ON CRYPTOGRAPHIC PROTOCOLS FOR NORWEGIAN INTERNET VOTING

Helger Lipmaa
Cybernetica AS and Tallinn University, Estonia

WHAT IS CRYPTOGRAPHY?

- (Mathematical) study of secure communication
- Goals: confidentiality, authenticity, …
- History?
  - "Writing was invented to conceal information, not to disseminate it"
- Any communication/computation can be performed in a secure way – in theory
- Split in (huuuuge) community
  - Practicians
  - Theoreticians

ESTONIAN CRYPTOGRAPHY

- WW2 – some Estonians worked in Finland, helped to break Soviet codes
- 1960+ Rein Turn in Rand Corporation
- 1992 – IOC founds department of data security
- 1996 – grant application by Buldas, me, Willemson
- 1998 – first major paper

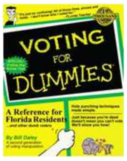
ESTONIAN CRYPTOGRAPHY

- Currently:
  - Dr Prof Buldas, Dr Prof Laud, Dr Prof Lipmaa, Dr Prof XXX, Dr Willemson, Dr Laur, Dr Tsahhirov, Dr Jürgenson, Dr Gonzalez, (Dr Elkind)
  - Most people working at Cybernetica AS + some university
  - Soon to doctor: Niitsoo, Bogdanov, Zhang (Tartu), Käsper (Leuven)
- MSc programs in security (Nordsecmob), cyber defense
- NATO Center of Excellence in Cyber Defense

ESTONIAN CRYPTOGRAPHY

VOTING FOR DUMMIES

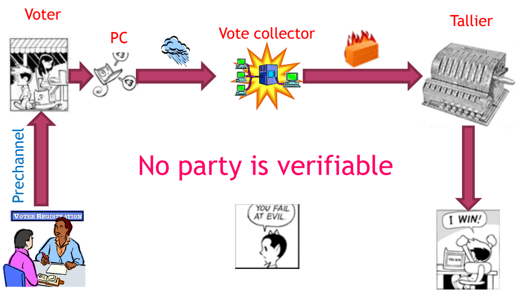
- Voting: one of the cornerstones of democracy
- A number of voters v who vote for a number of candidates c.
- Every candidate v has some preference list $c_{v1} \geq_v c_{v2} \geq_v \dots \geq_v c_{vn}$
- Voting mechanism: allows every voter v to cast some (ordered) list $(c'_1, \dots, c'_m)$ of votes. Given all such lists, computes winners of the election.
- In practice, voting stations/ballot boxes, postal voting, …



E-VOTING FOR DUMMIES

- "Booth voting" has known weaknesses
  - Accessibility
  - Cost
  - Security
- Postal voting
  - Accessibility++
  - Security/cost?
- E-voting / Internet voting
  - Accessibility++
  - Cost++ (?)
  - Security - relies (not only) on cryptography

ESTONIAN INTERNET VOTING: SETTING



NORWEGIAN (E-)VOTING: IN A NUTSHELL

- Universal suffrage:
  - 18+; women's suffrage from 1913
  - Foreigners (3+y) vote in local elections
- Cost/accessibility:
  - Large distances, small population
  - Huge expat community
- Solution: e-voting

NORWEGIAN (E-)VOTING: IN A NUTSHELL

- Organization started in 2009-
- 2011: first local Internet elections (11 municipalities)
- 2017: full parliamentary Internet election

OUR INVOLVEMENT, 1/2

- 2009: tender for organizing Norwegian Internet elections
- Norwegian government: security is paramount
  - Against malicious voting servers:
    - “we don’t want people to blame us”
    - “we want to be able to prove we did not cheat”
  - Against malicious voter PCs

OUR INVOLVEMENT

- Our consortium (Cybernetica AS + 3 more companies) proposed a new setting and cryptographic protocol
  - [HLV10] - Heiberg, Lipmaa, Van Laenen, ESORICS 2010
  - Showed that this setting (code-verification) can be efficiently implemented
- What is used in Norway?
  - Our setting
  - Protocol by Scytl and Kristian Gjøsteen
    - More efficient but less secure than [HLV10]
- New protocol [Lipmaa ’10] (unpublished)
  - As efficient as the Scytl protocol but considerably more secure

THE MEAT OF THIS TALK

- “Code-verification” setting
- Protocols
  - [Heiberg, Lipmaa, Van Laenen ’10]
    - Esorics 2010 and http://eprint.iacr.org/2010/195
  - Scytl protocol
    - http://eprint.iacr.org/2010/380
  - Lipmaa ‘10 protocol
    - under submission

CURIOSITY

- [HLV10] was submitted to Eurocrypt but rejected
- One of the reviewers stated:
  - “This paper is too practical for Eurocrypt, I recommend to resubmit it to ACM CCS/Esorics”

SECURITY CONSIDERATIONS

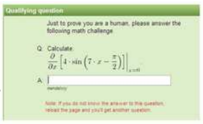
- All parties can be attacked/be malicious
  - Voter PC, Internet, different voting servers, …
- Goal: security against any party
  - Internet: encrypt/sign votes (… DDOS)
  - Voting servers:
    - large amount of cryptographic research since 1981
- Subject of this paper:
  - Security against malicious voter PC
  - Without sacrificing usability

SECURITY GOALS

- Privacy:
  - nobody knows how anybody else voted
- Correctness:
  - All votes are included correctly, and only once
  - Individual verifiability: voter is able to verify his/her vote was counter for
  - Universal verifiability: final tally includes votes of all legit voters exactly once
- Coercion/vote-buying:
  - No forced (or family) voting
  - Impossibility to sell votes
- All important but somewhat contradictory

MALICIOUS PC: PRIVACY?

- Privacy: how?
  - CAPTCHAs
  - Long random codes
    - Code sheets/PIN-calculators
  - IQ tests



- All known methods limit accessibility
- Example: code voting
  - For every c, v obtains two codes. He inputs first to PC, and obtains second back as check code
  - Too complicated for many users
  - Code sheet lost => can't vote

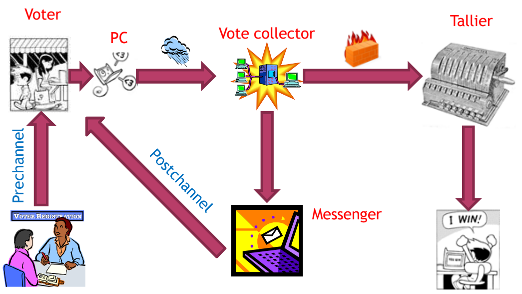
MALICIOUS PC: VERIFIABILITY!

- Verifiability against malicious PC: this presentation
  - NB: accessibility, usability
- "Code-verification" voting:
  - Voters receive check codes, showing that their vote was (in)correctly received by server
  - Voting consists of inputting the name/number of candidate by any preferred GUI
  - If code sheet is lost, can still vote -- but can't verify

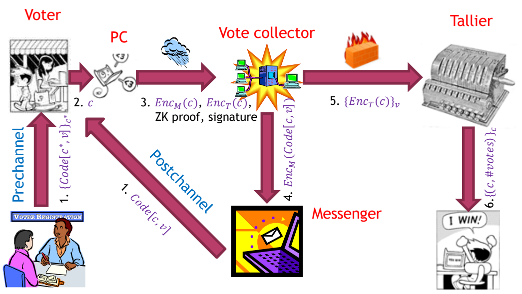
REST OF DESIDERATA

- Privacy against voting servers
  - Except the tally
- Correctness against voting servers
  - Individual verifiability
- Some coercion-resistance
  - Implemented as in Estonia:
    - People can revote several times, lastly on paper
    - Later vote revokes earlier vote
  - Adds complications to protocols

CODE-VERIFICATION VOTING: SETTING



CODE-VERIFICATION VOTING: CRYPTOGRAPHIC PROTOCOL



HEIBERG-LIPMAA-VAN LAENEN PROTOCOL

- Codes $Code[c, v]$ are randomly generated, and also sent to the vote collector (unordered)
- Based on Elgamal encryption
- Non-interactive zero-knowledge proof:
  - $e_1 = Enc_M(c)$ & $e_2 = Enc_T(c)$ & $c \in [0, \#cands - 1]$
  - Requires $\boldsymbol{\Theta}(\log \#cands)$ exponentiations
- VC maps $Enc_M(c) \to Enc_M(Code[c, v])$
  - Without knowing key or c
  - Our solution: proxy oblivious transfer

REMINDER: ELGAMAL ENCRYPTION

- Alice generates a new PK/SK pair $sk \leftarrow Z_q$, $pk \leftarrow g^{sk}$
- Bob encrypts a message $m \in \langle g \rangle$:
  - $r \leftarrow Z_q$, $Enc_{pk}(m; r) := (m \cdot pk^r, g^r)$
- Alice decrypts $Enc_{pk}(m; r) = (c, d)$ as $m' \leftarrow c/d^{sk}$
- Check: $m\, pk^r (g^r)^{-sk} = m \cdot pk^r \cdot pk^{-r} = m$
- Elgamal is
  - very efficient, especially over elliptic curves
  - Standard & well-known (1984, relies on DDH)
  - Available in some Hardware Security Modules

REMINDER: ZERO-KNOWLEDGE

- (Interactive) protocol between prover $P(x, w)$ and verifier $V(x)$ that $x \in L$.
- Correctness: $x \in L$ iff verifier accepts
- ZK: verifier only gets to know that $x \in L$
  - Exists simulator that can reproduce V's view, without knowning the witness
- $\boldsymbol{\Sigma}$-protocol: 3 round protocol with certain properties
- *Non-interactive* ZK proofs constructed from $\boldsymbol{\Sigma}$-protocols by applying Fiat-Shamir heuristic

EXAMPLE: NIZK PROOF

- $P(y, x)$ proves that he knows x such that $y = g^x$ (Schnorr's proof of knowledge of DL)

P: $r \leftarrow Z_q$, $a \leftarrow g^r$ $\xrightarrow{a}$ V

P $\xleftarrow{c}$ V: $c \leftarrow \{0,1\}^k$

P: $z \leftarrow r + cx$ $\xrightarrow{z}$ V: $g^z = ay^c$?

- Fiat-Shamir: $c \leftarrow H(y, a)$, P sends to V (a, c, z) - secure if H is a random oracle

REMINDER: SECURE PROTOCOLS

- Assume $(P_1, V_1, \ldots, P_r, V_r)$ is secure in semihonest model
  - All parties follow protocol but "listen in"
- How to make it secure in malicious model?
  - Parties can do arbitrary stuff
- Generic transformation:
  - With every message, add a ZK proof that this message was computed correctly

HLV10: ZERO-KNOWLEDGE PROOF

- $e_1 = Enc_M(c)$ & $e_2 = Enc_T(c)$ & $c \in [0, \#cands - 1]$
- For $\boldsymbol{\Sigma}$-protocol of $P(x)$ & $Q(x)$ or $P(x) \mid Q(x)$:
  - Construct Σ-protocols for $P(x)$ and $Q(x)$ separately, then use a "standard trick" to conjugate/disjungate
- $\boldsymbol{\Sigma}$-protocol for $e_i = Enc_*(c)$: variant of Schnorr's protocol

HLV10: ZERO-KNOWLEDGE PROOF

- $e_1 = Enc_M(c)\ \&\ e_2 = Enc_T(c)\ \&\ c \in [0, \#cands - 1]$
- $\boldsymbol{\Sigma}$-protocol (range proof) for $e_1 = Enc_M(c)\ \&\ c \in [0, \#cands - 1]$:
  - From [Lipmaa Asokan Niemi 2002]: $c \in [0, H]$ iff $\mathrm{c} = \sum_{i=0}^{\lfloor \log H \rfloor} \lfloor (H + 2^i)/2^{i+1} \rfloor \cdot c_i\ \&\ c_i \in [0,1]$
- Full protocol has complexity $\Theta(\log H)$

PROXY OBLIVIOUS TRANSFER

- Functionality:
  - Chooser has input x, sender has input $f = (f_0, \dots, f_{n-1})$, receiver has no input
  - Receiver obtains f_x
- Privacy:
  - Chooser and sender obtain no information
  - Receiver obtains f_x and nothing more
  - Strong POT: receiver obtains no other information even when knowing $\{f_0, \dots, f_{n-1}\}$
- E-voting:
  - PC = chooser ($x = c$)
  - VC = sender ($f = Code[\cdot, v]$),
  - messenger = receiver obtains $Code[c, v]$

PROXY OBLIVIOUS TRANSFER

- We proposed a new POT protocol with complexity $\Theta(\#cands)$, based on Elgamal
- New POT protocol looks simple…
  - … but it is the single most computationally expensive part of HLV10 e-voting protocol
  - (See paper)

SECURITY OF HLV10 PROTOCOL

- Malicious PC/correctness:
  - Can be verified from integrity check code
- Malicious VC:
  - Privacy guaranteed by protocol (under DDH)
  - Correctness --- can be guaranteed by additional protocol
- Malicious messenger:
  - Privacy guaranteed by protocol (under DDH)
  - Correctness --- by check code

PROS/CONS OF HLV10

- Pros:
  - Provable security
  - (Correctness against VC can be added)
  - Uses standard crypto (DDH)
- Cons:
  - Computational complexity $\Theta(\#cands)$
    - Ok in US presidential elections
    - Bad in Norway (max 80 candidates)

SCYTL PROTOCOL

- Idea: codes $Code[c, v]$ are **pseudorandom**
- Computed as $Code[c, v] = \mathrm{h}(gv^{f(c)})$
  - f is secret function computed by PCs
  - $gv = g^{tv}$ is secret voter-dependent function computed by VC
  - h is secret function computed by messenger
- Messenger and VC share tallier's secret key, $sk_M + sk_{VC} = sk_T$

SCYTL PROTOCOL

- Voter encrypts vote once, $a = Enc_T(g^{f(c)})$
- VC
  - "semidecrypts" $a = (g^{f(c)} g^{sk_T r}, g^r) = (g^{f(c)} g^{(sk_{VC}+sk_M)r}, g^r)$, obtaining $b = Enc_M(g^{f(c)})$
  - computes $b'' = Enc_M\left(gv^{f(c)}\right) = b^{tv}$.
  - Sends it with NIZK proof of correct decryption to messenger
- Messenger decrypts, and computes $Code[c, v] = h(gv^{f(c)})$, and sends it to voter

SCYTL PROTOCOL: PROS

- Very efficient, only a few exponentiations
- Easier to implement than HLV10
- Provably secure against malicious PC (only privacy), VC, messenger

SCYTL PROTOCOL: CONS

- Online servers share secret key of offline server
  - Easier to attack + tallier can be distributed
- Even *without* sharing the key, online servers can together breach voter privacy
- Need setup phase:
  - Codes need to be computed before voting starts by trusted servers who know all secrets of PC, VC and messenger
- Christian Bull, Swiss e-voting workshop 2010:
  - VC & M will be strongly separated (600 km + different organizations + …)

LIPMAA2010 – NEW PROTOCOL

- Desiderata:
  - As efficient as Scytl protocol but more secure
  - VC+M do not share tallier's secret key
  - VC+M coalition is not able to breach voter privacy
- Still has the setup phase ☹

L10 PROTOCOL

{Details omitted from web-published version of the slides}

L10 – SECURITY

- Same as in Scytl protocol
  - Privacy against malicious PC, security against VC, messenger
- In addition:
  - VC+M do not share tallier's key
  - L10/1: VC+M can breach voter privacy (as Scytl)
  - L10/2: VC+M can't breach

EFFICIENCY COMPARISON

Prot.	Voter PC	Vote collector	Messenger	Setup phase?	VC+M Pr. B.
HLV10	$(7g + 10)e + 1s$	$(2G + 6g + 8)e + 1v + 1s$	$Ge + 1v$	No	Yes
Scytl	$3e + 1s$	$\mathbf{8e + 1v + 1s}$	$\mathbf{10e + 1v}$	yes	Yes
L10/1	$12e + 1s$	$\mathbf{9e + 1v + 1s}$	$\mathbf{10e + 2v}$	yes	Yes
L10/2	$16e + 1s$	$17e + 1v + 1s$	$18e + 2v$	yes	No

G is number of candidates, $g = \log G$
In Norway, $G \leq 80$, in US presidential $G \leq 5$

EFFICIENCY COMPARISON

Prot.	Voter PC	Vote collector	Messenger	Setup phase?	VC+M Pr. B.
HLV10	$(7g + 10)e + 1s$	$(2G + 6g + 8)e + 1v + 1s$	$Ge + 1v$	No	Yes
Scytl	$3e + 1s$	$\mathbf{8e + 1v + 1s}$	$\mathbf{10e + 1v}$	yes	Yes
L10/1	$12e + 1s$	$\mathbf{9e + 1v + 1s}$	$\mathbf{10e + 2v}$	yes	Yes
L10/2	$16e + 1s$	$17e + 1v + 1s$	$18e + 2v$	yes	No

50000+ votes per hour

G is num
In Norway,

QUESTIONS?

- Happy elections for Latvian colleagues ☺