_{\mathsf{pub}(b)}$
	$\{x_a, x_{n_a}\}_{\mathsf{pub}(I)}$	$\rightsquigarrow \{x_{n_a}, n_b\}_{\mathsf{pub}(x_a)}$	2	$\{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$	$\rightsquigarrow \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}$

$$T_0 = \{a,\ b,\ I,\ \mathsf{priv}(I),\ \mathsf{pub}(I),\ \mathsf{pub}(b),\ \mathsf{pub}(a)\}$$

Example - Needham Schroeder's Protocol (2)

	Session 1: a / I			Session 2: a / b	
1		$\rightsquigarrow \{a, n_a\}_{\mathsf{pub}(I)}$			$\rightsquigarrow \{a, n'_a\}_{\mathsf{pub}(b)}$
3	$\{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x_{n_b}\}_{\mathsf{pub}(I)}$		$\{n'_a, x'_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x'_{n_b}\}_{\mathsf{pub}(b)}$
	$\{x_a, x_{n_a}\}_{\mathsf{pub}(I)}$	$\rightsquigarrow \{x_{n_a}, n_b\}_{\mathsf{pub}(x_a)}$	2	$\{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$	$\rightsquigarrow \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}$

$$T_0 = \{a,\ b,\ I,\ \mathsf{priv}(I),\ \mathsf{pub}(I),\ \mathsf{pub}(b),\ \mathsf{pub}(a)\}$$

Constraints System

$$T_0, \{a, n_a\}_{\mathsf{pub}(I)}$$

Example - Needham Schroeder's Protocol (2)

Session 1: a / I			Session 2: a / b		
1		$\rightsquigarrow \{a, n_a\}_{\mathsf{pub}(I)}$			$\rightsquigarrow \{a, n'_a\}_{\mathsf{pub}(b)}$
3	$\{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x_{n_b}\}_{\mathsf{pub}(I)}$		$\{n'_a, x'_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x'_{n_b}\}_{\mathsf{pub}(b)}$
	$\{x_a, x_{n_a}\}_{\mathsf{pub}(I)}$	$\rightsquigarrow \{x_{n_a}, n_b\}_{\mathsf{pub}(x_a)}$	2	$\{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$	$\rightsquigarrow \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}$

$$T_0 = \{a,\ b,\ I,\ \mathsf{priv}(I),\ \mathsf{pub}(I),\ \mathsf{pub}(b),\ \mathsf{pub}(a)\}$$

Constraints System

$$T_0, \{a, n_a\}_{\mathsf{pub}(I)} \Vdash \{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$$

Example - Needham Schroeder's Protocol (2)

Session 1: a / I			Session 2: a / b		
1		$\rightsquigarrow \{a, n_a\}_{\mathsf{pub}(I)}$			$\rightsquigarrow \{a, n'_a\}_{\mathsf{pub}(b)}$
3	$\{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x_{n_b}\}_{\mathsf{pub}(I)}$		$\{n'_a, x'_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x'_{n_b}\}_{\mathsf{pub}(b)}$
	$\{x_a, x_{n_a}\}_{\mathsf{pub}(I)}$	$\rightsquigarrow \{x_{n_a}, n_b\}_{\mathsf{pub}(x_a)}$	2	$\{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$	$\rightsquigarrow \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}$

$$T_0 = \{a,\ b,\ I,\ \mathsf{priv}(I),\ \mathsf{pub}(I),\ \mathsf{pub}(b),\ \mathsf{pub}(a)\}$$

Constraints System

$$T_0, \{a, n_a\}_{\mathsf{pub}(I)} \Vdash \{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$$

$$T_0, \{a, n_a\}_{\mathsf{pub}(I)}, \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}$$

Example - Needham Schroeder's Protocol (2)

Session 1: a / I			Session 2: a / b		
1		$\rightsquigarrow \{a, n_a\}_{\mathsf{pub}(I)}$			$\rightsquigarrow \{a, n'_a\}_{\mathsf{pub}(b)}$
3	$\{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x_{n_b}\}_{\mathsf{pub}(I)}$		$\{n'_a, x'_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x'_{n_b}\}_{\mathsf{pub}(b)}$
	$\{x_a, x_{n_a}\}_{\mathsf{pub}(I)}$	$\rightsquigarrow \{x_{n_a}, n_b\}_{\mathsf{pub}(x_a)}$	2	$\{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$	$\rightsquigarrow \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}$

$$T_0 = \{a,\ b,\ I,\ \mathsf{priv}(I),\ \mathsf{pub}(I),\ \mathsf{pub}(b),\ \mathsf{pub}(a)\}$$

Constraints System

$$T_0, \{a, n_a\}_{\mathsf{pub}(I)} \Vdash \{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$$
$$T_0, \{a, n_a\}_{\mathsf{pub}(I)}, \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)} \Vdash \{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$$

Example - Needham Schroeder's Protocol (2)

Session 1: a / I			Session 2: a / b		
1		$\rightsquigarrow \{a, n_a\}_{\mathsf{pub}(I)}$			$\rightsquigarrow \{a, n'_a\}_{\mathsf{pub}(b)}$
3	$\{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x_{n_b}\}_{\mathsf{pub}(I)}$		$\{n'_a, x'_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x'_{n_b}\}_{\mathsf{pub}(b)}$
	$\{x_a, x_{n_a}\}_{\mathsf{pub}(I)}$	$\rightsquigarrow \{x_{n_a}, n_b\}_{\mathsf{pub}(x_a)}$	2	$\{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$	$\rightsquigarrow \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}$

$$T_0 = \{a,\ b,\ I,\ \mathsf{priv}(I),\ \mathsf{pub}(I),\ \mathsf{pub}(b),\ \mathsf{pub}(a)\}$$

Constraints System

$$T_0, \{a, n_a\}_{\mathsf{pub}(I)} \Vdash \{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$$
$$T_0, \{a, n_a\}_{\mathsf{pub}(I)}, \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)} \Vdash \{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$$
$$T_0, \{a, n_a\}_{\mathsf{pub}(I)}, \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}, \{x_{n_b}\}_{\mathsf{pub}(I)}$$

Example - Needham Schroeder's Protocol (2)

Session 1: a / I			Session 2: a / b		
1		$\rightsquigarrow \{a, n_a\}_{\mathsf{pub}(I)}$			$\rightsquigarrow \{a, n'_a\}_{\mathsf{pub}(b)}$
3	$\{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x_{n_b}\}_{\mathsf{pub}(I)}$		$\{n'_a, x'_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x'_{n_b}\}_{\mathsf{pub}(b)}$
	$\{x_a, x_{n_a}\}_{\mathsf{pub}(I)}$	$\rightsquigarrow \{x_{n_a}, n_b\}_{\mathsf{pub}(x_a)}$	2	$\{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$	$\rightsquigarrow \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}$

$$T_0 = \{a,\ b,\ I,\ \mathsf{priv}(I),\ \mathsf{pub}(I),\ \mathsf{pub}(b),\ \mathsf{pub}(a)\}$$

Constraints System

$$T_0, \{a, n_a\}_{\mathsf{pub}(I)} \Vdash \{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$$
$$T_0, \{a, n_a\}_{\mathsf{pub}(I)}, \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)} \Vdash \{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$$
$$T_0, \{a, n_a\}_{\mathsf{pub}(I)}, \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}, \{x_{n_b}\}_{\mathsf{pub}(I)} \Vdash n'_b$$

Example - Needham Schroeder's Protocol (2)

Session 1: a / I			Session 2: a / b		
1		$\rightsquigarrow \{a, n_a\}_{\mathsf{pub}(I)}$			$\rightsquigarrow \{a, n'_a\}_{\mathsf{pub}(b)}$
3	$\{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x_{n_b}\}_{\mathsf{pub}(I)}$		$\{n'_a, x'_{n_b}\}_{\mathsf{pub}(a)}$	$\rightsquigarrow \{x'_{n_b}\}_{\mathsf{pub}(b)}$
	$\{x_a, x_{n_a}\}_{\mathsf{pub}(I)}$	$\rightsquigarrow \{x_{n_a}, n_b\}_{\mathsf{pub}(x_a)}$	2	$\{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$	$\rightsquigarrow \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}$

$$T_0 = \{a,\ b,\ I,\ \mathsf{priv}(I),\ \mathsf{pub}(I),\ \mathsf{pub}(b),\ \mathsf{pub}(a)\}$$

Constraints System

$$T_0, \{a, n_a\}_{\mathsf{pub}(I)} \Vdash \{x'_a, x'_{n_a}\}_{\mathsf{pub}(b)}$$
$$T_0, \{a, n_a\}_{\mathsf{pub}(I)}, \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)} \Vdash \{n_a, x_{n_b}\}_{\mathsf{pub}(a)}$$
$$T_0, \{a, n_a\}_{\mathsf{pub}(I)}, \{x'_{n_a}, n'_b\}_{\mathsf{pub}(x'_a)}, \{x_{n_b}\}_{\mathsf{pub}(I)} \Vdash n'_b$$

Solution

$$\sigma = \{x'_a \mapsto a,\ \ x'_{n_a} \mapsto n_a,\ \ x_{n_b} \mapsto n'_b\}$$

Roadmap of the Talk

I) Existing Results
- Results in the standard Dolev-Yao intruder model
- Constraints solving approach
- Relaxing the perfect cryptographic assumption

II) My Contributions
- Motivations
- Results
- Decision procedure

III) Works in Progress
- To develop a general framework
- To deal with non equivalence static constraints

Equational Theories and Existing Results

	Passive Attacker	Active Attacker
Exclusive or theory		
$x \oplus x = 0$	PTIME	Decidable / NP-complete
$x \oplus 0 = x$	[CLS03, CKRT03]	[CLS03] / [CKRT03]
Abelian group theory		
$x \times I(x) = 1$	NP	Decidable
$x \times 1 = x$	[CLS03]	[Shm04]
Diffie-Hellman theory		
$exp(x, 1) = x$	PTIME	Decidable / NP-complete
$exp(exp(x, y), z) = exp(x, y \times z)$	[CKRT03]	[Shm04] / [CKRT03]

Undecidability Result

Unification modulo an Equational Theory E

INPUT: Given 2 terms $u[x_1, \ldots, x_n]$ and $v[x_1, \ldots, x_n]$

OUTPUT: Yes iff there exists a substitution $\sigma = \{x_1 \mapsto M_1, \ldots, x_n \mapsto M_n\}$ such that

$$u\sigma =_E v\sigma$$

Undecidability Result

Unification modulo an Equational Theory E

INPUT: Given 2 terms $u[x_1, \ldots, x_n]$ and $v[x_1, \ldots, x_n]$

OUTPUT: Yes iff there exists a substitution $\sigma = \{x_1 \mapsto M_1, \ldots, x_n \mapsto M_n\}$ such that

$$u\sigma =_E v\sigma$$

Protocol

$$\begin{array}{llcl} B: & x_1, \ldots, x_n & \rightarrow & \{u[x_1, \ldots, x_n], v[x_1, \ldots, x_n]\}_{Kab} \\ A: & \{x, x\}_{Kab} & \rightarrow & secret \end{array}$$

$secret$ is secret $\Longleftrightarrow$ u and v have no E-unifier

Undecidability Result

Unification modulo an Equational Theory E

INPUT: Given 2 terms $u[x_1, \ldots, x_n]$ and $v[x_1, \ldots, x_n]$

OUTPUT: Yes iff there exists a substitution $\sigma = \{x_1 \mapsto M_1, \ldots, x_n \mapsto M_n\}$ such that

$$u\sigma =_E v\sigma$$

Protocol

$$\begin{array}{lcll} B: & x_1, \ldots, x_n & \rightarrow & \{u[x_1, \ldots, x_n], v[x_1, \ldots, x_n]\}_{Kab} \\ A: & \{x, x\}_{Kab} & \rightarrow & secret \end{array}$$

$secret$ is secret $\Longleftrightarrow$ u and v have no E-unifier

unification problem undecidable in $E \Rightarrow$ secrecy problem undecidable in E

Roadmap of the Talk

I) Existing Results
- Results in the standard Dolev-Yao intruder model
- Constraints solving approach
- Relaxing the perfect cryptographic assumption

II) My Contributions
- Motivations
- Results
- Decision procedure

III) Works in Progress
- To develop a general framework
- To deal with non equivalence static constraints

Protocol Specification (1)

Denning-Sacco Protocol (1981)

$$\begin{array}{lcll} A & \rightarrow & B: & A, \{\{K_{ab}\}^a_{\mathsf{priv}(A)}\}^a_{\mathsf{pub}(B)} \\ B & \rightarrow & A: & \{Secret\}^s_{K_{ab}} \end{array}$$

Usually, B's role: $x_A, \{\{x_{Kab}\}^a_{\mathsf{priv}(x_A)}\}^a_{\mathsf{pub}(b)} \quad \rightarrow \quad \{secret\}^s_{x_{Kab}}$

$\rightsquigarrow$ attack with 2 sessions between A/I and $I(A)/B$.

Here, B's role: $x \quad \rightarrow \quad \{secret\}^s_{da(da(\pi_2(x),\mathsf{pub}(b)^{-1}),\mathsf{pub}(\pi_1(x)))}$

Protocol Specification (1)

Denning-Sacco Protocol (1981)

$$\begin{array}{lcll} A & \rightarrow & B: & A, \{\{K_{ab}\}^a_{\mathsf{priv}(A)}\}^a_{\mathsf{pub}(B)} \\ B & \rightarrow & A: & \{Secret\}^s_{K_{ab}} \end{array}$$

Usually, B's role: $x_A, \{\{x_{Kab}\}^a_{\mathsf{priv}(x_A)}\}^a_{\mathsf{pub}(b)} \quad \rightarrow \quad \{secret\}^s_{x_{Kab}}$
$\rightsquigarrow$ attack with 2 sessions between A/I and $I(A)/B$.

Here, B's role: $x \quad \rightarrow \quad \{secret\}^s_{da(da(\pi_2(x),\mathsf{pub}(b)^{-1}),\mathsf{pub}(\pi_1(x)))} \qquad [\pi_1(x) = a]$

Protocol Specification (1)

Denning-Sacco Protocol (1981)

$$\begin{array}{rcll} A & \rightarrow & B: & A, \{\{K_{ab}\}^a_{\mathsf{priv}(A)}\}^a_{\mathsf{pub}(B)} \\ B & \rightarrow & A: & \{Secret\}^s_{K_{ab}} \end{array}$$

Usually, B's role: $x_A, \{\{x_{Kab}\}^a_{\mathsf{priv}(x_A)}\}^a_{\mathsf{pub}(b)} \quad \rightarrow \quad \{secret\}^s_{x_{Kab}}$

$\rightsquigarrow$ attack with 2 sessions between A/I and $I(A)/B$.

Here, B's role: $x \quad \rightarrow \quad \{secret\}^s_{da(da(\pi_2(x),\mathsf{pub}(b)^{-1}),\mathsf{pub}(\pi_1(x)))}$

$\rightsquigarrow$ attack with 1 session between $I(A)$ and B.

No ambiguities and **no implicit checks** allow us to found new attacks.

Protocol Specification (2)

Protocol [Tur03]

$$\begin{array}{lcll} A & \rightarrow & B: & \{M, B\}^s_K \\ B & \rightarrow & A: & B \\ A & \rightarrow & B: & K \\ B & \rightarrow & A: & M \end{array}$$

Usually, B's role can not be expressed.

Here, B's role: $\begin{array}{lcll} x & \rightarrow & x_B & \\ y & \rightarrow & \pi_1(sd(x,y)) & \quad [\pi_2(sd(x,y)) = x_B] \end{array}$

Explicit destructors and **equations** allow to address a broader class of protocol.

Summary

Goal:

Decidability and complexity results for the insecurity problems for general equational theories:

- flexible intruder power: $sd(\{x\}_y, y) = x$
- flexible cryptographic properties: $\{\{x\}^s_y\}^s_y = x$

Summary

Goal:

Decidability and complexity results for the insecurity problems for general equational theories:

- flexible intruder power: $sd(\{x\}_y, y) = x$
- flexible cryptographic properties: $\{\{x\}^s_y\}^s_y = x$

Related Work:

- Many works for a fixed theory (ex: exclusive or theory): [Comon-Lundh&Shmatikov,03][Chevalier *et al.*,03].
- For general theories, very few results: [Comon&Treinen,03][Abadi&Cortier,04].
  $\hookrightarrow$ they only consider the intruder deduction problem
- Explicit destructors: [Millen,03][Lynch&Meadows,04].
- Equality tests: [Corin&Estalle,02].

Roadmap of the Talk

I) Existing Results
- Results in the standard Dolev-Yao intruder model
- Constraints solving approach
- Relaxing the perfect cryptographic assumption

II) My Contributions
- Motivations
- Results
- Decision procedure

III) Works in Progress
- To develop a general framework
- To deal with non equivalence static constraints

Public-Collapsing Theories

We consider theories that can be represented by convergent public-collapsing rewriting systems, *i.e.* each rule $\ell \to r$ satisfies:

1. r is a variable or a ground term,
2. if $head(\ell) \in \mathcal{VF}$, then under each public symbol in ℓ, there is r.

Public-Collapsing Theories

We consider theories that can be represented by convergent public-collapsing rewriting systems, *i.e.* each rule $\ell \to r$ satisfies:

1. r is a variable or a ground term,
2. if $head(\ell) \in \mathcal{VF}$, then under each public symbol in ℓ, there is r.

Exemples:

Dolev-Yao theory: $sd(\{x\}^s_y, y) \to x, \;\; ad(\{x\}^a_y, y^{-1}) \to x, ad(\{x\}^a_{y^{-1}}, y) \to x$

$$\pi_i(\langle x_1, x_2 \rangle) \to x_i \; (i = 1, 2), x^{-1^{-1}} \to x$$

Inverse-key theory: $\{sd(x, y)\}^a_y \;\to\; x, \{ad(x, y)\}^a_{y^{-1}} \to x, \{ad(x, y^{-1})\}^a_y \to x.$

Theory of involution: $\{\{x\}^a_y\}^a_{y^{-1}} \to x, \;\; \{\{x\}^a_{y^{-1}}\}^a_y \to x.$

Probabilistic encryption: $dec(enc(m, k, r), k) \to m$

Our Main Result

Main Result:
A NP decision procedure for the resolution of well-formed sets of equations $(t_1 = t_2)$ and deduction constraints $(s_1, \ldots, s_m \Vdash t)$ modulo a convergent public-collapsing rewriting system $\mathcal{R}$.

Our procedure provides:

- a generic NP decision procedure to solve the insecurity problem for a whole class of equational theories (but not xor).
- the use of destructors symbols (ad, sd, π_1, π_2) and explicit checks (equations) in protocol specifications.

Roadmap of the Talk

I) Existing Results
- Results in the standard Dolev-Yao intruder model
- Constraints solving approach
- Relaxing the perfect cryptographic assumption

II) My Contributions
- Motivations
- Results
- Decision procedure

III) Works in Progress
- To develop a general framework
- To deal with non equivalence static constraints

Insecurity via Constraints Solving

Denning-Sacco Protocol (1981)

$$\begin{array}{lcll} A & \rightarrow & B: & A, \{\{K_{ab}\}^a_{\mathsf{priv}(A)}\}^a_{\mathsf{pub}(B)} \\ B & \rightarrow & A: & \{Secret\}^s_{K_{ab}} \end{array}$$

Insecurity via Constraints Solving

Denning-Sacco Protocol (1981)

$$\begin{array}{ccccl} A & \rightarrow & B: & A, \{\{K_{ab}\}^a_{\mathsf{priv}(A)}\}^a_{\mathsf{pub}(B)} \\ B & \rightarrow & A: & \{Secret\}^s_{K_{ab}} \end{array}$$

Session $A = a; B = b; K_{ab} = k; Secret = s$

$$\begin{array}{llccl} (Role\ A) & \mathbf{a}: & & \rightarrow & \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)} \rangle \\ (Role\ B) & \mathbf{b}: & x & \rightarrow & \{s\}^s_{da(da(\pi_2(x), \mathsf{pub}(b)^{-1}), \mathsf{pub}(\pi_1(x)))} \end{array}$$

Insecurity via Constraints Solving

Denning-Sacco Protocol (1981)

$$\begin{array}{lcll} A & \rightarrow & B: & A, \{\{K_{ab}\}^a_{\mathsf{priv}(A)}\}^a_{\mathsf{pub}(B)} \\ B & \rightarrow & A: & \{Secret\}^s_{K_{ab}} \end{array}$$

Session $A = a; B = b; K_{ab} = k; Secret = s$

$$\begin{array}{llccl} (Role\ A) & \mathbf{a}: & & \rightarrow & \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle \\ (Role\ B) & \mathbf{b}: & x & \rightarrow & \{s\}^s_{da(da(\pi_2(x),\mathsf{pub}(b)^{-1}),\mathsf{pub}(\pi_1(x)))} \end{array}$$

Constraint System $\mathcal{P}$

$$T_1 = IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle$$

$$T_2 = T_1, \{s\}^s_{da(da(\pi_2(x),\mathsf{pub}(b)^{-1}),\mathsf{pub}(\pi_1(x)))}$$

$$T_1 \Vdash x$$

$$T_2 \Vdash y \qquad y = s$$

Constraint Solving Problem

Input:

A finite set $\mathcal{P}$ of constraints:

- equations
- intruder contraints $s_1, \ldots, s_n \Vdash x$

Output:

Does there exist a $\mathcal{R}$-*solution*, ***i.e.*** a substitution θ such that:

- for each $t_1 = t_2 \in \mathcal{P}$, $t_1\theta =_{\mathcal{R}} t_2\theta$,
- for each $s_1, \ldots, s_n \Vdash x$ in $\mathcal{P}$, there exists a public context C such that $C[s_1\theta, \ldots, s_n\theta] =_{\mathcal{R}} x\theta$.

Constraint Solving System - $\mathcal{T}$

$$(\mathcal{P};\, \mathcal{C};\; \mathcal{S})$$

Constraint Solving System - $\mathcal{T}$

$$(\mathcal{P};\, \mathcal{C};\; \mathcal{S})$$

Narrowing
Unification Syntactic

$$\frac{\mathcal{P} \cup \{e[u]\};\; \mathcal{C};\; \mathcal{S}}{\mathcal{P} \cup \{e[r]\};\; \mathcal{C}\eta;\; \mathcal{S}\eta \cup \eta}$$

Narrowing
e is an equation or an intruder constraint,
$l \to r$ is a fresh variant of a rule of $\mathcal{R}$,
$\eta = mgu(l\mathcal{S}, u\mathcal{S})$,
$root(l) = root(u)$.

$$\frac{\mathcal{P} \cup \{t_1 = t_2\};\; \mathcal{C};\; \mathcal{S}}{\mathcal{P};\; \mathcal{C}\eta;\; \mathcal{S}\eta \cup \eta}$$

Unification Syntactic
$\eta = mgu(t_1\mathcal{S}, t_2\mathcal{S})$.

Constraint Solving System - $\mathcal{T}$

Blocking

$$(\mathcal{P};\ \mathcal{C};\ \mathcal{S})$$

Narrowing
Unification Syntactic

$$\frac{\mathcal{P} \cup \{c\};\ \mathcal{C};\ \mathcal{S}}{\mathcal{P};\ \mathcal{C} \cup \{c\mathcal{S}\};\ \mathcal{S}}$$

Blocking
c is an intruder constraint.

Constraint Solving System - $\mathcal{T}$

Blocking

$$(\mathcal{P};\ \mathcal{C};\ \mathcal{S})$$

Narrowing
Unification Syntactic

Var. Elim.
Ground

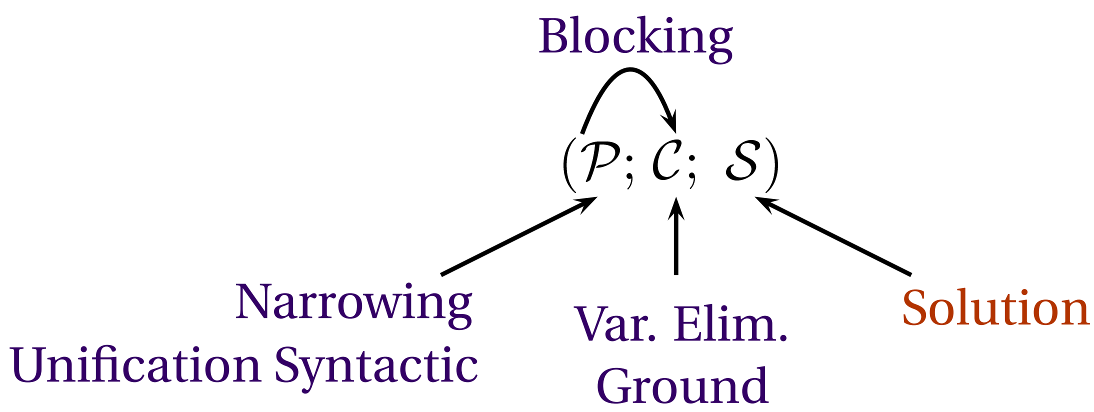
$$\frac{\mathcal{P};\ \mathcal{C};\ \mathcal{S}}{\mathcal{P};\ \mathcal{C}[x \mapsto t];\ \mathcal{S}[x \mapsto t] \cup [x \mapsto t]}$$

Variable Elimination
$x \in vars(\mathcal{C}), t \in st(\mathcal{C}) \setminus vars(\mathcal{C})$,
there is no occurence of x in t.

$$\frac{\mathcal{P};\ \mathcal{C} \cup \{T \Vdash u\};\ \mathcal{S}}{\mathcal{P};\ \mathcal{C};\ \mathcal{S}}$$

Ground
if $T \vdash_R u$.

Constraint Solving System - $\mathcal{T}$



$$\frac{\mathcal{P};\ \mathcal{C};\ \mathcal{S}}{\mathcal{P};\ \mathcal{C}[x \mapsto t];\ \mathcal{S}[x \mapsto t] \cup [x \mapsto t]}$$

Variable Elimination
$x \in vars(\mathcal{C}), t \in st(\mathcal{C}) \setminus vars(\mathcal{C})$,
there is no occurence of x in t.

$$\frac{\mathcal{P};\ \mathcal{C} \cup \{T \Vdash u\};\ \mathcal{S}}{\mathcal{P};\ \mathcal{C};\ \mathcal{S}}$$

Ground
if $T \vdash_R u$.

Denning-Sacco Example

$$
\begin{array}{ll}
\mathcal{P}: & IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle \Vdash x; \\
& IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle, \{s\}^s_{da(da(\pi_2(x),\mathsf{pub}(b)^{-1}),\mathsf{pub}(\pi_1(x)))} \Vdash y; \\
& y = s \\
\mathcal{C}: & \emptyset \\
\mathcal{S}: & \emptyset
\end{array}
$$

Denning-Sacco Example

$$\begin{array}{ll}
\mathcal{P}: & IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle \Vdash x; \\
 & IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle, \{s\}^s_{da(da(\pi_2(x),\mathsf{pub}(b)^{-1}),\mathsf{pub}(\pi_1(x)))} \Vdash y; \\
 & y = s \\
\mathcal{C}: & \emptyset \\
\mathcal{S}: & \emptyset
\end{array}$$

Step 1: Unification Syntactic

$$\frac{\mathcal{P} \cup \{t_1 = t_2\};\ \mathcal{C};\ \mathcal{S}}{\mathcal{P};\ \mathcal{C}\eta;\ \mathcal{S}\eta \cup \eta} \qquad \eta = mgu(t_1\mathcal{S}, t_2\mathcal{S})$$

Denning-Sacco Example

$\mathcal{P}$: $IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle \Vdash x;$

$IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle, \{s\}^s_{da(da(\pi_2(x),\mathsf{pub}(b)^{-1}),\mathsf{pub}(\pi_1(x)))} \Vdash y;$

$\mathcal{C}$: $\emptyset$

$\mathcal{S}$: $y = s$

Step 1: Unification Syntactic

$$\frac{\mathcal{P} \cup \{t_1 = t_2\};\ \mathcal{C};\ \mathcal{S}}{\mathcal{P};\ \mathcal{C}\eta;\ \mathcal{S}\eta \cup \eta} \qquad \eta = mgu(t_1\mathcal{S}, t_2\mathcal{S})$$

Denning-Sacco Example

$\mathcal{P}$: $IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle \Vdash x;$

$IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle, \{s\}^s_{da(da(\pi_2(x),\mathsf{pub}(b)^{-1}),\mathsf{pub}(\pi_1(x)))} \Vdash y;$

$\mathcal{C}$: $\emptyset$

$\mathcal{S}$: $y = s$

Step 2: Narrowing with $\pi_2(\langle x_1, x_2\rangle) \to x_2$

$$\frac{\mathcal{P} \cup \{e[u]\};\ \mathcal{C};\ \mathcal{S}}{\mathcal{P} \cup \{e[r]\};\ \mathcal{C}\eta;\ \mathcal{S}\eta \cup \eta}$$

$l \to r$ is a fresh variant of a rule of $\mathcal{R}$,
$\eta = mgu(l\mathcal{S}, u\mathcal{S})$,
$root(l) = root(u)$.

Denning-Sacco Example

$\mathcal{P}:$ $IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle \Vdash x;$

$IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle, \{s\}^s_{da(da(x_2,\mathsf{pub}(b)^{-1}),\mathsf{pub}(\pi_1(x)))} \Vdash y;$

$\mathcal{C}:$ $\emptyset$

$\mathcal{S}:$ $y = s,\ x = \langle x_1, x_2\rangle$

Step 2: Narrowing with $\pi_2(\langle x_1, x_2\rangle) \rightarrow x_2$

$$\frac{\mathcal{P} \cup \{e[u]\};\ \mathcal{C};\ \mathcal{S}}{\mathcal{P} \cup \{e[r]\};\ \mathcal{C}\eta;\ \mathcal{S}\eta \cup \eta}$$

$l \rightarrow r$ is a fresh variant of a rule of $\mathcal{R}$,
$\eta = mgu(l\mathcal{S}, u\mathcal{S})$,
$root(l) = root(u)$.

Denning-Sacco Example

$\mathcal{P}$: $IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle \Vdash x;$

$IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle, \{s\}^s_{da(da(x_2, \mathsf{pub}(b)^{-1}), \mathsf{pub}(\pi_1(x)))} \Vdash y;$

$\mathcal{C}$: $\emptyset$

$\mathcal{S}$: $y = s,\ x = \langle x_1, x_2\rangle$

Step 2: Narrowing with $\pi_2(\langle x_1, x_2\rangle) \to x_2$

$$\frac{\mathcal{P} \cup \{e[u]\};\ \mathcal{C};\ \mathcal{S}}{\mathcal{P} \cup \{e[r]\};\ \mathcal{C}\eta;\ \mathcal{S}\eta \cup \eta}$$

$l \to r$ is a fresh variant of a rule of $\mathcal{R}$,
$\eta = mgu(l\mathcal{S}, u\mathcal{S})$,
$root(l) = root(u)$.

. . . .

Denning Sacco Example

Initial System

$$\begin{array}{ll}\mathcal{P}: & IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle \Vdash x; \\ & IK, \langle a, \{\{k\}^a_{\mathsf{pub}(a)^{-1}}\}^a_{\mathsf{pub}(b)}\rangle, \{s\}^s_{da(da(\pi_2(x),\mathsf{pub}(b)^{-1}),\mathsf{pub}(\pi_1(x)))} \Vdash y; \\ & y = s \\ \mathcal{C}: & \emptyset \\ \mathcal{S}: & \emptyset \end{array}$$

Final System

$$\begin{array}{ll}\mathcal{P}: & \emptyset \\ \mathcal{C}: & \emptyset \\ \mathcal{S}: & y = s, x = \langle a, \{0\}_{\mathsf{pub}(b)}\rangle, \cdots \end{array}$$

Results

Given a convergent and public-collapsing system, we have the following results:

Termination: The application of the inference rules terminates and the depth, the branching degree and the size of the node of the derivation tree are polynomially bounded.

Soundness: The reduction rules do not introduce new solution.

Completeness: If the initial constraint set is satisfiable then among all the simple constraints sets produced by $\mathcal{T}$, there is at least one satisfiable set.

Corollary:
A NP decision procedure to solve the insecurity problem in presence of an active attacker as soon as the intruder capabilities and the algebraic properties of the cryptographic primitives can be represented by a public-collapsing rewriting system.

Roadmap of the Talk

I) Existing Results

- Results in the standard Dolev-Yao intruder model
- Constraints solving approach
- Relaxing the perfect cryptographic assumption

II) My Contributions

- Motivations
- Results
- Decision procedure

III) Works in Progress

- To develop a general framework
- To deal with non equivalence static constraints

Towards a General Compositional Result

Protocol rules

$$\begin{array}{ccc} u_1 & \to & v_1 \\ u_2 & \to & v_2 \\ & \cdots & \\ u_n & \to & v_n \end{array}$$

Intruder theory $\mathcal{I}$

$$\frac{T \vdash u \quad T \vdash v}{T \vdash \{u\}_v}$$

$$\frac{T \vdash u \quad u =_E v}{T \vdash v}$$

Equational theory E

$$\begin{array}{rcl} x + 0 & = & x \\ x + x & = & 0 \\ x + y & = & y + x \\ x + (y + z) & = & (x + y) + z \end{array}$$

Towards a General Compositional Result

Protocol rules

$$\begin{array}{ccc} u_1 & \to & v_1 \\ u_2 & \to & v_2 \\ & \cdots & \\ u_n & \to & v_n \end{array}$$

Intruder theory $\mathcal{I}$

$$\frac{T \vdash u \quad T \vdash v}{T \vdash \{u\}_v \downarrow}$$

$$\frac{T \vdash u \quad u =_E v}{T \vdash v}$$

Equational theory E

$$\left.\begin{array}{rcl} x + 0 & \to & x \\ x + x & \to & 0 \end{array}\right\} \mathcal{E}_1$$

$$\left.\begin{array}{rcl} x + y & = & y + x \\ x + (y + z) & = & (x + y) + z \end{array}\right\} \mathcal{E}_2$$

Towards a General Compositional Result

Protocol rules

$$\begin{array}{rcl} u_1 & \to & v_1 \\ u_2 & \to & v_2 \\ & \dots & \\ u_n & \to & v_n \end{array}$$

Intruder theory $\mathcal{I}$

$$\frac{T \vdash u \quad T \vdash v}{T \vdash \{u\}_v \downarrow}$$

$$\frac{T \vdash u \quad u =_E v}{T \vdash v}$$

Equational theory E

$$\left.\begin{array}{rcl} x + 0 & \to & x \\ x + x & \to & 0 \end{array}\right\} \mathcal{E}_1$$

$$\left.\begin{array}{rcl} x + y & = & y + x \\ x + (y + z) & = & (x + y) + z \end{array}\right\} \mathcal{E}_2$$

Constraint System

$$\left.\begin{array}{rcl} T_0 & \Vdash & u_1 \\ T_0, v_1 & \Vdash & u_2 \\ \dots & & \\ T_0, v_1, \dots, v_n & \Vdash & s \end{array}\right\} \quad \mathcal{C}$$

Towards a General Compositional Result

Protocol rules

$$\begin{array}{ccc} u_1 & \rightarrow & v_1 \\ u_2 & \rightarrow & v_2 \\ & \cdots & \\ u_n & \rightarrow & v_n \end{array}$$

Intruder theory $\mathcal{I}$

$$\frac{T \vdash u \quad T \vdash v}{T \vdash \{u\}_v \downarrow}$$

$$\frac{T \vdash u \quad u =_E v}{T \vdash v}$$

Equational theory E

$$\left.\begin{array}{rcl} x + 0 & \rightarrow & x \\ x + x & \rightarrow & 0 \end{array}\right\} \mathcal{E}_1$$

$$\left.\begin{array}{rcl} x + y & = & y + x \\ x + (y + z) & = & (x + y) + z \end{array}\right\} \mathcal{E}_2$$

Constraint System

$$\left.\begin{array}{rcl} T_0 & \Vdash & u_1 \\ T_0, v_1 & \Vdash & u_2 \\ \cdots & & \\ T_0, v_1, \ldots, v_n & \Vdash & s \end{array}\right\} \mathcal{C} \quad \begin{array}{c} \nearrow \\ \rightarrow \\ \searrow \end{array} \quad \left.\begin{array}{c} \mathcal{C}_1 \\ \vdots \\ \vdots \\ \mathcal{C}_n \end{array}\right\} \text{variants of } \mathcal{C}$$

- E satisfies the finite variant property $\Longrightarrow \forall \sigma \exists i, \tau$ such that $\mathcal{C}\sigma\downarrow =_{\mathcal{E}_2} \mathcal{C}_i\tau$

Towards a General Compositional Result

Protocol rules | Intruder theory $\mathcal{I}$ | Equational theory E

$$\begin{array}{ccc} u_1 & \to & v_1 \\ u_2 & \to & v_2 \\ & \cdots & \\ u_n & \to & v_n \end{array}$$

$$\frac{T \vdash u \quad T \vdash v}{T \vdash \{u\}_v \downarrow}$$

$$\frac{T \vdash u \quad u =_E v}{T \vdash v}$$

$$\left.\begin{array}{rcl} x + 0 & \to & x \\ x + x & \to & 0 \end{array}\right\} \mathcal{E}_1$$

$$\left.\begin{array}{rcl} x + y & = & y + x \\ x + (y + z) & = & (x + y) + z \end{array}\right\} \mathcal{E}_2$$

Constraint System

$$\left.\begin{array}{rcl} T_0 & \Vdash & u_1 \\ T_0, v_1 & \Vdash & u_2 \\ \cdots & & \\ T_0, v_1, \ldots, v_n & \Vdash & s \end{array}\right\} \mathcal{C} \quad \begin{array}{c} \nearrow \\ \to \\ \searrow \end{array} \quad \left.\begin{array}{c} \mathcal{C}_1 \\ \vdots \\ \vdots \\ \mathcal{C}_n \end{array}\right\}$$

variants of $\mathcal{C}$

- E satisfies the finite variant property $\Longrightarrow \forall \sigma \exists i, \tau$ such that $\mathcal{C}\sigma\downarrow =_{\mathcal{E}_2} \mathcal{C}_i\tau$
- $\mathcal{I}$ satisfies the locality properties

Roadmap of the Talk

I) Existing Results

- Results in the standard Dolev-Yao intruder model
- Constraints solving approach
- Relaxing the perfect cryptographic assumption

II) My Contributions

- Motivations
- Results
- Decision procedure

III) Works in Progress

- To develop a general framework
- To deal with non equivalence static constraints

Static Equivalence Constraints

Example

$$A \quad \to \quad S: \quad \{yes\}_{\mathsf{pub}(S)}$$

$\leadsto$ intruder's knowledge (frame): $\phi = \nu\tilde{n}\{M_1/_{x_1}, \ldots, M_n/_{x_l}\}$

Static Equivalence Constraints

Example

$$A \quad \to \quad S: \quad \{yes\}_{\mathsf{pub}(S)}$$

$\leadsto$ intruder's knowledge (frame): $\phi = \nu\tilde{n}\{M_1/_{x_1}, \ldots, M_n/_{x_l}\}$

Static equivalence $\approx_s$

$\phi_1 \approx_s \phi_2$ iff for every couple of term (M, N) s.t. $vars(M, N) \subseteq \{x_1, \ldots, x_l\}$,

$$M\phi_1 =_E N\phi_1 \Leftrightarrow M\phi_2 =_E N\phi_2.$$

$\phi_1 = \{\{yes\}_{\mathsf{pub}(S)}/_x\}$ and $\phi_2 = \{\{no\}_{\mathsf{pub}(S)}/_x\}$ are not statically equivalent

$$M = \{yes\}_{\mathsf{pub}(S)} \text{ and } N = x$$

Static Equivalence Constraints

Example

$$A \quad \rightarrow \quad S: \quad \{yes\}_{\mathsf{pub}(S)}$$

$\rightsquigarrow$ intruder's knowledge (frame): $\phi = \nu\tilde{n}\{M_1/_{x_1}, \dots, M_n/_{x_l}\}$

Static equivalence $\approx_s$

$\phi_1 \approx_s \phi_2$ iff for every couple of term (M, N) s.t. $vars(M, N) \subseteq \{x_1, \dots, x_l\}$,

$$M\phi_1 =_E N\phi_1 \Leftrightarrow M\phi_2 =_E N\phi_2.$$

$\phi_1 = \{\{yes\}_{\mathsf{pub}(S)}/_x\}$ and $\phi_2 = \{\{no\}_{\mathsf{pub}(S)}/_x\}$ are not statically equivalent

$$M = \{yes\}_{\mathsf{pub}(S)} \text{ and } N = x$$

Dictionary (guessing) attacks [Corin et al.,04]

Given:
- a frame $\nu\tilde{n}.\{M_1/_{x_1}, \dots, M_n/_{x_n}\}$
- a guessable data p

$$\nu\tilde{n}.\{M_1/_{x_1}, \dots, M_n/_{x_n}\} \overset{?}{\not\approx}_s \nu\tilde{n}, p.\{M_1/_{x_1}, \dots, M_n/_{x_n}\}$$