

# A Decision Procedure for the Verification of Security Protocols with Explicit Destructors

Stéphanie Delaune<sup>1</sup> Florent Jacquemard<sup>2</sup>

<sup>1</sup>France Télécom R & D and LSV  
Cachan, France

<sup>2</sup>INRIA and LSV  
Cachan, France

ACM Computer and Communications Security 2004

# Formal Methods for Cryptographic Protocols Verification

- **abstract model**: black box cryptographic functions (perfect cryptography assumption)
- restricted to logical attacks
- automatic proofs
- verification:
  - with general purpose techniques / tools (first / higher order theorem proving, model checking, symbolic constraint solving...)
  - with ad hoc algorithms

# Formal Methods for Cryptographic Protocols Verification

- abstract model: black box cryptographic functions (perfect cryptography assumption)
- restricted to **logical attacks**
- automatic proofs
- verification:
  - with general purpose techniques / tools (first / higher order theorem proving, model checking, symbolic constraint solving...)
  - with ad hoc algorithms

# Formal Methods for Cryptographic Protocols Verification

- abstract model: black box cryptographic functions (perfect cryptography assumption)
- restricted to logical attacks
- automatic proofs
- verification:
  - with general purpose techniques / tools (first / higher order theorem proving, model checking, symbolic constraint solving...)
  - with ad hoc algorithms

# Formal Methods for Cryptographic Protocols Verification

- abstract model: black box cryptographic functions (perfect cryptography assumption)
- restricted to logical attacks
- automatic proofs
- verification:
  - with general purpose techniques / tools (first / higher order theorem proving, model checking, symbolic constraint solving...)
  - with ad hoc algorithms

# Formal Methods for Cryptographic Protocols Verification

- abstract model: black box cryptographic functions (perfect cryptography assumption)
- restricted to logical attacks
- automatic proofs
- verification:
  - with general purpose techniques / tools (first / higher order theorem proving, model checking, symbolic constraint solving...)
  - with ad hoc algorithms

# Formal Methods for Cryptographic Protocols Verification

- abstract model: black box cryptographic functions (perfect cryptography assumption)
- restricted to logical attacks
- automatic proofs
- verification:
  - with general purpose techniques / tools (first / higher order theorem proving, model checking, symbolic constraint solving...)
  - with ad hoc algorithms

# Abstract Model

- definition of a signature: set of cryptographic primitives abstracted as function symbols,
- the messages are first order terms over the signature, ex:  $ae(\text{pair}(A, \text{hash}(M)), \text{pub}(B))$
- unique (insecure) communication channel,
- concurrent honest agents following sequences of rules:  $\text{rcv}(r).\text{snd}\langle s \rangle$ ,
- attacker's knowledge represented by a set of messages, updated by:
  - wire-tapping messages in the channel (passive attacker),
  - replaying known messages with impersonation (active att.),
  - deductions from the messages collected. e.g.  $\frac{\text{enc}(X, Y) \quad Y}{X}$

# Abstract Model

- definition of a **signature**: set of cryptographic primitives abstracted as function symbols,
- the messages are first order terms over the signature,  
ex:  $ae(\text{pair}(A, \text{hash}(M)), \text{pub}(B))$
- unique (insecure) communication channel,
- concurrent honest agents following sequences of rules:  
 $\text{rcv}(r).\text{snd}\langle s \rangle$ ,
- attacker's knowledge represented by a set of messages, updated by:
  - wire-tapping messages in the channel (passive attacker),
  - replaying known messages with impersonation (active att.),
  - deductions from the messages collected. e.g.  $\frac{\text{enc}(X, Y) \quad Y}{X}$

# Abstract Model

- definition of a **signature**: set of cryptographic primitives abstracted as function symbols,
- the **messages** are first order terms over the signature,  
ex:  $\text{ae}(\text{pair}(A, \text{hash}(M)), \text{pub}(B))$
- unique (insecure) communication channel,
- concurrent honest agents following sequences of rules:  
 $\text{rcv}(r).\text{snd}\langle s \rangle$ ,
- attacker's knowledge represented by a set of messages, updated by:
  - wire-tapping messages in the channel (passive attacker),
  - replaying known messages with impersonation (active att.),
  - deductions from the messages collected. e.g.  $\frac{\text{enc}(X, Y) \quad Y}{X}$

# Abstract Model

- definition of a **signature**: set of cryptographic primitives abstracted as function symbols,
- the **messages** are first order terms over the signature,  
ex:  $ae(\text{pair}(A, \text{hash}(M)), \text{pub}(B))$
- **unique (insecure) communication channel**,
- concurrent honest agents following sequences of rules:  
 $\text{rcv}(r).\text{snd}\langle s \rangle$ ,
- attacker's knowledge represented by a set of messages, updated by:
  - wire-tapping messages in the channel (passive attacker),
  - replaying known messages with impersonation (active att.),
  - deductions from the messages collected. e.g.  $\frac{\text{enc}(X, Y) \quad Y}{X}$

# Abstract Model

- definition of a **signature**: set of cryptographic primitives abstracted as function symbols,
- the **messages** are first order terms over the signature, ex:  $ae(\text{pair}(A, \text{hash}(M)), \text{pub}(B))$
- unique (insecure) communication **channel**,
- concurrent honest **agents** following sequences of rules:  $\text{rcv}(r).\text{snd}\langle s \rangle$ ,
- attacker's knowledge represented by a set of messages, updated by:
  - wire-tapping messages in the channel (passive attacker),
  - replaying known messages with impersonation (active att.),
  - deductions from the messages collected. e.g.  $\frac{\text{enc}(X, Y) \quad Y}{X}$

# Abstract Model

- definition of a **signature**: set of cryptographic primitives abstracted as function symbols,
- the **messages** are first order terms over the signature,  
ex:  $\text{ae}(\text{pair}(A, \text{hash}(M)), \text{pub}(B))$
- unique (insecure) communication **channel**,
- concurrent honest **agents** following sequences of rules:  
 $\text{rcv}(r).\text{snd}\langle s \rangle,$
- **attacker's** knowledge represented by a set of messages, updated by:
  - wire-tapping messages in the channel (passive attacker),
  - replaying known messages with impersonation (active att.),
  - deductions from the messages collected. e.g.  $\frac{\text{enc}(X, Y) \quad Y}{X}$

# Abstract Model

- definition of a **signature**: set of cryptographic primitives abstracted as function symbols,
- the **messages** are first order terms over the signature, ex:  $ae(\text{pair}(A, \text{hash}(M)), \text{pub}(B))$
- unique (insecure) communication **channel**,
- concurrent honest **agents** following sequences of rules:  $\text{rcv}(r).\text{snd}\langle s \rangle$ ,
- **attacker's** knowledge represented by a set of messages, updated by:
  - wire-tapping messages in the channel (**passive** attacker),
  - replaying known messages with impersonation (active att.),
  - deductions from the messages collected. e.g.  $\frac{\text{enc}(X, Y) \quad Y}{X}$

# Abstract Model

- definition of a **signature**: set of cryptographic primitives abstracted as function symbols,
- the **messages** are first order terms over the signature,  
ex:  $ae(\text{pair}(A, \text{hash}(M)), \text{pub}(B))$
- unique (insecure) communication **channel**,
- concurrent honest **agents** following sequences of rules:  
 $\text{rcv}(r).\text{snd}\langle s \rangle$ ,
- **attacker's** knowledge represented by a set of messages, updated by:
  - wire-tapping messages in the channel (**passive** attacker),
  - replaying known messages with impersonation (**active** att.),
  - deductions from the messages collected. e.g.  $\frac{\text{enc}(X, Y) \quad Y}{X}$

# Abstract Model

- definition of a **signature**: set of cryptographic primitives abstracted as function symbols,
- the **messages** are first order terms over the signature,  
ex:  $\text{ae}(\text{pair}(A, \text{hash}(M)), \text{pub}(B))$
- unique (insecure) communication **channel**,
- concurrent honest **agents** following sequences of rules:  
 $\text{rcv}(r).\text{snd}\langle s \rangle,$
- **attacker's** knowledge represented by a set of messages, updated by:
  - wire-tapping messages in the channel (**passive** attacker),
  - replaying known messages with impersonation (**active** att.),
  - **deductions** from the messages collected. e.g.  $\frac{\text{enc}(X, Y) \quad Y}{X}$

# Results

**Problem of secrecy:** reachability of a state where a secret data fell in the attacker's knowledge set.

# Results

**Problem of secrecy:** reachability of a state where a secret data fell in the attacker's knowledge set.

- undecidable in the deduction model of *Dolev-Yao*  
[Durgin, Lincoln, Mitchell, Scedrov 1999]
- NP-complete if the number of agents is bounded,  
[M. Rusinowitch, M. Turuani 2001].
- NP-complete in a model extended with equations for exclusive or, Diffie-Hellman exponentiation. . .  
[Y. Chevalier, R. Kuester, M. Rusinowitch, M. Turuani 2003].

# Results

**Problem of secrecy:** reachability of a state where a secret data fell in the attacker's knowledge set.

- undecidable in the deduction model of *Dolev-Yao*  
[Durgin, Lincoln, Mitchell, Scedrov 1999]
- NP-complete if the number of agents is bounded,  
[M. Rusinowitch, M. Turuani 2001].
- NP-complete in a model extended with equations for exclusive or, Diffie-Hellman exponentiation. . .  
[Y. Chevalier, R. Kuester, M. Rusinowitch, M. Turuani 2003].

Each result for a particular deduction model.

# General Framework

Based on an [equational theory](#)  $\mathcal{E}$  specifying the cryptographic primitives (following [applied pi-calculus](#)).

# General Framework

Based on an [equational theory](#)  $\mathcal{E}$  specifying the cryptographic primitives (following [applied pi-calculus](#)).

ex.:

$$\begin{aligned} \text{dec}(\text{enc}(x, y), y) &= x \\ \text{ad}(\text{ae}(x, y), \text{inv}(y)) &= x & \text{inv}(\text{inv}(x)) &= x \\ \text{ad}(\text{ae}(x, \text{inv}(y)), y) &= x \\ \text{fst}(\text{pair}(x_1, x_2)) &= x_1 \\ \text{snd}(\text{pair}(x_1, x_2)) &= x_2 \end{aligned}$$

# General Framework (2)

- More flexible.

Attacker deduction abilities:

- apply public symbols to known messages
- apply equations of  $\mathcal{E}$ .

$$\text{ex: } \frac{\text{enc}(x, y) \quad y}{\text{dec}(\text{enc}(x, y), y) =_{\mathcal{E}} x}$$

- More expressive.

Agents are sequence of

$\text{rcv}(x).\text{if } u_1 = v_1, \dots \text{ then } \text{snd}\langle s \rangle \text{ else abort}$

where the terms  $u_i, v_i, s$  can contain explicit destructor symbols  $\text{dec}, \text{ad}, \text{fst}, \text{snd} \dots$

# General Framework (2)

- More flexible.

**Attacker** deduction abilities:

- apply **public** symbols to known messages
- apply equations of  $\mathcal{E}$ .

$$\text{ex: } \frac{\text{enc}(x, y) \quad y}{\text{dec}(\text{enc}(x, y), y) =_{\mathcal{E}} x}$$

- More expressive.

Agents are sequence of

$\text{rcv}(x).\text{if } u_1 = v_1, \dots \text{ then } \text{snd}\langle s \rangle \text{ else abort}$

where the terms  $u_i, v_i, s$  can contain explicit destructor symbols  $\text{dec}, \text{ad}, \text{fst}, \text{snd} \dots$

# General Framework (2)

- More flexible.

Attacker deduction abilities:

- apply **public** symbols to known messages
- apply equations of  $\mathcal{E}$ .

$$\text{ex: } \frac{\text{enc}(x, y) \quad y}{\text{dec}(\text{enc}(x, y), y) =_{\mathcal{E}} x}$$

- More expressive.

**Agents** are sequence of

$\text{rcv}(x).\text{if } u_1 = v_1, \dots \text{ then } \text{snd}\langle s \rangle \text{ else abort}$

where the terms  $u_i, v_i, s$  can contain **explicit destructor** symbols  $\text{dec}, \text{ad}, \text{fst}, \text{snd} \dots$

It permits to capture more attacks.

- [J. Millen 2003], [C. Lynch, C. Meadows 2004].

# Example

Denning and Sacco Protocol (1981)

Exchange of a signed symmetric key  $K$ .

Protocol messages:

0.  $A \rightarrow B$ :  $\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$
1.  $B \rightarrow A$ :  $\text{enc}(S, K)$

# Example

Denning and Sacco Protocol (1981)

Exchange of a signed symmetric key  $K$ .

Protocol messages:

0.  $A \rightarrow B$ :  $\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$
1.  $B \rightarrow A$ :  $\text{enc}(S, K)$

agent  $A$ :

$$\nu k.\text{snd}\langle \text{pair}(A, \text{ae}(\text{ae}(k, \text{inv}(\text{pub}(A))), \text{pub}(B))) \rangle.\text{rcv}(x)$$

agent  $B$ :

$$\nu s.\text{rcv}(y).\text{snd}\langle \text{enc}(s, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$$

$\text{rcv}(y)$  corresponds to message reception without integrity checking.

# Example

Denning and Sacco Protocol (1981)

Exchange of a signed symmetric key  $K$ .

Protocol messages:

0.  $A \rightarrow B$ :  $\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$
1.  $B \rightarrow A$ :  $\text{enc}(S, K)$

agent  $A$ :

$$\nu k. \text{snd}\langle \text{pair}(A, \text{ae}(\text{ae}(k, \text{inv}(\text{pub}(A))), \text{pub}(B))) \rangle. \text{rcv}(x)$$

agent  $B$ :

$$\nu s. \text{rcv}(y). \text{snd}\langle \text{enc}(s, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$$

$\text{rcv}(y)$  corresponds to message reception without integrity checking.

# Example

Denning and Sacco Protocol (1981)

Exchange of a signed symmetric key  $K$ .

Protocol messages:

0.  $A \rightarrow B$ :  $\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$
1.  $B \rightarrow A$ :  $\text{enc}(S, K)$

agent  $A$ :

$$\text{snd}\langle \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))) \rangle.\text{rcv}(x)$$

agent  $B$ :

$$\nu s.\text{rcv}(y).\text{snd}\langle \text{enc}(s, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$$

$\text{rcv}(y)$  corresponds to message reception without integrity checking.

# Example

Denning and Sacco Protocol (1981)

Exchange of a signed symmetric key  $K$ .

Protocol messages:

0.  $A \rightarrow B$ :  $\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$
1.  $B \rightarrow A$ :  $\text{enc}(S, K)$

agent  $A$ :

$$\text{snd}\langle \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))) \rangle.\text{rcv}(x)$$

agent  $B$ :

$$\nu s.\text{rcv}(y).\text{snd}\langle \text{enc}(s, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$$

$\text{rcv}(y)$  corresponds to message reception without integrity checking.

# Example

Denning and Sacco Protocol (1981)

Exchange of a signed symmetric key  $K$ .

Protocol messages:

0.  $A \rightarrow B$ :  $\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$
1.  $B \rightarrow A$ :  $\text{enc}(S, K)$

agent  $A$ :

$$\text{snd}\langle \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))) \rangle.\text{rcv}(x)$$

agent  $B$ :

$$\text{rcv}(y).\text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$$

$\text{rcv}(y)$  corresponds to message reception without integrity checking.

# Example

Alice

$$\text{snd} \langle \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))) \rangle . \text{rcv}(x)$$


---

$K$

Insecure network

Bob

$$\text{rcv}(y) . \text{snd} \langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$$


---

$S$

# Example

Alice

$$\text{snd}\langle \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))) \rangle . \text{rcv}(x)$$


---

$K$

Insecure network

Bob

$$\text{rcv}(y). \text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$$


---

$S$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

Bob

$\text{rcv}(y).\text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$

---

$S$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

Bob

$\text{rcv}(y).\text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$

---

$S$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

---

Bob

$\text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

Bob

$\text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

rcv( $x$ )

---

$K$

Insecure network

$$\begin{aligned} & \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))), \\ & \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))), \\ & \quad \text{inv}(\text{pub}(B))), \\ & \quad \text{pub}(\text{fst}(\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))))) \end{aligned}$$

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$$\begin{aligned} & \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))), \\ & \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))), \\ & \quad \text{inv}(\text{pub}(B))), \\ & \quad \text{pub}(\text{fst}(\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))))) \end{aligned}$$

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

rcv( $x$ )

---

$K$

Insecure network

$$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))),$$

$$\text{enc}(S, \text{ad}(\text{ad}(\text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))),$$

$$\text{inv}(\text{pub}(B))),$$

$$\text{pub}(\text{fst}(\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))))))$$

Bob

---


$$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$$

# Example

Alice

rcv( $x$ )

---

$K$

Insecure network

pair( $A$ , ae(ae( $K$ , inv(pub( $A$ ))), pub( $B$ ))),  
 enc( $S$ , ad(ad(ae(ae( $K$ , inv(pub( $A$ ))), pub( $B$ ))), inv(pub( $B$ ))),

pub(fst(pair( $A$ , ae(ae( $K$ , inv(pub( $A$ ))), pub( $B$ ))))))

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))),$   
 $\text{enc}(S, \text{ad}(\text{ad}(\text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)), \text{inv}(\text{pub}(B))),$

$\text{pub}(\text{fst}(\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))))))$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))),$   
 $\text{enc}(S, \text{ad}(\text{ae}(K, \text{inv}(\text{pub}(A))),$

$\text{pub}(\text{fst}(\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))))))$

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))),$   
 $\text{enc}(S, \text{ad}(\text{ae}(K, \text{inv}(\text{pub}(A))),$

$\text{pub}(\text{fst}(\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))))))$

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))),$   
 $\text{enc}(S, \text{ad}(\text{ae}(K, \text{inv}(\text{pub}(A))),$

$\text{pub}(A))$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))),$   
 $\text{enc}(S, \text{ad}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(A)))$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))),$   
 $\text{enc}(S, \text{ad}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(A)))$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

$\text{rcv}(x)$

---

$K$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))),$   
 $\text{enc}(S, K)$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

$rcv(x)$

---

$K$

Insecure network

$pair(A, ae(ae(K, inv(pub(A))), pub(B))),$   
 $enc(S, K)$

---

Bob

---

$S, y = pair(A, ae(ae(K, inv(pub(A))), pub(B)))$

# Example

Alice

---

$K, \text{enc}(S, K)$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))),$   
 $\text{enc}(S, K)$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Example

Alice

---

$K, \text{enc}(S, K), S$

Insecure network

$\text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B))),$   
 $\text{enc}(S, K)$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(\text{ae}(K, \text{inv}(\text{pub}(A))), \text{pub}(B)))$

# Attack with one agent

---

Insecure network

$0, A, \text{pub}(A), \text{pub}(B)$

---

Bob

$\text{rcv}(y).\text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$

---

$S$

# Attack with one agent

---

Insecure network

$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

---

Bob

$\text{rcv}(y). \text{snd} \langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$

---

$S$

# Attack with one agent

---

Insecure network

$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

---

Bob

$\text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$

---

$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

# Attack with one agent

---

Insecure network

$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

---

Bob

$\text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$

---

$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

# Attack with one agent

---

Insecure network

$$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$$

$$\text{enc}(S, \text{ad}(\text{ad}(\text{snd}(\text{pair}(A, \text{ae}(0, \text{pub}(B))))), \text{inv}(\text{pub}(B))),$$

$$\text{pub}(\text{fst}(\text{pair}(A, \text{ae}(0, \text{pub}(B))))))$$


---

Bob

---


$$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$$

# Attack with one agent

---

Insecure network

$$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$$

$$\text{enc}(S, \text{ad}(\text{ad}(\text{snd}(\text{pair}(A, \text{ae}(0, \text{pub}(B))))) , \text{inv}(\text{pub}(B))),$$

$$\text{pub}(\text{fst}(\text{pair}(A, \text{ae}(0, \text{pub}(B)))))$$


---

Bob

---


$$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$$

# Attack with one agent

---

Insecure network

$$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$$
$$\text{enc}(S, \text{ad}(\text{ad}(\text{ae}(0, \text{pub}(B))), \text{inv}(\text{pub}(B))),$$
$$\text{pub}(\text{fst}(\text{pair}(A, \text{ae}(0, \text{pub}(B))))))$$

---

Bob

---

$$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$$

# Attack with one agent

---

Insecure network

$$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$$

$$\text{enc}(S, \text{ad}(\text{ad}(\text{ae}(0, \text{pub}(B)), \text{inv}(\text{pub}(B))),$$

$$\text{pub}(\text{fst}(\text{pair}(A, \text{ae}(0, \text{pub}(B))))))$$


---

Bob

---


$$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$$

# Attack with one agent

---

Insecure network

$$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$$

$$\text{enc}(S, \text{ad}(0, \text{pub}(\text{fst}(\text{pair}(A, \text{ae}(0, \text{pub}(B)))))))$$


---

Bob

---


$$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$$

# Attack with one agent

---

Insecure network

$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$   
 $\text{enc}(S, \text{ad}(0,$   
 $\quad \text{pub}(\text{fst}(\text{pair}(A, \text{ae}(0, \text{pub}(B)))))$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

# Attack with one agent

---

Insecure network

$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$   
 $\text{enc}(S, \text{ad}(0,$   
 $\quad \text{pub}(A)))$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

# Attack with one agent

---

Insecure network

$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$   
 $\text{enc}(S, \text{ad}(0, \text{pub}(A)))$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

# Attack with one agent

---

Insecure network

$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$   
 $\text{enc}(S, \text{ad}(0, \text{pub}(A))), \text{ad}(0, \text{pub}(A))$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

# Attack with one agent

---

Insecure network

$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$   
 $\text{enc}(S, \text{ad}(0, \text{pub}(A))), \text{ad}(0, \text{pub}(A)),$   
 $\text{ad}(\text{enc}(S, \text{ad}(0, \text{pub}(A))), \text{ad}(0, \text{pub}(A)))$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

# Attack with one agent

---

Insecure network

$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$   
 $\text{enc}(S, \text{ad}(0, \text{pub}(A))), \text{ad}(0, \text{pub}(A)),$   
 $\text{ad}(\text{enc}(S, \text{ad}(0, \text{pub}(A))), \text{ad}(0, \text{pub}(A)))$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

# Attack with one agent

---

Insecure network

$0, A, \text{pub}(A), \text{pub}(B), \text{pair}(A, \text{ae}(0, \text{pub}(B)))$   
 $\text{enc}(S, \text{ad}(0, \text{pub}(A))), \text{ad}(0, \text{pub}(A)),$   
 $S$

---

Bob

---

$S, y = \text{pair}(A, \text{ae}(0, \text{pub}(B)))$

# Amended Denning Sacco Protocol (Lowe 1996)

Protocol messages:

0.  $A \rightarrow B$ :  $\text{pair}(A, \text{ae}(\text{ae}(\text{tup}(A, B, K), \text{inv}(\text{pub}(A))), \text{pub}(B)))$
1.  $B \rightarrow A$ :  $\text{enc}(S, K)$

# Amended Denning Sacco Protocol (Lowe 1996)

Protocol messages:

0.  $A \rightarrow B$ :  $\text{pair}(A, \text{ae}(\text{ae}(\text{tup}(A, B, K), \text{inv}(\text{pub}(A))), \text{pub}(B)))$
1.  $B \rightarrow A$ :  $\text{enc}(S, K)$

agent  $A$ :

$\text{snd}\langle \text{pair}(A, \text{ae}(\text{ae}(\text{tup}(A, B, K), \text{inv}(\text{pub}(A))), \text{pub}(B))) \rangle.\text{rcv}(x)$

# Amended Denning Sacco Protocol (Lowe 1996)

Protocol messages:

0.  $A \rightarrow B$ :  $\text{pair}(A, \text{ae}(\text{ae}(\text{tup}(A, B, K), \text{inv}(\text{pub}(A))), \text{pub}(B)))$
1.  $B \rightarrow A$ :  $\text{enc}(S, K)$

agent  $A$ :

$\text{snd}\langle \text{pair}(A, \text{ae}(\text{ae}(\text{tup}(A, B, K), \text{inv}(\text{pub}(A))), \text{pub}(B))) \rangle.\text{rcv}(x)$

agent  $B$ : (with equality tests)

$\text{rcv}(y).\text{if } \text{fst}(y) = \text{fst}(\text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y))))$   
 then  $\text{if } \text{snd}(\text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) = B$   
 then  $\text{snd}\langle \text{enc}(S, \text{thd}(\text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$   
 else abort  
 else abort

# Amended Denning Sacco Protocol (Lowe 1996)

Protocol messages:

0.  $A \rightarrow B$ :  $\text{pair}(A, \text{ae}(\text{ae}(\text{tup}(A, B, K), \text{inv}(\text{pub}(A))), \text{pub}(B)))$
1.  $B \rightarrow A$ :  $\text{enc}(S, K)$

agent  $A$ :

$\text{snd}\langle \text{pair}(A, \text{ae}(\text{ae}(\text{tup}(A, B, K), \text{inv}(\text{pub}(A))), \text{pub}(B))) \rangle.\text{rcv}(x)$

agent  $B$ : (with equality tests)

$\text{rcv}(y).\text{if } \text{fst}(y) = \text{fst}(\text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y))))$   
 then  $\text{if } \text{snd}(\text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) = B$   
 then  $\text{snd}\langle \text{enc}(S, \text{thd}(\text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$   
 else abort  
 else abort

integrity checking is possible with e.g.  $\text{match}(\text{enc}(x, y)) = \text{true}$

# Problems & Results

Given:

- a set of collapsing equations  $\mathcal{E}$  of the form  $\ell = x$  or  $\ell = a$ , presented by a convergent term rewriting system,
- some terms initially known to the attacker,
- agents (with explicit destructors and equality tests),
- an interleaving  $I$  (sequence of agents actions),
- a secret term  $t$ .

# Problems & Results

Given:

- a set of **collapsing** equations  $\mathcal{E}$  of the form  $\ell = x$  or  $\ell = a$ , presented by a convergent term rewriting system,
- some terms initially known to the attacker,
- agents (with explicit destructors and equality tests),
- an interleaving  $I$  (sequence of agents actions),
- a secret term  $t$ .

# Problems & Results

Given:

- a set of **collapsing** equations  $\mathcal{E}$  of the form  $\ell = x$  or  $\ell = a$ , presented by a convergent term rewriting system,
- **some terms initially known to the attacker**,
- agents (with explicit destructors and equality tests),
- an interleaving  $I$  (sequence of agents actions),
- a secret term  $t$ .

# Problems & Results

Given:

- a set of **collapsing** equations  $\mathcal{E}$  of the form  $\ell = x$  or  $\ell = a$ , presented by a convergent term rewriting system,
- some terms initially known to the attacker,
- **agents** (with explicit destructors and equality tests),
- an interleaving  $I$  (sequence of agents actions),
- a secret term  $t$ .

# Problems & Results

Given:

- a set of **collapsing** equations  $\mathcal{E}$  of the form  $\ell = x$  or  $\ell = a$ , presented by a convergent term rewriting system,
- some terms initially known to the attacker,
- agents (with explicit destructors and equality tests),
- an **interleaving**  $I$  (sequence of agents actions),
- a secret term  $t$ .

# Problems & Results

Given:

- a set of **collapsing** equations  $\mathcal{E}$  of the form  $\ell = x$  or  $\ell = a$ , presented by a convergent term rewriting system,
- some terms initially known to the attacker,
- agents (with explicit destructors and equality tests),
- an interleaving  $I$  (sequence of agents actions),
- a **secret** term  $t$ .

# Problems & Results

Given:

- a set of **collapsing** equations  $\mathcal{E}$  of the form  $\ell = x$  or  $\ell = a$ , presented by a convergent term rewriting system,
- some terms initially known to the attacker,
- agents (with explicit destructors and equality tests),
- an interleaving  $I$  (sequence of agents actions),
- a secret term  $t$ .

Question (Protocol Insecurity):

Is the interleaving  $I$  feasible (in the given environment)?  
In the state reached following  $I$ , is  $t$  in the attacker knowledge?

# Problems & Results

Given:

- a set of **collapsing** equations  $\mathcal{E}$  of the form  $\ell = x$  or  $\ell = a$ , presented by a convergent term rewriting system,
- some terms initially known to the attacker,
- agents (with explicit destructors and equality tests),
- an interleaving  $I$  (sequence of agents actions),
- a secret term  $t$ .

Question (Protocol Insecurity):

Is the interleaving  $I$  feasible (in the given environment)?  
In the state reached following  $I$ , is  $t$  in the attacker knowledge?

Theorem 1

Polynomial time for a passive attacker.

# Problems & Results

Given:

- a set of **collapsing** equations  $\mathcal{E}$  of the form  $\ell = x$  or  $\ell = a$ , presented by a convergent term rewriting system,
- some terms initially known to the attacker,
- agents (with explicit destructors and equality tests),
- an interleaving  $I$  (sequence of agents actions),
- a secret term  $t$ .

Question (Protocol Insecurity):

Is the interleaving  $I$  feasible (in the given environment)?  
In the state reached following  $I$ , is  $t$  in the attacker knowledge?

Theorem 1

Polynomial time for a passive attacker.

Theorem 2

NP-complete for an active attacker.

# Passive Attacker

The messages exchanged during the interleaving are ground terms  $s_1, \dots, s_n$ . In this case, *Protocol Insecurity* is equivalent to:

Is the attacker able to deduce  $t$  from  $s_1, \dots, s_n$  by application of public function symbols and of equations of  $\mathcal{E}$ ?

denoted  $t \in \mathcal{I}_{\mathcal{E}}(s_1, \dots, s_n)$ , where  $\mathcal{I}_{\mathcal{E}}(\bar{s})$  is the smallest (w.r.t.  $\subseteq$ ) set of ground terms containing  $s_1, \dots, s_n$  and such that:

- $\forall f$  public,  $\forall t_1, \dots, t_n \in \mathcal{I}_{\mathcal{E}}(\overline{s_i})$ ,  $f(t_1, \dots, t_n) \in \mathcal{I}_{\mathcal{E}}(\bar{s})$ .
- if  $u \in \mathcal{I}_{\mathcal{E}}(\bar{s})$  and  $u =_{\mathcal{E}} v$ , then  $v \in \mathcal{I}_{\mathcal{E}}(\bar{s})$ .

# Passive Attacker

The messages exchanged during the interleaving are ground terms  $s_1, \dots, s_n$ . In this case, *Protocol Insecurity* is equivalent to:  
Is the attacker able to deduce  $t$  from  $s_1, \dots, s_n$  by application of public function symbols and of equations of  $\mathcal{E}$ ?

denoted  $t \in \mathcal{I}_{\mathcal{E}}(s_1, \dots, s_n)$ , where  $\mathcal{I}_{\mathcal{E}}(\bar{s})$  is the smallest (w.r.t.  $\subseteq$ ) set of ground terms containing  $s_1, \dots, s_n$  and such that:

- $\forall f$  public,  $\forall t_1, \dots, t_n \in \mathcal{I}_{\mathcal{E}}(\overline{s_i})$ ,  $f(t_1, \dots, t_n) \in \mathcal{I}_{\mathcal{E}}(\bar{s})$ .
- if  $u \in \mathcal{I}_{\mathcal{E}}(\bar{s})$  and  $u =_{\mathcal{E}} v$ , then  $v \in \mathcal{I}_{\mathcal{E}}(\bar{s})$ .

## Theorem 1

$t \in \mathcal{I}_{\mathcal{E}}(s_1, \dots, s_n)$  is decidable in polynomial time.

*proof.* by a **locality** lemma, the problem is equivalent to the satisfiability of a set of ground Horn clauses.

# Active attacker

In this case, *Protocol Insecurity* is equivalent to simultaneous solving a set of **symbolic constraints**.

	Equations	Deduction Constraints
$\mathcal{E}$ -solution = ground substitution $\sigma$ s.t.:	$s = t$ $s\sigma =_{\mathcal{E}} t\sigma$	$s_1, \dots, s_n \Vdash t$ $t\sigma \in \mathcal{I}_{\mathcal{E}}(s_1\sigma, \dots, s_n\sigma)$

Solved by a procedure based on techniques of **narrowing** with basic strategy.

- [C. Meadows, 1989], *Using Narrowing in the Analysis of Key Management Protocols*
- [J. Millen, H-P. Ko, 1996] *Narrowing Terminates for Encryption*

# Active attacker: example

one agent hence only one choice of interleaving!

$0, A, \text{pub}(A), \text{pub}(B)$

Bob

$\text{rcv}(y).\text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$

---

# Active attacker: example

one agent hence only one choice of interleaving!

$0, A, \text{pub}(A), \text{pub}(B)$

Bob

$\text{rcv}(y).\text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$

---

# Active attacker: example

one agent hence only one choice of interleaving!

$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$

Bob

$\text{snd}\langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$

Bob

$$\text{snd} \langle \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \rangle$$

# Active attacker: example

one agent hence only one choice of interleaving!

$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$

$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \Vdash$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$
$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \Vdash S$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$
$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \Vdash S$$

narrowing with  $\text{snd}(\text{pair}(x_1, x_2)) = x_2$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$

$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(\text{ad}(\text{snd}(y), \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \Vdash S$$

narrowing with  $\text{snd}(\text{pair}(x_1, x_2)) = x_2$

$$y = \text{pair}(x_1, x_2)$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$

$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(\text{ad}(x_2, \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \Vdash S$

$y = \text{pair}(x_1, x_2)$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$
$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(\text{ad}(x_2, \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \Vdash S$$

narrowing with  $\text{fst}(\text{pair}(x'_1, x'_2)) = x'_1$

$$y = \text{pair}(x_1, x_2)$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$

$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(\text{ad}(x_2, \text{inv}(\text{pub}(B))), \text{pub}(\text{fst}(y)))) \Vdash S$$

narrowing with  $\text{fst}(\text{pair}(x'_1, x'_2)) = x'_1$

$$y = \text{pair}(x_1, x_2) \quad y = \text{pair}(x'_1, x'_2)$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$

$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(\text{ad}(x_2, \text{inv}(\text{pub}(B))), \text{pub}(x'_1))) \Vdash S$$

$$y = \text{pair}(x_1, x_2) \quad y = \text{pair}(x'_1, x'_2)$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$
$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(\text{ad}(x_2, \text{inv}(\text{pub}(B))), \text{pub}(x_1))) \Vdash S$$
$$y = \text{pair}(x_1, x_2)$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$

$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(\text{ad}(x_2, \text{inv}(\text{pub}(B))), \text{pub}(x_1))) \Vdash S$$

narrowing with  $\text{ad}(\text{ae}(z_1, z_2), \text{inv}(z_2)) = z_1$

$$y = \text{pair}(x_1, x_2)$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$

$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(\text{ad}(x_2, \text{inv}(\text{pub}(B))), \text{pub}(x_1))) \Vdash S$$

narrowing with  $\text{ad}(\text{ae}(z_1, z_2), \text{inv}(z_2)) = z_1$

$$y = \text{pair}(x_1, x_2) \quad x_2 = \text{enc}(z_1, \text{pub}(B))$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$
$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(z_1), \text{pub}(x_1))) \Vdash S$$
$$y = \text{pair}(x_1, x_2) \quad x_2 = \text{enc}(z_1, \text{pub}(B))$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash y$$
$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(z_1), \text{pub}(x_1))) \Vdash S$$
$$y = \text{pair}(x_1, x_2) \quad x_2 = \text{enc}(z_1, \text{pub}(B))$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$0, A, \text{pub}(A), \text{pub}(B) \Vdash \text{pair}(x_1, \text{enc}(z_1, \text{pub}(B)))$

$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(z_1), \text{pub}(x_1)) \Vdash S$

$y = \text{pair}(x_1, x_2) \quad x_2 = \text{enc}(z_1, \text{pub}(B))$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash \text{pair}(x_1, \text{enc}(z_1, \text{pub}(B)))$$
$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(z_1), \text{pub}(x_1)) \Vdash S$$
$$y = \text{pair}(x_1, x_2) \quad x_2 = \text{enc}(z_1, \text{pub}(B))$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash \text{pair}(x_1, \text{enc}(z_1, \text{pub}(B)))$$
$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(z_1), \text{pub}(x_1)) \Vdash S$$
$$y = \text{pair}(x_1, x_2) \quad x_2 = \text{enc}(z_1, \text{pub}(B)) \quad z_1 = 0$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$0, A, \text{pub}(A), \text{pub}(B) \Vdash \text{pair}(x_1, \text{enc}(0, \text{pub}(B)))$

$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(0, \text{pub}(x_1))) \Vdash S$

$y = \text{pair}(x_1, x_2) \quad x_2 = \text{enc}(z_1, \text{pub}(B)) \quad z_1 = 0$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash \text{pair}(x_1, \text{enc}(0, \text{pub}(B)))$$
$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(0), \text{pub}(x_1)) \Vdash S$$
$$y = \text{pair}(x_1, x_2) \quad x_2 = \text{enc}(z_1, \text{pub}(B)) \quad z_1 = 0 \quad x_1 = A$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash \text{pair}(A, \text{enc}(0, \text{pub}(B)))$$

$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(0), \text{pub}(A)) \Vdash S$$

$$y = \text{pair}(x_1, x_2) \quad x_2 = \text{enc}(z_1, \text{pub}(B)) \quad z_1 = 0 \quad x_1 = A$$

Bob

---

# Active attacker: example

one agent hence only one choice of interleaving!

$$0, A, \text{pub}(A), \text{pub}(B) \Vdash \text{pair}(A, \text{enc}(0, \text{pub}(B)))$$

$$0, A, \text{pub}(B), \text{enc}(S, \text{ad}(0), \text{pub}(A)) \Vdash S$$

$$y = \text{pair}(x_1, x_2) \quad x_2 = \text{enc}(z_1, \text{pub}(B)) \quad z_1 = 0 \quad x_1 = A$$

Bob

---

# Inference System

$$\begin{array}{c}
 \frac{\mathcal{P} \cup \{e[f(u_1, \dots, u_n)]\}; \mathcal{C}; \sigma}{\mathcal{P} \cup \{e[r]\}; \mathcal{C}\eta; \sigma\eta \cup \eta} \text{N} \\
 \\
 \frac{\mathcal{P} \cup \{s = t\}; \mathcal{C}; \sigma}{\mathcal{P}; \mathcal{C}\eta; \sigma\eta \cup \eta} \text{U} \\
 \\
 \frac{\mathcal{P} \cup \{c\}; \mathcal{C}; \sigma}{\mathcal{P}; \mathcal{C} \cup \{c\sigma\}; \sigma} \text{B} \\
 \\
 \frac{\mathcal{P}; \mathcal{C}; \sigma}{\mathcal{P}; \mathcal{C}[x \mapsto t]; \sigma[x \mapsto t] \cup [x \mapsto t]} \text{VE} \\
 \\
 \frac{\mathcal{P}; \mathcal{C} \cup \{s_1, \dots, s_n \Vdash t\}; \sigma}{\mathcal{P}; \mathcal{C}; \sigma} \text{G}
 \end{array}$$

## Narrowing

$$\begin{aligned}
 f(l_1, \dots, l_n) &= r \in \mathcal{E} \\
 \eta &= \text{mgu}(f(l_1, \dots, l_n)\sigma, \\
 &\quad f(u_1, \dots, u_n)\sigma)
 \end{aligned}$$

## Syntactic Unification

$$\eta = \text{mgu}(s\sigma, t\sigma)$$

## Blocking

$c$  deduction constraint

## Variable Elimination

$x \in \text{vars}(\mathcal{C}), t \in \text{st}(\mathcal{C}) \setminus \text{vars}$

## Ground

$t \in \mathcal{I}_{\mathcal{E}}(s_1, \dots, s_n)$

# Correctness, completeness, termination

## Lemma

The application of the inferences rules to  $\mathcal{P}; \emptyset; \emptyset$

- terminates  
(and the depth, branching deg. and dag-size of nodes of the derivation tree are polynomial in  $\|\mathcal{P}\| + \|\mathcal{E}\|$ ),
- is correct,
- is complete.

# Correctness, completeness, termination

## Lemma

The application of the inferences rules to  $\mathcal{P}; \emptyset; \emptyset$

- terminates  
(and the depth, branching deg. and dag-size of nodes of the derivation tree are polynomial in  $\|\mathcal{P}\| + \|\mathcal{E}\|$ ),
- is correct,
- is complete.

## Lemma (completeness)

A minimal  $\mathcal{E}$ -solution  $\sigma$  of a *well-formed* set  $\mathcal{C}$  of constraints is made of non-variable subterms of  $\mathcal{C}$ .

# Correctness, completeness, termination

## Lemma

The application of the inferences rules to  $\mathcal{P}; \emptyset; \emptyset$

- terminates  
(and the depth, branching deg. and dag-size of nodes of the derivation tree are polynomial in  $\|\mathcal{P}\| + \|\mathcal{E}\|$ ),
- is correct,
- is complete.

## Lemma (completeness)

A minimal  $\mathcal{E}$ -solution  $\sigma$  of a *well-formed* set  $\mathcal{C}$  of constraints is made of non-variable subterms of  $\mathcal{C}$ .

## Corollary

The  $\mathcal{E}$ -solvability of *well formed* sets of constraints is NP.

# Correctness, completeness, termination (cnt)

## Corollary

Protocol insecurity for an active attacker is NP-complete.

**Rem.:** completeness by reduction of 3-SAT.

# Conclusion & Further Work

Decidability and complexity of the problem of protocol insecurity for an passive/active attacker and a bounded number of agents with explicit destructors and equality tests.

Based on classical techniques of first order automatic deduction and constraints solving.

# Conclusion & Further Work

Decidability and complexity of the problem of protocol insecurity for an passive/active attacker and a bounded number of agents with explicit destructors and equality tests.

Based on classical techniques of first order automatic deduction and constraints solving.

## Extensions

- Extension of the procedure to [Associativity/Commutativity](#) e.g. [XOR](#) theory with AC for  $+$  and the collapsing equations:

$$\begin{aligned}x + x &= 0 \\x + 0 &= x \\x + x + y &= y\end{aligned}$$

- Efficient new decision procedure based on a translation into first order Horn clauses with equality, using a [superposition based](#) automatic deduction systems.