Symbolic Verification of Cryptographic Protocols

Protocol Analysis in the Applied Pi-Calculus

David Baelde

LSV, ENS Paris-Saclay

2019

Deducibility constraints

Definition

A deducibility constraint system is either $\perp$ or a (possibly empty) conjunction of deducibility constraints of the form

$$T_1 \vdash^? u_1 \wedge \ldots \wedge T_n \vdash^? u_n$$

such that

- $T_1 \subseteq T_2 \subseteq \ldots \subseteq T_n$ (monotonicity)
- for every i, $\mathrm{fv}(T_i) \subseteq \mathrm{fv}(u_1, \ldots, u_{i-1})$ (origination)

Definition

The substitution σ is a solution of $\mathcal{C} = T_1 \vdash^? u_1 \wedge \ldots \wedge T_n \vdash^? u_n$ when $T_i\sigma \vdash u_i\sigma$ for all i and $\mathrm{img}(\sigma) \subseteq T_c(\mathcal{N})$.

Example: Needham-Schroeder

- $S_1 := \langle sk_i, \mathrm{pub}(sk_a), \mathrm{pub}(sk_b)\rangle, \mathrm{aenc}(\langle \mathrm{pub}(sk_a), n_a\rangle, \mathrm{pub}(sk_i))$
  $S_1 \vdash^? x$

Example: Needham-Schroeder

- $S_1 := \langle sk_i, \mathrm{pub}(sk_a), \mathrm{pub}(sk_b)\rangle, \mathrm{aenc}(\langle \mathrm{pub}(sk_a), n_a\rangle, \mathrm{pub}(sk_i))$
  $S_1 \vdash^? \mathrm{aenc}(\langle x_a, x_{na}\rangle, \mathrm{pub}(sk_b))$

Example: Needham-Schroeder

- $S_1 := \langle sk_i, \mathrm{pub}(sk_a), \mathrm{pub}(sk_b)\rangle, \mathrm{aenc}(\langle \mathrm{pub}(sk_a), n_a\rangle, \mathrm{pub}(sk_i))$
  $S_1 \vdash^? \mathrm{aenc}(\langle x_a, x_{na}\rangle, \mathrm{pub}(sk_b))$
- $S_2 := S_1, \mathrm{aenc}(\langle x_{na}, n_b\rangle, x_a)$
  $S_2 \vdash^? \mathrm{aenc}(\langle n_a, x_{nb}\rangle, \mathrm{pub}(sk_a))$

Example: Needham-Schroeder

- $S_1 := \langle sk_i, \mathsf{pub}(sk_a), \mathsf{pub}(sk_b)\rangle, \mathsf{aenc}(\langle \mathsf{pub}(sk_a), n_a\rangle, \mathsf{pub}(sk_i))$
  $S_1 \vdash^? \mathsf{aenc}(\langle x_a, x_{na}\rangle, \mathsf{pub}(sk_b))$
- $S_2 := S_1, \mathsf{aenc}(\langle x_{na}, n_b\rangle, x_a)$
  $S_2 \vdash^? \mathsf{aenc}(\langle n_a, x_{nb}\rangle, \mathsf{pub}(sk_a))$
- $S_3 := S_2, \mathsf{aenc}(x_{nb}, \mathsf{pub}(sk_i))$
  $S_3 \vdash^? \mathsf{aenc}(n_b, \mathsf{pub}(sk_b))$

Example: Needham-Schroeder

- $S_1 := \langle sk_i, \mathrm{pub}(sk_a), \mathrm{pub}(sk_b)\rangle, \mathrm{aenc}(\langle \mathrm{pub}(sk_a), n_a\rangle, \mathrm{pub}(sk_i))$
  $S_1 \vdash^? \mathrm{aenc}(\langle x_a, x_{na}\rangle, \mathrm{pub}(sk_b))$
- $S_2 := S_1, \mathrm{aenc}(\langle x_{na}, n_b\rangle, x_a)$
  $S_2 \vdash^? \mathrm{aenc}(\langle n_a, x_{nb}\rangle, \mathrm{pub}(sk_a))$
- $S_3 := S_2, \mathrm{aenc}(x_{nb}, \mathrm{pub}(sk_i))$
  $S_3 \vdash^? \mathrm{aenc}(n_b, \mathrm{pub}(sk_b))$
- $S_4 := S_3, \mathrm{senc}(\mathrm{secret}, n_b)$ and $x_a = \mathrm{pub}(sk_a)$
  $S_4 \vdash^? \mathrm{secret}$

Constraint resolution

Solved form

A system is solved if it is of the form

$$T_1 \vdash^? x_1 \wedge \ldots \wedge T_n \vdash^? x_n$$

Proposition

If $\mathcal{C}$ is solved, then it admits a solution.

Constraint resolution

Solved form

A system is solved if it is of the form

$$T_1 \vdash^? x_1 \wedge \ldots \wedge T_n \vdash^? x_n$$

Proposition

If $\mathcal{C}$ is solved, then it admits a solution.

Theorem

There exists a terminating relation $\rightsquigarrow$ such that for any $\mathcal{C}$ and θ, $\theta \in \mathrm{Sol}(\mathcal{C})$ iff there is $\mathcal{C} \rightsquigarrow^*_\sigma \mathcal{C}'$ solved and $\theta = \sigma\theta'$ for some $\theta' \in \mathrm{Sol}(\mathcal{C}')$.

Simplification of constraint systems

Here systems are considered modulo AC of $\wedge$.

$$(R_1) \quad \mathcal{C} \wedge T \vdash^? u \;\rightsquigarrow\; \mathcal{C} \qquad \text{if } T \cup \{x \mid (T' \vdash^? x) \in \mathcal{C},\, T' \subsetneq T\} \vdash u$$

$$(R_2) \quad \mathcal{C} \wedge T \vdash^? u \;\rightsquigarrow_\sigma\; \mathcal{C}\sigma \wedge T\sigma \vdash^? u\sigma$$
$$\text{if } \sigma = \mathrm{mgu}(t,u),\, t \in \mathrm{st}(T),\, t \neq u, \text{ and } t, u \notin \mathcal{X}$$

$$(R_3) \quad \mathcal{C} \wedge T \vdash^? u \;\rightsquigarrow_\sigma\; \mathcal{C}\sigma \wedge T\sigma \vdash^? u\sigma$$
$$\text{if } \sigma = \mathrm{mgu}(t_1,t_2),\, t_1, t_2 \in \mathrm{st}(T),\, t_1 \neq t_2$$

$$(R_4) \quad \mathcal{C} \wedge T \vdash^? u \;\rightsquigarrow\; \bot \qquad \text{if } \mathrm{fv}(T \cup \{u\}) = \emptyset,\, T \nvdash u$$

$$(R_f) \quad \mathcal{C} \wedge T \vdash^? f(u_1,\ldots,u_n) \;\rightsquigarrow\; \mathcal{C} \wedge \textstyle\bigwedge_i T \vdash^? u_i \qquad \text{for } f \in \Sigma_c$$

$$(R_{\mathrm{pub}}) \quad \mathcal{C} \;\rightsquigarrow\; \mathcal{C}[x := \mathrm{pub}(x)] \quad \text{if } \mathrm{aenc}(t,x) \in T \text{ for some } (T \vdash^? u) \in \mathcal{C}$$

Examples of simplifications

1. $\mathsf{senc}(n, k) \vdash^? \mathsf{senc}(x, k)$
2. $\mathsf{senc}(\mathsf{senc}(t_1, k), k) \vdash^? \mathsf{senc}(x, k)$ (two opportunities for R_2)
3. $S \vdash^? x \ \wedge \ S, n \vdash^? y \ \wedge \ S, n, \mathsf{senc}(m, \mathsf{senc}(x, k)), \mathsf{senc}(y, k) \vdash^? m$
4. $S \vdash^? x \wedge S \vdash^? \langle x, x \rangle$
5. $n \vdash^? x \wedge n \vdash^? \mathsf{senc}(x, k)$

Constraint simplification proof (1)

Proposition (Validity)

If $\mathcal{C}$ is a deducibility constraint system, and $\mathcal{C} \rightsquigarrow_\sigma \mathcal{C}'$, then $\mathcal{C}'$ is a deducibility constraint system.

Constraint simplification proof (1)

Proposition (Validity)

If $\mathcal{C}$ is a deducibility constraint system, and $\mathcal{C} \rightsquigarrow_\sigma \mathcal{C}'$, then $\mathcal{C}'$ is a deducibility constraint system.

Proposition (Soundness)

If $\mathcal{C} \rightsquigarrow_\sigma \mathcal{C}'$ and $\theta \in \mathsf{Sol}(\mathcal{C}')$ then $\sigma\theta \in \mathsf{Sol}(\mathcal{C})$.

Proposition (Termination)

Simplifications are terminating, as shown by the termination measure $(v(\mathcal{C}), p(\mathcal{C}), s(\mathcal{C}))$ where:

- *$v(\mathcal{C})$ is the number of variables occurring in $\mathcal{C}$;*
- *$p(\mathcal{C})$ is the number of terms of the form $aenc(u, x)$ occurring on the left of constraints in $\mathcal{C}$;*
- *$s(\mathcal{C})$ is the total size of the right-hand sides of constraints in $\mathcal{C}$.*

Constraint simplification proof (2)

Left-minimality & Simplicity

A derivation Π of $T_i \vdash u$ is left-minimal if, whenever $T_j \vdash u$, Π is also a derivation of $T_j \vdash u$.
A derivation is simple it is non-repeating
and all its subderivations are left-minimal.

Proposition

If $T_i \vdash u$, then it has a simple derivation.

Lemma

Let $\mathcal{C} = \bigwedge_j T_j \vdash^? u_j$ be a constraint system, $\theta \in \mathsf{Sol}(\mathcal{C})$,
and i be such that $u_j \in \mathcal{X}$ for all $j < i$.
If $T_i\theta \vdash u$ with a simple derivation starting with an axiom or a decomposition, then there is $t \in \mathsf{subterm}(T_i) \setminus \mathcal{X}$ such that $t\theta = u$.

Constraint simplification proof (3)

Lemma

Let $\mathcal{C} = \bigwedge_j T_j \vdash^? u_j$, $\sigma \in \mathsf{Sol}(\mathcal{C})$.
Let i *be a minimal index such that* $u_i \notin \mathcal{X}$.
Assume that:

- T_i *does not contain two subterms* $t_1 \neq t_2$ *such that* $t_1\sigma = t_2\sigma$;
- T_i *does not contain any subterm of the form* $aenc(t, x)$;
- u_i *is a non-variable subterm of* T_i.

Then $T_i' \vdash u_i$, *where* $T_i' = T_i \cup \{x \mid (T \vdash^? x) \in \mathcal{C}, T \subsetneq T_i\}$.

Constraint simplification proof (3)

Lemma

Let $\mathcal{C} = \bigwedge_j T_j \vdash^? u_j$, $\sigma \in \mathsf{Sol}(\mathcal{C})$.
Let i *be a minimal index such that* $u_i \notin \mathcal{X}$.
Assume that:

- T_i *does not contain two subterms* $t_1 \neq t_2$ *such that* $t_1\sigma = t_2\sigma$*;*
- T_i *does not contain any subterm of the form* $aenc(t, x)$*;*
- u_i *is a non-variable subterm of* T_i*.*

Then $T_i' \vdash u_i$*, where* $T_i' = T_i \cup \{x \mid (T \vdash^? x) \in \mathcal{C}, T \subsetneq T_i\}$*.*

Proposition (Completeness)

If $\mathcal{C}$ *is unsolved and* $\theta \in \mathsf{Sol}(\mathcal{C})$*, there is* $\mathcal{C} \leadsto_\sigma \mathcal{C}'$ *and* $\theta' \in \mathsf{Sol}(\mathcal{C}')$ *such that* $\theta = \sigma\theta'$*.*

Concluding remarks

Improvements

- A complete strategy can yield a polynomial bound, hence a small attack property
- Equalities and disequalities may be added
- Several variants and extensions may be considered: sk instead of pub, signatures, xor, etc.

Symbolic execution

To decide secrecy for processes without replication,
one has to consider all constraint systems produced by symbolic execution.
Enumerating all interleavings of concurrent actions is uselessly costly.