Analysis of Verifiability in Electronic Voting

Mark Ryan
University of Birmingham

based on joint work with

Ben Smyth
Steve Kremer
Mounira Kourjieh

VOTE-ID, Luxembourg, 2009

Desired properties

Fault tolerance

Declared tally

Inalterability

Robustness

Correctness

Scalability

Availability

Eligibility

Secret ballot

Eligibility verifiability

Universal verifiability

Privacy

Verifiability

Receipt freeness

Coercion resistance

Individual verifiability

Thanks for the pic: Ben Smyth / Cătălin Hrițcu

Election verifiability

verifiability

Election verifiability

verifiability
auditability

Election verifiability

$$\text{end-to-end} \left\{ \begin{array}{l} \text{verifiability} \\ \text{auditability} \end{array} \right.$$

Election verifiability

$$\text{end-to-end} \left\{ \begin{array}{l} \text{verifiability} \\ \text{auditability} \end{array} \right.$$

- Election results can be fully verified by voters/observers
- The software provided by election authorities does not need to be trusted
- The software used to perform the verification can be sourced independently

Dagstuhl Accord

Participants of the 2007 Dagstuhl Conference on Frontiers of E-Voting agree that:

Taking advantage of technology to improve large-scale elections has recently captured the interest of researchers coming from a number of disciplines. The basic requirements pose an apparently irreconcilable challenge: while voter confidence hinges on transparently ensuring integrity of the outcome, ballot secrecy must also be ensured. Current systems can only address these essential requirements by relying on trust in those conducting the election or by trust in the machines and software they use. Some promising new systems dramatically reduce the need for such trust.

What are called “end-to-end” voting systems, for example, allow each voter to ensure that his or her vote cast in the booth is recorded correctly. They then allow anyone to verify that all such recorded votes are included in the final tally correctly. Surprisingly, typically through use of encryption, these systems can also provide privacy of votes. They do this without introducing any danger of “improper influence” of voters, as in vote buying and coercion. Moreover, such systems offer all these properties without relying on trust in particular persons, manual processes, devices, or software.

History

Electronic voting

- FOO [Fujioka/Okamoto/Ohta 92]
- Civitas [Juels/Catalano/Jakobsson 05] [Clarkson/Chong/Myers 08]
- Helios [Adida 08] [Adida/deMarneffe/Pereira/Quisq. 09]

Paper-and-scan

- Visual crypto [Chaum 04]
- Prêt-à-Voter [P.Ryan/Schneider/Chaum 05]
- Punchscan [Chaum/Clark/Popoveniuc 06]
- ThreeBallot [Rivest 06]

Election of president at University of Louvain

The election

- Based on Helios
  - but many modifications
- 25,000 potential voters
  - 5000 registered, 4000 voted
  - Educated, but not technical
- 30% voters checked their vote
  - No valid complaints
- erifiability
  - Anyone can write code to verify the election
  - Sample python code provided

No coercion resistance

- Only recommended for low-coercion environments
- Re-votes are allowed, but don't help w.r.t. "insider" coercer

[Adida/deMarneffe/Pereira/-Quisquater 09]

OPEN-AUDIT OF THE RESULTS OF THE RECTOR ELECTION 2009

The voting system used for this election provides *universally verifiable elections*. This means that:

1. a voter can verify that her ballot is cast as intended (her ballot reflects her own opinion),
2. a voter can verify that her ballot is included *unmodified* in the collection of ballots to be used at tally time,
3. anyone can verify that the election result is consistent with that collection of ballots.

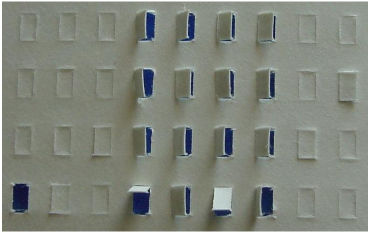

Where is my
VOTE?
GETTY IMAGES


Démocratie
& Liberté
WHERE
IS
MY VOTE


Where is
my vote ?


WHERE
IS MY
VOTE?


TURN OVER FOR PAGE




SYSTEMS
Which state is the first in the nation in efforts to reform elections?
(Vote for One)
New York
California
Georgia
Michigan

Election verifiability

Individual verifiability

A voter can check her own vote is included in the tally.

Universal verifiability

Anyone can check that the declared outcome corresponds to the tally.

Eligibility verifiability

Anyone can check that only eligible votes are included in the declared outcome.

Remarks

- Verifiability $\neq$ correctness
- What system components need to be trusted in order to carry out these checks?

- Formalisation of election verifiability
- Analysis of systems:
  - FOO
  - JCJ/Civitas
  - Helios/UCL [in progress]

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], but more general w.r.t. cryptography

Advantages:

- naturally models a Dolev-Yao attacker
- allows us to model less classical cryptographic primitives
- both reachability-bases 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

Equations to model the cryptography: examples

1. *Encryption and signatures*

```
   decrypt( encrypt(m,pk(k)), k )   =   m
checksign( sign(m,k), m, pk(k) )   =   ok
```

2. *Blind signatures*

```
unblind( sign( blind(m,r), sk ), r )   =   sign(m,sk)
```

3. *Designated verifier proof of re-encryption*
   The term dvp(x,renc(x,r),r,pkv) represents a proof designated for the owner of pkv that x and renc(x,r) have the same plaintext.

```
checkdvp(dvp(x,renc(x,r),r,pkv),x,renc(x,r),pkv) = ok
       checkdvp( dvp(x,y,z,skv), x, y, pk(skv) ) = ok.
```

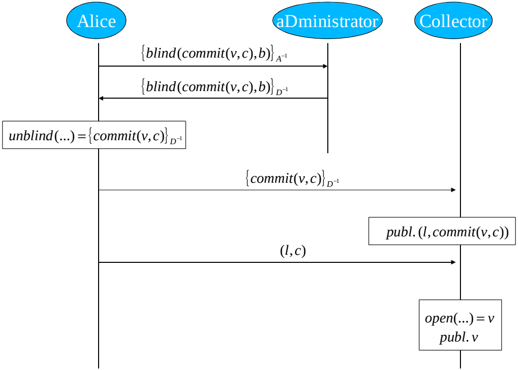
4. *Zero-knowledge proofs of knowledge*
   pf(k,x,y) represents proof that I know k such that dec(x,k)=y.

```
checkpf( pf(k,x,dec(x,k)), x, dec(x,k) ) = ok.
```

Coding protocols as processes

Example ([FOO'92]):

```
processV =
  new b; new c;
  let bcv = blind(commit(v,c),b) in
  out(ch, (sign(bcv, skv)));
  in(ch,m2);
  if getMess(m2,pka)=bcv then
  let scv = unblind(m2,b) in
  str_phase 1;
  out(ch, scv);
  in(ch,(l, =scv));
  str_phase 2;
  out(ch,(l,c)).
```



Formalisation of privacy-type properties

Definition (Privacy)

A voting protocol respects privacy if
$S[V_A\{^a/_v\} \mid V_B\{^b/_v\}] \approx_\ell$
$S[V_A\{^b/_v\} \mid V_B\{^a/_v\}]$

Definition (Receipt-freeness)

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

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

Definition (Coercion resistance)

VP is coercion resistant if there exists a process V' such that for any $C = \nu c_1.\nu c_2.(\_ \mid P)$ satisfying

- $\tilde{n} \cap fn(C) = \emptyset$
- $S[C[V_A\{^?/_v\}^{c1,c2}] \mid V_B\{^a/_v\}] \approx_\ell S[V_A\{^c/_v\}^{chc} \mid V_B\{^a/_v\}]$

we have

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

[Delaune/Kremer/Ryan 08]

Election verifiability

We suppose that the protocol involves

- Voter credentials (typically, a public part and a private part for each voter)
- A bulletin board, on which are placed entries corresponding to voter's outputs.

Election verifiability

A protocol satisfies *election verifiability* if

- Each voter's credentials are unique
- Each voter's bulletin board entry is unique
- There are tests R^{IV}, R^{UV} and R^{EV} satisfying certain acceptability conditions.

Individual verifiability

Intuition: a protocol satisfies individual verifiability if there is a test

$$R^{IV}(\text{my\_vote}\,,\ \text{my\_data}\,,\ \text{bb\_entry})$$

that a voter can apply after the election.

The test succeeds iff the bulletin board entry corresponds to the voter's vote and data.

Acceptability conditions for R^{IV}

- For all votes s, there is an execution of the protocol that produces $\tilde{M}$ such that some bulletin board entry T satisfies $R^{IV}(s, \tilde{M}, T)$.
- The bulletin board entry determines the vote, that is:

$$\forall s, t, \tilde{M}, \tilde{N}, T \ \left(R^{IV}(s, \tilde{M}, T) \wedge R^{IV}(t, \tilde{N}, T) \Rightarrow s = t \right)$$

Universal verifiability

Intuition: a protocol satisfies universal verifiability if there is a test

$$R^{UV}(\text{declared\_outcome}\,,\ \text{bb\_entries}\,,\ \text{proof})$$

that an observer can apply after the election.

The test succeeds iff the declared outcome is correct w.r.t. the bb entries and the proof.

Acceptability conditions for R^{UV}

- $\tilde{T}$ determines $\tilde{s}$, that is,

$$R^{UV}(\tilde{s}_1, \tilde{T}, p_1) \wedge R^{UV}(\tilde{s}_2, \tilde{T}, p_2) \Rightarrow \tilde{s}_1 = \tilde{s}_2$$

- The observer opens the bb entry the same way as the voter:

$$R^{IV}(s, \tilde{M}, T) \wedge R^{UV}(\tilde{s}, \tilde{T}, p') \Rightarrow \exists p'.\, R^{UV}(\tilde{s} \circ s, \tilde{T} \circ T, p')$$

"Pointwise" universal verifiability

In some cases, the proof may be a bijection $p : \underline{n} \to \underline{n}$ such that

$$R^{UV}(\tilde{s}, \tilde{T}, p) = \bigwedge_{i=1}^{n} R^{UV}_{\bullet}(s_i, T_{p(i)})$$

This is the case for FOO and JCJ/Civitas, but not for Helios/UCL.

In this case, the verification is slightly simpler:

Equivalent acceptability conditions for $R^{UV}_{\bullet}$

- $R^{UV}_{\bullet}(s, T) \wedge R^{UV}_{\bullet}(t, T) \Rightarrow s = t$
- $R^{IV}(s, \tilde{M}, T) \Rightarrow R^{UV}_{\bullet}(s, T)$

This is the case we have implemented, although the more general case is probably straightforward.

Eligibility verifiability

Intuition: a protocol satisfies eligibility verifiability if there is a test

$$R^{EV}(\text{public\_credentials}\,,\ \text{bb\_entries}\,,\ \text{proof})$$

that an observer can apply after the election.

Again, for some protocols, the proof may consist of a bijection $p : \underline{n} \longrightarrow \underline{n}$ that allows the verifier to perform the test pointwise:

Acceptability conditions for $R^{EV}_{\bullet}$:

- $R^{EV}_{\bullet}(U, T) \wedge R^{EV}_{\bullet}(V, T) \Rightarrow U = V$
- If voter voting s with credential U and voting data $\tilde{M}$ generates bulletin board entry T, then

$$R^{IV}(s, \tilde{M}, T) \Leftrightarrow R^{EV}_{\bullet}(U, T)$$

Election verifiability

A voting process $C[!\nu\tilde{a}.(P \mid Q[\overline{c}\langle U\rangle])]$ satisfies *election verifiability* if voter's credentials and bulletin board entries are unique and there exists tests R^{IV}, R^{UV}, R^{EV} with

- $fv(R^{IV}) \subseteq bv(P) \cup \{v, z\}$
- $fv(R^{UV}) \subseteq \{v, z\}$
- $fv(R^{EV}) \subseteq \{y, z\}$
- $(fn(R^{UV}) \cup fn(R^{EV})) \cap bn(P) = \emptyset$

such that the augmented voting process satisfies the following conditions:

- the *un*reachability assertion: $\overline{\texttt{fail}}\langle\text{true}\rangle$.
- the reachability assertion: $\overline{\texttt{pass}}\langle\text{true}, x\rangle$.

Augmented process

Given a voting process $C[!\nu\tilde{a}.(P \mid Q[\overline{c}\langle U\rangle])]$ and tests R^{IV}, R^{UV}, R^{EV}, the *augmented voting process* is

$$\nu b.(C[!\nu\tilde{a}, b'.(\hat{P} \mid \hat{Q})] \mid R \mid R') \mid R'' \mid R'''$$

where

$$\begin{array}{lcl}
\hat{P} & = & b(v).P.c(z).b'(y).(\overline{\texttt{pass}}\langle R^{IV}, z\rangle \mid \overline{\texttt{fail}}\langle\psi\rangle) \\
\hat{Q} & = & Q[\overline{b'}\langle U\rangle \mid \overline{\mathcal{D}}\langle U\rangle \mid \overline{c}\langle U\rangle] \\
R & = & !\nu s.((!\overline{b}\langle s\rangle) \mid \overline{c}\langle s\rangle) \\
R' & = & b(v').b(v'').c(x').c(x'').c(y').c(y'').c(z').\overline{\texttt{fail}}\langle\phi' \vee \phi'' \vee \phi'''\rangle \\
R'' & = & \texttt{pass}(e).\texttt{pass}(e').\overline{\texttt{fail}}\langle e_1 \wedge e'_1 \wedge (e_2 = e'_2)\rangle \\
R''' & = & \mathcal{D}(e).\mathcal{D}(e').\overline{\texttt{fail}}\langle\neg(e = e')\rangle
\end{array}$$

$$\begin{array}{lcl}
\psi & = & (R^{IV} \wedge \neg R^{UV}) \vee (R^{IV} \wedge \neg R^{EV}) \vee (\neg R^{IV} \wedge R^{EV}) \\
\phi' & = & R^{IV}\{^{v',\tilde{x'},z'}/_{v,\tilde{x},z}\} \wedge R^{IV}\{^{v'',\tilde{x''},z'}/_{v,\tilde{x},z}\} \wedge \neg(v' = v'') \\
\phi'' & = & R^{UV}\{^{v',z'}/_{v,z}\} \wedge R^{UV}\{^{v'',z'}/_{v,z}\} \wedge \neg(v' = v'') \\
\phi''' & = & R^{EV}\{^{y',z'}/_{y,z}\} \wedge R^{EV}\{^{y'',z'}/_{y,z}\} \wedge \neg(y' =_E y'')
\end{array}$$

Case study: FOO

Bulletin board entries are of the form

$$z' = (\ell, com, sig) \;\text{ and }\; z = (\ell, com, sig, rand, vote).$$

Individual verifiability

$$R^{IV} = \mathrm{eq}(z, \langle z'_1, \mathrm{commit}(v, r), \mathrm{unblind}(y', r'), r, v\rangle) \wedge \\ \mathrm{checksign}(z'_3, z'_2, \mathrm{pk}(sk_R))$$

Universal verifiability

$$R^{UV} = \mathrm{eq}(z_2, \mathrm{commit}(z_5, z_4)) \wedge \mathrm{checksign}(z_3, z_2, \mathrm{pk}(sk_R)) \wedge \mathrm{eq}(z5, v)$$

Eligibility verifiability

Not satisfied.

Case study: JCJ/Civitas

Voter *Registrar* *Tallier*

$k \quad <\cdots> \quad U = \mathrm{penc}(k, r'', \mathrm{pk}(sk_T))$

$\xrightarrow{(M, M', spk_F)}$

$\xrightarrow{z = (spk_F, spk_{F'}, petKey)}$

where

- $M = \mathrm{penc}(s, r, \mathrm{pk}(sk_T))$
- $M' = \mathrm{penc}(k, r', \mathrm{pk}(sk_T))$.
- spk_F = proof that M, M' are properly constructed
- $spk_{F'}$ = proof that decryption by Tallier properly performed
- petKey = PET key demonstrating that M' and U have same plaintext

Case study: JCJ/Civitas

Individual verifiability

$$R^{IV} = \phi' \wedge \mathrm{eq}(z_1, \mathrm{spk}_{4,3+l}((v, r, k, r'), (M, M', \mathrm{pk}(sk_T), s_1, \ldots, s_l), \mathcal{F}))$$

Universal verifiability

$$R^{UV}_{\bullet} = \phi \ \wedge \mathrm{eq}(\mathrm{dec}(\mathrm{public}_2(z_2), \mathrm{public}_1(z_2)), v)$$

Eligibility verifiability

$$R^{EV}_{\bullet} = \phi' \wedge \mathrm{ver}_{4,3+l}(\mathcal{F}, z_1)$$

where

$$\begin{aligned}
\phi &= \mathrm{ver}_{1,2}(\mathcal{F}', z_2) \wedge \mathrm{eq}(\mathrm{public}_1(z_1), \mathrm{public}_2(z_2)) \\
\phi' &= \phi \wedge \mathrm{pet}(y, \mathrm{public}_2(z_1), z_3))
\end{aligned}$$

Case study: Helios/UCL

Work in progress; caused us to generalise R^{UV}, R^{EV} to non-pointwise case.

Individual verifiability

Straightforward.

Universal verifiability

Probably straightforward :-)

Eligibility verifiability

Not satisfied.

Results and trustworthiness requirements

Property	*FOO'92*	*Civitas '08*	*Helios/UCL '09*
Vote-privacy	✓	✓	✓
trusted compnts	client	client	client
Receipt-freeness	×	✓	×
trusted compnts		client	
Coercion resist.	×	✓	×
trusted compnts		client	
Individual verif.	✓	✓	✓
trusted compnts	client	client	client
Universal verif.	✓	✓	✓
trusted compnts			
Elig. verif.	×	✓	×
trusted compnts			

Conclusions and future work

Conclusions

- First *generic* formal definitions of election verifiability.
- Suitable for automation.
- Automatic verification for PostalBallot, FOO, Civitas.

Future work

- Completion of homomorphic cases (Helios/UCL)
- Voting systems that are not client-crypto-based.