Scytl
Innovating Democracy

Return Codes: challenges and new approaches

E-Voting PhD Days, 2013, November 14–15, Münchenwiler, Switzerland

Alex Escala, Sandra Guasch
Alex.escala@scytl.com
Sandra.guasch@scytl.com

Scytl
Innovating Democracy

Index

What is a Return Code?

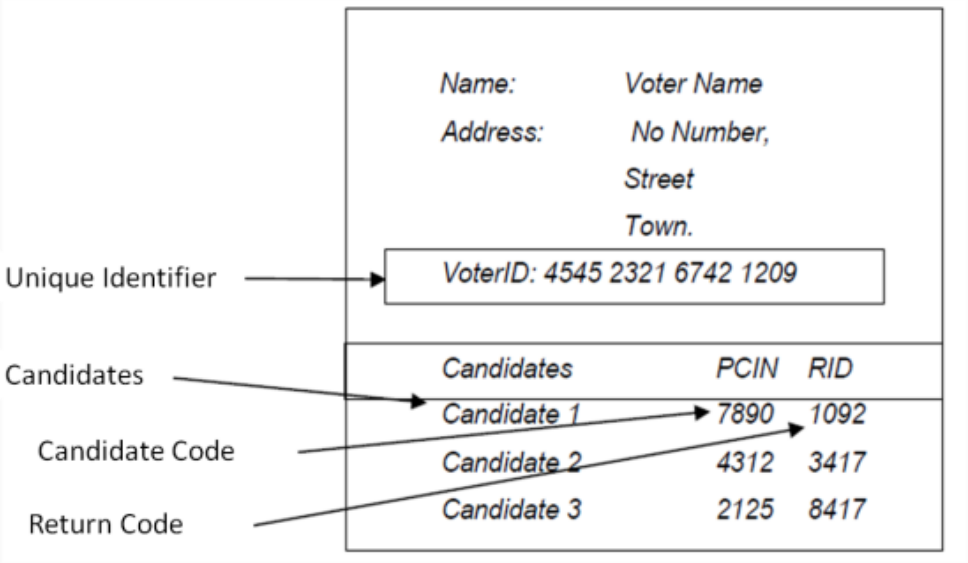
Scytl
Innovating Democracy

What is a Return Code?

Pollsterless or Code Voting

Voter receives a Voting Card with two codes per voting option:

- **Candidate Codes:** used to select a specific candidate.
- **Return Codes:** used to verify that the server properly registered the selection.



Scytl
Innovating Democracy

What is a Return Code?

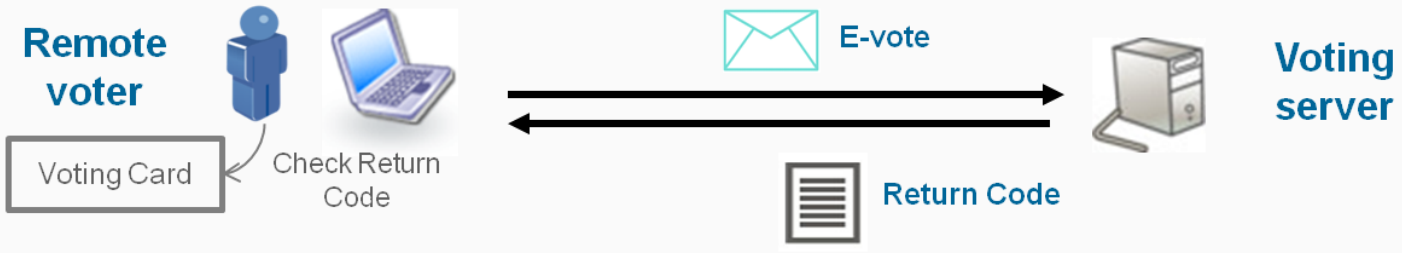
Pollsterless or Code Voting

Cast-as-intended verifiability:

The server generates the Return Codes from the Candidate Codes received.

Only the server can generate such Return Codes.

Voter checks the Return Codes received match those in the Voting Card for the selected options.



Scytl
Innovating Democracy

What is a Return Code?

Norway Scheme

Voting Card with one **Return Code** per voting option.

The voter selects her voting options with a **click-and-select** interface (no Candidate codes).

VALG — Kommunestyre- og fylkestingvalget 2011

Brukernavn 02045698156

Passord: sdppa3

Sikkerhetskoder 5ert – df8y – csd9 – u7lo - jfdb

Parti	Returkode
Disney Party	186479
Marvel Party	657294

Kandidat	Returkode
Donald Duck, f. 1934, Disneyland	453782
Peter Pan, f. 1904, Disneyland	083128
Mickey Mouse, f. 1928, Disneyland	537287
Minnie Mouse, f. 1928, Disneyland	975659
Aurora Princess, f. 1959 , Disneyland	736181

Scytl
Innovating Democracy

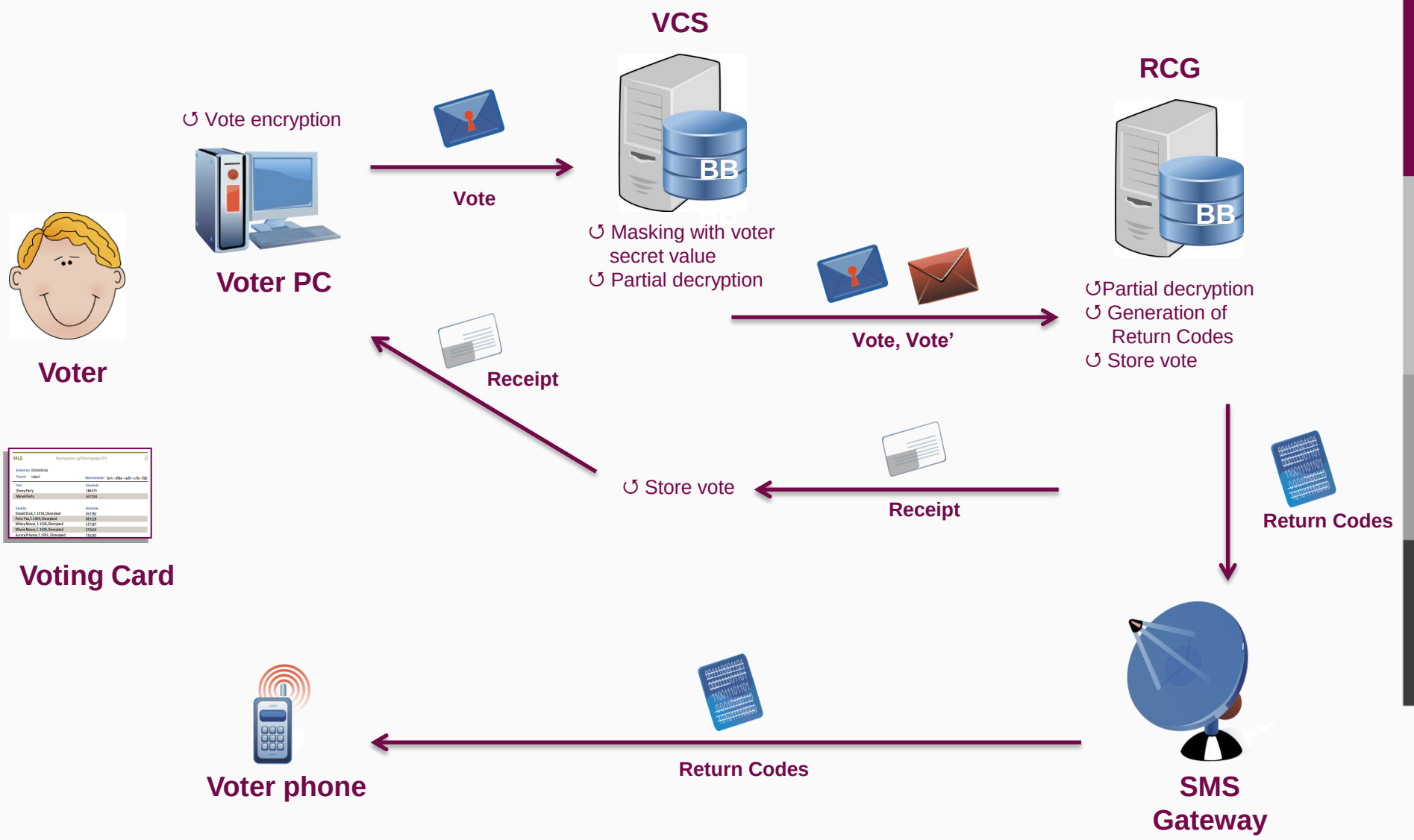
Oops, my return code doesn’t match.
What should I do?

Cast-as-intended in Norway

Scytl
Innovating Democracy

Cast-as-intended in Norway

Overview



Scytl
Innovating Democracy

Cast-as-intended in Norway

Crypto operations

ElGamal encryption scheme:

- Electoral Board public key: $p, g, h_{eb} = g^{x_{eb}} \bmod p$
- Electoral Board private key: x_{eb}
- VCS and RCG private keys: $x_{vcs} + x_{rcg} = x_{eb}$

Voting Client (Voter PC)

- Encryption of the voting options with ElGamal:

$$(g^r, h_{eb}^r \cdot v_i)$$

VCS

- Masking with voter secret value:

$$((g^r)^{s_j}, (h_{eb}^r \cdot v_i)^{s_j})$$

- Partial decryption:

$$(h_{eb}^r \cdot v_i)^{s_j} \cdot (g^{r \cdot s_j})^{-x_{vcs}} = (h_{rcg}^r \cdot v_i)^{s_j}$$

RCG

- Partial decryption:

$$(h_{rcg}^r \cdot v_i)^{s_j} \cdot (g^{r \cdot s_j})^{-x_{rcg}} = v_i^{s_j}$$

- Return Code generation:

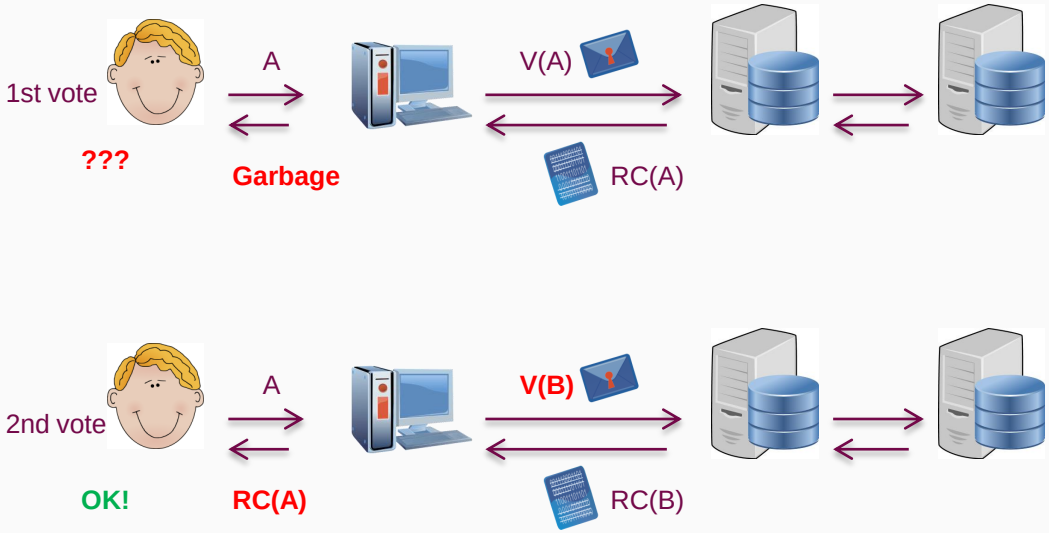
RC → $v_i^{s_j}$

Cast-as-intended in Norway

What do we do if the Return Codes do not match?

- #1: Multiple voting:
  - Voter can cast as many electronic votes as desired, only the last one counts.
  - Voter can cast a preferential vote from a poll-site.
  - Voter can override her electronic votes by voting in paper.

The voting client shall not learn the Return Code values:



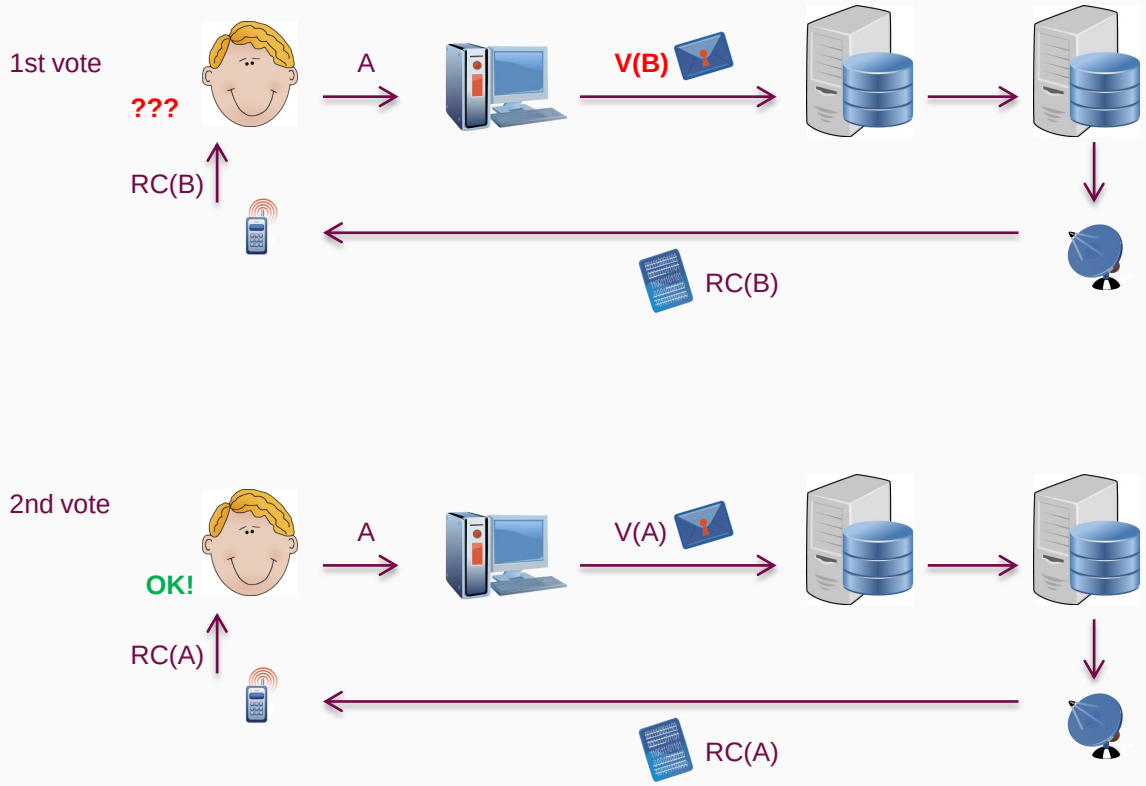
Voter thinks her vote cast for A will count.
However, a vote cast for B will count instead.

Scytl
Innovating Democracy

Cast-as-intended in Norway

What do we do if the Return Codes do not match?

- #2: Independent channel for sending back the Return Codes: SMS.



Voter thinks her vote cast for A will count, and it's true.

Cast-as-intended in Norway

INDEPENDENT?

Reto E. Koenig, Philipp Locher, Rolf Haenni:

A security flaw in the verification code mechanism of the Norwegian Internet Voting System.

- Man in the Browser & SMS-Channel.
- Attack: Man in the Browser informs Man in the SMS Channel to withold a second SMS → silent second vote.

However, in Norway, only one vote can be cast per voter session.

- Tamper also with voter authentication:
  - A TAN is sent to the voter through SMS.
  - The Man in the SMS Channel sends the TAN to the Man in the Browser without the voter detecting it.

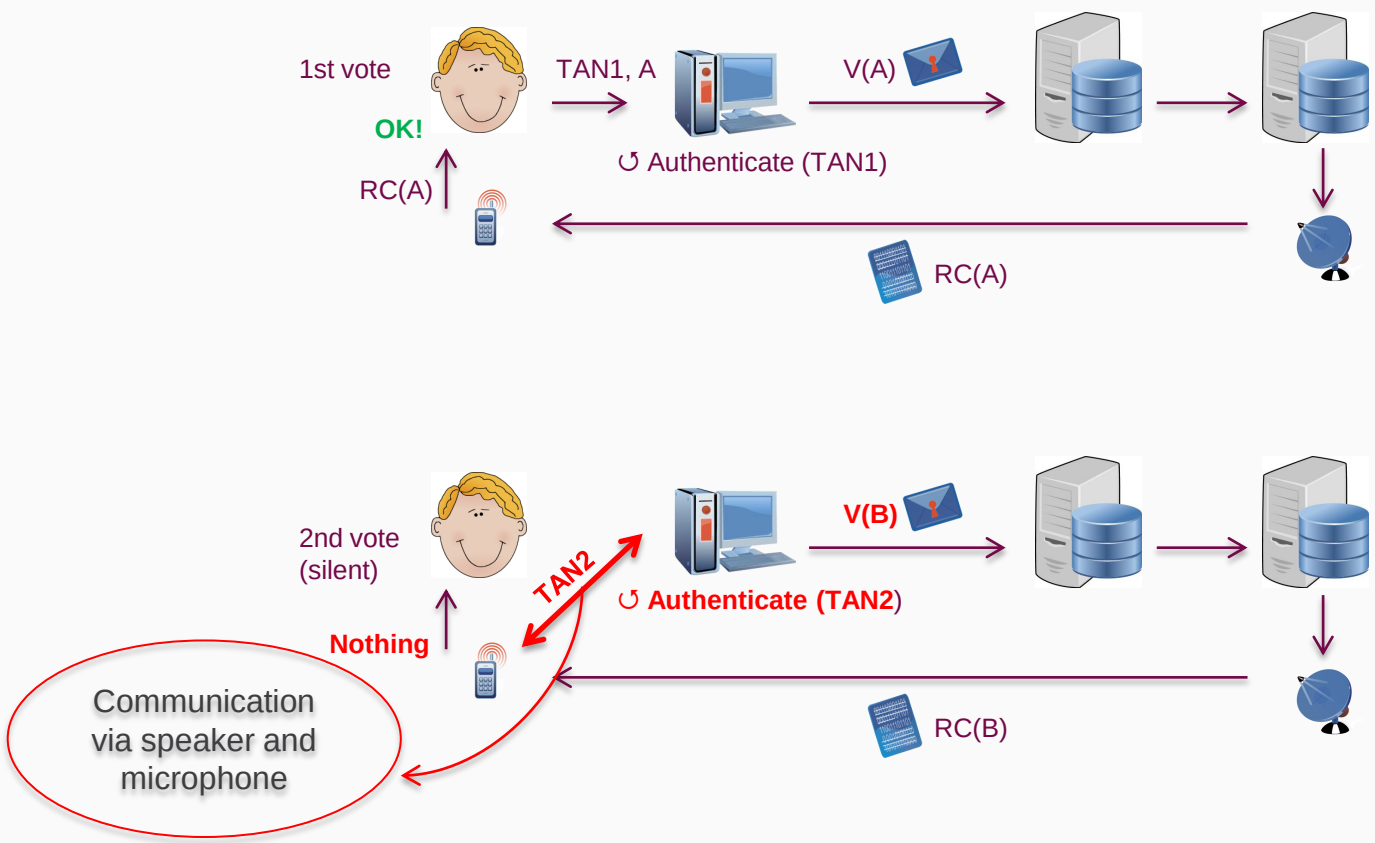
Scytl
Innovating Democracy

Cast-as-intended in Norway

Independent?

A security flaw in the verification code mechanism of the Norwegian Internet Voting System.

Attack example:



Voter does not know that a vote for B has been cast on her behalf. The vote for B will count.

Scytl
Innovating Democracy

Solutions to the attack

Trusted device

- Trusted in the sense of:
  - Providing a secure channel between the server and the voter.
    - Not colliding with third entities.
    - Showing the information as is.

- Functionalities:
  - Receive SMS's
    - So that it is ensured that all the messages sent by the platform are received by the voter.
  - Read encrypted messages shown by the voting client
    - So that a second channel is not needed, since the voting client does not learn the contents of the Return Codes (encrypted).
  - User can enter one-time confirmation values to be sent to the server
    - So that the server knows that the Return Codes have been received and accepted by the voter.

- Examples:

CrontoSign Device:
- Transaction authentication.
- Encrypted data visually transmitted.
- Transaction data displayed in the device.
- Generation of confirmation code.

ZTIC (Zone Trusted Information Channel):
- Smart-card reader for online baking transactions.
- Direct SSL connection with the server.
- Embedded confirm/abort buttons.

NagraID:
- Smart-card with display and keypad.
- Authentication by means of OTP: event-based, time-based, challenge-response.

Scytl
Innovating Democracy

Solutions to the attack

No multiple voting

The voter is only able to cast **one** vote.

What do we do if we receive wrong Return Codes?

→ **Not to accept the vote.**

Let’s see the e-Voting scenario in Switzerland…

E-Voting in Switzerland

If you don’t like your Return Codes, vote in paper!

Scytl
Innovating Democracy

E-Voting in Switzerland

- Highly participative democracy: cantons hold lots of referendums a year, plus canton and federal elections.

- Remote e-Voting official pilots since 2003 (Geneva).
- Initially, 3 cantons started with pilots: Geneva, Zurich, Neuchâtel (Scytl). Now, all of them.
- Systems can only be used by up to 10% of the electorate.

- New e-Voting regulation to be applied from January 1st 2014:

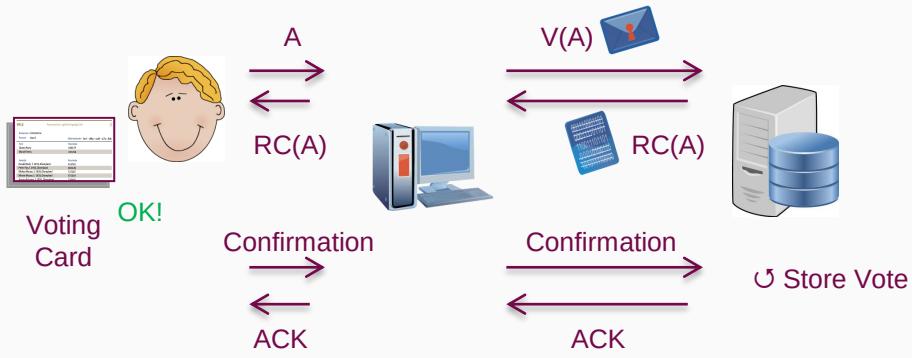
  - Requirements for e-Voting systems to be used by up to the 50% of the electorate.
    - Simplified individual verifiability model:
      - Proof for individual verifiability.
      - Voting client and channel between voting client and server not trusted.
      - Server side trusted.
  - Requirements for e-Voting systems to be used by up to 100% of the electorate.
    - Improved individual verifiability model.
      - Control components check correct server-side functionality. Their collaboration is needed to compute the proof.
      - Bulletin Board so that voters can check their cast vote is in the Ballot Box.
    - Universal verifiability.
      - Verification of the counting process.
      - Publication of proofs in the Bulletin Board.

Scytl
Innovating Democracy

E-Voting in Switzerland

- Simplified individual verifiability model:
  - Proof for individual verifiability (cast-as-intended verification).
  - Voting Client and channel between Voting Client and server not trusted.
  - Server-side trusted.

- Cast-as-intended verification scheme:
  - Server provides a proof of the contents of the vote cast.
  - The proof is shown to the voter through the voting client.
  - **Only one vote per voter can be cast. → Legal requirement.**
  - The vote can be confirmed after the proof has been verified.
  - **If no confirmation, the voter can vote on paper.**



Dynamic generation of Return Codes

Can we have multiple voting back?

Scytl
Innovating Democracy

Dynamic generation of Return Codes

Problems with multiple voting & Return Codes

Can we have multiple voting back?

Let's remember the problem(s):

With the silent-revote attacks:

- The adversary can send arbitrary votes (blocking the Return Codes)
- A confirmation code unknown to the adversary might prevent this

But the adversary can capture the confirmation code during the first vote

- Solution attempt #1: give the voter multiple confirmation codes
- Voting TANs in Reto's words

Scytl
Innovating Democracy

Dynamic generation of Return Codes

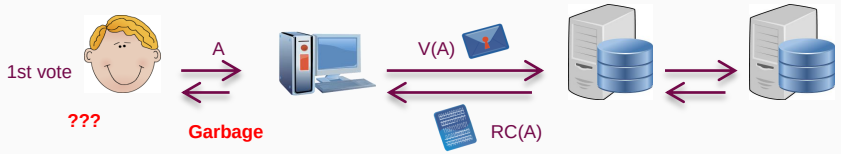
Solution attempt #1

Using Voting TAN might look like a feasible solution

Remember the attack which does not rely on Voting TAN:

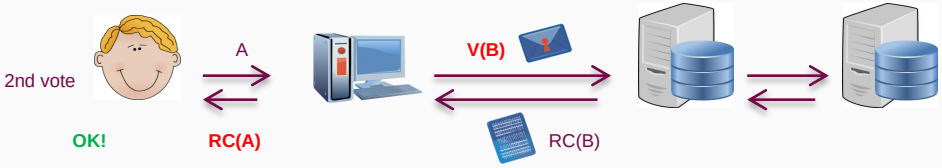
Step 1

- Voter casts vote for *Candidate A*
- Computer receives Return Code for *Candidate A*, but shows *garbage*
- Voter thinks something went wrong, so votes again



Step 2

- Voter casts a vote (again) for *Candidate A*
- Computer knows Return Code for *Candidate A*
- Computer casts a vote for *Candidate B*, but shows Return Code for *Candidate A*



Scytl
Innovating Democracy

Dynamic generation of Return Codes

Solution attempt #2

To defend against attacks, we need:

- Voting TANs
- Different Return Codes for each voting attempt

We could send multiple voting cards to each voter

- costly solution
- limits the number of vote updates (makes coercers happy)

OR

We could have each voter generate his set of voting cards

- The encrypted vote would depend on the generated voting card
- The server would generate a different return code for each voting attempt

How to do that? We don’t know (yet)

Scytl
Innovating Democracy

Dynamic generation of Return Codes

Side-effects

Paradigm-changing point

In the **Norwegian protocol,** voting servers generate Return Codes

- VCS and RCG collaborate to generate each Return Code

In **Solution Attempt #2**, voting client participates as well.

- Voting Client must link the encrypted vote with a generated Return Code card

Downside: we need to get the voter to introduce this linking value

- Can become an usability problem

Can we use the value included in the encrypted vote for other purposes?

Sure! (See next slide).

Getting rid of one (of two) voting servers

Scytl
Innovating Democracy

Getting rid of one (of two) voting servers

Problems with using two servers

In the Norwegian project, VCS and RCG jointly computed the Return Codes

Privacy assumption

The two servers won't collude

- **Heavy assumption**
  - Insider attackers could control both servers
  - Both servers could have similar vulnerabilities
- Having two physical servers is **expensive**
  - Could be tempted to use virtualization on the same machine…

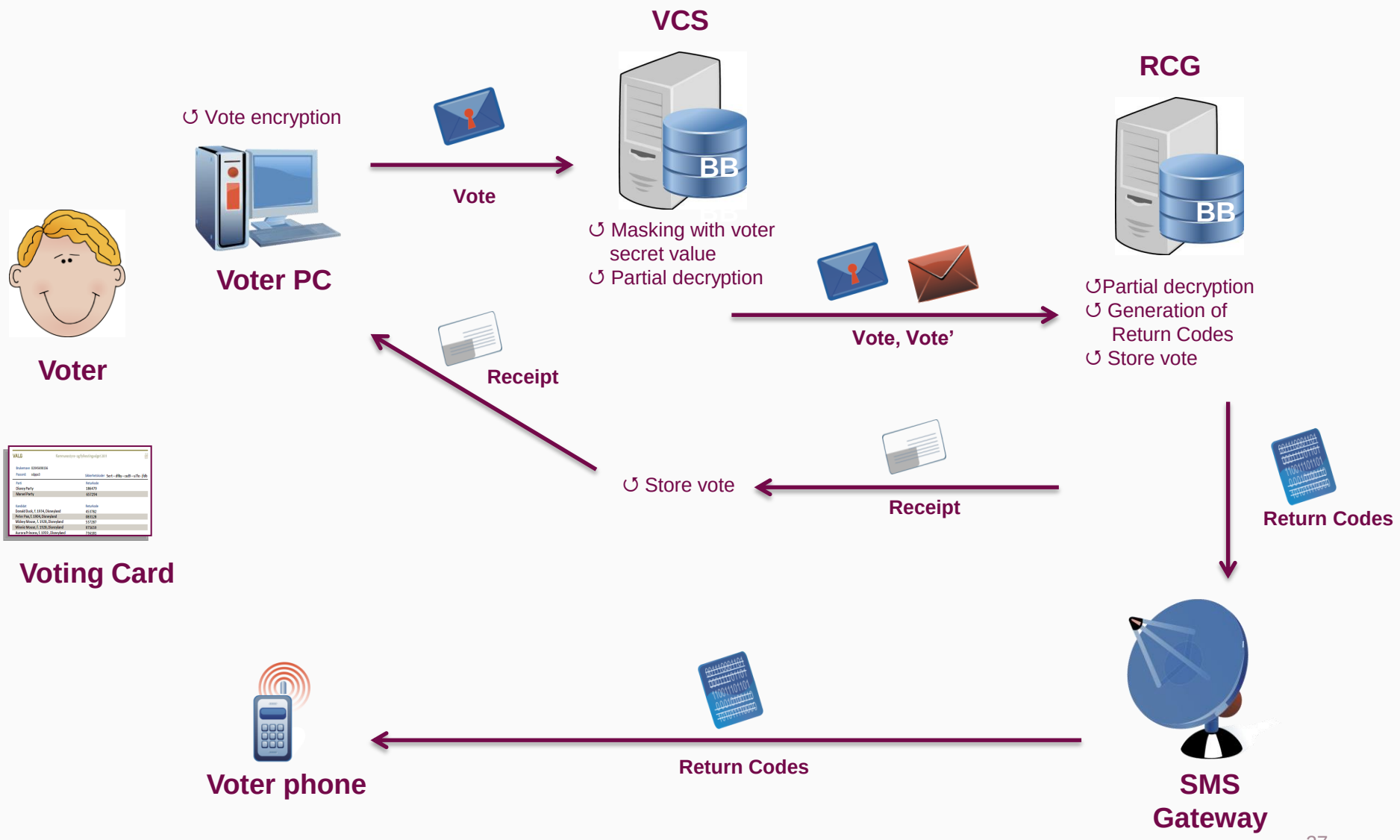
Note: even if the assumption is broken, integrity is guaranteed

We can have the voter act as one of the two servers

Scytl
Innovating Democracy

Getting rid of one (of two) voting servers

Norway's scenario



Scytl
Innovating Democracy

Getting rid of one (of two) voting servers

New scenario

RCG

↺ Vote encryption
↺ Masking with voter secret value
↺ ZK Proof

Vote, Vote'

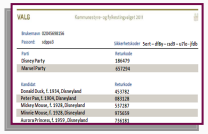
Voter PC

Receipt

↺Partial decryption
↺ Generation of Return Codes
↺ Store vote

Voter

Return Codes



Voting Card

SMS Gateway

Return Codes

Voter phone

Scytl
Innovating Democracy

Getting rid of one (of two) voting servers

Advantages & Disadvantages

By having the Voting Client contribute to the Return Code generation:

- Move some cost from the back-end to the voting client
- Explicitly require the voter to input some secret data to her computer
- (The Norwegians preferred to have increased usability)

Next step: a voting system with one server, no trusted device and multiple voting

- Cheap solution
- Higher security (no heavy assumptions)
- Requires more voter's interaction than other approaches
    - Only if she wants to vote multiple (many) times

Conclusion

Scytl
Innovating Democracy

Conclusion

We have presented different approaches towards Return Codes

- Approaches with trusted devices
- Approaches without trusted devices
    - Without multiple voting
    - With multiple voting
- Approaches with just one voting server

But there is still a lot of work

- Design a system with multiple voting but no trusted device
- Find the minimum trust assumptions for a secure system
- Research audit mechanisms for 100% integrity (even if *everyone* colludes!)

Scytl
Innovating Democracy

Questions?

Innovating Democracy