

Work in Progress: On the Deducibility Game for Computational Soundness of Equational Theories

Hubert Comon-Lundh Steve Kremer Joe-Kai Tsay

15 April 2010

CosyProof 2010

Our Work

- General deducibility criterion
 - Can be considered under any equational theory
 - Independent of other aspects of the protocol language
- Trace Mapping Property for generic protocol execution models
 - Implied by security w.r.t. deducibility game
- Goal: easier combination of proofs for different equational theories
 - Only consider theories w.r.t. deducibility game
- No parsing assumption

Some Related Work

- [Micciancio, Warinschi '04], [Cortier, Warinschi '05], [Janvier, Lakhnech, Mazaré '05], [Cortier, Kremer, Küsters, Warinschi '06], . . .
 - Trace mapping, public-key encryption, signatures, secrecy with hash
- [Backes, Hofheinz, Unruh '09]
 - CoSP, minimizing effort of establishing soundness of multiple primitives for different calculi

Outline

- 1 Deducibility Game
- 2 Deduc. Game \rightarrow Trace Mapping
- 3 INDCCA \rightarrow Deduc. Game

Symbolic Algebras

- Similar to Steve's talk on Monday:
 - Term algebra $\mathcal{T}(\Sigma)$ for names \mathcal{N} and signature Σ under Equational theory E
 - Frames $\varphi = \nu \bar{n}. \sigma$
 - $\varphi \vdash_E t$ iff \exists term M such that $M\varphi =_E t$, for $names(M) \cap names(\varphi) = \emptyset$ and $var(M) \subset dom(\varphi)$
- No types on terms
 - Some arguments of function marked as “random positions”

Computational Interpretation of Terms

- Mapping $\tau : \mathcal{N} \longrightarrow \{0, 1\}^{\rho(\eta)}$
- \forall function symbol $f \exists$ poly-time computable function $\llbracket f \rrbracket$
- Instantiation function $\llbracket \cdot \rrbracket_{\eta}^{\tau}$ mapping terms to bitstrings
 - $\llbracket f(a_1, \dots, a_m) \rrbracket_{\eta}^{\tau} := \llbracket f \rrbracket(\llbracket a_1 \rrbracket_{\eta}^{\tau}, \dots, \llbracket a_m \rrbracket_{\eta}^{\tau})$
 - for names, $\llbracket n \rrbracket_{\eta}^{\tau} = \tau(n)$
 - if $t =_E s$ then $\llbracket t \rrbracket_{\eta}^{\tau} = \llbracket s \rrbracket_{\eta}^{\tau}$

Deducibility Game: Bitstring Query

Experiment $\mathbf{Exp}_{E, \mathcal{T}(\Sigma), \mathcal{A}}^{\perp}(\eta)$, between probabilistic poly-time attacker \mathcal{A} and oracle \mathcal{O}_{\perp}

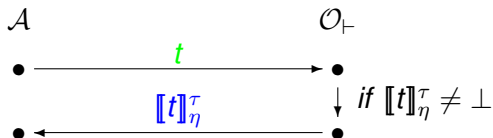
- \mathcal{A} can query *bistrings* $bs \in \{0, 1\}^*$ to \mathcal{O}_{\perp}



- If $\exists u \in st(\varphi)$ s.t. $\llbracket u \rrbracket_{\eta}^{\tau} = bs$ then \mathcal{A} loses,
- else n added to φ

Deducibility Game: Term Query

- \mathcal{A} can submit query *terms* $t \in \mathcal{T}(\Sigma)$ to \mathcal{O}_\top



- \exists conditions on random positions
- $t \downarrow$ added to φ

Deducibility Game: Winning Condition

- In order to win, \mathcal{A} outputs a *challenge pair* (bs, ψ) ,
 - bs a bitstring and
 - ψ a finite boolean expression
- \mathcal{A} wins iff $bs \models^c \psi \wedge \forall t. [t \models \psi \implies \varphi \not\models t]$
 - $\mathbf{Adv}_{E, \mathcal{I}(\Sigma), \mathcal{A}}^{\dagger}(\eta) := \Pr[\mathcal{A} \text{ wins } \mathbf{Exp}_{E, \mathcal{I}(\Sigma), \mathcal{A}}^{\dagger}(\eta)]$

Outline

- 1 Deducibility Game
- 2 Deduc. Game \rightarrow Trace Mapping
- 3 INDCCA \rightarrow Deduc. Game

Protocol Specification

- Protocols are specified from a term algebra over Σ , \mathcal{N} , and variables \mathcal{X}
- Let Q be a set of states q_i
- Map each (q, q') to context $C_{q,q'}[.]$, and finite boolean expressions $\psi_{q,q'}$

Symbolic Protocol Execution Model

- The global states are given by (q, σ, φ) for $q \in Q$, where σ is an instantiation of variables in the specification with terms $\in \mathcal{T}(\Sigma)$ and $\varphi \subset \mathcal{T}(\Sigma)$. The initial state is $(q_0, \emptyset, \emptyset)$.
- State transitions, denoted by $(q, \sigma, \varphi) \longrightarrow (q', \sigma', \varphi')$, are enabled iff $\exists t$.
 - $\varphi \vdash t$
 - $t \models \psi_{q,q'}\sigma$
 - $\sigma' = \sigma \cup \{x_{q,q'} \mapsto t\}$
 - $\varphi' = \varphi \cup \{t, \mathcal{C}_{q,q'}\sigma' \downarrow\}$

Concrete Protocol Execution Model

- The global states are given by $(q, \theta, \varphi^c, s)$ for $q \in Q$, where θ is an instantiation of variables in the specification with bitstrings $\in \{0, 1\}^*$, $\varphi^c \subset \{0, 1\}^*$, and s some state information of the attacker. The initial state is $(q_0, \emptyset, \emptyset, \emptyset)$.
- Let $\mathcal{R}_{\mathcal{A}}$ be the coins of attacker \mathcal{A} , and $\mathcal{R}_{\mathcal{C}}$ the coins of $\llbracket \cdot \rrbracket_{\eta}^{\tau}$
- Given p.p.t. \mathcal{A} , state transitions $(q, \theta, \varphi^c, s) \xrightarrow{\mathcal{A}, \mathcal{R}_{\mathcal{A}}} (q', \theta', \varphi'^c, s')$ are enabled iff
 - $\mathcal{A}(\eta, \varphi^c, s) = bs, s'$
 - $bs \models^c \psi_{q, q'} \theta$
 - $\theta' = \theta \cup \{x_{q, q'} \mapsto bs\}$
 - $\varphi'^c = \varphi \cup \{bs, \llbracket C_{q, q'} \rrbracket_{\eta}^{\tau} \theta\}$

Trace Mapping Property

- Define **Trace** $_{E, \mathcal{T}(\Sigma), \mathcal{A}}^Q(\eta) := \Pr[\mathcal{R}_{\mathcal{A}}(\eta), \mathcal{R}_c(\eta) \mid \exists n \exists q_1, \dots, q_n \exists \sigma_1, \dots, \sigma_n, \theta_1, \dots, \theta_n \exists \varphi_1, \dots, \varphi_n \exists s_1, \dots, s_n \text{ s.t.}$
 - $(q_0, \emptyset, \emptyset, \emptyset) \xrightarrow{\mathcal{A}, \mathcal{R}_{\mathcal{A}}} (q_1, \theta_1, \varphi_1^c, s_1) \xrightarrow{\mathcal{A}, \mathcal{R}_{\mathcal{A}}} \dots \xrightarrow{\mathcal{A}, \mathcal{R}_{\mathcal{A}}} (q_n, \theta_n, \varphi_n^c, s_n)$
 - $\neg((q_0, \emptyset, \emptyset) \longrightarrow (q_1, \sigma_1, \varphi_1) \longrightarrow \dots \longrightarrow (q_n, \sigma_n, \varphi_n))]$

Theorem (\vdash Security \implies Trace Mapping)

If, for all \mathcal{A} , $\text{Adv}_{E, \mathcal{T}(\Sigma), \mathcal{A}}^{\perp}(\eta)$ negligible in η , then, for all \mathcal{A} , $\text{Trace}_{E, \mathcal{T}(\Sigma), \mathcal{A}}^Q(\eta)$ negligible.

Outline

- 1 Deducibility Game
- 2 Deduc. Game \rightarrow Trace Mapping
- 3 INDCCA \rightarrow Deduc. Game

Symbolic Public-Key Encryption

- Signature contains pairing $(., .)$, and associated left/right-projections $\pi_l(.)$ and $\pi_r(.)$
- Public keys $pk(sk)$ formalized via unary function symbol.
- The equational theory E_{asym} generated by
 - $dec(\{x\}_{pk(sk)}^r, sk) = x$
 - $\pi_i((x_1, x_2)) = x_i$ for $i = 1, 2$
 - $extpk(\{u\}_{pk_i}^r) = pk_i$
- The random seed of the encryption symbol is a *random position*
- \exists ordering of subterms of ciphertexts

Concrete Public-Key Encryption

- Let $(\mathcal{K}_\eta, \mathcal{E}_\eta, \mathcal{D}_\eta)$ be an IND-CCA Public-key encryption scheme
 - $\text{Adv}_{\mathcal{E}, \mathcal{A}}^{\text{ind-cca}}(\eta) :=$ advantage of $\mathcal{A}(\eta)$ for distinguishing IND-CCA oracle pair
- Domains of $\mathcal{D}_\eta(\cdot, sk_i)$ and $\mathcal{D}_\eta(\cdot, sk_j)$ disjoint for $i \neq j$

Deducibility Game for Asymmetric Encryption

Consider the deducibility game $\mathbf{Exp}_{E_{\text{asym}}, \mathcal{T}(\Sigma), \mathcal{A}}^{\dagger}(\eta)$ for E_{asym}

- Concrete key pairs $(\bar{p}k_j, \bar{s}k_j)$ are generated by $\mathcal{O}_{\dagger, E_{\text{asym}}}$ with key generator \mathcal{K}_{η} whenever $pk(sk_j)$, for $sk_j \in \mathcal{N}$, occurs in a query for the first time and $\llbracket sk_j \rrbracket_{\eta}^{\tau}$ was not previously defined.
 - The instantiation mapping is then extended s.t.
 $\llbracket pk(sk_j) \rrbracket_{\eta}^{\tau} := \bar{p}k_j$ and $\llbracket sk_j \rrbracket_{\eta}^{\tau} := \bar{s}k_j$
- No adaptive corruption
- No key cycles
- If query term t contains destructor symbols, then $\mathcal{O}_{\dagger, E_{\text{asym}}}$ adds to φ only its “shortest” term representation

Security under IND-CCA Encryption

- Formula ψ of the challenge pair consists of conjunctions and disjunctions of equations over variables, destructors, subterms of φ , without containing free names

Lemma (IND-CCA Security $\implies \vdash$ Security)

If for all \mathcal{A} , $\mathbf{Adv}_{E_{\text{asym}}, \mathcal{A}}^{\text{ind-cca}}(\eta)$ is negligible, then for all \mathcal{A} , $\mathbf{Adv}_{E_{\text{asym}}, \mathcal{I}(\Sigma), \mathcal{A}'}^{\vdash}(\eta)$ is negligible.

Conclusion and Future Work

- Presented a deducibility game for arbitrary equational theories
 - Implies a mapping lemma (\rightarrow soundness)
 - Demonstrated it's usefulness through examples
- Consider different equational theories
- Prove composition results for multiple equational theories
- Investigate relation to observational equivalence

XOR

- Consider the signature consisting of pairing, the binary function symbol \oplus with constants 0, 1.
- The equational theory E_{xor} of XOR is generated by the equations

$$x \oplus y = y \oplus x \quad (1)$$

$$x \oplus (y \oplus z) = (x \oplus y) \oplus z \quad (2)$$

$$x \oplus x = 0 \quad (3)$$

$$x \oplus 0 = x \quad (4)$$

- The concrete semantics of names are given by bitstrings in $\{0, 1\}^\eta$ and the instantiations $\llbracket \oplus \rrbracket_\eta^\tau$, $\llbracket 0 \rrbracket_\eta^\tau$, $\llbracket 1 \rrbracket_\eta^\tau$ of the symbolic \oplus , 0, and 1, are, respectively, the usual exclusive-or of bitstrings in $\{0, 1\}^\eta$, 0^η , and 1^η .

Security for XOR

- Consider the deducibility game between an attacker \mathcal{A} and oracle $\mathcal{O}_{\vdash, E_{xor}}$ for E_{xor}
- If, for the attacker's challenge pair (bs, ψ) , ψ consists conjunctions and disjunctions of equations over variables, destructors, and subterms of the frame φ generated during the game execution, then

Lemma (XOR is \vdash secure)

$\text{Adv}_{E_{xor}, \mathcal{I}(\Sigma), \mathcal{A}}^{\vdash}(\eta)$ is negligible for all \mathcal{A}