A Survey: Electronic Voting Development and Trends

Komminist Weldemariam, Adolfo Villafiorita

Fondazione Bruno Kessler,
Center for Scientific and Technological Research (FBK-IRST)
Via Sommarive 18
I-38050 Trento, Italy
(sisai,adolfo}@fbk.eu

Abstract: Any practitioner working on electronic voting (e-voting) seem to have stumbled upon the main issues that seem to affect the area. On the one hand — given the criticality and the risk e-voting systems potentially pose to the democratic process — e-voting systems are permanently under a magnifying glass that amplifies any glitch, be it significant or not. On the other hand, given the interest e-voting raises with the general public, there seems to be a tendency to generalize and oversimplify. This tendency leads to attribute specific problems to all systems, regardless of context, situation, and actual systems used. Additionally, scarce know-how about the electoral context often contributes to make matters even more confused.

This is not to say all e-voting systems show the security and reliability characteristics that are necessary for system of such a criticality. On the contrary, a lot of work still has to be done.

Starting from previous experiences and from a large-scale experimentation we conducted in Italy, this paper provides some directions, issues, and trends in e-voting. Getting a clearer view of the research activities in the area, highlighting both positive and negative results, and emphasizing some trends, could help, in our opinion, drawing a neater line between opinion from facts and contribute to the construction of a next generation of e-voting machines to be safely and more confidently employed for elections.

1 Introduction

The advantages that e-voting systems can bring cannot be achieved without an observable cost (e.g., risks) one of which is opening security vulnerabilities for attackers [Mer01, GGR07, BBC+08, BBC+10]. In that respect, recently we have seen that most currently deployed e-voting systems share critical failures in their design and implementation, which render their technical and procedural controls insufficiency to guarantee trustworthy voting [LKK+03, KSRW04]. The lack of trust can also render even more secure and more reliable e-voting systems completely useless.

Clearly, the abundance of security threats in e-voting system and their increasing popularity make a strong case for the need of proposing new designs, protocols/schemes, as well as techniques and tools for their design, development as well as their security as-
The application and usage of known techniques such as business process modeling and formal techniques and tools in voting in general and in the development of an e-voting solution in particular, however, are very limited and unsatisfactory. Additionally, work to rigorously define e-voting properties, and attack models and languages for describing the counter-measurements is still more preliminary.

Although some progresses have been made in understanding and supporting better development of e-voting systems e.g., [MN03, XM05b, XM07, WVM07, VWT09, DKR09], there is no classification to understand the common characteristics, objectives, and limitations of these approaches. Thus, the lack of a comprehensive comparative study provides little or no direction on choosing the appropriate development techniques for particular needs.

In this paper, we classify the most important development approaches for e-voting systems and compare them among each other with respect to motivations, methods, and logics. More specifically, we have classified them in four major categories, according to what we believe to be their major contribution for the development of e-voting systems: UNDERSTANDING (the risks posed by the introduction of e-voting systems in the polling stations), REQUIREMENT (developing requirements for e-voting), IMPLEMENTATION (designing voting schemes, protocols and/or techniques), and ASSURANCE (using techniques and tools to analyze the security of existing systems, by giving lower-level and higher-level assurances). We hope the work contributes to designers, developers, certification authorities, as well as to technical election officials.

The paper is organized as follows. In Section 2 we review the usage of (business) process modeling and redesigning for understanding the context and risks due to the introduction of electronic solutions in the poll stations. In Section 3, we briefly surveying progress made in developing requirements for e-voting systems. We continue, in Section 4, by briefly surveying progress made in designing and implementing voting schemes. In Section 5, we focus on the application of formal methods and techniques and tools to assess the security of e-voting systems. We conclude, in Section 6, by presenting some conclusive considerations and viewpoints.

2 Understanding Risks

Understanding the “context” of elections is very important prior to introduce e-voting solutions. The obvious reason is that this helps to understand and discuss the possible risks that can be resulted by the introduction of new system. Previous work in this area focused on the understanding, representation, and effective implementation of e-voting procedures. That is, using BPR for understanding what changes could be introduced in the conventional voting procedures to allow safe and secure transition to electronic elections.

The BPR concept pertains to the redesign in the context of existing business rules, such that the introduction of e-voting solution can be evaluated. As it is critical to define roles
and responsibilities within the e-voting process which could furnish a better understanding of who is responsible for doing what in the different process stages to produce election results, it is also equally important to provide systematic methodology to reason on what can go wrong on this procedural rich workflow, instead of detecting the weaknesses until well after attacks have been taken action(s).

As far as we are aware, the first usage of BPR to evaluate the transition to e-voting is that proposed by Xenakis and Macintosh in [XM05b, XM07]. The authors investigated the need for applying business process re-engineering to electoral process in order to propose a possible transition into e-voting system. Risks and difficulties while introducing e-voting solution are discussed, in more detail, in [XM04a, XM04b]. Furthermore, the same authors in [XM04c, XM05a] discussed the need for procedural security in electronic elections and provided various examples of procedural risks occurred during trials in UK. The approach can obviously highlight some of the security implications of the administrative workflow in e-voting, such as, as discussed in [LKK+03]. However, these approaches lack a machinery to systematically model and analysis the alternatives nor provide precise notations for the redesigning activities, including modeling and analysis on the aspects needed for the transitions, such as systematic analysis of procedures.

In references [Mat06, WVM07], the authors developed a UML based methodology for modeling and analyzing electoral processes. The methodology is supported by a tool named VLPM [CMV09], that helps the modeling, analysis and structuring of electoral procedures as business process models. Beyond that, the VLPM tool helps assisting a law-maker to link laws with the process models, and a process engineer to analyze the effects of the changes due to the introduction of a new law (or law modification) on the models to maintain the “synchronization” of laws with models, as the same time by fostering collaboration between them, i.e. lawmaker and the process analyst. The methodology and the tool have been demonstrated for the development of an e-voting system named ProVotE [VWT09]. An approach to reason on security properties of the to-be models (which are derived from as-is model) in order to evaluate procedural alternatives in e-voting systems is presented [BDF+09]. In particular, using Datalog and the underlying analysis tool the authors expressed and analyzed security concerns, such as delegation of responsibility among untrusted parties and trust conflict. The aim is that of understanding problematic trust/delegation relationships and eventually finding ways to adopt a solution to the detected security properties violations.

3 Developing Requirements for E-voting

Various international documents such as European Union (EU) Venice Commission recommendations [Cou04] and the U.S. Federal Election Commission (FEC) Voting Systems Standard (VSS) [Fed02, Fed05], which describe a set of principles for voting systems. These documents mainly specify principles about the behaviors of each component of a voting system should respect, as well as the related procedures. The FEC-VSS, for instance, provides details about the standards to be used for performance and tests of the voting machines. It also describes non-functional requirements (e.g., audits log features) and specifications for various hardware components. However, these kinds
of requirements often make difficult the development and implementation of the actual system. Moreover, the way these documents describe (security) requirements is hard to understand, and sometimes they contain contradicting/conflicting requirements; specifically, the conflict between the requirements for secrecy and accuracy. If the e-voting system needs to be developed in a safe and secure way, there must be an appropriate requirements definition for it. We have surveyed a dozen of work in this area. Because of space limit, however, we are able to present few of them that we think are the most important and complete.

Reference [Mer01] presents a thorough discussion on three gaps that must be comprehended prior to developing (security) requirements for e-voting systems. These gaps are the technological gap — that is, between hardware and software, the socio-technical gap — that is, between social and computer policies, and the social gap — that is, between social policies and human behavior. The same author also coined the term audit trails, which is often used in DRE machines. Namely, the type of DRE equipped with printed audit trails is often called DRE-VVPAT. That is, a touch-screen-based machine that produces a printout of each vote, verified directly by the voter, to maintain a physical and verifiable record of the votes cast. Thus, an essential activity to ensure e-voting system behaves correctly is laying down what behaving correctly means for that system. This cannot be achieved without a proper engineering approach, such as requirements engineering techniques. The author (in [McG08]) presented an approach to address the mentioned problems by proposing a methodological approach for analyzing the root causes of the conflicts, organizational barriers (or procedural barriers), and requirements for critical election. The approach comprises of two strategies for the development of requirements, namely top-down and bottom-up. The first one is aimed at developing a set of requirements from an existing catalogue. The latter, instead, is aimed at developing a new catalogue.

Subsequent to [McG08], Volkamer has provided, “a standardized, consistent, and exhaustive list of requirements for e-voting systems” [Vol09]. Specifically, these requirements are mostly for stand-alone DRE and remote e-voting systems. Such requirements not only describe requirements that the system should meet, but also specify the corresponding laws or regulations for the evaluation of the systems themselves. The author developed a methodology for the requirement development process. The results of the methodology include system requirements (divided into functional, security, usability requirements), organizational requirements, and assurance requirements for both stand-alone DRE voting machines and remote e-voting systems. Furthermore, the methodology comprises of cross checks existing catalogues, election principles, and the possible threats. This could allow software engineers and developers to easily understand how their systems meet these requirements. Following that, the author proposed evaluation and certification procedure mostly for remote voting system by complementing the Common Criteria common evaluation methodology and also developed protection profile for remote voting.

In reference [WMV09], the authors showed the management and structuring of requirements using finite state machines (FSMs). That is, by defining relationships between requirements and system architecture based on FSMs. More specifically, the methodol-
ogy they followed allowed them to understand the election processes, identify constraints, and distinguish both common and event specific requirements from various requirements sources, e.g. from those mentioned above. These are then refined into fine-grained requirements using FSMs. The decomposition from high-level to low-level requirements and the logical dependencies among them have been demonstrated. Additionally, the separation between generic and election or configuration specific requirements is concrete and detailed enough to function as a general reference schema that could be adopted by other solutions. In other words, this approach is fairly general to be used for other e-voting systems and, possibly, to provide a roadmap — rough and draft as it might be — for bridging the gap between higher-level principles and lower level system specifications.

4 Designing Voting Schemes and/or Protocols

Prior works with respect to this area focused on the design of cryptographic schemes, protocols, and/or techniques for better designing of voting machines. The ultimate goals of these approaches include ensuring a voter can be certain about her/his vote has been recorded correctly and accurately (voter verifiability), no voter can prove to anyone else how s/he voted (receipt freeness), and an independent body can verify that the recorded votes exactly match with the published tally after the election [Ive91, CFSY95, Cha04]. What is most common to all these approaches is that they rely on the underlying cryptographic principles to various degree of complexity.

PunchScan [CPS+07, ECCP07] is a cryptographic voting system that is easy to use by the voter as well as by election officials while, at the same time providing transparent and reliable process. It also provides public verifiability, election integrity and enhanced voter privacy. Scantegrity [CEC+08, CCC+09] is a successor of PunchScan that meets industrial standard by providing end-to-end verifiability of the integrity of critical steps in the voting process and election results. Pre’t a Voter (verifiable electronic elections) [RBH+09] is a type of electronic voting system that uses paper based ballot forms that are converted to encrypted receipts to provide security and auditability, at the same time by keeping coercion resistant and making easy to use. The Scratch & Vote is another cryptographic voting method proposed in [Adi06]. It provides public election auditability using simple, immediately deployable technology. The method combines a variety of existing cryptographic voting ideas such as homomorphic encryption — e.g., which allows votes to be tallied without decrypting individual votes, the cut-and-choose at the precinct approach, and so on. Additionally, works like [FOO93, BT94, RRN01, SCM08] attempt to provide (maximum) secrecy and/or anonymity for the vote and voter.

We cannot, however, say that cryptographic schemes and/or protocols address the current situation in the democratic process for several reasons. For example, the protocols that have been proposed so far do not yet overcome all of the barriers to their use in critical elections [McG08]; although DRE machines are most popular in public elections in some U.S. states, the applicability and scope of the proposed schemes are very limited in these machines. Moreover, as noted in [KSW05], some cryptographic protocols live with some security holes, such that sensitive information about the election can be
leaked in one or another way. Therefore, we must analyze their security by considering the system in its entirety since these protocols are only one part of a larger system composed of voting machines, software design and implementations, and complex election procedures [KSW05].

In reference [Sas07], the author presents the concept of “designing voting machines for verification”, aimed at providing techniques to help vendors, independent testing agencies, and others verify critical security properties DRE voting machines. The basis idea of the approach consists of two interesting techniques. The first focuses on creating a trustworthy vote confirmation process, where the author proposed an architecture that splits the vote confirmation code into separate modules whose integrity are protected using hardware isolation techniques. The second focuses on helping ensuring a very important property in voting, that is, “None of a voter’s interactions with the voting machine, including the final ballot, can affect any subsequent voter’s sessions”. In order to do that, the author used hardware resets technique that restores the state of modules components to a consistent initial value between consecutive voters. With this, it could be possible to eliminate the risk of privacy breaches and ensure that all voters are treated equally by the systems.

Other works, such as [SKW06, Yee07] apply techniques used in other domains —like pre- rendering user interface and hardware separation— to build a higher assurance e-voting systems with accessible, verifiable, and secure voting system. The design of a trustworthy DRE-based voting system by exploring the TPM (Trusted Platform Module) infrastructures (e.g., PKI, hardware protection of cryptographic keys) is presented in [PT09]. Additionally, the authors present a scheme that improves registration integrity, and introduced a design that prioritizes election integrity. Their voting system has a nine-step as a whole, which takes place from an election’s inception to its final conclusion

5 Providing Assurances

With respect to the assurance of e-voting system, existing works focus in two main directions to access the security of e-voting systems. While the first one focuses on providing lower-level assurances, the other focuses on providing higher-level assurance, both using powerful techniques and tools.

5.1 Applying Formal Methods for e-voting

The usage of formal methods in the specification and verification of e-voting systems is relatively new. Existing works in this area present formal specification and verification of an e-voting system at different level of abstractions. These works practicality aimed to demonstrate the feasibility of formal verification of voting machine logic, thereby providing a higher level of assurance on the security of the system. In this area the trends focus in three closely related directions, mainly according to the perspective of the aim of the verification. These are verifying cryptographic protocols, system behavior, and procedures.
The references [DKR09, KR05] present a framework for formal specification and verification of three privacy-type e-voting protocol properties. These properties are vote-privacy, receipt-freeness, and coercion-resistance. The authors used applied π-calculus [AF01] to formalize these properties as observational equivalence, after being formalized the voting protocol as a set of processes using the same machinery. In [CFM+08], the authors used a CCS (Calculus of Communicating Systems) like process algebra with cryptographic primitives to specify and analyze some properties of the e-voting system they built. More specifically, they presented a small mobile implementation of an e-voting system named M-SEAS (Mobile Secure E-voting Applet System) and used formal verification technique to validate the security property of the system.

The authors in [VWT09] demonstrate the integration of formal methods in the development process of a voting system. In particular, the authors specified the behaviors of voting control logic using UML finite state machine and developed a tool named FSMC⁺¹ that automatically generates NuSMV [CCG+02] code corresponding to the specified FSM (this helped the requirements discussed in [VWT09]). Then they performed the verification using the NuSMV model checker. The results of the model checker, presented in the form of counter-measurement, are then analyzed. This enabled the authors to incorporate the analysis results of the verification into the actual development process of the core application.

In references [WKV09, WKV10], the authors show how formal methods can be used to reverse synthesis existing e-voting system (named ES&S voting system). They used ASTRAL language to specify the ES&S voting process and used PVS analysis tool. A number of critical security requirements that the machines should respect have been specified and analyzed against the specification. Subsequently, the authors specified known attacks for the system (as demonstrated in [MBV07]) using the same machinery and extended the original specification, and then preformed the analysis on the extended model against on the same set of critical security requirements as the original specification should respect. The two main lessons drawn from their work are: formal methods help get a better understanding of the security “boundaries” of e-voting systems, and the role of open specifications play for the development of more secure e-voting systems. The reference [SJSW09] presents an approach for the design and analysis of an e-voting machine based on combination of formal verification and systematic testing. They formally verify the correctness of each of the individual component of voting machine, as well as verifying some crucial correctness properties of their composition. Their work is targeted to the following verification goals: ensuring that each individual component of the voting machine and their composition should meet the specification of the individual components and their composition respectively; voting machine should be structured to enable sound systematic system testing; ensuring that the voting machine must behave and store votes according to the voters selection when configured with a particular election definition file. For each module, they construct a formal specification that fully

¹ http://ict4g.fbk.eu/fsmpc/last/
characterizes the intended behavior of that component. A number of properties related to the structural and functional aspects that the machine should satisfy are identified and specified. They used Verilog [TM91] for the implementation of their specification and SMV\(^2\) analysis tool and satisfiability solving (especially, the SMT solver) to verify that their Verilog implementation meets the specifications.

Finally, in reference [WV08], the authors proposed an approach to formally analyze procedures. Namely, they proposed a methodology based on NuSMV [CCG+02] machine to analyze procedures systematically.

### 5.2 Assessing exiting e-voting systems

Some e-voting systems currently deployed in elections have recently undergone a thorough and independent scrutiny to evaluate their security and quality. This is because, in the recent years, the DRE machines raised serious security concerns. These machines make the election process less verifiable and greatly expand the aspects of an election for which voters must rely solely on trust. Security vulnerabilities have been reported in each aspect of security—that is, technological, socio-technical, and social aspects, as noted prior in [Mer01]. These vulnerabilities have been practically investigated and proved by various academic researches. This creates an enigma in the trustworthiness of the machine and the voting process as well.

In line with this, we mention the following academic researches [Jon03, KSRW04, GGR07, BBC+08, ASH+08]. These works assess both hardware and software of different forms of e-voting machines (e.g., Diebold/Premier, ES&S, InterCivic), mostly used in some U.S. states. The studies identified serious design and implementation flaws, which are notable for their level of egregiousness. More specifically, these analyses have showed that the current e-voting systems are vulnerable to very serious attacks, as well as they have produced a catalogue of vulnerabilities and possible attacks. Some analyzes also suggested a drastic change in the way in which e-voting systems are designed, developed, and tested (e.g., by identifying procedures to eliminate or mitigate the discovered issues, by developing a precise methodology and toolsets for the assessment). The assessment methodology presented in [BBC+08, MBV07] is particularly astonishing as it provides various insights on each individual and in-depth step of the analysis, to be reparable also. It can be used for other complex-security critical systems evaluation and assessment as well as to the software testing community.

### 6 Discussions and Conclusion

There are a number of established approaches for modeling, specifying, and verifying a system satisfies a set of properties. One important contributor to the security of any system is the way in which the software is designed and developed. Standards for software
engineering developed over the last 40 plus years require that a system undergo a rigorous process of requirements definition, structured design and review, and careful programming and testing [Som95]. Like proper engineering leads to cars of higher quality, so too does better software engineering lead to more secure, robust software computer systems. Systems that are designed without this kind of careful design and implementation are almost certain to have flaws and security issues.

BPR techniques help to understand, model, and analyze the high-level context of the electoral processes. This gives information about the context of the business architecture (as-is) and software delivery (to-be) prior to the subsequent development activities for the introduction of e-voting solution. It also helps assessing the effectiveness of the processes as experienced and evaluated by the citizens outside the development and support organizations. However, it is not always possible to transform business solution to e-voting as a solution [AO05]. This is because, unlikely from firm processes, the electoral processes are tightly bounded by legal framework and are usually more regulated than business processes. Thus, we need a proper methodology and tools that abet such reengineering activities. However, some approaches such as the one given in [CMV09] can be a starting point to extend and reuse in the reengineering process of e-voting projects.

The usage of formal methods has been shown to improve the security and quality of complex systems3. These approaches allow designers to prove, test, or otherwise examine interesting properties of a complex process whose behavior is specified abstractly, and then interactively refine the behavioral specification to be as close to an implementation as appropriate for a given assurance level. Since the technique has been recognized as powerful and effective approaches for improving the security and quality of complex systems (such as in flight (software) system), drawing straight connection with this can help making better the current development of e-voting machines.

Moreover, the studies of experimental data about the e-voting machines’ security, performance and their evolution with respect to the social and technical aspects are still unsatisfactory. This limits their usage in a larger scope. For example, data set on observing security threats to voters anonymity by following standard procedures that illustrate each machine’s behavior during elections can help raise the transparency in elections using electronic devices and increase the confidence of voters in the democratic system. Data sets related to the process of setting up experiments, running an election, and performing security evaluation across various voting machines (e.g., as in Diebold and ES&$S$) provide information about the behavior of machines under malicious circumstances, if they are not designed carefully, and provide need to consider recommendations for design alternatives.

Developing and deploying e-voting systems in a safe and secure manner requires ensuring the technical and procedural levels of assurance with respect to social and regulatory frameworks. In this paper, we have presented techniques mainly in three areas (namely, BPR, formal methods, and security) and showed how these techniques are effectively exercised for correct design and implementation of e-voting systems. Thus, therefore, the success of the next generation of e-voting machines depends upon being able to capi-
talize lessons learned from different disciplines. The work we have presented in this paper is one way in which we can get a better understanding of the strengths and the weaknesses of existing techniques and thus lay the foundations for engineering, designing, implementing, as well as deploying a new generation of more secure and robust technologies for polling stations.

Bibliography


