Inductive Analysis of Security Protocols in Isabelle/HOL with Applications to Electronic Voting

Denis Butin

Introduction

- Network communication sensitive: banking, private correspondence, business-critical data
- Cryptography contributes to network security…
- …But not sufficient in itself!

Security Layers

Several levels at which attacks can and have been led:

- Hardware (e.g. side-channel attacks)
- Cryptographic primitives
- Security protocols
- Ceremonies

Security Protocol Goals

- Classically: authentication, secret sharing, electronic payment...
- New, more complex needs: electronic voting, secure multiparty computation, electronic cash...

Analysing Security Protocols

Many methods:

- Model checking
- Automated / interactive theorem proving
- Static analysis, applied pi calculus, strand spaces...

Tools with automation: ProVerif, AVISPA, Scyther, AKiSs...

Interactive Theorem Proving

- Uses mathematical reasoning to determine if protocol reaches its security goals
- Unlike model checking, population unbounded
- Doesn't provide explicit attack but may give clues
- Interactive
- Our choice — Isabelle

The Inductive Method

- Application of Isabelle ("generic proof assistant"!) to security protocol verification
- $\star$ Paulson 1996, then Bella
- Uses mathematical induction to model and verify protocols + goals

Principles of the Inductive Method

- Unbounded number of agents
- Dedicated datatypes (keys, hashes, nonces. . .)
- Events for message sending, reception, agent knowledge
- Inductive reasoning over network event lists (traces)
- Cryptographic algorithms idealised

Threat Model

- Attacker = "Spy"
- Controls network (Dolev-Yao)
- Eavesdropping + dynamic behaviour, can also act like normal agent

Goal Definition and Proving

- Protocol security goals $\longleftrightarrow$ predicates over all possible traces
- User specifies techniques to use: basic induction, rewriting, automatic prover...
- In most cases, several subgoals generated and user input required again

Modelling Properties — Example

Authentication of an agent:

$$\llbracket A \notin bad;\ B \notin bad;\ evs \in ns\_public \rrbracket \Longrightarrow$$
$$Crypt\ (pubEK\ A)\ \{\!|Nonce\ NA,\ Nonce\ NB,\ Agent\ B|\!\} \in parts\ (spies\ evs) \longrightarrow$$
$$Says\ A\ B\ (Crypt\ (pubEK\ B)\ \{\!|Nonce\ NA,\ Agent\ A|\!\}) \in set\ evs \longrightarrow$$
$$Says\ B\ A\ (Crypt\ (pubEK\ A)\ \{\!|Nonce\ NA,\ Nonce\ NB,\ Agent\ B|\!\}) \in set\ evs$$

Protocols Verified in Isabelle So Far

Protocol	**Class**	**Year**	**Author(s)**
Yahalom	Key sharing, authentication	1996	Paulson
NS symmetric	Key sharing	1996	Paulson & Bella
Otway-Rees (with variants)	Authentication	1996	Paulson
Woo-Lam	Authentication	1996	Paulson
Otway-Bull	Authentication	1996	Paulson
NS asymmetric	Authentication	1997	Paulson
TLS	Multiple	1997	Paulson
Kerberos IV	Mutual authentication	1998	Bella
Kerberos BAN	Mutual authentication	1998	Paulson & Bella
SET suite	Multiple	2000+	Bella *et al.*
Abadi *et al.* certified e-mail	Accountability	2003	Bella *et al.*
Shoup-Rubin smartcard	Key distribution	2003	Bella
Zhou-Gollmann	Non-repudiation	2003	Paulson & Bella
Kerberos V	Mutual authentication	2007	Bella
TESLA	Broadcast authentication	2009	Schaller *et al.*
Meadows distance bounding	Physical	2009	Basin *et al.*
Multicast NS symmetric	Key sharing	2011	Martina
Franklin-Reiter	Byzantine	2011	Martina
Onion routing	Anonymising	2011	Li & Pang

New Applications — General Approach

- Adapt Isabelle theory framework (specifications of messages, events, keys, knowledge. . .)
- Model protocol steps
- Formalise novel guarantees: sometimes hardest step
- Proofs (interactive)

Analysing Composed Protocols

- Typical real-world scenario of security protocol use
- Analysis issue not solved in general, partially supported by Scyther
- Not done before in the Inductive Method

Protocol Composition Paradigm

- Certificate distribution sequenced with authentication
- Specified by two linked inductive models
- Better guarantee availability (implicit public key binding)

Protocol Composition — Discussion

- Scalable semantics, not limited to two protocols
- No compositionality theorem as for Scyther
- Case study extendable to detailed PKI

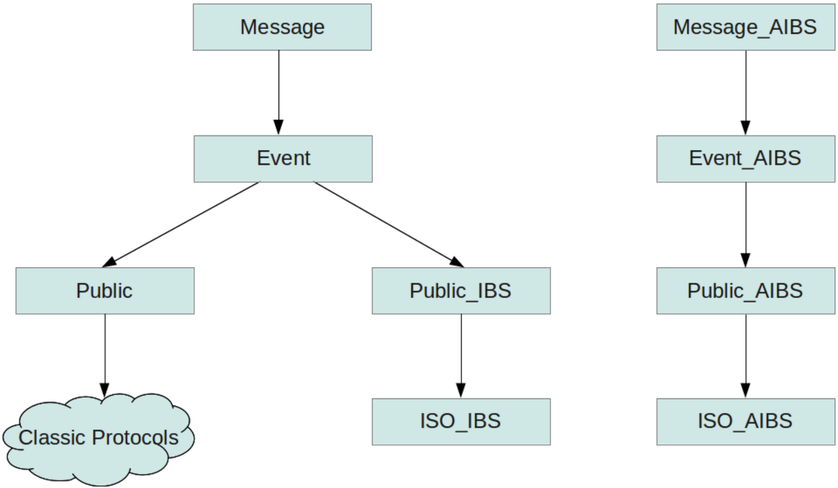
Auditable Identity-Based Signatures

- Proposed by David Gray in 2007
- Provide stronger non-repudiation than “standard” IBS (mitigate key escrow)
- Separate audit step allows third party to ensure signature origin
- Relies on additional audit key-pair; private one required to sign and registered with KGC

ISO/IEC 9798-3

- 2010 Amendment presents new authentication protocols
- We study *Five-pass mutual authentication with TTP, initiated by A*
- Side-by-side specification of IBS and AIBS versions
- Focus is not on the protocol itself but on AIBS

Auditable Identity-Based Signatures – Theories



Auditable Identity-Based Signatures – Modelling

- Key package datatype: *datatype pack* $=$ *Pack key key*
- Auditable signature structure:
  $Crypt\ (priSK\ A)\ \{\!|Crypt\ (priEK\ A)\ M,\ M|\!\}$
- Can only sign with key package + private key:
  $$[\![evss \in iso;\ X \in synth(analz\ (spies\ evss));\ Key\ (priEK\ A) \in analz\ (spies\ evss);\ Pkg\ (KP\ A\ B) \in analz\ (spies\ evss)]\!] \Longrightarrow Notes\ Spy\ Crypt\ (priSK\ B)\ \{\!|Crypt\ (priEK\ A)\ X,\ X|\!\}\ \#\ evss \in iso$$

Auditable Identity-Based Signatures – Modelling

- *candidates* function — input agent name, output set of potential signers who leave a trace
- Classic authentication results + focus on signatures
- Comparative analysis shows operational auditable feature of AIBS

Extensions for E-voting Protocols — Introduction

- E-voting use is spreading quickly in the EU and elsewhere
- Sensitive, need for formal guarantees
- Inductive Method: protocol verification through theorem proving + mathematical induction
- Toolbox built with FOO as example protocol

Extensions for E-voting Protocols — Motivation

- Analysis of e-voting dominated by ProVerif automatic verifier
- Powerful, but sometimes limited
- Motivation to fill in the gaps with complementary, alternative approach

Related Work

- Ryan / Kremer / Delaune: applied pi calculus, partially mechanized through ProVerif
- Observational equivalence: traces in which two voters swap their votes are equivalent in a sense
- Parts of the proof done by hand

E-voting Protocols

- New properties : privacy, verifiability, coercion-resistance. . .
- Partially studied with applied pi calculus, but with little mechanisation
- Often require modelling new crypto primitives

E-voting protocols: properties

- Eligibility
- Fairness
- Privacy / Receipt freeness / Coercion resistance – linkability concept (hard)
- Individual / Universal verifiability

The FOO Protocol

- Fujioka, Okamoto and Ohta, 1992
- Two election officials, bit commitment, blind signatures
- Signed, blinded commitment on a vote
- 6 steps

Specifying Blind Signatures

- Directly in `Message.thy` — limitation of operators interplay
- Solution: as part of inductive model

$$
\begin{array}{l}
[\![\mathit{evsb} \in \mathit{foo};\ \mathit{Crypt}\ (\mathit{priSK}\ V)\ \mathit{BSBody} \in \mathit{analz}\ (\mathit{spies}\ \mathit{evsb});\\
\ \mathit{BSBody} = \mathit{Crypt}\ b\ (\mathit{Crypt}\ c\ (\mathit{Nonce}\ N));\ b \in \mathit{symKeys};\\
\ \mathit{Key}\ b \in \mathit{analz}\ (\mathit{spies}\ \mathit{evsb})]\!]\\
\ \Longrightarrow \mathit{Notes}\ \mathit{Spy}\ (\mathit{Crypt}\ (\mathit{priSK}\ V)\ (\mathit{Crypt}\ c\ (\mathit{Nonce}\ N)))\ \#\ \mathit{evsb} \in \mathit{foo}
\end{array}
$$

What Is Privacy in E-Voting?

- Crucial point: privacy is NOT confidentiality of vote...
- ...But unlinkability of voter and vote
- In Pro-Verif, done with observational equivalence between swapped votes

Privacy in the Inductive Method: *aanalz*

```
primrec aanalz :: "agent => event list => msg set set"
where
  aanalz_Nil:   "aanalz A [] = {}"
| aanalz_Cons:
"aanalz A (ev # evs) =
 (if A = Spy then
  (case ev of
    Says A' B X ⇒
     (if A' ∈ bad  then aanalz Spy evs
      else if isAnms X
          then insert              ({Agent B} ∪ (analzplus {X} (analz(knows Spy evs)))) (aanalz Spy evs)
          else insert ({Agent A'} Un {Agent B} ∪ (analzplus {X} (analz(knows Spy evs)))) (aanalz Spy evs)
 )
  | Gets A' X ⇒ aanalz Spy evs
  | Notes A' X ⇒ aanalz Spy evs)
  else aanalz A evs)"
```

Extract associations from honest agent's messages

Privacy in the Inductive Method: *asynth*

inductive_set
asynth :: *msg set set* $\Rightarrow$ *msg set set*
for as :: *msg set set where*
asynth_Build [intro]:
$[\![a1 \in as;\ a2 \in as;\ m \in a1;\ m \in a2;\ m \neq Agent\ Adm;\ m \neq Agent\ Col]\!]$
$\Longrightarrow a1 \cup a2 \in asynth\ as$

Build up association sets from associations with common elements. Only pairwise so far!

Privacy in the Inductive Method: Theorem Statement

theorem foo_V_privacy_asynth:

$$
\begin{array}{l}
[\![\mathit{Says}\ V\ \mathit{Adm}\ \{\!|\mathit{Agent}\ V,\\
\qquad\qquad \mathit{Crypt}\ (\mathit{priSK}\ V)\ (\mathit{Crypt}\ b\ (\mathit{Crypt}\ c\ (\mathit{Nonce}\ \mathit{Nv})))|\!\}\in \mathit{set}\ \mathit{evs};\\
\ a \in (\mathit{asynth}\ (\mathit{aanalz}\ \mathit{Spy}\ \mathit{evs}));\\
\ \mathit{Nonce}\ \mathit{Nv} \in a;\ V \notin \mathit{bad};\ V \neq \mathit{Adm};\ V \neq \mathit{Col};\ \mathit{evs} \in \mathit{foo}]\!]\\
\ \Longrightarrow \mathit{Agent}\ V \neq a
\end{array}
$$

If a regular voter started the protocol, the corresponding vote and identity are unlinkable.

Privacy in the Inductive Method: Proving Process

- Genericity of steps 2 and 4 yields proof complexity
- Genericity is natural consequence of respecting guarantee availability
- Strategy: map components in asynth to possible origins in aanalz
- Taxonomy of structures of elements in aanalz
- Divide & conquer

Contributions

- Conference publications:
  - *Holistic Analysis of Mix Protocols* — International Conference on Information Assurance and Security (IAS 2011)
  - *Verifying Privacy by Little Interaction and No Process Equivalence* — International Conference on Security and Cryptography (SECRYPT 2012)
- Workshop talk:
  - *Electronic Voting Protocol Analysis with the Inductive Method* — 2011 miniWorkshop on Security Frameworks (mWSF11)

Conclusions

- Flexibility of Inductive Method confirmed...
- ...but limitations related to message datatype extension
- Very different approach from most used tools (ProVerif, Scyther)...
- ...hence potential for complementarity!

Future Work

- Focus on the e-voting part of the work
- Need stronger association synthesis — proof complexity challenge
- Analyse more recent e-voting protocols
- Article on AIBS chapter
- Long-term goal: reengineer message datatype completely for broader primitive support