Towards a Practical Mobile Application for Election Authorities

Stephan Neumann, Oksana Kulyk, Lulzim Murati, Melanie Volkamer

Technische Universität Darmstadt, Germany
name.surname@cased.de

1 Motivation

The most crucial security criteria of voting schemes are secrecy and integrity of the votes. In order to ensure an adequate level of security, cryptographic voting schemes generally distribute trust among several election authorities. Independent of the concrete Internet voting instantiation (shuffling based, homomorphic tallying), cryptographic key distribution and verifiable distributed decryption is of central importance to most of these schemes. Requiring several election authorities to collaborate in order to decrypt votes bears, however, the risk of robustness issues because failures of single authorities might prevent the system from calculating the election result. Consequently, when running elections, an adequate trade-off between secrecy and robustness must be achieved.

Consider the following examples of real-world elections: During the Internet voting that was run in parallel to 2003 parliamentary elections in Catalonia, Spain, a secret key was centrally generated and distributed among n (in their case n was equal to 7) electoral board members [5]. During the university presidential election at Université catholique de Louvain [3], which used the Helios Internet voting scheme version 2.0 (the original Helios 1.0 was invented in [2]), a key was generated in a fully distributed manner. Afterwards, in order to improve robustness, a copy of each individual secret key was generated such that each secret key was in possession of two authorities. In the key generation scheme used in the 2011 elections in Norway [6], the election key was generated distributively between two machines and the two secret key shares were further distributed onto the smart cards of election authorities via a (t, n) threshold sharing scheme. While these examples seem reasonable for the election organizers, with respect to secrecy and robustness trade-offs, significant shortcomings can be identified. Either trade-offs are not adequately considered, e.g., because of high organizational effort, or trade-offs are *not optimal*, i.e., improvements and decreases of secrecy and robustness respectively are not proportional. The former case corresponds to the Norwegian election where secrecy could be improved by running further generating machines. This however results in a significant organizational effort as election authorities would have to obtain (via their smart cards) key shares from *all* machines. The latter case is seen at the KU Leuven election, where through copying robustness is slightly increased by 1 to $(2, 2n)$ (2 out of $2n$ authorities might collaborate in order to violate robustness) rather than $(1, n)$ without copying, while secrecy drops down by one half to $(n, 2n)$ rather than (n, n) without copying.

On the other side, scientific works providing *optimal trade-offs* with respect to secrecy and robustness have been invented, e.g., [4]. In spite of their implementation, e.g. [1], their usage is not yet widely seen in practice. One reason for this lack is that implementations are generally optimized for desktop machines, rely on wireless LAN connections to establish distributed keys, are generally integrated into specific voting schemes, and in summary are not tailored towards modular, spontaneous, and practical use.

2 A Smartphone Application as a Solution for Election Authorities

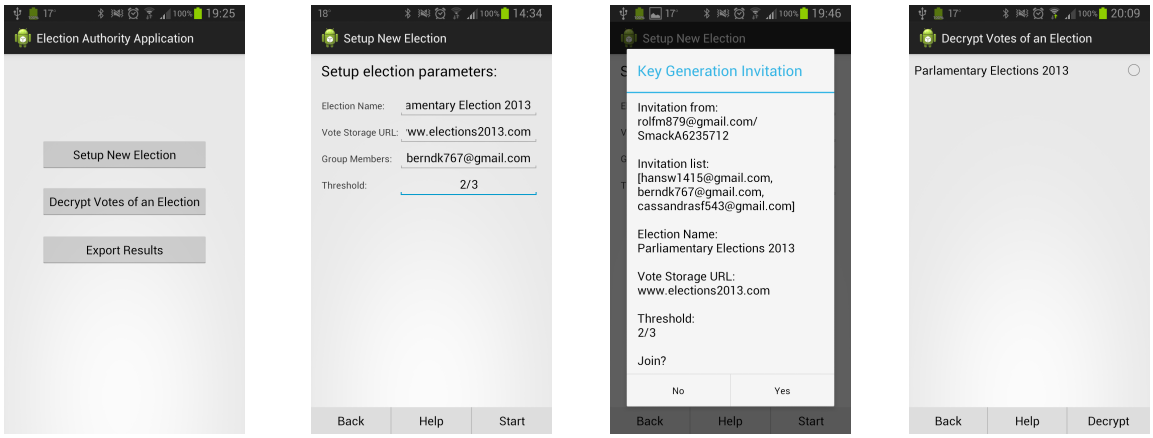
Motivated by these insights and the increasing spread of smartphones, our goal is to develop a practical and usable modular smartphone application providing an optimal secrecy-robustness trade-off. The application we are developing runs on Android OS, which is open for developers and enjoys wide-spread use on modern smartphones. The XMPP protocol, which is one of the standard communication protocols, is used for communication between the users, and messages are stored in XML format, which is also a standard in saving data. As XMPP enables creation of unique message formats, we made use of this feature as well. Messages are signed with RSA keys, while the underlying PKI (at this point) is hard-coded. Currently, our application makes use of Gmail accounts for user authentication as Android users generally have one by default. The two important components of our scheme are the robust distributed key generation and robust distributed decryption.

Key Generation and Decryption. Our application builds upon Gennaro et al.'s distributed key generation scheme [4]. The scheme we implement for the decryption is described in [1]. The commitments,

This work has been presented at the voting system demonstration session of the The 4th International Conference on e-Voting and Identity (VoteID 2013, http://www.voteid13.org/)

computed in the key distribution, are used in order to verify the partial decryptions, which ultimately ensures *universal verifiability* of the decrypted votes. After the partial decryptions are exchanged among the authorities, each authority can verify them and reconstruct the decryption of the votes.

User Process. After starting the application (see Figure 1(a)), the authority has the possibility to setup a new election, initiate the decryption of a running election, or export the results and correctness proofs of a tallied election. In order to setup an election, the initiating authority fixes the election parameters (see Figure 1(b)) by providing an election name, the URL where encrypted votes are stored, nominating other election authorities, and fixing the threshold value. Afterwards, nominated authorities receive an invitation indicating the initiating authority and other election parameters fixed by the initiator (see Figure 1(c)). After the key has been distributively generated between the authorities, the election is stored as running election. After the voting phase has finished, any authority can initiate the tallying process of this election (see Figure 1(d)) which starts the distributed decryption among involved election authorities. Eventually, election results and their integrity proofs can be exported.



(a) Main Menu. (b) Parameter Choice. (c) Invitation. (d) Running Elections

Fig. 1. The Smartphone Application for Election Authorities.

Status and Future Work. To date, the developed prototype application represents a first step in the right direction. Nevertheless, several challenges are to be addressed in the future: The application is not limited to a specific election type, though in some cases additional legal criteria should be considered. Since the application is intended for election authorities and not for voters, the aspects of accessibility so far were beyond our scope, but we plan to consider them in future. We currently investigate how the application can be integrated within established PKIs of states or companies. Furthermore, we are currently supported by usability experts and election authorities in making the application fit the expectations of end-users. In the future, we plan to improve the application's performance by integrating techniques like batch proofs and optimized user synchronization. For further information, do not hesitate to contact one of the authors.

Acknowledgment. This work has been partially developed within the project 'ModIWa2' - Juristisch-informatische Modellierung von Internetwahlen - which is funded by the Deutsche Forschungsgemeinschaft (DFG, German Science Foundation) and partially within the project 'BoRoVo' - Board Room Voting - which is funded by the German Federal Ministry of Education and Research (BMBF).

References

1. Adam M. Davis, D. Chmelev, and M. R. Clarkson. Civitas: Implementation of a Threshold Cryptosystem, 2008.
2. B. Adida. Helios: web-based open-audit voting. In *USENIX Security Symposium*, pages 335–348, 2008.
3. B. Adida, O. Pereira, O. D. Marneffe, and J. jacques Quisquater. Electing a University President using Open-Audit Voting: Analysis of real-world use of Helios. 2009.
4. R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin. Secure Distributed Key Generation. 2007.
5. A. Riera and G. Cervelló. Experimentation on Secure Internet Voting in Spain. *Electronic Voting in Europe Technology, Law, Politics and Society*, pages 91–100, 2004.

6. O. Spycher, M. Volkamer, and R. Koenig. Transparency and Technical Measures to Establish Trust in Norwegian Internet Voting. *E-Voting and Identity*, pages 19–35, 2012.