When reachability-based secrecy implies equivalence-based secrecy in security protocols

Véronique Cortier

joint work with Eugen Zălinescu and Michaël Rusinowitch

LORIA, CNRS & INRIA project CASSIS, Nancy, France

Artist2 Workshop

Pisa, May 18th 2006

Context

Verifying security protocols

- programs that ensure secure communications
- notoriously difficult to design

The intruder controls the network

- can see all messages
- can modify and send new messages
- can intercept messages

Two standard notions of secrecy

Reachability-based (syntactic) secrecy	Equivalence-based (strong) secrecy
$P \rightarrow^* s$	$P(M) \approx P(M') \quad \forall M, M'$
decidable classes	stronger security notion
many available tools	closer to computational secrecy

Two standard notions of secrecy

Reachability-based (syntactic) secrecy	Equivalence-based (strong) secrecy
$P \longrightarrow^* s$	$P(M) \approx P(M') \quad \forall M, M'$
decidable classes	stronger security notion
many available tools	closer to computational secrecy

Goal: Relating the two notions of secrecy

Motivations:

- Few results for strong secrecy
- [ESOP'05] in a cryptographic setting, accessibility-based secrecy implies indistinguishability, for asymmetric encryption.

Goal

syntactic secrecy implies strong secrecy

$$P \rightarrow^* s \quad \Rightarrow \quad P(M) \approx P(M')$$

Passive case:

- probabilistic encryption
- the secret does not occur in keys

Active case:

- probabilistic encryption
- ground keys
- no tests on the secret

Messages

Messages are modeled by terms.

- concatenation: $\langle m_1, m_2 \rangle$
- probabilistic symmetric encryption: $\mathrm{enc}(m, k, r)$
- probabilistic asymmetric encryption: $\mathrm{enca}(m, \mathrm{pub}(a), r)$
- digital signatures: $\mathrm{sign}(m, \mathrm{priv}(a))$
- + constants, variables and names (from a set $\mathcal{N}$).

Messages

Messages are modeled by terms.

- concatenation: $\langle m_1, m_2 \rangle$
- probabilistic symmetric encryption: $\mathrm{enc}(m, k, r)$
- probabilistic asymmetric encryption: $\mathrm{enca}(m, \mathrm{pub}(a), r)$
- digital signatures: $\mathrm{sign}(m, \mathrm{priv}(a))$
- + constants, variables and names (from a set $\mathcal{N}$).

We equip the algebra with an equational theory:

$$\begin{cases} \pi_1(\langle z_1, z_2 \rangle) = z_1 \\ \pi_2(\langle z_1, z_2 \rangle) = z_2 \\ \mathrm{dec}(\mathrm{enc}(z_1, z_2, z_3), z_2) = z_1 \\ \mathrm{deca}(\mathrm{enca}(z_1, \mathrm{pub}(z_2), z_3), \mathrm{priv}(z_2)) = z_1 \\ \mathrm{check}(z_1, \mathrm{sign}(z_1, \mathrm{priv}(z_2)), \mathrm{pub}(z_2)) = \mathrm{ok} \\ \mathrm{retrieve}(\mathrm{sign}(z_1, z_2)) = z_1 \end{cases}$$

Deducibility

Frame: $\nu\tilde{n}\{{}^{M_1}/_{x_1}, \ldots, {}^{M_l}/_{x_l}\}$

$\underbrace{\nu\tilde{n}}_{\text{fresh values}}\underbrace{\{{}^{M_1}/_{x_1}, \ldots, {}^{M_l}/_{x_l}\}}_{\text{sequence of messages (terms)}}$

Deduction System:

$$\frac{}{\nu\tilde{n}.\sigma \vdash x\sigma}\ x \in \mathrm{dom}(\sigma) \qquad \frac{}{\nu\tilde{n}.\sigma \vdash s}\ s \in \mathcal{N}\backslash\widetilde{n}$$

$$\frac{\nu\tilde{n}.\sigma \vdash t_1 \quad \cdots \quad \nu\tilde{n}.\sigma \vdash t_r}{\nu\tilde{n}.\sigma \vdash f(t_1, \ldots, t_r)} \qquad \frac{\nu\tilde{n}.\sigma \vdash t \quad t =_E t'}{\nu\tilde{n}.\sigma \vdash t'}$$

Deducibility

Frame: $\nu\tilde{n}\{^{M_1}/_{x_1}, \ldots, ^{M_l}/_{x_l}\}$

$\underbrace{\nu\tilde{n}}_{\text{fresh values}}\underbrace{\{^{M_1}/_{x_1}, \ldots, ^{M_l}/_{x_l}\}}_{\text{sequence of messages (terms)}}$

Deduction System:

$$\frac{}{\nu\tilde{n}.\sigma \vdash x\sigma}\; x \in \mathrm{dom}(\sigma) \qquad \frac{}{\nu\tilde{n}.\sigma \vdash s}\; s \in \mathcal{N}\backslash\tilde{n}$$

$$\frac{\nu\tilde{n}.\sigma \vdash t_1 \quad \cdots \quad \nu\tilde{n}.\sigma \vdash t_r}{\nu\tilde{n}.\sigma \vdash f(t_1, \ldots, t_r)} \qquad \frac{\nu\tilde{n}.\sigma \vdash t \quad t =_E t'}{\nu\tilde{n}.\sigma \vdash t'}$$

Example: k and $\langle k, k'\rangle$ are deducible from the frame
$\nu k, k', r.\{^{\mathrm{enc}(k,k',r)}/_x, ^{k'}/_y\}$.

Deducibility

Frame: $\underbrace{\nu\tilde{n}}_{\text{fresh values}}\underbrace{\{^{M_1}/_{x_1}, \ldots, ^{M_l}/_{x_l}\}}_{\text{sequence of messages (terms)}}$

Deduction System:

$$\frac{}{\nu\tilde{n}.\sigma \vdash x\sigma}\ x \in \mathrm{dom}(\sigma) \qquad \frac{}{\nu\tilde{n}.\sigma \vdash s}\ s \in \mathcal{N}\backslash\widetilde{n}$$

$$\frac{\nu\tilde{n}.\sigma \vdash t_1 \quad \cdots \quad \nu\tilde{n}.\sigma \vdash t_r}{\nu\tilde{n}.\sigma \vdash f(t_1, \ldots, t_r)} \qquad \frac{\nu\tilde{n}.\sigma \vdash t \quad t =_E t'}{\nu\tilde{n}.\sigma \vdash t'}$$

Example: k and $\langle k, k'\rangle$ are deducible from the frame

$$\nu k, k', r.\{^{\mathrm{enc}(k,k',r)}/_x, ^{k'}/_y\}.$$

Definition: A term M is *syntactically secret* in φ if $\varphi \not\vdash M$.

Static equivalence $\approx_s$

Definition: $\phi_1 \approx_s \phi_2$ iff $\mathrm{dom}(\phi_1) = \mathrm{dom}(\phi_2)$ and for every couple of terms (M, N) such that $\mathsf{var}(M, N) \subseteq \mathrm{dom}(\phi_1)$,

$$M\phi_1 = N\phi_1 \quad \Leftrightarrow \quad M\phi_2 = N\phi_2.$$

Static equivalence $\approx_s$

Definition: $\phi_1 \approx_s \phi_2$ iff $\mathrm{dom}(\phi_1) = \mathrm{dom}(\phi_2)$ and for every couple of terms (M, N) such that $\mathsf{var}(M, N) \subseteq \mathrm{dom}(\phi_1)$,

$$M\phi_1 = N\phi_1 \quad \Leftrightarrow \quad M\phi_2 = N\phi_2.$$

Example:
$\phi_1 = \nu k\{\mathrm{enc}(\mathsf{yes}, k)/x, k/y\}$ and $\phi_2 = \nu k\{\mathrm{enc}(\mathsf{no}, k)/x, k/y\}$
are not statically equivalent
since $\mathrm{dec}(x, y) = \mathsf{yes}$ for ϕ_1, while $\mathrm{dec}(x, y) = \mathsf{no}$ for ϕ_2.

Static equivalence $\approx_s$

Definition: $\phi_1 \approx_s \phi_2$ iff $\mathrm{dom}(\phi_1) = \mathrm{dom}(\phi_2)$ and for every couple of terms (M, N) such that $\mathsf{var}(M, N) \subseteq \mathrm{dom}(\phi_1)$,

$$M\phi_1 = N\phi_1 \quad \Leftrightarrow \quad M\phi_2 = N\phi_2.$$

Example:
$\phi_1 = \nu k\{\mathrm{enc}(\mathsf{yes}, k)/x, k/y\}$ and $\phi_2 = \nu k\{\mathrm{enc}(\mathsf{no}, k)/x, k/y\}$
are not statically equivalent
since $\mathrm{dec}(x, y) = \mathsf{yes}$ for ϕ_1, while $\mathrm{dec}(x, y) = \mathsf{no}$ for ϕ_2.

If the key is not revealed: $\nu k\{\mathrm{enc}(\mathsf{yes}, k)/x\} \approx_s \nu k\{\mathrm{enc}(\mathsf{no}, k)/x\}$

Static equivalence $\approx_s$

Definition: $\phi_1 \approx_s \phi_2$ iff $\mathrm{dom}(\phi_1) = \mathrm{dom}(\phi_2)$ and for every couple of terms (M, N) such that $\mathsf{var}(M, N) \subseteq \mathrm{dom}(\phi_1)$,

$$M\phi_1 = N\phi_1 \quad \Leftrightarrow \quad M\phi_2 = N\phi_2.$$

Example:
$\phi_1 = \nu k\{\mathrm{enc}(\mathsf{yes}, k)/x, k/y\}$ and $\phi_2 = \nu k\{\mathrm{enc}(\mathsf{no}, k)/x, k/y\}$
are not statically equivalent
since $\mathrm{dec}(x, y) = \mathsf{yes}$ for ϕ_1, while $\mathrm{dec}(x, y) = \mathsf{no}$ for ϕ_2.

If the key is not revealed: $\nu k\{\mathrm{enc}(\mathsf{yes}, k)/x\} \approx_s \nu k\{\mathrm{enc}(\mathsf{no}, k)/x\}$

Definition: s is *strongly secret* in ϕ if

$$\phi({}^{M}/_{\mathsf{s}}) \approx_s \phi({}^{M'}/_{\mathsf{s}}) \quad \forall M, M'$$

Syntactic secrecy *is* weaker than strong secrecy!

Examples of ψ_i s.t. $\psi_i \nvdash \mathtt{s}$ and $\psi_i(M) \not\approx_s \psi_i(M')$ for some M, M'.

Probabilistic encryption $\quad \psi_1 = \nu k, r. \{^{\mathrm{enc}(\mathtt{s},k,r)}/_x, {}^{\mathrm{enc}(n,k,r)}/_y\} \quad x = y$

Syntactic secrecy *is* weaker than strong secrecy!

Examples of ψ_i s.t. $\psi_i \nvdash \mathtt{s}$ and $\psi_i(M) \not\approx_s \psi_i(M')$ for some M, M'.

Probabilistic encryption $\quad \psi_1 = \nu k, r. \{^{\mathrm{enc}(\mathtt{s},k,r)}/_{x}, {}^{\mathrm{enc}(n,k,r)}/_{y}\} \quad x = y$

Key position $\quad \psi_2 = \nu r. \{^{\mathrm{enc}(n,\mathtt{s},r)}/_{x}\} \quad \mathrm{dec}(x, k) = n$

Syntactic secrecy *is* weaker than strong secrecy!

Examples of ψ_i s.t. $\psi_i \not\vdash \mathtt{s}$ and $\psi_i(M) \not\approx_s \psi_i(M')$ for some M, M'.

Probabilistic encryption	$\psi_1 = \nu k, r. \{^{\mathrm{enc}(\mathtt{s},k,r)}/_x, {}^{\mathrm{enc}(n,k,r)}/_y\}$	$x = y$
Key position	$\psi_2 = \nu r. \{^{\mathrm{enc}(n,\mathtt{s},r)}/_x\}$	$\mathrm{dec}(x, k) = n$
Destructors	$\psi_3 = \{^{\pi_1(\mathtt{s})}/_x\}$	$x = a$

Syntactic secrecy *is* weaker than strong secrecy!

Examples of ψ_i s.t. $\psi_i \not\vdash \mathtt{s}$ and $\psi_i(M) \not\approx_s \psi_i(M')$ for some M, M'.

Probabilistic encryption	$\psi_1 = \nu k, r.\{^{\mathrm{enc}(\mathtt{s},k,r)}/_x, {}^{\mathrm{enc}(n,k,r)}/_y\}$	$x = y$
Key position	$\psi_2 = \nu r.\{^{\mathrm{enc}(n,\mathtt{s},r)}/_x\}$	$\mathrm{dec}(x, k) = n$
Destructors	$\psi_3 = \{^{\pi_1(\mathtt{s})}/_x\}$	$x = a$
Retrieve rule	$\psi_4 = \{^{\mathrm{sign}(\mathtt{s},\mathrm{priv}(a))}/_x, {}^{\mathrm{pub}(a)}/_y\}$	$\mathrm{check}(n, x, y) = \mathrm{ok}$

Syntactic secrecy vs strong secrecy (passive case)

A frame $\varphi = \nu\widetilde{n}.\sigma$ is *well-formed* if

- encryption is probabilistic,
- $\mathtt{s}$ is not part of a key and,
- φ does not contain destructor symbols.

Theorem 1 For any well-formed frame, week secrecy is equivalent to strong secrecy, that is

$$\varphi \not\vdash \mathtt{s} \quad \text{iff} \quad \varphi(^{M}/_{\mathtt{s}}) \approx_s \varphi(^{M'}/_{\mathtt{s}})$$

for all M, M' closed terms public wrt φ.

Proof sketch (passive case)

Base case Lemma: $u\sigma({}^{M}/_{\mathtt{s}}) = v\sigma({}^{M}/_{\mathtt{s}})$ implies $u\sigma = v\sigma$.

Transfer Lemma: $\varphi = \nu\widetilde{n}.\sigma$ well-formed frame, $\varphi \nvdash \mathtt{s}$.
If $u\sigma({}^{M}/_{\mathtt{s}}) \rightarrow w$, then

- there exists $\varphi' = \nu\widetilde{n}.\sigma'$ extending φ, preserving deducible terms and,
- such that $w = w'\sigma'({}^{M}/_{\mathtt{s}})$ and $u\sigma \rightarrow w'\sigma'$.

Hence, $u\sigma({}^{M}/_{\mathtt{s}})\!\downarrow = u'\sigma'({}^{M}/_{\mathtt{s}})$ and $v\sigma({}^{M}/_{\mathtt{s}})\!\downarrow = v'\sigma'({}^{M}/_{\mathtt{s}})$.

Applied-pi calculus (1) [Abadi Fournet]

(Plain) processes are defined by the grammar:

$P, Q, R :=$	processes
$\mathbf{0}$	null process
$P \vert Q$	parallel composition
$!P$	replication
$\nu n.P$	name restriction
$[M = N].P$	conditional
$p(z).P$	message input
$\overline{p}\langle M\rangle.P$	message output

Structural equivalence rules:

$$A \equiv A \mid \mathbf{0}$$
$$A \mid (B \mid C) \equiv (A \mid B) \mid C$$
$$A \mid B \equiv B \mid A$$
$$!P \equiv P \mid !P$$
$$\nu n.\mathbf{0} \equiv \mathbf{0}$$
$$\nu u.\nu v.A \equiv \nu v.\nu u.A$$
$$A \mid \nu u.B \equiv \nu u.(A \mid B) \quad \text{if } u \notin \mathrm{fv}(A) \cup \mathrm{fn}(A)$$

Internal reduction is given by the rules:

COMM	$\overline{p}\langle x\rangle.P \mid p(x).Q \rightarrow P \mid Q$
COND	$[M = M].P \rightarrow P$

Applied-pi calculus (2)

Extended processes are defined by the grammar:

$A, B :=$	extended processes
P	plain process
$A \mid B$	parallel composition
$\nu n.A$	name restriction
$\nu x.A$	variable restriction
$\{^M/_x\}$	active substitution

Additional structural equivalence rules:

ALIAS	$\nu x.\{^M/_x\} \equiv \mathbf{0}$
SUBST	$\{^M/_x\} \mid A \equiv \{^M/_x\} \mid A\{^M/_x\}$
REWRITE	$\{^M/_x\} \equiv \{^N/_x\}$ if $M =_E N$

Labeled reduction is defined by the following rules:

IN $\quad p(x).P \xrightarrow{p(M)} P\{^M/_x\}$

OUT-ATOM $\quad \overline{p}\langle u \rangle.P \xrightarrow{\overline{p}\langle u \rangle} P$

OPEN-ATOM $\quad \dfrac{A \xrightarrow{\overline{p}\langle u \rangle} A'}{\nu u.A \xrightarrow{\nu u.\overline{p}\langle u \rangle} A'} \quad u \neq p$

SCOPE $\quad \dfrac{A \xrightarrow{\alpha} A'}{\nu u.A \xrightarrow{\alpha} \nu u.A'} \quad u$ not in α

PAR $\quad \dfrac{A \xrightarrow{\alpha} A'}{A \mid B \xrightarrow{\alpha} A' \mid B}$ condition (*)

STRUCT $\quad \dfrac{A \equiv B \qquad B \xrightarrow{\alpha} B' \qquad B' \equiv A'}{A \xrightarrow{\alpha} A'}$

where u is a metavariable that ranges over names and variables.

Modeling protocols

The Yahalom protocol:

$$\begin{aligned} A \Rightarrow B : &\quad A, N_a \\ B \Rightarrow S : &\quad B, \{A, N_a, N_b\}_{K_{bs}} \\ S \Rightarrow A : &\quad \{B, K_{ab}, N_a, N_b\}_{K_{as}}, \{A, K_{ab}\}_{K_{bs}} \\ A \Rightarrow B : &\quad \{A, K_{ab}\}_{K_{bs}} \end{aligned}$$

$$\begin{aligned} P_A = &\quad \nu n_a.\overline{p}\langle a, n_a\rangle.p(z_a).[b = \pi_1(\mathrm{dec}(\pi_1(z_a), k_{as}))]. \\ &\quad [n_a = \pi_1(\pi_2(\pi_2(\mathrm{dec}(\pi_1(z_a), k_{as}))))].\overline{p}\langle\pi_2(z_a)\rangle \\ \xrightarrow{\nu z.\overline{p}\langle z\rangle} &\quad \nu n_a.(\{^{\langle a,n_a\rangle}/_z\} \mid p(z_a).[b = u_b].[n_a = u_{n_a}].\overline{p}\langle\pi_2(z_a)\rangle) \\ \xrightarrow{p(\langle b,z\rangle)} &\quad \nu n_a.(\{^{\langle a,n_a\rangle}/_z\} \mid [b = \pi_1(\mathrm{dec}(b, k_{as}))].[n_a = u'_{n_a}].\overline{p}\langle a, n_a\rangle) \end{aligned}$$

Definitions of secrecy

We say that $\mathtt{s}$ is *syntactically secret* in P if, for every P' such that $P \Rightarrow^* P'$, $\mathtt{s}$ is not deducible from P', that is $\varphi(P') \not\vdash \mathtt{s}$.

Definitions of secrecy

We say that $\mathtt{s}$ is *syntactically secret* in P if, for every P' such that $P \Rightarrow^* P'$, $\mathtt{s}$ is not deducible from P', that is $\varphi(P') \not\vdash \mathtt{s}$.

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

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

We say that $\mathtt{s}$ is *strongly secret* in P if $P({}^{M}/_{\mathtt{s}}) \approx_l P({}^{M'}/_{\mathtt{s}})$ for any closed terms M, M' public wrt P.

Hypotheses

A process P is *well-formed* if:

- encryption is probabilistic
- there are no destructors above constructors, nor above s
- the keys are ground
- for any test $[M = N]$, the terms M, N are
  - name,
  - constant,
  - or of the form $\pi^1(\mathrm{dec}(\ldots\pi^n(\mathrm{dec}(\pi^{n+1}(z), k_n))\ldots, k_1))$, where the π^i are words on $\{\pi_1, \pi_2\}$.

Ground keys

Counter-example for non ground keys:

$$P = \nu k, r, r'.(\overline{c}\langle \mathrm{enc}(\mathtt{s}, k, r)\rangle \mid c(z).\overline{c}\langle \mathrm{enc}(a, \mathrm{dec}(z, k), r')\rangle)$$

$$\rightarrow \nu k, r, r'.(\{^{\mathrm{enc}(\mathtt{s},k,r)}/_{z}\} \mid \overline{c}\langle \mathrm{enc}(a, \mathtt{s}, r')\rangle)$$

$$\xrightarrow{\nu z'.\overline{c}\langle z'\rangle} \nu k, r, r'.\{^{\mathrm{enc}(\mathtt{s},k,r)}/_{z}, {}^{\mathrm{enc}(a,\mathtt{s},r')}/_{z'}\}$$

Tests over $\mathtt{s}$

Conditionals should not test on the secret:

$$P = \nu k, r.(\overline{p}\langle \mathrm{enc}(\mathtt{s}, k, r)\rangle \mid p(z).[\mathrm{dec}(z, k) = a].\overline{p}\langle \mathrm{ok}\rangle)$$
$$\rightarrow \nu k, r.(\{^{\mathrm{enc}(\mathtt{s},k,r)}/_{z}\} \mid [\mathtt{s} = a].\overline{p}\langle \mathrm{ok}\rangle)$$

$P(^{a}/_{\mathtt{s}}) \not\approx_{l} P(^{b}/_{\mathtt{s}})$.

There may be hidden tests on the secret. Yahalom protocol, again:

$$\begin{array}{ll} A \Rightarrow B: & A, N_a \\ B \Rightarrow S: & B, \{A, N_a, N_b\}_{K_{bs}} \\ S \Rightarrow A: & \{B, K_{ab}, N_a, N_b\}_{K_{as}}, \{A, K_{ab}\}_{K_{bs}} \\ A \Rightarrow B: & \{A, K_{ab}\}_{K_{bs}} \end{array}$$

Hence we mark the test $[n_a = \pi_1(\pi_2(\pi_2(\mathrm{dec}(\pi_1(z_a), k_{as}))))]$.

$\longrightarrow$ We construct a set of ”potentially dangerous“ tests $\mathcal{M}_t$.

Syntactic secrecy vs strong secrecy (active case)

Definition: A protocol *does not test over* $\mathtt{s}$ if for any test $[M = N]$ or $[N = M]$ such that $M \in \mathcal{M}_t$, N is a restricted name.

Theorem 2 For well-formed processes

- which do not test over $\mathtt{s}$
- + some syntactic condition to ensure that messages sent through the network do not contain destructor directly above the secret.

then syntactic secrecy is equivalent with strong secrecy.

Syntactic secrecy vs strong secrecy (active case)

Definition: A protocol *does not test over* s if for any test $[M = N]$ or $[N = M]$ such that $M \in \mathcal{M}_t$, N is a restricted name.

Theorem 2 For well-formed processes

- which do not test over s
- + some syntactic condition to ensure that messages sent through the network do not contain destructor directly above the secret.

then syntactic secrecy is equivalent with strong secrecy.

Proof elements:

- Any frame produced by the protocol is an extended well-formed frame.
- If $[T_1 = T_2]$ is a test in P, then $T_1\sigma(^{M}/_{\mathtt{s}}) =_E T_2\sigma(^{M}/_{\mathtt{s}})$ implies $T_1\sigma(^{M'}/_{\mathtt{s}}) =_E T_2\sigma(^{M'}/_{\mathtt{s}})$.

Conclusion

We have proved that syntactic secrecy implies strong secrecy

- in the passive case, for symmetric and asymmetric encryption and digital signatures;
- in the active case, for symmetric encryption, under some (rather tight) conditions;

Application: Yahalom, Wide Mouthed Frog, symmetric key Needham-Schroeder protocols are strongly secret.

Conclusion

We have proved that syntactic secrecy implies strong secrecy

- in the passive case, for symmetric and asymmetric encryption and digital signatures;
- in the active case, for symmetric encryption, under some (rather tight) conditions;

Application: Yahalom, Wide Mouthed Frog, symmetric key Needham-Schroeder protocols are strongly secret.

Further work:

- analyse more primitives (symmetric encryptiom, ...)
- relax some conditions (allow more tests, ...)

Related work:

- H. Hüttel. *Deciding framed bisimilarity.*
- B. Blanchet. *Automatic Proof of Strong Secrecy for Security Protocols.*