Homomorphic Encryption in Maude-NPA

Santiago Escobar	Universidad Politécnica de Valencia (Spain)
Deepak Kapur	University of New Mexico (USA)
Christopher Lynch	Clarkson University (USA)
Catherine Meadows	Naval Research Laboratory (USA)
José Meseguer	University of Illinois at Urbana-Champaign (USA)
Paliath Narendran	University at Albany - SUNY (USA)

PX Meeting, NSA Museum, MD

Beyond the Dolev-Yao Model

Crypto protocol analysis in the standard model free algebra (Dolev-Yao) well understood. But, it fails to account for algebraic identities of crypto operations:

- Diffie-Hellman,
- exclusive-or,
- homomorphism (one-sided distributivity)

used to break a protocol or to specify modern protocols. These operations are beginning to be understood in the bounded sessions case

- Decidability results for exclusive-or, exponentiation, homomorphisms, etc.

What is lacking:

(1) more general understanding, especially for unbounded sessions,

(2) tool support.

Our approach

- Use rewriting logic as general theoretical framework
  - protocols and intruder rules specified as rewrite rules
  - crypto properties as oriented equational properties and axioms
- Use narrowing modulo equational theories in two ways
  - as a symbolic reachability analysis method
  - as an extensible equational unification method
- Combine with state reduction techniques (grammars, optimizations, etc.)
- Implement in Maude programming environment
  - Rewriting logic gives us theoretical framework and understanding
  - Maude implementation gives us tool support

Maude-NPA

- A tool to find or prove the absence of attacks using backwards search
- Analyzes infinite state systems:
  - Active Dolev-Yao intruder
  - No abstraction or approximation of nonces
  - Unbounded number of sessions
- Intruder and honest protocol transitions using variant of strand space model: strands with a marker denoting the current state
  - Searches backwards through strands from final state.
  - Set of rewrite rules governs how search is conducted
  - Sensitive to past and future

Our Plans

(1) Start by formalizing NPA techniques in rewriting logic (done)

- Prove soundness and completeness theorems (done)
- Implement in Maude (the Maude-NPA tool) (done)

(2) Include state reduction techniques present in NPA and new (done)

(3) Document and distribute the tool (done)

(4) Extend model to different types of equational theories

- Explicit Encryption and Decryption (done)
- Bounded Associativity (done)
- AC-unification (done)
- Diffie-Hellman Exponentiation (done)
- Exclusive-or (done)
- Homormophism (one-sided distributivity) (current)

Explicit Encryption and Decryption

- Most formal models lack explicit decryption operator
- If a principal knows an encrypted message and a key, assume principal can decrypt message
  - Implicit assumption that principal never decrypts a message that wasn't encrypted in the first place
  - Usually justified by assumption that principals can check format of decrypted message
- What if format checking isn't implemented? Or what if it is, but you are trying to verify that it works properly?
- In that case, need to model both encryption and decryption explicitly, plus their cancellation, e.g. $d(K, e(K, Y)) = Y$.

Modular Exponentiation in Diffie-Hellman

- Basic DH protocol (each nonzero residue mod P is a power of g)
  1. $A \rightarrow B : g^{N_A}\ mod\ P$

     B computes $(g^{N_A})^{N_B}\ mod\ P$
  2. $B \rightarrow A : g^{N_B}\ mod\ P$

     A and B compute $(g^{N_B})^{N_A} = (g^{N_A})^{N_B}\ mod\ P$ and get a shared secret key.
- Properties:

$$(g^X)^Y = g^{X*Y} = g^{Y*X} = (g^Y)^X$$

$$(X * Y) * Z = X * (Y * Z) \qquad X * Y = Y * X$$

of modular exponentiation in order to faithfully represent this protocol

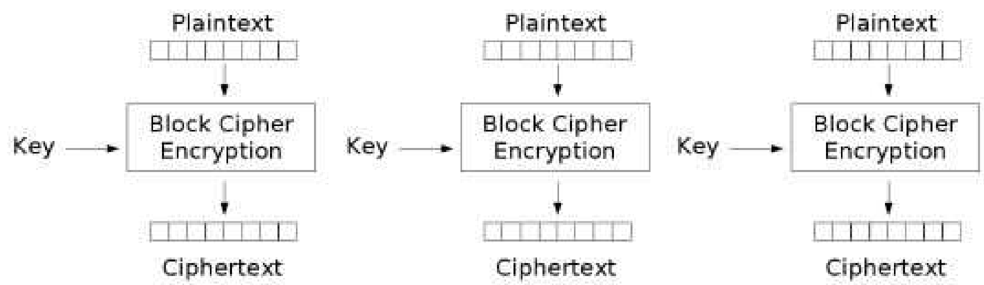
Exclusive-Or

- Cheap and has provable security properties
  - If we send $X \oplus R$, where R a random secret, observer learns no more about X than before it saw message
- On the other hand, commutativity and cancellation properties make it tricky to reason about

$$
\begin{array}{ll}
X \oplus Y = Y \oplus X & X \oplus X = 0 \\
(X \oplus Y) \oplus Z = X \oplus (Y \oplus Z) & X \oplus 0 = X
\end{array}
$$

Homomorphism

- The electronic codebook (ECB) encryption splits a message into blocks and cyphers the blocks using the same key



Electronic Codebook (ECB) mode encryption

- Identical plaintext blocks are encrypted into identical ciphertext blocks (does not hide data patterns well). Sensitive to the property:

$$e(K, X; Y) = e(K, X); e(K, Y)$$

Outline

① Introduction to Rewriting Logic and Narrowing
② Equational Unification
③ Maude-NPA Integration & Demo
④ Conclusions & Future Work

Rewriting Logic in a Nutshell

A rewrite theory $\mathcal{R}$ is a triple $\mathcal{R} = (\Sigma, E, R)$, with:

- (Σ, R) a set of rewrite rules of the form $t \to s$ (i.e., protocol transitions)
  e.g. $e(K, N_A; X) \to e(K, X)$
- (Σ, E) a set of equations of the form $t = s$ (i.e., cryptographic properties)
  e.g. $d(K, e(K, Y)) = Y$

Intuitively, $\mathcal{R}$ specifies a concurrent system,
whose states are elements of the initial algebra $T_{\Sigma/E}$ specified by (Σ, E), and
whose concurrent transitions are specified by the rules R.

R, E-rewriting

Let R a set of rewrite rules and E an equational theory

Rewriting: $t \rightarrow_{R,E} s$ if there is

- a non-variable position $p \in Pos(t)$;
- a rule $l \rightarrow r \in R$;
- a matching σ (modulo E) such that $t|_p =_E \sigma(l)$, and $s = t[\sigma(r)]_p$.

Example:

- $R = \{X \oplus X \rightarrow 0, X \oplus 0 \rightarrow X, X \oplus X \oplus Y \rightarrow Y\}$
- $E = \{X \oplus Y = Y \oplus X, (X \oplus Y) \oplus Z = X \oplus (Y \oplus Z)\}$
- $(a \oplus b) \oplus a \quad \rightarrow_{R,E} \quad b$

Narrowing and Backwards Narrowing

Narrowing: $t \rightsquigarrow_{\sigma,R,E} s$ if there is

- a non-variable position $p \in Pos(t)$;
- a rule $l \to r \in R$;
- a unifier σ (modulo E) such that $\sigma(t|_p) =_E \sigma(l)$, and $s = \sigma(t[r]_p)$.

Example:

- $R = \{\ e(K, N_A; X) \to e(K, X)\ \}$
- $E = \{\ d(K, e(K, Y)) = Y\ \}$
- $$\begin{array}{lll} e(k,\ X) & \rightsquigarrow_{\{X \mapsto N_A;X'\},R,E} & e(k, X') \\ d(k,\ X) & \rightsquigarrow_{\{X \mapsto e(k,e(K,N_A;X'))\},R,E} & e(K, X') \end{array}$$

Backwards Narrowing: narrowing with rewrite rules reversed

Narrowing Reachability Analysis

Narrowing can be used as a general deductive procedure for solving reachability problems of the form

$$(\exists \vec{x})\; t_1(\vec{x}) \to t'_1(\vec{x}) \;\wedge \ldots \wedge\; t_n(\vec{x}) \to t'_n(\vec{x})$$

in a given rewrite theory.

- The terms t_i and t'_i denote sets of states.
- For what subset of states denoted by t_i are the states denoted by t'_i reachable?
- No finiteness assumptions about the state space.
- Sound and complete for topmost rewrite theories.

Outline

① Introduction to Rewriting Logic and Narrowing
② Equational Unification
③ Maude-NPA Integration & Demo
④ Conclusions & Future Work

A Little Background on Unification

- Given a signature Σ and an equational theory E, and two terms s and t built from Σ:
- A unifier of s and t is a substitution σ to the variables in s and t such that σs can be transformed into σt by applying equations from E to s and its subterms
- Example: $\Sigma = \{d/2, e/2, m/0, k/0\}, E = \{d(K, e(K, X) = X\}$. The substitution $\sigma = \{X/e(K, Y)\}$ is a unifier of $d(K, X)$ and Y.
- The set of most general unifiers of s and t is the set Γ such that any unifier σ is of the form $\rho\tau$ for some ρ, and some τ in Γ.
- Example, $\{X/e(K, Y), Y/d(K, X)\}$ is the set of mgu's of $e(K, X)$ and Y.

- Given the theory, can have:
  - at most one mgu (empty theory)
  - a finite number (AC)
  - an infinite number (associativity)
- Problem in general undecidable, so different algorithms devised for different theories
- Compare to syntactic unification:
  - $f(a, X) = f(Y, b)$ has solution $X \mapsto b, Y \mapsto a$
  - $f(a, X) = f(b, Y)$ has no solution
  - $f(a, X) =_{AC} f(b, Y)$ has solution $X \mapsto b, Y \mapsto a$
  - $X + 0 = X$ has no solution
  - $X + 0 =_{ACU} X$, where 0 is the identity, has solution id

Equational Unification

- Equational unification is the solving of formulas $\exists \vec{x}\ t =_E t'$
- When E convergent TRS, narrowing provides a complete E-unification procedure [Hullot80] e.g. cancellation $d(K, e(K, Y)) = Y$
- When $E = \Delta \uplus B$ and Δ convergent and coherent modulo B, narrowing provides a complete E-unification procedure [Jouannaud-Kirchner-Kirchner-83] e.g. exclusive-xor

$$(g^X)^Y = g^{X*Y} = g^{Y*X} = (g^Y)^X$$

$$(X * Y) * Z = X * (Y * Z) \qquad X * Y = Y * X$$

- Narrowing provides semi-decidable E-unification procedure, since it may not terminate even for simple cases

Equational Unification - Strategies

- Narrowing is very inefficient. Strategies have been studied in the literature.
- When E convergent TRS, basic narrowing strategy [Hullot80] is complete for normalized substitutions.
- Cases where basic narrowing terminates have been studied in order to provide decidable E-unification procedures.
- When $E = \Delta \uplus B$ and Δ convergent and coherent modulo B, no many strategies have been studied in practice and AC-narrowing is highly non-terminating.
- Variant-narrowing [Escobar-Meseguer-Sasse] is the most promising strategy for equational unification. It is partially implemented in Maude-NPA.

Variant-Narrowing

1. Complete narrowing strategy modulo axioms B with smaller search space than full B-narrowing.
2. Decidable narrowing-based E-unification procedure

- Based on E-*variant* of [Comon-Delaune-RTA05], we define variant narrowing strategy:
  1. it only uses substitutions in normal form
  2. complete under very general assumptions on B and Δ
  3. if Δ has the finite variant property of [Comon-Delaune-RTA05],
     (a) compute all E-variants of a term in a space-effective way
     (b) obtain a finitary E-unification procedure

Equational Theories & Finite Variant Property

1. [Escobar-Meseguer-Sasse-TechRep07]
   $Lhs \to Rhs$ where Rhs is a variable or a constant (bound 1)
   plus some extra conditions
2. [Comon-Delaune-RTA05]
   Exclusive Or (max. bound 1)
3. [Comon-Delaune-RTA05]
   Abelian group (max. bound 2)
4. [Comon-Delaune-RTA05]
   Diffie-Hellman (max. bound 4)
5. [Comon-Delaune-RTA05]
   Homomorphism (NOT)

Outline

① Introduction to Rewriting Logic and Narrowing
② Equational Unification
③ Maude-NPA Integration & Demo
④ Conclusions & Future Work

Protocol Verification: Maude-NPA

- Maude-NPA uses backwards search from an insecure state to find attacks or to prove unreachability of cryptographic protocols $(\Sigma, \Delta \uplus B, R)$
- Narrowing at two levels in Maude-NPA
  1. a theory $(\Sigma, \Delta \uplus B, R)$: ($\Delta \uplus B$-narrowing with rules R)
  2. for $\Delta \uplus B$-unification (B-narrowing with rules Δ)
- $\Delta \uplus B$-unification for each backwards step using R
  1. Built-in unification algorithms desirable
  2. Our hybrid approach: built-in algorithms for B, and a generic algorithm (variant narrowing) for Δ.

Demo

Outline

① Introduction to Rewriting Logic and Narrowing
② Equational Unification
③ Maude-NPA Integration & Demo
④ Conclusions & Future Work

Conclusions

- Equational unification is critical for cryptographic protocol analysis
- Equational unification can be:
  - unitary (empty theory),
  - finitary (AC),
  - infinitary (associativity)
- Equational unification in Maude-NPA:
  - built-in unification algorithms
    (AC –Core Maude–, homomorphism –Meta Level–),
  - our hybrid approach: built-in algorithms for B, and a generic algorithm (variant narrowing) for Δ.
- When a theory E has the finite variant property (modulo AC), variant narrowing provides an efficient equational unification algorithm

Future work

- Prototype implementation developed (already used in Maude-NPA)
- Homomorphism algorithm (from Narendran-Lynch) does not support AC properties but Maude-NPA relies on that for states $\Rightarrow$ infrastructure for combining different unification procedures (AC & homomorphism)
- Homomorphism algorithm (from Narendran-Lynch) does not support sorts (order-sorted) $\Rightarrow$ order-sorted filter (Hendrix-Meseguer)
- Is there variant-narrowing modulo homomorphism possible??? (e.g. homomorphism with exclusive-or)