Rewriting in Protocol Verification

Stéphanie Delaune

Univ Rennes, CNRS, IRISA, France

Monday, June 29th, 2020

Cryptographic protocols everywhere !

Cryptographic protocols

- small programs designed to secure communication (*e.g.* secrecy, authentication, anonymity, . . .)
- use cryptographic primitives (*e.g.* encryption, signature, ...)

The network is unsecure!

Communications take place over a public network like the Internet.

Cryptographic protocols everywhere !

Cryptographic protocols

- small programs designed to secure communication (*e.g.* secrecy, authentication, anonymity, . . .)
- use cryptographic primitives (*e.g.* encryption, signature, ...)

It becomes more and more important to protect our privacy.

How cryptographic protocols can be attacked?

Cryptanalysis

- Differential attacks,
- Boomerang attacks,
- Cube attacks,
- ...

How cryptographic protocols can be attacked?

Logical attacks

- can be mounted even assuming perfect cryptography,
  $\hookrightarrow$ replay attack, man-in-the middle attack, ...
- subtle and hard to detect by "eyeballing" the protocol

This is the so-called **Dolev-Yao attacker** !

How cryptographic protocols can be attacked?

Logical attacks

- can be mounted even assuming perfect cryptography,
  $\hookrightarrow$ replay attack, man-in-the middle attack, ...
- subtle and hard to detect by "eyeballing" the protocol

Example: An authentication flaw on the Needham Schroeder protocol

$$A \to B : \{A, N_A\}_{\mathsf{pub}(B)}$$
$$B \to A : \{N_A, N_B\}_{\mathsf{pub}(A)}$$
$$A \to B : \{N_B\}_{\mathsf{pub}(B)}$$

NS protocol (1978)

How cryptographic protocols can be attacked?

Logical attacks

- can be mounted even assuming perfect cryptography,
  $\hookrightarrow$ replay attack, man-in-the middle attack, ...
- subtle and hard to detect by "eyeballing" the protocol

Example: An authentication flaw on the Needham Schroeder protocol

$$A \to B : \{A, N_A\}_{\mathrm{pub}(B)}$$
$$B \to A : \{N_A, N_B\}_{\mathrm{pub}(A)}$$
$$A \to B : \{N_B\}_{\mathrm{pub}(B)}$$

NS protocol (1978)

$$A \to B : \{A, N_A\}_{\mathrm{pub}(B)}$$
$$B \to A : \{N_A, N_B, B\}_{\mathrm{pub}(A)}$$
$$A \to B : \{N_B\}_{\mathrm{pub}(B)}$$

NS-Lowe protocol (1995)

How cryptographic protocols can be attacked?

Logical attacks

- can be mounted even assuming perfect cryptography,
  $\hookrightarrow$ replay attack, man-in-the middle attack, ...
- subtle and hard to detect by "eyeballing" the protocol

Example: A traceability attack on the BAC protocol (2010)

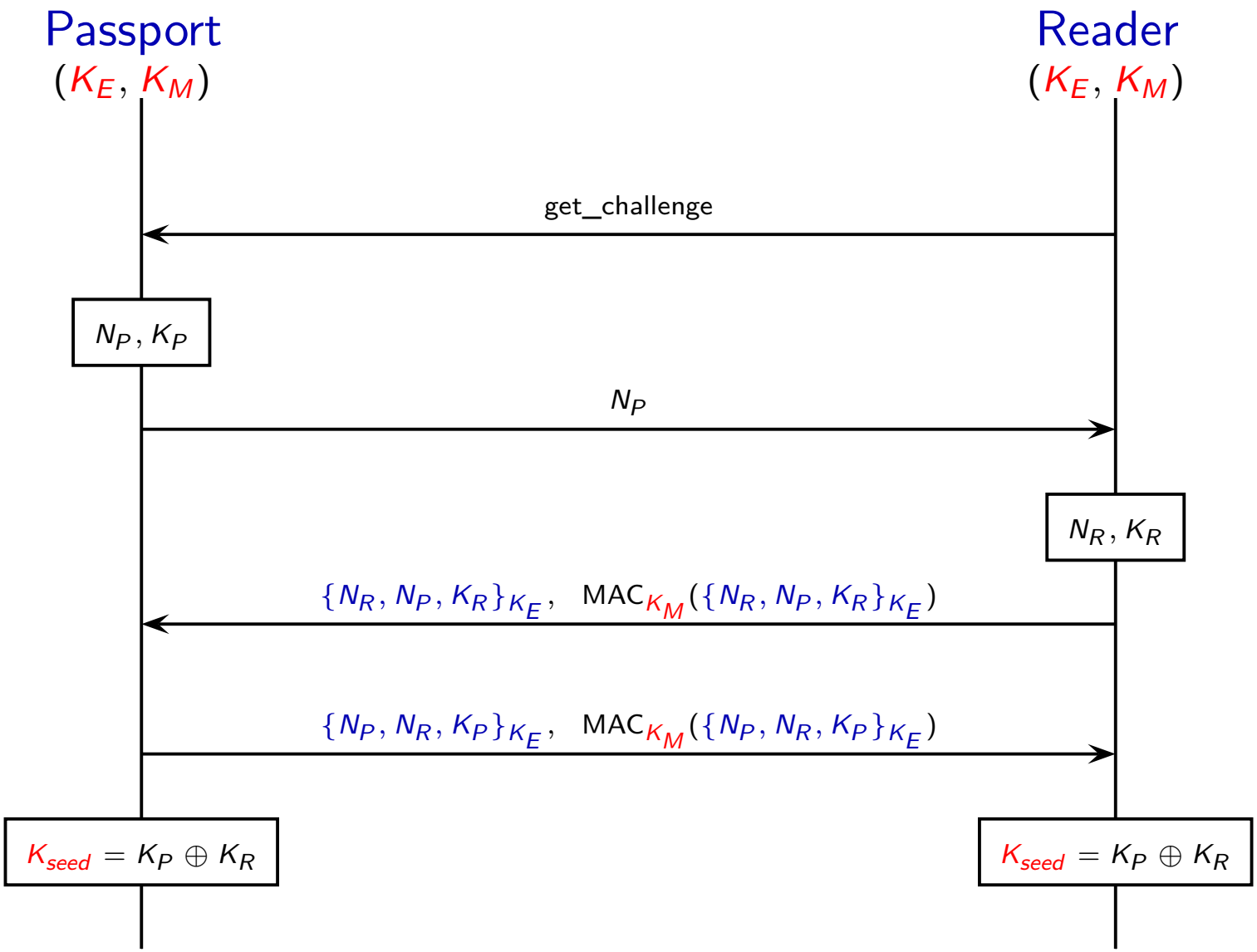
Security

Defects in e-passports allow real-time tracking

This threat brought to you by RFID

The register - Jan. 2010

Basic Acccess Control (BAC) protocol



Unlinkability/Untraceability

Informally, an observer/attacker can not observe the difference between the two following situations:

1. a situation where the same passport may be used twice (or even more);
2. a situation where each passport is used at most once.

Unlinkability/Untraceability

Informally, an observer/attacker can not observe the difference between the two following situations:

1. a situation where the same passport may be used twice (or even more);
2. a situation where each passport is used at most once.

More formally,

$$!\text{new } ke.\text{new } km.(!P_{\text{BAC}} \mid !R_{\text{BAC}}) \stackrel{?}{\approx} !\text{new } ke.\text{new } km.(P_{\text{BAC}} \mid R_{\text{BAC}})$$

many sessions for each passport

only one session for each passport

(we still have to formalize the notion of equivalence)

How rewriting and unification theory

can help us

in protocol verification?

Messages as terms - Back to the BAC protocol

Nonces n_r, n_p, and keys k_r, k_p, k_e, k_m are modelled using names

Cryptographic primitives are modelled using function symbols

- encryption/decryption: senc/2, sdec/2
- concatenation/projections: $\langle , \rangle/2$, $\mathrm{proj}_1/1$, $\mathrm{proj}_2/1$
- mac construction: mac/2
- exclusive or: $\oplus/2$, 0 (neutral element)

Messages as terms - Back to the BAC protocol

Nonces n_r, n_p, and keys k_r, k_p, k_e, k_m are modelled using names

Cryptographic primitives are modelled using function symbols

- encryption/decryption: senc/2, sdec/2
- concatenation/projections: $\langle , \rangle/2$, $\mathsf{proj}_1/1$, $\mathsf{proj}_2/1$
- mac construction: mac/2
- exclusive or: $\oplus/2$, 0 (neutral element)

$$\mathsf{sdec}(\mathsf{senc}(x,y),y) = x \quad \mathsf{proj}_1(\langle x,y\rangle) = x \quad \mathsf{proj}_2(\langle x,y\rangle) = y$$

$$\begin{array}{rcl@{\qquad}rcl} x \oplus (y \oplus z) & = & (x \oplus y) \oplus z & x \oplus x & = & 0 \\ x \oplus y & = & y \oplus x & x \oplus 0 & = & x \end{array}$$

Messages as terms - Back to the BAC protocol

Nonces n_r, n_p, and keys k_r, k_p, k_e, k_m are modelled using names

Cryptographic primitives are modelled using function symbols

- encryption/decryption: senc/2, sdec/2
- concatenation/projections: $\langle\,,\,\rangle/2$, $\mathrm{proj}_1/1$, $\mathrm{proj}_2/1$
- mac construction: mac/2
- exclusive or: $\oplus/2$, 0 (neutral element)

$$\mathrm{sdec}(\mathrm{senc}(x,y),y) = x \quad \mathrm{proj}_1(\langle x,y\rangle) = x \quad \mathrm{proj}_2(\langle x,y\rangle) = y$$

$$\begin{array}{rcl@{\qquad}rcl} x \oplus (y \oplus z) & = & (x \oplus y) \oplus z & x \oplus x & = & 0 \\ x \oplus y & = & y \oplus x & x \oplus 0 & = & x \end{array}$$

Equational theories are useful to model algebraic properties of cryptographic primitives.

It was 15 years ago!

The finite variant property: How to get rid of some algebraic properties. ⋆

Hubert Comon-Lundh[2] and Stéphanie Delaune[1,2]

[1] France Télécom R&D
[2] Laboratoire Spécification & Vérification
ENS de Cachan & CNRS UMR 8643
61, avenue du Président Wilson,
94235 CACHAN Cedex, FRANCE
email: comon,delaune@lsv.ens-cachan.fr

Abstract. We consider the following problem: Given a term t, a rewrite system $\mathcal{R}$, a finite set of equations E' such that $\mathcal{R}$ is E'-convergent, compute finitely many instances of t: $t_1, \ldots, t_n$ such that, for every substitution σ, there is an index i and a substitution θ such that $t\sigma\downarrow =_{E'} t_i\theta$ (where $t\sigma\downarrow$ is the normal form of $t\sigma$ w.r.t. $\rightarrow_{E'\setminus\mathcal{R}}$).
The goal of this paper is to give equivalent (resp. sufficient) conditions for the finite variant property and to systematically investigate this property for equational theories, which are relevant to security protocols verification. For instance, we prove that the finite variant property holds for Abelian Groups, and a theory of modular exponentiation and does not hold for the theory *ACUNh* (Associativity, Commutativity, Unit, Nilpotence, homomorphism).

$\longrightarrow$ published at RTA in 2005

Main motivation behind the Finite Variant Property

Goal: to reduce the decidability of a problem modulo E into a (supposedly) simpler theory E'.

Security Problem (reachability properties):

Input: a protocol $\mathcal{P}$ (finite set of rules), a set $\mathcal{I}$ of inference rules, an equational theory E

Output: Does there exist an attack on $\mathcal{P}$ with $\mathcal{I}$ modulo E?

Main motivation behind the Finite Variant Property

Goal: to reduce the decidability of a problem modulo E into a (supposedly) simpler theory E'.

Security Problem (reachability properties):

Input: a protocol $\mathcal{P}$ (finite set of rules), a set $\mathcal{I}$ of inference rules, an equational theory E

Output: Does there exist an attack on $\mathcal{P}$ with $\mathcal{I}$ modulo E?

Theorem

There is an attack on $\mathcal{P}$ in $\mathcal{I}$ modulo E
if, and only if
there exists $\mathcal{P}_i \in \mathsf{variant}(\mathcal{P})$ which admits an attack in $\mathsf{variant}(\mathcal{I})$ modulo E'.

This was not so clear for us 15 years ago ...

$\longrightarrow$ our RTA paper in 2005

1 Introduction

In our recent work on the verification of cryptographic protocols [3, 5] we came twice across the following problem:

> Given an AC-convergent rewrite system $\mathcal{R}$, is it possible (and how) to compute from any term t a finite set of instances $t\sigma_1, \ldots, t\sigma_n$ such that
>
> $$\{t\sigma{\downarrow_{\mathcal{R}}} \mid \sigma \in \Sigma\} = \bigcup_{i=1}^{n} \{t\sigma_i{\downarrow_{\mathcal{R}}}\theta \mid \theta \in \Sigma\}$$
>
> where Σ is the set of normalized substitutions and $u{\downarrow_{\mathcal{R}}}$ is the AC-normal form of u w.r.t. $\mathcal{R}$.

This was not so clear for us 15 years ago ...

$\longrightarrow$ our RTA paper in 2005

1 Introduction

In our recent work on the verification of cryptographic protocols [3, 5] we came twice across the following problem:

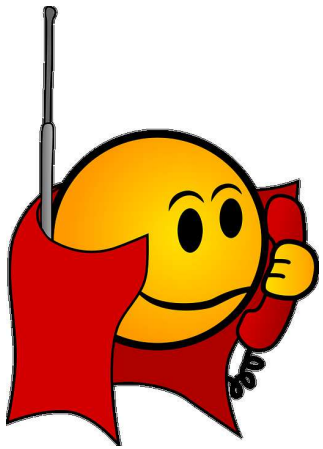
> Given an AC-convergent rewrite system $\mathcal{R}$, is it possible (and how) to compute from any term t a finite set of instances $t\sigma_1, \ldots, t\sigma_n$ such that
>
> $$\{t\sigma{\downarrow_{\mathcal{R}}} \mid \sigma \in \Sigma\} = \bigcup_{i=1}^{n} \{t\sigma_i{\downarrow_{\mathcal{R}}}\theta \mid \theta \in \Sigma\}$$
>
> where Σ is the set of normalized substitutions and $u{\downarrow_{\mathcal{R}}}$ is the AC-normal form of u w.r.t. $\mathcal{R}$.

We have seen over the time the usefulness of the FVP

TAMARIN
Tamarin prover interactive mode
[Meier et al., 13]

Maude-NPA
[Escobar et al., 07]

DEEPSEC
[Cheval et al., 18]



Static equivalence

$\longrightarrow$ this is the so-called passive attacker

The static equivalence problem ($\phi \sim \psi$)

- Input: two substitutions (called frames) ϕ and ψ

  $$\phi = \{w_1 \triangleright u_1, \ldots, w_\ell \triangleright u_\ell\} \qquad \psi = \{w_1 \triangleright v_1, \ldots, w_\ell \triangleright v_\ell\}$$

- Output: Can the attacker distinguish the two frames, *i.e.* does there exist a test $R_1 \stackrel{?}{=} R_2$ such that:

  $$R_1\phi =_E R_2\phi \text{ but } R_1\psi \neq_E R_2\psi \text{ (or the converse).}$$

The static equivalence problem ($\phi \sim \psi$)

- Input: two substitutions (called frames) ϕ and ψ

  $$\phi = \{w_1 \triangleright u_1, \ldots, w_\ell \triangleright u_\ell\} \qquad \psi = \{w_1 \triangleright v_1, \ldots, w_\ell \triangleright v_\ell\}$$

- Output: Can the attacker distinguish the two frames, *i.e.* does there exist a test $R_1 \stackrel{?}{=} R_2$ such that:

  $$R_1\phi =_E R_2\phi \text{ but } R_1\psi \neq_E R_2\psi \text{ (or the converse).}$$

Example: $\mathsf{adec}(\mathsf{aenc}(x, \mathsf{pk}(y)), y) = x$

- $\phi = \{w_1 \triangleright \mathsf{pk}(sks);\ w_2 \triangleright \mathsf{aenc}(yes, \mathsf{pk}(sks))\}$; and
- $\psi = \{w_1 \triangleright \mathsf{pk}(sks);\ w_2 \triangleright \mathsf{aenc}(no, \mathsf{pk}(sks))\}$.

The static equivalence problem ($\phi \sim \psi$)

- Input: two substitutions (called frames) ϕ and ψ

  $$\phi = \{w_1 \triangleright u_1, \ldots, w_\ell \triangleright u_\ell\} \qquad \psi = \{w_1 \triangleright v_1, \ldots, w_\ell \triangleright v_\ell\}$$

- Output: Can the attacker distinguish the two frames, *i.e.* does there exist a test $R_1 \stackrel{?}{=} R_2$ such that:

  $$R_1\phi =_E R_2\phi \text{ but } R_1\psi \neq_E R_2\psi \text{ (or the converse).}$$

Example: $\mathsf{adec}(\mathsf{aenc}(x, \mathsf{pk}(y)), y) = x$

- $\phi = \{w_1 \triangleright \mathsf{pk}(sks);\ w_2 \triangleright \mathsf{aenc}(yes, \mathsf{pk}(sks))\}$; and
- $\psi = \{w_1 \triangleright \mathsf{pk}(sks);\ w_2 \triangleright \mathsf{aenc}(no, \mathsf{pk}(sks))\}$.

$\longrightarrow$ They are not in static equivalence: $\mathsf{aenc}(yes, w_1) \stackrel{?}{=} w_2$.

Static equivalence – some existing results

Theory E	Deduction	**Static Equivalence**
subterm convergent	PTIME	
blind signature, homo. encryption	decidable [Abadi & Cortier, 06]	
ACU	NP-complete	PTIME [Cortier & D., 10]
ACUN/AG	PTIME [Chevalier et al, 03]	PTIME [Cortier & D., 10]
ACUNh/AGh	PTIME [D., 06]	decidable [Cortier & D., 10]

Static equivalence – some existing results

Theory E	Deduction	**Static Equivalence**
subterm convergent	PTIME	
blind signature, homo. encryption	decidable [Abadi & Cortier, 06]	
ACU	NP-complete	PTIME [Cortier & D., 10]
ACUN/AG	PTIME [Chevalier et al, 03]	PTIME [Cortier & D., 10]
ACUNh/AGh	PTIME [D., 06]	decidable [Cortier & D., 10]

Unification theory was an inspiration for us

Monoidal equational theories

1. We associate a semi-ring S_E to a monoidal theory E:

   $\mathbb{Z}/2\mathbb{Z}$ for ACUN, $\mathbb{Z}$ for AG, $\mathbb{Z}/2\mathbb{Z}[X]$ for ACUNh, ...

2. We reduce the static equivalence problem to the problem of deciding whether two sets of linear equations have the same set of solutions.

$\longrightarrow$ inspired by [Nutt, 90] and [Baader & Schulz, 96]

Unification theory was an inspiration for us

Monoidal equational theories

1. We associate a semi-ring S_E to a monoidal theory E:

   $\mathbb{Z}/2\mathbb{Z}$ for ACUN, $\mathbb{Z}$ for AG, $\mathbb{Z}/2\mathbb{Z}[X]$ for ACUNh, ...

2. We reduce the static equivalence problem to the problem of deciding whether two sets of linear equations have the same set of solutions.

$\longrightarrow$ inspired by [Nutt, 90] and [Baader & Schulz, 96]

Combination result for disjoint theories [Cortier & D., 10]

If deduction and static equivalence are decidable for two disjoint theories E_1 and E_2 then they are also decidable for $E_1 \cup E_2$.

$\longrightarrow$ inspired by [Schmidt-Schauß, 89; Baader & Schluz, 96]

Conclusion

Many UNIF topics are of interest for protocol verification:

- Equational unification and unification modulo theories
- Narrowing
- Higher-Order Unification
- Contraint Solving
- Disunification
- ...

Conclusion

Many UNIF topics are of interest for protocol verification:

- Equational unification and unification modulo theories
- Narrowing
- Higher-Order Unification
- Contraint Solving
- Disunification
- ...

Challenging theory: A useful equational theory on which existing tools behave badly is homomophic encryption (e-voting protocols):

$$\{x\}_{\mathsf{pk}(s)} \star \{y\}_{\mathsf{pk}(s)} = \{x + y\}_{\mathsf{pk}(s)}$$

Norvegian e-voting protocol

$\longrightarrow$ a proof "by hand" of ballot secrecy [Cortier & Wiedling, JCS'17]

$$
\begin{aligned}
\mathsf{fst}(\mathsf{pair}(x,y)) &= x \\
\mathsf{snd}(\mathsf{pair}(x,y)) &= y \\
\mathsf{dec}(\mathsf{penc}(x,r,\mathsf{pk}(k)),k) &= x \\
\mathsf{dec}(\mathsf{blind}(\mathsf{penc}(x,r,\mathsf{pk}(k)),b),k) &= \mathsf{blind}(x,b) \\
\mathsf{penc}(x,r_1,k_p) \circ \mathsf{penc}(y,r_2,k_p) &= \mathsf{penc}(x \diamond y, r_1 * r_2, k_p) \\
\mathsf{renc}(\mathsf{penc}(x,r,\mathsf{pk}(k_1)),k_2) &= \mathsf{penc}(x,r,\mathsf{pk}(k_1+k_2)) \\
\mathsf{unblind}(\mathsf{blind}(x,b),b) &= x \\
\mathsf{checksign}(x,\mathsf{vk}(id),\mathsf{sign}(x,id)) &= \mathsf{ok} \\
\mathsf{checkpfk}_1(\mathsf{vk}(id),ball,\mathsf{pfk}_1(id,r,x,ball)) &= \mathsf{ok} \\
\text{where } ball &= \mathsf{penc}(x,r,k_p) \\
\mathsf{checkpfk}_2(\mathsf{vk}(id),x,ball,\mathsf{pfk}_2(\mathsf{vk}(id),k,b,x,ball)) &= \mathsf{ok} \\
\text{where } ball &= \mathsf{blind}(\mathsf{renc}(x,k),b)
\end{aligned}
$$

with associativity and commutativity for $+$, $*$, $\circ$, and $\diamond$.

Thank you for listening