Bern University of Applied Sciences

Broadcast channel with Memory and Bulletin Board Service

Severin Hauser

E-Voting Seminar, Bern (Switzerland), Jan 24th, 2017

Content

- Broadcast channel with memory
- Bulletin Board Service
- Summary and Outlook

Broadcast channel with memory

Distributed System

- Parties $\Omega = \{p_1, \ldots, p_n\}$
- Channels $\Gamma = \{c_1, \ldots, c_m\}$
- Messagespace $\mathcal{M}$
- Parties in the system provide services

Channel

- Noiseless and unlimited capacity
- Sender domain $S_c \subseteq \Omega$
- Receiver domain $R_c \subseteq \Omega$
- Messagespace $\mathcal{M}_c \subseteq \mathcal{M}$
- Parties outside the receiver domain can see the sending of messages
- Parties outside the System can not see anything

Channel cont.

- Public $S_c = \Omega$
- Broadcast $R_c = \Omega$
- Authentic $S_c = \{s\}$
- Confidential $R_c = \{r\}$
- Closed Group $S_c = R_c$
- Untappable channel $\Omega = \{s, r\}$

Channel with memory

- We introduce two operations
  - $s : \text{Send}_c(m)$
  - $r : \mathbf{M} \leftarrow \text{Receive}_c()$
- List of messages $\mathbf{M} = \langle m_1, \ldots, m_t \rangle$
- Order of the list same as the Send operations performed

Bundled input channel

- Is a non-empty set of channels $C = \{c_1, \ldots, c_r\} \subseteq \Gamma$
- Have a common receiver domain $R_C = R_{c_i}$
- The sender domain is defined as $S_C = \cup_{i=1}^{r} S_{c_i}$
- Message space is $\mathcal{M}_C = \cap_{i=1}^{r} \mathcal{M}_{c_i}$
- Sender $s \in S_C$ selects all channels $c_i \in C$ where $s \in S_{c_i}$ and $m \in \mathcal{M}_{c_i}$

Bundled input channel with memory

- What most voting protocols require
- Provides an total order over all bundled channels
- $(\mathbf{M}, \mathbf{S}) \leftarrow \text{Receive}_C()$
- $\mathbf{M} = \langle m_1, \ldots, m_t \rangle$ and $\mathbf{S} = \langle S_1, \ldots, S_t \rangle$
- $S_i = \cap_{c \in C_{s_i}} S_c$

Bern University
of Applied Sciences

Bulletin Board Service

Why?

- There exists no BCM in reality
- Substitution with similar guarantees
- Service because in general it is provided by multiple parties.

Goals

- Authentication
- Non-discrimination
- Message ordering
- Message well-formedness
- Uniqueness
- δ completeness

Authentication

- Only $s \in S_c$ is allowed to send messages
- Needs to be transferable

Non-discrimination

- A channel c provides every party in S_c or R_c equal access to the operations Send and Receive
- Can only be solved with a trust assumption
  - The entry point party p is non-discriminating
  - At least t entry point parties of n are non-discriminating
- Can be defused by hybrid voting systems for Send

Message ordering

- A channel with memory has $\mathbf{M} = \langle m_1, \ldots, m_t \rangle$
- The BBS needs to provide a total order
- Hash-chains not mandatory

Message well-formedness

- The channel accepts only messages in its message space $\mathcal{M}_c$
- The BBS must provide some message validation
- In a voting protocol it has only to accept messages defined by the protocol

Uniqueness

- The channel keeps exactly one **M**
- The BBS must provide a similar guarantee

δ completeness

- The channel always provides the complete **M**
- On the channel the transmission is instantaneous
- For the BBS there is some time δ needed

How to provide U & δ-C

- Can only be achieved by trust or comparing
- Spread the trust to a group in the BBS
- Clients create a peer2peer network and compare

peer2peer

- Requires high availability of the clients
- Better suited for protocols that run all the time

Group of identical parties

- Every party has the same role
- Every party stores and updates **M**
- Pros:
  - Provides threshold for all goals
  - Well understood
- Cons:
  - High synchronization cost, big δ
  - Not a specific solution for one goal

Our Proposal

- Different roles
- Provide solution per goal
- Pros:
  - More flexible
  - In general better performance
- Cons:
  - More room for errors when building the BBS

Controllers

- A group of parties
- Guarantee uniqueness
- Check and sign the order created by the BB
- Attached / Detached

Providing completeness

- A group similar to the controllers sign queries with timestamp
- Know the current **M**
- Could also be controllers

Providing completeness

- Publish at a fixed interval i a message
- δ can be at max i

Distributors

- Help providing uniqueness and completeness
- Spread the latest messages the BBS shown them
- Similar to the peer2peer

Summary and Outlook

Questions?

http://e-voting.bfh.ch