Annotated biography

Free & fair elections

Ben Smyth

December 30, 2024

Foundational ideas are explored in tutorials:

[Smy18] Ben Smyth. A foundation for secret, verifiable elections. Technical Report 2018/225, Cryptology ePrint Archive, 2018

[QS18] Elizabeth A. Quaglia and Ben Smyth. A short introduction to secrecy and verifiability for elections. Technical Report 1702.03168, arXiv, 2018

Theoretical foundations appear in technical publications:

Secrecy for freedom: Indistinguishability games

You're the cool kid. Bravado grants freedom of expression. Elsewhere there's fear. The game is the game. You ain't the cool kid in every room; some opulent power observing, social constraints presiding. Freedom to express one's conscience, morals, will, without fear, a century-old innovation: Eighteen-hundreds closed with realisation, we can't vote freely in public. Secret ballots are better. Instead of declaring preferences in public, mark ballots privately. Tally votes. Reveal collective decisions in public, keep individual votes private.

Vendors claim their systems ensure secrecy of ballots. Hackers devise breaks. Designers argue, "that's not what I meant by security." Formalising security notions enables verification of vendor claims. "[Without] precise definitions, security claims would be a moving target for analysts," explain Koblitz & Menezes.

Cryptographic games are a formalisation tool: A malicious adversary engages in a series of interactions with a benign challenger operating a cryptographic scheme. The adversary wins orchestrating an execution that breaks security. For example, voting system soundness is expressed as the inability of an adversary to dupe voters into accepting the wrong result. Secrecy can be expressed as indistinguishability between any election and any other election with the same outcome.

I formalise ballot secrecy as a game tasking the adversary to distinguish between an instance of a voting system with voters casting some votes, from another instance with voters casting a permutation of those votes:

[Smy21] Ben Smyth. Ballot secrecy: Security definition, sufficient conditions, and analysis of Helios. *Journal of Computer Security*, 29(6):551–611, 2021

In my game, the adversary supplies two lists of votes. The challenger constructs ballots for votes in one of those lists, which list being decided by a coin flip. The adversary casting a set of challenger- and adversary-constructed ballots. And the challenger tallying that set to derive an election outcome, along with any evidence of correct tallying. The game ends with the adversary being tasked to (non-trivially) determine the result of the coin flip from the election outcome and evidence, when votes cast from each list are permutations of each other.

I also provide a blueprint for construction of voting systems from asymmetric encryption schemes, and prove that the blueprint produces voting systems delivering on secrecy needs, when the underlying asymmetric encryption scheme is perfectly correct, non-malleable, and ill-formed ciphertexts are distinguishable from well-formed ciphertexts. I construct one such secure-by-design voting system using a non-malleable derivative of ElGamal built by myself and Toshiba colleagues:

[SH19] Ben Smyth and Yoshikazu Hanatani. Non-malleable encryption with proofs of plaintext knowledge and applications to voting. *International Journal of Security and Networks*, 14(4):191–204, 2019

[SHM15] Ben Smyth, Yoshikazu Hanatani, and Hirofumi Muratani. NM-CPA secure encryption with proofs of plaintext knowledge. In *IWSEC'15: 10th International Workshop on Security*, volume 9241 of *LNCS*. Springer, 2015

I claimed non-malleability wasn't strictly necessary [Smy21], I was wrong:

[Smy24] Ben Smyth. Ballot secrecy: Security definition, sufficient conditions, and analysis of Helios. Technical Report 2015/942, Cryptology ePrint Archive, 2024

Should secrecy be preserved when ballots are suppressed? *Bien sûr*! It follows that non-malleable ballots are necessary: Tallying a meaningfully related ballot suffices to violate secrecy. In the extreme, suppose an adversary observes your ballot, suppresses all ballots including yours, and casts a ballot related to yours. You're led to believe your ballot wasn't successfully cast, yet the adversary can establish your vote.

Stronger privacy notions for voting. Ballot secrecy requires voters to protect their privacy: Voters must follow the prescribed voting procedure. Freedom may be sacrificed, votes may be sold. With Ashley and Liz, we surveyed stronger formulations of privacy that demand voters be unable to convince adversaries of their votes, even when they reveal secrets generated during the voting procedure.

[FQS19] Ashley Fraser, Elizabeth A. Quaglia, and Ben Smyth. A critique of game-based definitions of receipt-freeness for voting. In *ProvSec'19: 13th International Conference on Provable and Practical Security*, volume 11821 of *LNCS*, pages 189–205. Springer, 2019

Coercion resistance goes further: Regardless of what a voter does, they cannot reveal how they voted, or whether they abstained, even when they follow an adversary's instructions. Seminal work by Juels, Catalano & Jakobsson proposed a coercion resistance voting system, albeit complexity is quadratic in the number of cast ballots. I achieve linear complexity with Athena:

[Smy19] Ben Smyth. Athena: A verifiable, coercion-resistant voting system with linear complexity. Technical Report 2019/761, Cryptology ePrint Archive, 2019

I proved verifiability, and wanted to prove coercion resistance too, initiating a survey of definitions.

[HS20] Thomas Haines and Ben Smyth. Surveying definitions of coercion resistance. Technical Report 2019/822, Cryptology ePrint Archive, 2020

We found all definitions unsuitable: One is unsatisfiable, two label systems coercion resistance when they aren't, another is unsuitable for analysis of Athena.

Verifiability for legitimacy: Reachability games

Public forums create legitimacy. Everyone can count raised hands. Secrecy creates a quandary: Decisions must reflect stakeholder will. In public, stakeholders vote openly. Legitimacy is undeniable; anyone present can determine the outcome. But in public, you aren't free. Conversely, private decisions protect freedom, but lack legitimacy, there's no visible proof. Everybody lies. Don't trust. Verify. Decisions must demonstrably reflect stakeholder will. There must be an absolute path from decision to stakeholder votes, whilst protecting voter privacy.

I first studied verifiability using mathematical logic:

[SRKK10] Ben Smyth, Mark D. Ryan, Steve Kremer, and Mounira Kourjieh. Towards automatic analysis of election verifiability properties. In *ARSPA-WITS'10: Joint Workshop on Automated Reasoning for Security Protocol Analysis and Issues in the Theory of Security*, volume 6186 of *LNCS*, pages 165–182. Springer, 2010

[KRS10] Steve Kremer, Mark D. Ryan, and Ben Smyth. Election verifiability in electronic voting protocols. In *ESORICS'10: 15th European Symposium on Research in Computer Security*, volume 6345 of *LNCS*, pages 389–404. Springer, 2010

Later I surveyed computational definitions:

[Smy20b] Ben Smyth. Surveying global verifiability. *Information Processing Letters*, 163, 2020

[SC22] Ben Smyth and Michael R. Clarkson. Surveying definitions of election verifiability. *Information Processing Letters*, 177, 2022

We identify two verifiability principles: Verification must only accept tallies corresponding to votes expressed in collected ballots (soundness), and verification must always accept tallies corresponding to votes expressed in collected ballots (completeness). These principles aren't captured by existing definitions, which are satisfied by voting systems vulnerable to attack, fuelling exploration for a new definition. With Michael & Steven, I provide one:

[SFC21] Ben Smyth, Steven Frink, and Michael R. Clarkson. Election Verifiability: Cryptographic Definitions and an Analysis of Helios, Helios-C, and JCJ. Technical Report 2015/233, Cryptology ePrint Archive, 2021

We require cast ballots be published, allowing each voter to check the presence of their ballot (individual verifiability). We also require tallies be coupled with proofs, allowing anyone to check whether a tally represents votes expressed in collected ballots (universal verifiability), when aforementioned soundness and completeness principles are achieved. Taken together, these facets enable voters to check whether tallying includes the vote expressed by their ballot.

We mere mortals, even the most studious, can't compute digital ballots. Voters are at the mercy of machines, which mightn't even compute ballots, let alone correctly compute ballots expressing voters' votes. Individual- and universal-verifiability only suffice for verifiable voting systems when voters can compute their own ballots, which is atypical of digital ballots. Further aspects of verifiability are necessary when machines cannot be trusted:

[RRS21] Peter B. Rønne, Peter Y. A. Ryan, and Ben Smyth. Cast-as-intended: A formal definition and case studies. In *FC'21: 25th International Conference on Financial Cryptography and Data Security*, volume 12676 of *LNCS*, pages 251–262. Springer, 2021

Cast-as-intended demands a voter be able to check whether a machine-generated ballot expresses their vote.

Digital ballot construction typically mandates sampling bits. If a machine abandons the prescribed sampling procedure, computation delivers something resembling a ballot, rather than a correctly computed ballot: A ballot constructed in disregard for the prescribed procedure is not correctly computed. It may resemble a ballot. A ballot may even be computable in that way. However, ignoring the construction mandate means the result cannot a priori be considered a correctly computed ballot. Herein lies the rub: Voters cannot determine whether machines compute ballots or things resembling ballots. Cast-as-intended and individual verifiability don't compose to enable determination of whether collected ballots express voters' votes, since voters cannot determine whether machines even compute ballots correctly, there's a gap:

[Smy20a] Ben Smyth. Mind the Gap: Individual- and universal-verifiability plus cast-as-intended don't yield verifiable voting systems. Technical Report 2020/1054, Cryptology ePrint Archive, 2020

There's a mismatch between assumptions underpinning cast-as-intended and individual verifiability: Cast-as-intended assures a ballot expresses a vote, not

whether the ballot is correctly computed, whilst individual verifiability assures correctly computed ballots are collected. Cast-as-intended and individual verifiability don't compose in the expected way. Individual- and universal-verifiability plus cast-as-intended don't yield verifiable voting systems, when ballots are computed on untrusted voting kiosks. Identifying a suitable security notion to bridge the gap is a direction for future research.

References

[FQS19] Ashley Fraser, Elizabeth A. Quaglia, and Ben Smyth. A critique of game-based definitions of receipt-freeness for voting. In *ProvSec'19: 13th International Conference on Provable and Practical Security*, volume 11821 of *LNCS*, pages 189–205. Springer, 2019.

[HS20] Thomas Haines and Ben Smyth. Surveying definitions of coercion resistance. Technical Report 2019/822, Cryptology ePrint Archive, 2020.

[KRS10] Steve Kremer, Mark D. Ryan, and Ben Smyth. Election verifiability in electronic voting protocols. In *ESORICS'10: 15th European Symposium on Research in Computer Security*, volume 6345 of *LNCS*, pages 389–404. Springer, 2010.

[QS18] Elizabeth A. Quaglia and Ben Smyth. A short introduction to secrecy and verifiability for elections. Technical Report 1702.03168, arXiv, 2018.

[RRS21] Peter B. Rønne, Peter Y. A. Ryan, and Ben Smyth. Cast-as-intended: A formal definition and case studies. In *FC'21: 25th International Conference on Financial Cryptography and Data Security*, volume 12676 of *LNCS*, pages 251–262. Springer, 2021.

[SC22] Ben Smyth and Michael R. Clarkson. Surveying definitions of election verifiability. *Information Processing Letters*, 177, 2022.

[SFC21] Ben Smyth, Steven Frink, and Michael R. Clarkson. Election Verifiability: Cryptographic Definitions and an Analysis of Helios, Helios-C, and JCJ. Technical Report 2015/233, Cryptology ePrint Archive, 2021.

[SH19] Ben Smyth and Yoshikazu Hanatani. Non-malleable encryption with proofs of plaintext knowledge and applications to voting. *International Journal of Security and Networks*, 14(4):191–204, 2019.

[SHM15] Ben Smyth, Yoshikazu Hanatani, and Hirofumi Muratani. NM-CPA secure encryption with proofs of plaintext knowledge. In *IWSEC'15: 10th International Workshop on Security*, volume 9241 of *LNCS*. Springer, 2015.

[Smy18] Ben Smyth. A foundation for secret, verifiable elections. Technical Report 2018/225, Cryptology ePrint Archive, 2018.

[Smy19] Ben Smyth. Athena: A verifiable, coercion-resistant voting system with linear complexity. Technical Report 2019/761, Cryptology ePrint Archive, 2019.

[Smy20a] Ben Smyth. Mind the Gap: Individual- and universal-verifiability plus cast-as-intended don't yield verifiable voting systems. Technical Report 2020/1054, Cryptology ePrint Archive, 2020.

[Smy20b] Ben Smyth. Surveying global verifiability. *Information Processing Letters*, 163, 2020.

[Smy21] Ben Smyth. Ballot secrecy: Security definition, sufficient conditions, and analysis of Helios. *Journal of Computer Security*, 29(6):551–611, 2021.

[Smy24] Ben Smyth. Ballot secrecy: Security definition, sufficient conditions, and analysis of Helios. Technical Report 2015/942, Cryptology ePrint Archive, 2024.

[SRKK10] Ben Smyth, Mark D. Ryan, Steve Kremer, and Mounira Kourjieh. Towards automatic analysis of election verifiability properties. In *ARSPA-WITS'10: Joint Workshop on Automated Reasoning for Security Protocol Analysis and Issues in the Theory of Security*, volume 6186 of *LNCS*, pages 165–182. Springer, 2010.