Satisfiability of Dolev-Yao Constraints

Laurent Mazaré

laurent.mazare@imag.fr

Laboratoire VERIMAG

Grenoble, France

Motivation

- Constraints are used to verify secrecy with a bounded number of sessions.

Motivation

- Constraints are used to verify secrecy with a bounded number of sessions.
- Needham-Schroeder constraint:

$$
\begin{aligned}
NS = \quad & E, \{N_C, A\}_{K_C} \Vdash \{x, A\}_{K_B} \\
\wedge \quad & E, \{N_C, A\}_{K_C}, \{x, N_B\}_{K_A} \Vdash \{N_C, y\}_{K_A} \\
\wedge \quad & E, \{N_C, A\}_{K_C}, \{x, N_B\}_{K_A}, \{y\}_{K_C} \Vdash N_B
\end{aligned}
$$

Motivation

- Constraints are used to verify secrecy with a bounded number of sessions.
- Needham-Schroeder constraint:

$$
\begin{array}{rcl}
NS = & & E, \{N_C, A\}_{K_C} \Vdash \{x, A\}_{K_B} \\
 & \wedge & E, \{N_C, A\}_{K_C}, \{x, N_B\}_{K_A} \Vdash \{N_C, y\}_{K_A} \\
 & \wedge & E, \{N_C, A\}_{K_C}, \{x, N_B\}_{K_A}, \{y\}_{K_C} \Vdash N_B
\end{array}
$$

- Extensions: Inequations, Multiple Intruders, Opacity (Strong Secrecy).

Messages and Constraints

- Messages are defined by:

$$m ::= a|x|f(m_1,\ldots,m_n)|\langle m_1,m_2\rangle|\{m_1\}_{m_2}$$

a : atom, x: variable, f : first order symbol

Messages and Constraints

- Messages are defined by:

$$m ::= a|x|f(m_1,\ldots,m_n)|\langle m_1,m_2\rangle|\{m_1\}_{m_2}$$

a : atom, x: variable, f : first order symbol

- Constraints are defined by:

$$C ::= \perp|\top|C\vee C|C\wedge C|C_A$$

$$C_A ::= T \Vdash m[U]|m \neq n$$

m, n: messages, T, U: finite sets of messages.

Models

A substitution σ is a model of constraint C iff $\sigma \models C$ where $\models$ is the smallest relation defined by:

- Usual definitions for $\top$, $\wedge$, $\vee$

Models

A substitution σ is a model of constraint C iff $\sigma \models C$ where $\models$ is the smallest relation defined by:

- Usual definitions for $\top$, $\wedge$, $\vee$
-
$$\frac{m\sigma \neq n\sigma}{\sigma \models m \neq n}$$

Models

A substitution σ is a model of constraint C iff $\sigma \models C$ where $\models$ is the smallest relation defined by:

- Usual definitions for $\top$, $\wedge$, $\vee$
- $$\frac{m\sigma \neq n\sigma}{\sigma \models m \neq n}$$
- $$\frac{T\sigma \vdash m\sigma[U\sigma]}{\sigma \models T \Vdash m[U]}$$

  where $T \vdash m[U]$ is defined as $\vdash$ except the decode rule:

  $$\frac{T \vdash \{m\}_u[U] \qquad u \in U}{T \vdash m[U]}$$

Well-Formed Constraints

- A constraint C is well-formed iff $C = \bigvee Co$ and for each Co:

Well-Formed Constraints

- A constraint C is well-formed iff $C = \bigvee Co$ and for each Co:
  - If $T \Vdash m[U]$ and $T' \Vdash m'[U']$ are in Co, then $T \subseteq T'$ or $T' \subseteq T$ (Environment Inclusion).

Well-Formed Constraints

- A constraint C is well-formed iff $C = \bigvee Co$ and for each Co:
  - If $T \Vdash m[U]$ and $T' \Vdash m'[U']$ are in Co, then $T \subseteq T'$ or $T' \subseteq T$ (Environment Inclusion).
  - If $T \Vdash m[U] \in Co$ and $x \in var(T)$, then there exists $T' \Vdash m'[U'] \in Co$ such that $x \in var(m')$, $U' \subseteq U$ and $T' \subsetneq T$ (Variable Introduction).

Well-Formed Constraints

- A constraint C is well-formed iff $C = \bigvee Co$ and for each Co:
  - If $T \Vdash m[U]$ and $T' \Vdash m'[U']$ are in Co, then $T \subseteq T'$ or $T' \subseteq T$ (Environment Inclusion).
  - If $T \Vdash m[U] \in Co$ and $x \in var(T)$, then there exists $T' \Vdash m'[U'] \in Co$ such that $x \in var(m')$, $U' \subseteq U$ and $T' \subsetneq T$ (Variable Introduction).
- A constraint C quasi-well-formed iff

Well-Formed Constraints

- A constraint C is well-formed iff $C = \bigvee Co$ and for each Co:
  - If $T \Vdash m[U]$ and $T' \Vdash m'[U']$ are in Co, then $T \subseteq T'$ or $T' \subseteq T$ (Environment Inclusion).
  - If $T \Vdash m[U] \in Co$ and $x \in var(T)$, then there exists $T' \Vdash m'[U'] \in Co$ such that $x \in var(m')$, $U' \subseteq U$ and $T' \subsetneq T$ (Variable Introduction).
- A constraint C quasi-well-formed iff
  - Variable Introduction.

Well-Formed Constraints

- A constraint C is well-formed iff $C = \bigvee Co$ and for each Co:
  - If $T \Vdash m[U]$ and $T' \Vdash m'[U']$ are in Co, then $T \subseteq T'$ or $T' \subseteq T$ (Environment Inclusion).
  - If $T \Vdash m[U] \in Co$ and $x \in var(T)$, then there exists $T' \Vdash m'[U'] \in Co$ such that $x \in var(m')$, $U' \subseteq U$ and $T' \subsetneq T$ (Variable Introduction).
- A constraint C quasi-well-formed iff
  - Variable Introduction.
  - There exists a closed message m that occurs in any environment of Co.

Constraints

Reducing usual well-formed constraints to our constraints:

$$
\begin{aligned}
& \bigwedge_{1\le i\le n} T_i\sigma \vdash m_i\sigma \Leftrightarrow \\
\sigma \models \quad & \bigvee_{\substack{k_1,\ldots,k_\alpha\in keys(T,m)\\ 1\le i_1\le i_2\le\ldots\le i_n\le n}} \left(T_1 \Vdash k_1[] \wedge \ldots \wedge T_1 \Vdash m_1[k_1,\ldots,k_{i_1}]\right) \\
& \wedge \left(T_2 \Vdash k_{i_1+1}[k_1,\ldots,k_{i_1}] \wedge \ldots \wedge T_2 \Vdash m_2[k_1,\ldots,k_{i_2}]\right) \\
& \wedge \ldots \\
& \wedge \left(T_n \Vdash k_{i_{n-1}+1}[k_1,\ldots,k_{i_{n-1}}] \wedge \ldots \wedge T_n \Vdash m_n[k_1,\ldots,k_{i_n}]\right)
\end{aligned}
$$

Satisfiability 1 : Rewriting

$$
\begin{array}{rcl}
T \Vdash a[U] & \rightarrow & \top \\
T \Vdash a[U] & \rightarrow & \bot \\
T \Vdash f(m_1, \ldots, m_n)[U] & \rightarrow & \top \\
T \Vdash f(m_1, \ldots, m_n)[U] & \rightarrow & \bot \\
T \Vdash \langle m, n\rangle[U] & \rightarrow & T \Vdash m[U] \wedge T \Vdash n[U] \\
T \Vdash \{m\}_n[U] & \rightarrow & \top \\
T \Vdash \{m\}_n[U] & \rightarrow & T \Vdash m[U] \wedge T \Vdash n[U]
\end{array}
$$

Satisfiability 2 : Properties

- $C_1 \to C_2$, if C_1 is well-formed, then C_2 is well-formed too.

Satisfiability 2 : Properties

- $C_1 \to C_2$, if C_1 is well-formed, then C_2 is well-formed too.
- Correctness and Completeness

Satisfiability 2 : Properties

- $C_1 \to C_2$, if C_1 is well-formed, then C_2 is well-formed too.
- Correctness and Completness
- Termination

Satisfiability 2 : Properties

- $C_1 \to C_2$, if C_1 is well-formed, then C_2 is well-formed too.
- Correctness and Completness
- Termination
- Normal Forms:

$$\bigvee \Big(\big(T \Vdash x[U] \big) \wedge \big(\bigwedge m \neq n \big) \Big)$$

Satisfiability 3 : Inequations

- Let P be the constraint

$$m_1 \neq n_1 \wedge \ldots \wedge m_j \neq n_j$$

If P is satisfiable, then for any substitution σ such that $P\sigma$ is closed and $x\sigma = y\sigma \Rightarrow x = y$,
There exists an integer k such that $k \leq j+1$ and σ^k is a model of P.

Satisfiability 3 : Inequations

- Let P be the constraint

$$m_1 \neq n_1 \wedge \ldots \wedge m_j \neq n_j$$

  If P is satisfiable, then for any substitution σ such that $P\sigma$ is closed and $x\sigma = y\sigma \Rightarrow x = y$,
  There exists an integer k such that $k \leq j+1$ and σ^k is a model of P.
- Application:

$$x\sigma = \langle m, \ldots, m \rangle$$

Satisfiability 4 : Results

- Satisfiability for well-formed constraints is decidable (and NP-complete).

Satisfiability 4 : Results

- Satisfiability for well-formed constraints is decidable (and NP-complete).
- Same thing for quasi-well-formed constraints.

Satisfiability 4 : Results

- Satisfiability for well-formed constraints is decidable (and NP-complete).
- Same thing for quasi-well-formed constraints.
- Security for protocols with inequations (bounded number of sessions).

Opacity: Definitions

- An intruder C observes a protocol session between A and B. Could he deduce any property on the parameters of this protocol ?

Opacity: Definitions

- An intruder C observes a protocol session between A and B. Could he deduce any property on the parameters of this protocol ?
- Example : electronic vote.

$$A \to B : \{vote\}_k$$

If $vote \in \{yes, no\}$, the intruder could infer yes or no using Dolev-Yao, could he guess the value of $vote$?

Opacity: Definitions

- An intruder C observes a protocol session between A and B. Could he deduce any property on the parameters of this protocol ?
- Example : electronic vote.

$$A \rightarrow B : \{vote\}_k$$

If $vote \in \{yes, no\}$, the intruder could infer yes or no using Dolev-Yao, could he guess the value of $vote$?

- Intuitive definition of opacity of property ϕ.

Similarity

- Simultaneous Deductions: $E, E' \vdash m, m'$

$$\frac{}{\{n_1,\ldots,n_k\}_j,\{n'_1,\ldots,n'_k\}_j \vdash n_i,n'_i} \qquad \frac{E,E' \vdash n_1,n'_1 \quad E,E' \vdash n_2,n'_2}{E,E' \vdash \langle n_1,n_2\rangle,\langle n'_1,n'_2\rangle}$$

$$\frac{E,E' \vdash \langle n_1,n_2\rangle,\langle n'_1,n'_2\rangle}{E,E' \vdash n_1,n'_1} \qquad \frac{E,E' \vdash \langle n_1,n_2\rangle,\langle n'_1,n'_2\rangle}{E,E' \vdash n_2,n'_2}$$

$$\frac{E,E' \vdash \{n_1\}_{n_2},\{n'_1\}_{n'_2} \quad E,E' \vdash n_2,n'_2}{E,E' \vdash n_1,n'_1} \qquad \frac{E,E' \vdash n_1,n'_1 \quad E,E' \vdash n_2,n'_2}{E,E' \vdash \{n_1\}_{n_2},\{n'_1\}_{n'_2}}$$

Similarity

- Simultaneous Deductions: $E, E' \vdash m, m'$

$$\frac{}{\{n_1,\ldots,n_k\}_j,\{n'_1,\ldots,n'_k\}_j \vdash n_i,n'_i} \qquad \frac{E,E'\vdash n_1,n'_1 \quad E,E'\vdash n_2,n'_2}{E,E'\vdash \langle n_1,n_2\rangle,\langle n'_1,n'_2\rangle}$$

$$\frac{E,E'\vdash \langle n_1,n_2\rangle,\langle n'_1,n'_2\rangle}{E,E'\vdash n_1,n'_1} \qquad \frac{E,E'\vdash \langle n_1,n_2\rangle,\langle n'_1,n'_2\rangle}{E,E'\vdash n_2,n'_2}$$

$$\frac{E,E'\vdash \{n_1\}_{n_2},\{n'_1\}_{n'_2} \quad E,E'\vdash n_2,n'_2}{E,E'\vdash n_1,n'_1} \qquad \frac{E,E'\vdash n_1,n'_1 \quad E,E'\vdash n_2,n'_2}{E,E'\vdash \{n_1\}_{n_2},\{n'_1\}_{n'_2}}$$

- Similarity:

$$\frac{a\in Atomes}{a\sim a} \qquad \frac{u_1\sim u_2 \quad v_1\sim v_2}{\langle u_1,v_1\rangle\sim\langle u_2,v_2\rangle} \qquad \frac{env_1,env_2\vdash k,k \quad u\sim v}{\{u\}_k\sim\{v\}_k} \qquad \frac{\neg env_1\vdash k \quad \neg env_2\vdash k'}{\{u\}_k\sim\{v\}_{k'}}$$

Hypothesis

- The intruder C follows Dolev-Yao model against a protocol implying A and B (Active Intruder).

Hypothesis

- The intruder C follows Dolev-Yao model against a protocol implying A and B (Active Intruder).
- The intruder knows the protocol used and the trace of the protocol followed during this session.

Hypothesis

- The intruder C follows Dolev-Yao model against a protocol implying A and B (Active Intruder).
- The intruder knows the protocol used and the trace of the protocol followed during this session.
- The intruder has an initial knowledge c_0. For example :

$$c_0 = (k_1 \neq k_2)$$

Hypothesis

- The intruder C follows Dolev-Yao model against a protocol implying A and B (Active Intruder).
- The intruder knows the protocol used and the trace of the protocol followed during this session.
- The intruder has an initial knowledge c_0. For example :

$$c_0 = (k_1 \neq k_2)$$

- Protocols without branching (if).

Opacity Constraints

- Two sessions are not similar iff C is satisfiable.

Opacity Constraints

- Two sessions are not similar iff C is satisfiable.
- Constraint related to the active intruder is:

$$C_{AI} = T_1, T_1' \Vdash m_1, m_1' \wedge T_2, T_2' \Vdash m_2, m_2' \wedge \ldots \Vdash m_\alpha$$

Opacity Constraints

- Two sessions are not similar iff C is satisfiable.
- Constraint related to the active intruder is:

$$C_{AI} = T_1, T_1' \Vdash m_1, m_1' \wedge T_2, T_2' \Vdash m_2, m_2' \wedge \ldots \Vdash m_\alpha$$

- Sessions are not similar:

$$C_S = \left(\bigvee_i T_\alpha, T_\alpha' \Vdash n_i \not\sim n_i' \right)$$

Opacity Constraints

- Two sessions are not similar iff C is satisfiable.
- Constraint related to the active intruder is:

$$C_{AI} = T_1, T_1' \Vdash m_1, m_1' \wedge T_2, T_2' \Vdash m_2, m_2' \wedge \ldots \Vdash m_\alpha$$

- Sessions are not similar:

$$C_S = \left(\bigvee_i T_\alpha, T_\alpha' \Vdash n_i \not\sim n_i' \right)$$

- Eventually,

$$C = C_{AI} \wedge C_S$$

Restricted Version: Vote Protocol

- Only two possible sessions: $v = y$ (for T) and $v = n$ (for T').

Restricted Version: Vote Protocol

- Only two possible sessions: $v = y$ (for T) and $v = n$ (for T').
- New constraint:

$$C_S = T_\alpha, T'_\alpha \Vdash y, n \vee T_\alpha, T'_\alpha \Vdash n, y$$

Restricted Version: Vote Protocol

- Only two possible sessions: $v = y$ (for T) and $v = n$ (for T').
- New constraint:

$$C_S = T_\alpha, T'_\alpha \Vdash y, n \vee T_\alpha, T'_\alpha \Vdash n, y$$

- The vote value is opaque ($\phi = (v = y)$) iff C is not satisfiable.

Restricted Version: Vote Protocol

- Only two possible sessions: $v = y$ (for T) and $v = n$ (for T').
- New constraint:

$$C_S = T_\alpha, T'_\alpha \Vdash y, n \vee T_\alpha, T'_\alpha \Vdash n, y$$

- The vote value is opaque ($\phi = (v = y)$) iff C is not satisfiable.
- Constraint C only uses $\Vdash$ so satisfiability is decidable.

Dumb Vote Protocol

- S is the authority collecting votes, A is a voter.

$$\begin{aligned} S \rightarrow A : &\quad k \\ A \rightarrow S : &\quad \{v\}_k \end{aligned}$$

And $T = T' = \{y, n, k_C, k_C^{-1}\}$.

Dumb Vote Protocol

- S is the authority collecting votes, A is a voter.

$$\begin{aligned} S \rightarrow A: &\quad k \\ A \rightarrow S: &\quad \{v\}_k \end{aligned}$$

  And $T = T' = \{y, n, k_C, k_C^{-1}\}$.

- Constraint C is:

$$T, T' \Vdash x, x' \wedge T; \{y\}_x \;\;,\;\; T'; \{n\}_{x'} \Vdash y, n$$

Dumb Vote Protocol

- S is the authority collecting votes, A is a voter.

$$\begin{aligned} S \to A: &\quad k \\ A \to S: &\quad \{v\}_k \end{aligned}$$

And $T = T' = \{y, n, k_C, k_C^{-1}\}$.

- Constraint C is:

$$T, T' \Vdash x, x' \wedge T; \{y\}_x \ , \ T'; \{n\}_{x'} \Vdash y, n$$

- C is satisfiable ($x = x' = k_C$). Immediate attack.

Conclusion

- Decision procedure for constraint with inequations, first order symbols, multiple "knowledges".

Conclusion

- Decision procedure for constraint with inequations, first order symbols, multiple "knowledges".
- Decision procedure for opacity with an active (Dolev-Yao) intruder.

Conclusion

- Decision procedure for constraint with inequations, first order symbols, multiple "knowledges".
- Decision procedure for opacity with an active (Dolev-Yao) intruder.
- Future Works:

Conclusion

- Decision procedure for constraint with inequations, first order symbols, multiple "knowledges".
- Decision procedure for opacity with an active (Dolev-Yao) intruder.
- Future Works:
    - General opacity problem.

Conclusion

- Decision procedure for constraint with inequations, first order symbols, multiple "knowledges".
- Decision procedure for opacity with an active (Dolev-Yao) intruder.
- Future Works:
  - General opacity problem.
  - General constraints (allow to modelize Fairness/Protocol Composition...).

Conclusion

- Decision procedure for constraint with inequations, first order symbols, multiple "knowledges".
- Decision procedure for opacity with an active (Dolev-Yao) intruder.
- Future Works:
  - General opacity problem.
  - General constraints (allow to modelize Fairness/Protocol Composition...).
  - Add equational theories.