How not to VoteAgain: Pitfalls of Scalable Coercion-Resistant E-Voting

Johannes Müller[1] [0000−0003−2134−3099]

[1] SnT, University of Luxembourg, Luxembourg

Abstract. Designing secure e-voting systems is notoriously hard, and this is even more the case when coercion-resistance comes into play. Recently, Lueks, Querejeta-Azurmendi, and Troncoso proposed VoteAgain (Usenix Security 2020) which aims to provide coercion-resistance for real practical elections where usability and efficiency are particularly important. To this end, VoteAgain is based on the re-voting paradigm to protect voters against coercion, and it employs a novel tallying mechanism with quasilinear complexity to achieve high efficiency.

In this paper, we revisit VoteAgain from a security perspective. We show that for each security property, i.e., ballot privacy, verifiability, and coercion-resistance, there exists (at least) one attack which breaks the respective property under the trust assumptions for which the property was claimed to hold true. But our results are even more disillusioning: first, there exists a voting authority in VoteAgain which needs to be trusted for all security properties; second, all voting authorities in VoteAgain need to be trusted for coercion-resistance.

It will be interesting and challenging future work to mitigate, or even remove, these undesirably strong trust assumptions without affecting the usability and superior efficiency of VoteAgain.

1 Pitfalls

We show that VoteAgain [8] is not secure under the trust assumptions for which it was claimed to be. More precisely, for each security property, i.e., ballot privacy, verifiability, and coercion-resistance, we describe (at least) one attack which breaks the respective property under the trust assumptions for which the property was claimed to hold true.

In particular, our results will demonstrate that there exists a voting authority in VoteAgain (namely, the PA) which needs to be trusted for all security properties, and that all voting authorities in VoteAgain need to be trusted for coercion-resistance.

Remark. The current version of this report is not self-contained.[1] We assume that the reader is familiar with the details of VoteAgain, as presented in [8]. We will use the notation of [8] for variables, protocol participants, references, and so on.

[1] We intend to publish an extended version with full technical details of the report soon.

1.1 Verifiability

It is claimed that VoteAgain provides verifiability if the PA is honest, while the TS, PBB, and the trustees can be malicious. Similarly to an attack on Belenios [3] that was recently presented in [6], we will now describe an attack of a malicious PBB which breaks verifiability of VoteAgain.

The verifiability definition [2] that was applied to analyze VoteAgain in [8] requires that an adversary cannot drop choices of voters who verify successfully without being detected. Due to the attack described below, VoteAgain is not verifiable according to [2] if PBB can be malicious.

Malicious PBB. Let V be an arbitrary voter. In Step 4 of Procedure 3, the PBB shows a faked view on the bulletin board to voter V which includes V's ballot β. Then, the voter verifies successfully that β was appended to the "faked" bulletin board. However, the PBB does not append β to the "real" bulletin board. Effectively, V's choice is dropped even though V's individual verification was successful.

1.2 Ballot privacy

It is claimed that VoteAgain provides ballot privacy if k out of t trustees are honest, while the PA, TS, and PBB can be malicious. We will now describe two attacks which break ballot privacy of VoteAgain. The first attack assumes a malicious PA, and the second one a malicious PBB.

The privacy definition [1] that was applied to analyze VoteAgain in [8] (essentially) requires that an adversary, who is asked to first output two vectors of choices that equal modulo permutation, cannot efficiently distinguish between those protocol runs in which all honest voters vote according to either the first or the second vector. Due to the attacks described below, VoteAgain does not provide ballot privacy according to [1] if the PA or the PBB can be malicious.

Malicious PA. Let $V_1, \ldots, V_N$ be the voters. In Step 2 of Procedure 1, the PA assign the same identifier *vid* to each voter $V_2, \ldots, V_N$. However, it is never verified whether two or more voters use the same identifier. Therefore, the final election result consists of V_1's vote plus one vote of *one* of the voters $V_2, \ldots, V_N$.

Now, in the privacy game, the adversary chooses vectors $(A, B, B, \ldots, B)$ and $(B, A, B, \ldots, B)$ for $V_1, \ldots, V_N$. Then the adversary returns that the first vector was chosen if and only if the final result contains a vote for A. By this, the adversary wins the game, and hence breaks ballot privacy, with high probability.

Malicious PBB. It is easy to see that by the attack on verifiability described above, the adversary can break privacy. We refer to [4] for more details.

1.3 Coercion-resistance

It is claimed that VoteAgain provides coercion-resistance if the PA and TS are honest, while the PBB (if voters submit anonymously) and the trustees can be malicious. We will now describe two attacks which break coercion-resistance of VoteAgain. The first attack assumes a malicious PBB, and the second one malicious trustees.

Malicious PBB. (Assume that voters use anonymous channels to submit their ballots.) The adversary chooses a candidate c and instructs an arbitrary voter V to submit a ballot for this candidate. Furthermore, the adversary asks the voter to reveal the submitted ballot β (including all secret information). The malicious PBB can now identify the incoming ballot β, append it, and drop all subsequently incoming ballots by any voter (see attack on verifiability). By this, the affected voter can no longer "overwrite" the adversary's choice c even if the adversary is absent for the rest of submission phase.

Malicious trustees. The adversary chooses a sequence $(c_j)_{j=1}^{l}$ over the set of candidates $\mathcal{C}$ uniformly at random. The adversary instructs (and supervises) a targeted voter V to first submit a ballot β_j for each element c_j of this sequence (preserving the order of the sequence) and then a ballot β for the adversary's favorite candidate c. Since the trustees are malicious, the adversary can decrypt all $v_{i,\star}$ for each vid_i in the grouped ballots (see Figure 6 in [8]). The adversary (removes all 0-votes injected by TA and) verifies whether there is vid_i which contains the chosen sequence of candidates. If this is the case, the adversary knows (with high probability if l is sufficiently high) that the voter obeyed.

2 Conclusion

Our attacks show that there exists a single voting authority in VoteAgain [8], namely the PA, which needs to be trusted for *all* security properties, i.e., ballot privacy, verifiability, and coercion-resistance.[2] We further demonstrated that *all* voting authorities need to be trusted for coercion-resistance. We can therefore conclude that VoteAgain is not secure under reasonable trust assumptions.

It will be interesting and challenging future work to mitigate, or even remove, these undesirably strong trust assumptions without affecting the superior efficiency of VoteAgain.

Acknowledgements

The author acknowledges support from the Luxembourg National Research Fund (FNR) and the Research Council of Norway for the joint INTER project SURCVS (Number 11747298).

[2] The PBB needs to be trusted for all security properties as well but this assumption can be mitigated by techniques independent of the specific e-voting protocol (see, e.g., [5–7]).

Bibliography

[1] David Bernhard, Véronique Cortier, David Galindo, Olivier Pereira, and Bogdan Warinschi. SoK: A Comprehensive Analysis of Game-Based Ballot Privacy Definitions. In *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17-21, 2015*, pages 499–516, 2015.

[2] Véronique Cortier, David Galindo, Stéphane Glondu, and Malika Izabachène. Election Verifiability for Helios under Weaker Trust Assumptions. In Miroslaw Kutylowski and Jaideep Vaidya, editors, *Computer Security - ESORICS 2014 - 19th European Symposium on Research in Computer Security, Wroclaw, Poland, September 7-11, 2014. Proceedings, Part II*, volume 8713 of *Lecture Notes in Computer Science*, pages 327–344. Springer, 2014.

[3] Véronique Cortier, Pierrick Gaudry, and Stéphane Glondu. Belenios: A Simple Private and Verifiable Electronic Voting System. In Joshua D. Guttman, Carl E. Landwehr, José Meseguer, and Dusko Pavlovic, editors, *Foundations of Security, Protocols, and Equational Reasoning - Essays Dedicated to Catherine A. Meadows*, volume 11565 of *Lecture Notes in Computer Science*, pages 214–238. Springer, 2019.

[4] Véronique Cortier and Joseph Lallemand. Voting: You Can't Have Privacy without Individual Verifiability. In *ACM Conference on Computer and Communications Security*, pages 53–66. ACM, 2018.

[5] Chris Culnane and Steve A. Schneider. A Peered Bulletin Board for Robust Use in Verifiable Voting Systems. In *IEEE 27th Computer Security Foundations Symposium, CSF 2014, Vienna, Austria, 19-22 July, 2014*, pages 169–183. IEEE, 2014.

[6] Lucca Hirschi, Lara Schmid, and David A. Basin. Fixing the Achilles Heel of E-Voting: The Bulletin Board. *IACR Cryptol. ePrint Arch.*, 2020:109, 2020.

[7] Aggelos Kiayias, Annabell Kuldmaa, Helger Lipmaa, Janno Siim, and Thomas Zacharias. On the Security Properties of e-Voting Bulletin Boards. In *Security and Cryptography for Networks - 11th International Conference, SCN 2018, Proceedings*, pages 505–523, 2018.

[8] Wouter Lueks, Iñigo Querejeta-Azurmendi, and Carmela Troncoso. VoteAgain: A Scalable Coercion-Resistant Voting System. In Srdjan Capkun and Franziska Roesner, editors, *29th USENIX Security Symposium, USENIX Security 2020, August 12-14, 2020*, pages 1553–1570. USENIX Association, 2020.