Relating Strands and Multiset Rewriting For Security Protocol Analysis

Iliano Cervesato
Nancy Durgin, Patrick Lincoln
John Mitchell, Andre Scedrov
Several recent proposal based on the Dolev-Yao model:

- Strand spaces
- Multiset rewriting
- Spi-calculus, ...

How are they related?
Roadmap

- MSR
  - Restricted MSR
  - Canonical MSR
- Strands
  - Dynamic Strands
  - Decorated Strands

Relating Strands and Multiset Rewriting for Security Protocols
Running Example

Needham-Schroeder Protocol

\[
\begin{align*}
A & \rightarrow B: \{N_A, A\}_{KB} \\
B & \rightarrow A: \{N_A, N_B\}_{KA} \\
A & \rightarrow B: \{N_B\}_{KB}
\end{align*}
\]
**MSR**

- **Executable** specification language
- Adapts multiset rewriting with $\exists$
  - Solid logical foundation
  - Ties with linear logic and process algebra
- **Flexible and fully precise**
- **Follows the Dolev-Yao model**

Relating Strands and Multiset Rewriting for Security Protocols
Multiset rewriting ...

- Multiset: set with repetitions allowed
- Rewrite rule:
  \[ r: N_1 \rightarrow N_2 \]
- Application
  \[ M_1 \overset{r}{\rightarrow} M_2 \]
  \[ M', N_1 \overset{r}{\rightarrow} M', N_2 \]
- Multi-step transition, reachability
... with existentials

- msets of $1^{st}$-order atomic formulas
- Rules:
  \[ r: F(x) \rightarrow \exists n. G(x,n) \]
- Application

\[
\begin{align*}
M_1 \xrightarrow{r} M_2 \\
M', F(t) \xrightarrow{r} M', G(t,c)
\end{align*}
\]
\( c \) not in \( M_1 \)
MSR predicates

- \( N(m) \)  
  Network messages

- \( I(m) \)  
  Intruder info.

- \( A_i(t_1, \ldots, t_{n_i}) \)  
  Role states

- \( Pr, PrvK, PubK, \ldots \)  
  Persistent info.
Protocol Theories

- Initialization rules
- For each role
  - 1 role generation rule
  - n execution rules
• Assume initialization has already happened

• Initial info: $\Pi$

? No initialization in strands
NS: MSR rules for Alice

\[ \pi_{A_0}(A) \rightarrow A_0(A), \pi_{A_0}(A) \]

\[ A_0(A), \pi_{A_1}(B) \rightarrow \exists N_A. A_1(A,B,N_A), N(\{N_A,A\}_K_B), \pi_{A_1}(B) \]

\[ A_1(A,B,N_A), N(\{N_A,N_B\}_K_A) \rightarrow A_2(A,B,N_A,N_B) \]

\[ A_2(A,B,N_A,N_B) \rightarrow A_3(A,B,N_A,N_B), N(\{N_B\}_K_B) \]

where

\[ \pi_{A_0}(A) = Pr(A), PrvK(A,K_A^{-1}) \]

\[ \pi_{A_1}(B) = Pr(B), PubK(B,K_B) \]
Relating Strands and Multiset Rewriting for Security Protocols

**NS: MSR rules for Bob**

\[ \pi_{B0}(B) \rightarrow B_0(B), \pi_{B0}(B) \]

\[ B_0(A), \pi_{B1}(A), N(\{N_A, A\}_K) \rightarrow B_1(A, B, N_A), \pi_{B1}(A) \]

\[ B_1(A, B, N_A) \rightarrow \exists N_B. B_2(A, B, N_A, N_B), N(\{N_A, N_B\}_K) \]

\[ B_2(A, B, N_A, N_B), N(\{N_B\}_K) \rightarrow B_3(A, B, N_A, N_B) \]

where

\[ \pi_{B0}(B) = \text{Pr}(B), \text{PrvK}(B, K_{B^{-1}}) \]

\[ \pi_{B1}(A) = \text{Pr}(A), \text{PubK}(A, K_A) \]
MSR Intruder

- Implement the Dolev-Yao model
  - Decryption/Encryption
  - Decomposition/composition
  - Nonce generation
  - ...

- Expressed within the language
Strands

- Graphical representation of execution
- Designed for after-the-fact analysis
- Very simple
- Follow the Dolev-Yao model
- Related to
  - Lamport’s causality
  - Mazurkiewicz’s traces
NS: A Bundle

\[ \{N_A, A\}_{KB} \rightarrow \{N_A, A\}_{KB} \]

\[ \downarrow \]

\[ \{N_A, N_B\}_{KA} \leftarrow \{N_A, N_B\}_{KA} \]

\[ \downarrow \]

\[ \{N_B\}_{KB} \rightarrow \]
Penetrator Strands

• Implement the Dolev-Yao model
  - Decryption/Encryption
  - Decomposition/composition
  - Nonce generation
  - ...

• Expressed within the language
Support executable specifications

- Specification language
  - Parametric strands

- Execution capabilities
  - Configurations
  - Transitions
Parametric strands

• Strands are instances of roles
• Parameters: instantiable information
• Constraints:
  ➢ Nonces
  ➢ Persistent info.
NS: Parametric Strand for *Alice*

Alice \((A,B,N_A,N_B)\):

- \(N_A\) Fresh, \(\pi_A(A,B)\)

where

\[
\pi(A,B) = Pr(A), PrvK(A,K_A^{-1}), Pr(B), PubK(B,K_B)
\]

Relating Strands and Multiset Rewriting for Security Protocols
NS: Parametric Strand for Bob

Bob \((A, B, N_A, N_B)\):

- \(N_B\) Fresh, \(\pi_B(A, B)\)

where

\[
\pi(A, B) = Pr(B), PrvK(B, K_B^{-1}), Pr(A), PubK(A, K_A)
\]

\[
\{N_A, A\}_{KB} \quad \{N_A, N_B\}_{KA} \quad \{N_B\}_{KB}
\]
Configurations

- **Extension**: bundle + remaining actions
- **Configuration**: bundle + extension
- **Fringe**: crossing arrows

\[
\text{Configuration} = \text{bundle} + \text{extension} + \text{fringe}
\]
NS: Configuration

\[ \{N_A, A\}_{KB} \rightarrow \{N_A, A\}_{KB} \]

\[ \{N_A, N_B\}_{KA} \leftarrow \{N_A, N_B\}_{KA} \]

\[ \{N_B\}_{KB} \rightarrow \{N_B\}_{KB} \]

\[ \{N_C, C\}_{KD} \rightarrow \{N_C, C\}_{KD} \]

\[ \{N_C, N_D\}_{KC} \leftarrow \{N_C, N_D\}_{KC} \]

\[ \{N_D\}_{KD} \rightarrow \{N_D\}_{KD} \]

Relating Strands and Multiset Rewriting for Security Protocols
Relating Strands and Multiset Rewriting for Security Protocols

**Fresh**

**Instantiate**

**Send**

**Receive**
Bundles vs. Transition Sequences

- 1 bundle $\Rightarrow O(n!)$ transition sequences
- 1 transition sequence $\Rightarrow$ 1 bundle

- Bundles represent execution more compactly
Relating Strands and Multiset Rewriting for Security Protocols

- **Restr. MSR** ➔ **Can. MSR**

- **Merge role gen. with 1st exec. rule**
- **Choose nonces upfront**
- **Guess persistent info. upfront**

**Conversion to canonical form preserves reachability**
NS: Canonical MSR rules for Alice

\[ \pi_A(A,B) \rightarrow \exists N_A. A_1(A,B,N_A), N(\{N_A,A\}_{KB}), \pi_A(A,B) \]

\[ A_1(A,B,N_A), N(\{N_A,N_B\}_{KA}) \rightarrow A_2(A,B,N_A,N_B) \]

\[ A_2(A,B,N_A,N_B) \rightarrow A_3(A,B,N_A,N_B), N(\{N_B\}_{KB}) \]

where \[ \pi_A(A,B) = Pr(A), PrvK(A,K_{A^{-1}}) \]
\[ Pr(B), PubK(B,K_B) \]
Relating Strands and Multiset Rewriting for Security Protocols

Can. MSR \rightarrow\rightarrow Dyn. Strands

- Rules \Rightarrow nodes
- Role state predicates \Rightarrow arrows
- Nonces, persistent info. \Rightarrow constraints

- Configuration \Leftrightarrow state

Reachable states
\Downarrow
Reachable configurations
\[ \pi_A(A, B) \rightarrow \exists N_A. \]

\[ A_1(A, B, N_A), \]

\[ N(\{N_A, A\}_{KB}), \]

\[ \pi_A(A, B) \]

**where** \[ \pi(A, B) = Pr(A), PrvK(A, K_A^{-1}), Pr(B), PubK(B, K_B) \]

Relating Strands and Multiset Rewriting for Security Protocols
Relating Strands and Multiset Rewriting for Security Protocols

NS: MSR \( \rightarrow \) Strands

\[ A_1(A,B, N_A), \]
\[ N(\{N_A, N_B\}_{KA}) \]
\[ \rightarrow \]
\[ A_2(A,B,N_A,N_B) \]

\[ \{N_A, A\}_{KB} \rightarrow \]

\[ \{N_A, N_B\}_{KA} \leftarrow \]

\[ \vdots \]
Relating Strands and Multiset Rewriting for Security Protocols
Relating Strands and Multiset Rewriting for Security Protocols

- Add initial (T) and final node (⊥)
- Add labels $A_i(t_1, \ldots, t_{n_i})$ to arrows $t_1, \ldots, t_{n_i}$ from
  - Constraints
  - Arguments of $A_{i-1}$
Relating Strands and Multiset Rewriting for Security Protocols

- Transitions

Decoration preserves reachability
NS: Decorated Strand for Alice

Alice \((A,B,N_A,N_B)\):

- **\(N_A\) Fresh, \(\pi_A(A,B)\)**

where

\[
\pi(A,B) = Pr(A), PrvK(A,K_A^{-1}), Pr(B), PubK(B,K_B)
\]

Relating Strands and Multiset Rewriting for Security Protocols
Dec. Strands $\rightarrow$ Restr. MSR

- Labels $\Rightarrow$ role state predicates
- Events $\Rightarrow$ network messages
- Constraints $\Rightarrow$ nonces, persistent info.

State $\leftrightarrow$ fringe

Reachable configurations

Reachable states

Relating Strands and Multiset Rewriting for Security Protocols
Relating Strands and Multiset Rewriting for Security Protocols

NS: Strands $\rightarrow$ MSR

Alice $\langle A, B, N_A, N_B \rangle$

1. $N_A$ Fresh, $\pi_A(A, B)$

where

$\pi(A, B) = Pr(A), PrvK(A, K_A^{-1}), Pr(B), PubK(B, K_B)$

2. $A_0(A)$

3. $\pi_{A_0}(A) \rightarrow A_0(A), \pi_{A_0}(A)$

where $\pi_{A_0}(A) = Pr(A), PrvK(A, K_A^{-1})$
Relating Strands and Multiset Rewriting for Security Protocols

\[ \text{Alice (} A, B, N_A, N_B \text{)} \]

\[ \{N_A, A\}_{KB} \rightarrow A_0(A) \]

\[ A_1(A, B, N_A) \rightarrow \exists N_A. A_1(A, B, N_A), N(\{N_A, A\}_{KB}), \pi_{A_1}(B) \]

\[ \text{where } \pi_{A_1}(B) = Pr(B), PubK(B, K_B) \]

\[ N_A \text{ Fresh}, \pi_A(A, B) \]

\[ \text{where } \pi(A, B) = Pr(A), PrvK(A, K_A^{-1}), Pr(B), PubK(B, K_B) \]
Relating Strands and Multiset Rewriting for Security Protocols

\[ \text{NS: Strands} \rightarrow \text{MSR} \]

Alice \((A, B, N_A, N_B)\)

\[ N_A \text{ Fresh, } \pi_A(A, B) \]

where

\[ \pi(A, B) = Pr(A), PrvK(A, K_A^{-1}), Pr(B), PubK(B, K_B) \]

\[ (3) \]
What did we learn?

- Substantial equivalence of
  - MSR
  - Strands

- Strands as executable spec. language
  - Parametric strands
  - Configurations, transitions

- Computational traces
  - Bundles
  - Transition sequences