Towards Verifying Voter Privacy Through Unlinkability

Denis Butin (Inria), David Gray (DCU) & Giampaolo Bella (Catania)

Introduction

- E-voting protocols increasingly used — need for formal verification!
- Key property: voter privacy / ballot secrecy
- Inductive Method: protocol verification through theorem proving
- Extension for e-voting privacy analysis through unlinkability

Extensions for E-voting Protocols — Motivation

- Analysis of e-voting dominated by the indistinguishability approach, with automated tools: ProVerif, more recently AKiSs
- Powerful, but sometimes limited (approximations / termination issues)
- Motivation for complementary, alternative approach
- This work: first specification of voter privacy in an interactive theorem prover

E-voting Protocols

- New properties when compared to classic security protocols: privacy, verifiability, coercion-resistance. . .
- Partially studied with applied pi calculus, but with little mechanisation
- Often require modelling new cryptographic primitives (e.g. blind signatures)

Privacy in E-Voting

- Crucial point: privacy is not confidentiality of ballot. . .
- . . . But unlinkability between voter and ballot! Operational view / natural threat model
- In ProVerif, done with observational equivalence between swapped votes

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

Method: the Inductive approach

- Mathematical induction on protocol steps: one subgoal per step
- Dolev-Yao threat model
- Tool support: Isabelle, a generic interactive theorem prover, using HOL

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

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

$$\llbracket evsb \in foo;\ Crypt\ (priSK\ V)\ BSBody \in analz\ (spies\ evsb);$$
$$BSBody = Crypt\ b\ (Crypt\ c\ (Nonce\ N));\ b \in symKeys;$$
$$Key\ b \in analz\ (spies\ evsb)\rrbracket$$
$$\Longrightarrow Notes\ Spy\ (Crypt\ (priSK\ V)\ (Crypt\ c\ (Nonce\ N)))\ \#\ evsb \in foo$$

Plain signature obtained from knowledge of blind signature and corresponding (symmetric) blinding factor

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 B} ∪ {Agent A'} ∪
                         (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 (Spy's point of view)

Privacy in the Inductive Method: *asynth*

```
inductive_set
 asynth :: msg set set ⇒ msg set set
 for as :: msg set set where
  asynth_Build [intro]:
  ⟦a1 ∈ as; a2 ∈ as; m ∈ a1; m ∈ a2; m ≠ Agent Adm; m ≠ Agent Col⟧
  ⟹ a1 ∪ a2 ∈ 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:
$\llbracket$*Says V Adm* $\{\!|$*Agent V,*
Crypt (priSK V) (Crypt b (Crypt c (Nonce Nv)))$|\!\}$ $\in$ *set evs;*
a $\in$ *(asynth (aanalz Spy evs));*
Nonce Nv $\in$ *a; V* $\notin$ *bad; V* $\neq$ *Adm; V* $\neq$ *Col; evs* $\in$ *foo*$\rrbracket$
$\Longrightarrow$ *Agent V* $\neq$ *a*

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

Privacy in the Inductive Method: Proving Ingredients

- *asynth_insert*: splits the association synthesis set — three disjunctions yielding simpler subgoals
- Third disjunction bulk of work: structure of sets in *aanalz*, needs more specialised lemmas
- Family of lemmas stating that fresh nonces do not appear in association syntheses
- *aanalz_traffic*: relates non-agent names elements in associations with traffic

Privacy in the Inductive Method — Lessons Learned

- Initial proof effort significant, magnitude larger than effort for reuse (even between protocol subgoals)
- Coherent line of reasoning emerged — hope for re-usability
- Protocol-independent results about crypto operators
- Greater insight into protocol intricacies
- Main issue: association synthesis not general enough

Conclusions

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

Future Work

- Need stronger association synthesis — proof complexity challenge
- Modelling and analysis of related properties: receipt-freeness, coercion-resistance
- Investigation of recent e-voting protocols that are problematic for existing tools

Questions?