Coercion-Resistance and Receipt-Freeness in Electronic Voting

Stéphanie Delaune[1,2], Steve Kremer[2] and Mark Ryan[3]

[1] LSV, ENS de Cachan, CNRS & INRIA, France

[2] France Télécom R&D

[3] School of Computer Science, University of Birmingham, UK

Electronic voting

Advantages:

- Convenient,
- Efficient facilities for tallying votes.

Drawbacks:

- Risk of large-scale and undetectable fraud,
- Such protocols are extremely error-prone.

> *"A 15-year-old in a garage could manufacture smart cards and sell them on the Internet that would allow for multiple votes"* Avi Rubin

Possible issue: formal methods

abstract analysis of the protocol against formally-stated properties

Expected properties

Privacy: the fact that a particular voted in a particular way is not revealed to anyone

Receipt-freeness: a voter cannot prove that she voted in a certain way (this is important to protect voters from coercion)

Coercion-resistance: same as receipt-freeness, but the coercer interacts with the voter during the protocol, e.g. by preparing messages

Summary

Observations:

- Definitions of security propeties are often insufficiently precise
- No clear distinction between receipt-freeness and coercion-resistance

Goal:

Propose the first "formal methods" definitions of receipt-freeenes and coercion-resistance

Results:

- Formalisation of receipt-freenes and coercion-resistance as some kind of observational equivalence in the applied pi-calculus,
- Coercion-Resistance $\Rightarrow$ Receipt-Freeness $\Rightarrow$ Privacy,
- Case study: protocol due to Lee *et al.* [Lee *et al.*, 03]

Summary

Observations:

- Definitions of security propeties are often insufficiently precise
- No clear distinction between receipt-freeness and coercion-resistance

Goal:

Propose the first "formal methods" definitions of receipt-freeenes and coercion-resistance

Results:

- Formalisation of receipt-freenes and coercion-resistance as some kind of observational equivalence in the applied pi-calculus,
- Coercion-Resistance $\Rightarrow$ Receipt-Freeness $\Rightarrow$ Privacy,
- Case study: protocol due to Lee *et al.* [Lee *et al.*, 03]

Outline of the talk

1. Introduction
2. Applied π-calculus
3. Formalisation of Privacy and Receipt-Freeness
4. Formalisation of Coercion-Resistance
5. Conclusion and Future Works

Outline of the talk

Motivation for using the applied π-calculus

Applied pi-calculus: [Abadi & Fournet, 01]
basic programming language with constructs for concurrency and communication

- based on the π-calculus [Milner *et al.*, 92]
- in some ways similar to the spi-calculus [Abadi & Gordon, 98]

Advantages:

- allows us to model less classical cryptographic primitives
- both reachability and equivalence-based specification of properties
- automated proofs using ProVerif tool [Blanchet]
- powerful proof techniques for hand proofs
- successfully used to analyze a variety of security protocols

Motivation for using the applied π-calculus

Applied pi-calculus: [Abadi & Fournet, 01]

basic programming language with constructs for concurrency and communication

- based on the π-calculus [Milner *et al.*, 92]
- in some ways similar to the spi-calculus [Abadi & Gordon, 98]

Advantages:

- allows us to model less classical cryptographic primitives
- both reachability and equivalence-based specification of properties
- automated proofs using ProVerif tool [Blanchet]
- powerful proof techniques for hand proofs
- successfully used to analyze a variety of security protocols

The applied π-calculus on an example

Syntax:

- Equational theory: $dec(enc(x, y), y) = x$
- Process:

$$P = \nu s, k.(\mathrm{out}(c_1, enc(s, k)) \mid \mathrm{in}(c_1, y).\mathrm{out}(c_2, dec(y, k))).$$

Semantics:

- Operational semantics $\rightarrow$:

$$P \rightarrow \nu s, k.\mathrm{out}(c_2, s)$$

- Operational labeled semantics $\xrightarrow{\alpha}$:

$$P \xrightarrow{\nu x_1.out(c_1,x_1)} \nu s, k.(\mathrm{in}(c_1, y).\mathrm{out}(c_2, dec(y, k))) \mid \{enc(s, k)/x_1\})$$
$$\xrightarrow{in(c_1,x_1)} \nu s, k.(\mathrm{out}(c_2, s) \mid \{enc(s, k)/x_1\}$$
$$\ldots$$

The applied π-calculus on an example

Syntax:

- Equational theory: $dec(enc(x, y), y) = x$
- Process:

$$P = \nu s, k.(\mathrm{out}(c_1, enc(s, k)) \mid \mathrm{in}(c_1, y).\mathrm{out}(c_2, dec(y, k))).$$

Semantics:

- Operational semantics $\rightarrow$:

$$P \;\rightarrow\; \nu s, k.\mathrm{out}(c_2, s)$$

- Operational labeled semantics $\xrightarrow{\alpha}$:

$$\begin{array}{lll} P & \xrightarrow{\nu x_1.out(c_1, x_1)} & \nu s, k.(\mathrm{in}(c_1, y).\mathrm{out}(c_2, dec(y, k))) \mid \{enc(s, k)/x_1\}) \\ & \xrightarrow{in(c_1, x_1)} & \nu s, k.(\mathrm{out}(c_2, s) \mid \{enc(s, k)/x_1\} \\ & \ldots & \end{array}$$

Frame

A frame is a process of the form $\nu\tilde{n}.(\{^{M_1}/_{x_1}\} \mid \ldots \mid \{^{M_n}/_{x_n}\})$.

Example

$$P = \nu s, k.(\text{out}(c_2, s) \mid \{enc(s, k)/x_1\} \qquad \phi(P) = \nu s, k.\{enc(s, k)/x_1\}$$

Static equivalence on frames ($\approx_s$)

$\varphi \approx_s \psi$ when

- $dom(\varphi) = dom(\psi)$ (the frames coincide on unrestricted variables),
- for all terms U, V, $(U =_E V)\varphi$ iff $(U =_E V)\psi$

Static equivalence on frames – passive attacker

Frame

A frame is a process of the form $\nu\tilde{n}.(\{^{M_1}/_{x_1}\} \mid \ldots \mid \{^{M_n}/_{x_n}\})$.

Example

$$P = \nu s, k.(\mathrm{out}(c_2, s) \mid \{enc(s, k)/x_1\} \qquad \phi(P) = \nu s, k.\{enc(s, k)/x_1\}$$

Static equivalence on frames ($\approx_s$)

$\varphi \approx_s \psi$ when

- $dom(\varphi) = dom(\psi)$ (the frames coincide on unrestricted variables),
- for all terms U, V, $(U =_E V)\varphi$ iff $(U =_E V)\psi$

Example 1: $\nu k.(\{^{enc(a,k)}/_x\} \mid \{^k/_y\}) \not\approx_s \nu n.(\{^{enc(b,k)}/_x\} \mid \{^k/_y\})$

Static equivalence on frames – passive attacker

Frame

A frame is a process of the form $\nu\tilde{n}.(\{^{M_1}/_{x_1}\} \mid \ldots \mid \{^{M_n}/_{x_n}\})$.

Example

$$P = \nu s, k.(\mathrm{out}(c_2, s) \mid \{enc(s, k)/x_1\} \qquad \phi(P) = \nu s, k.\{enc(s, k)/x_1\}$$

Static equivalence on frames ($\approx_s$)

$\varphi \approx_s \psi$ when

- $dom(\varphi) = dom(\psi)$ (the frames coincide on unrestricted variables),
- for all terms U, V, $(U =_E V)\varphi$ iff $(U =_E V)\psi$

Example 2: $\nu k.\{^{enc(a,k)}/_x\} \approx_s \nu n.\{^{enc(b,k)}/_x\}$

Labeled bisimulation on processes – active attacker

Labeled bisimulation ($\approx_\ell$)

Labeled bisimilarity is the largest symmetric relation $\mathcal{R}$ on closed extended processes, such that $A \mathrel{\mathcal{R}} B$ implies

1. $\phi(A) \approx_s \phi(B)$,
2. if $A \rightarrow A'$, then $B \rightarrow^* B'$ and $A' \mathrel{\mathcal{R}} B'$ for some B',
3. if $A \xrightarrow{\alpha} A'$, then $B \rightarrow^* \xrightarrow{\alpha} \rightarrow^* B'$ and $A' \mathrel{\mathcal{R}} B'$ for some B'.

Theorem (Abadi & Fournet, 01)

$A \approx_\ell B \Leftrightarrow$ *no context can distinguish the two processes* A *and* B.

Definition (Voting process)

$$VP \equiv \nu\tilde{n}.(V\sigma_1 \mid \cdots \mid V\sigma_n \mid A_1 \mid \cdots \mid A_m)$$

- $V\sigma_i$: voter process and $v \in dom(\sigma_i)$ refers to the value of his vote
- A_j: election authority
- $\tilde{n}$: channel names

The outcome of the vote is made public, *i.e.* there exists B such that

$$VP \;\; (\rightarrow^* \xrightarrow{\alpha} {}^*)^* \;\; B$$

with $\phi(B) \equiv \varphi \;\mid \{^{v\sigma_1}/_{x_1}, \ldots, {}^{v\sigma_n}/_{x_n}\}$ for some φ.

$\hookrightarrow$ S is a context which is as VP but has a hole instead of two of the $V\sigma_i$

Outline of the talk

Formalisation of privacy

Classically modeled as observational equivalences between two slightly different processes P_1 and P_2, but

- changing the identity does not work, as identities are revealed
- changing the vote does not work, as the votes are revealed at the end

Solution: [Kremer & Ryan, 05]
$\hookrightarrow$ consider 2 honest voters and swap their votes

A voting protocol respects privacy if

$$S[V_A\{^a/_v\} \mid V_B\{^b/_v\}] \approx_\ell S[V_A\{^b/_v\} \mid V_B\{^a/_v\}].$$

Formalisation of privacy

Classically modeled as observational equivalences between two slightly different processes P_1 and P_2, but

- changing the identity does not work, as identities are revealed
- changing the vote does not work, as the votes are revealed at the end

Solution: [Kremer & Ryan, 05]
$\hookrightarrow$ consider 2 honest voters and swap their votes

A voting protocol respects privacy if

$$S[V_A\{^a/_v\} \mid V_B\{^b/_v\}] \approx_\ell S[V_A\{^b/_v\} \mid V_B\{^a/_v\}].$$

Leaking secrets to the coercer

To model receipt-freeness we need to specify that a coerced voter cooperates with the coercer by leaking secrets on a channel ch

We denote by V^{ch} the process built from the process V as follows:

- $0^{ch} \mathrel{\widehat{=}} 0$,
- $(P \mid Q)^{ch} \mathrel{\widehat{=}} P^{ch} \mid Q^{ch}$,
- $(\nu n.P)^{ch} \mathrel{\widehat{=}} \nu n.\mathsf{out}(ch, n).P^{ch}$,
- $(\mathsf{in}(u, x).P)^{ch} \mathrel{\widehat{=}} \mathsf{in}(u, x).\mathsf{out}(ch, x).P^{ch}$,
- $(\mathsf{out}(u, M).P)^{ch} \mathrel{\widehat{=}} \mathsf{out}(u, M).P^{ch}$,
- ...

We denote by $V^{\backslash out(chc,\cdot)} \mathrel{\widehat{=}} \nu chc.(V \mid !\mathsf{in}(chc, x)$.

Receipt-freeness

Definition (Receipt-freeness)

A voting protocol is receipt-free if there exists a process V', satisfying

- $V'^{\backslash out(chc,\cdot)} \approx_\ell V_A\{^a/_v\}$,
- $S[V_A\{^c/_v\}^{chc} \mid V_B\{^a/_v\}] \approx_\ell S[V' \mid V_B\{^c/_v\}]$.

Intuitively, there exists a process V' which

- does vote a,
- leaks (possibly fake) secrets to the coercer,
- and makes the coercer believe he voted c

Let VP be a voting protocol. We have formally shown that:

$$VP \text{ is receipt-free} \implies VP \text{ respects privacy.}$$

Case study: Lee *et al.* protocol

We have proved receipt-freeness by

- exhibiting V'
- showing that $V'^{\backslash out(chc,\cdot)} \approx_\ell V_A\{^a/_v\}$
- showing that $S[V_A\{^c/_v\}^{chc} \mid V_B\{^a/_v\}] \approx_\ell S[V' \mid V_B\{^c/_v\}]$

Outline of the talk

Interacting with the coercer

To model coercion-resistance, we need to model interaction between the coercer and the voter:

1. secrets are leaked to the coercer on a channel c_1, and
2. outputs are prepared by the coercer and given to the voter via c_2.

We denote by V^{c_1,c_2} the process built from V as follows:

- $0^{c_1,c_2} \mathrel{\widehat{=}} 0$,
- $(P \mid Q)^{c_1,c_2} \mathrel{\widehat{=}} P^{c_1,c_2} \mid Q^{c_1,c_2}$,
- $(\nu n.P)^{c_1,c_2} \mathrel{\widehat{=}} \nu n.\mathsf{out}(c_1, n).P^{c_1,c_2}$,
- $(\mathsf{in}(u,x).P)^{c_1,c_2} \mathrel{\widehat{=}} \mathsf{in}(u,x).\mathsf{out}(c_1,x).P^{c_1,c_2}$,
- $(\mathsf{out}(u,M).P)^{c_1,c_2} \mathrel{\widehat{=}} \mathsf{in}(c_2,x).\mathsf{out}(u,x).P^{c_1,c_2}$ (x is a fresh variable),
- ...

Coercion-resistance (1)

First approximation:
VP is coercion-resistant if there exists a process V' such that

$$S[V_A\{^c/_v\}^{c_1,c_2} \mid V_B\{^a/_v\}] \approx_\ell S[V' \mid V_B\{^c/_v\}].$$

Problem:

- the coercer could oblige $V_A\{^c/_v\}^{c_1,c_2}$ to vote $c' \neq c$,
- the process $V_B\{^c/_v\}$ would not counterbalance the outcome

Solution:
$\hookrightarrow$ a new relation we have called adaptive simulation ($A \preceq_a B$)

Coercion-resistance (1)

First approximation:
VP is coercion-resistant if there exists a process V' such that

$$S[V_A\{^c/_v\}^{c_1,c_2} \mid V_B\{^a/_v\}] \approx_\ell S[V' \mid V_B\{^c/_v\}].$$

Problem:

- the coercer could oblige $V_A\{^c/_v\}^{c_1,c_2}$ to vote $c' \neq c$,
- the process $V_B\{^c/_v\}$ would not counterbalance the outcome

Solution:
$\hookrightarrow$ a new relation we have called adaptive simulation (A $\preceq_a$ B)

Coercion-resistance (2)

Definition (Coercion-resistance)

A voting protocol is coercion-resistant if there exists a process V' and an evaluation context C satisfying

- $S[V_A\{^c/_v\}^{c_1,c_2} \mid V_B\{^a/_v\}] \preceq_a S[V' \mid V_B\{^x/_v\}]$,
- $\nu c_1, c_2.C[V_A\{^c/_v\}^{c_1,c_2}] \approx_\ell V_A\{^c/_v\}^{chc}$,
- $\nu c_1, c_2.C[V']^{\setminus out(chc,\cdot)} \approx_\ell V_A\{^a/_v\}$,

where x is a fresh free variable.

Intuitively,

- $V_B\{^x/_v\}$ can adapt his vote and counter-balance the outcome,
- we require that when we apply a context C (the coercer requesting $V_A\{^c/_v\}^{c_1,c_2}$ to vote c) the process V' in the same context C votes a.

Some results

Let *VP* be a voting protocol. We have formally shown that:

$$VP \text{ is coercion-resistant} \implies VP \text{ respects receipt-free.}$$

↪ reflects the intuition but the proof is technical

Case study: Lee *et al.* protocol

Coersion-resistance depends on implementation details:

- encryption with integrity check
  ↪ fault attack: the protocol is not coercion-resistant
- encryption without integrity check
  ↪ the protocol is coercion-resistant

Conclusion and Future Works

Conclusion:

- first formal definitions of receipt-freeness and coercion-resistance
- coercion-resistance $\Rightarrow$ receipt-freeness $\Rightarrow$ privacy,
- a case study giving interesting insights

Future Works:

- decision procedure for observational equivalence for processes without replication
- other properties based on *not being able to prove*
- individual/universal verifiability

Conclusion and Future Works

Conclusion:

- first formal definitions of receipt-freeness and coercion-resistance
- coercion-resistance $\Rightarrow$ receipt-freeness $\Rightarrow$ privacy,
- a case study giving interesting insights

Future Works:

- decision procedure for observational equivalence for processes without replication
- other properties based on *not being able to prove*
- individual/universal verifiability