POSTER: Enabling User-Accountable Mechanisms in Decision Systems

Rosario Giustolisi and Carsten Schürmann

IT University of Copenhagen, Denmark

Abstract. Decision systems are at the core of our democratic and meritocratic processes. Systems for voting, procurement, grant management, and competitive examinations all rest on *submission*, *evaluation*, and *ranking*. Computer assistance is a critical part of modern decision systems and so are cybersecurity challenges. As decision systems get increasingly complex, the classic approach of enforcing security through fail-safe mechanisms *preventing* cybersecurity attacks becomes infeasible. A recent trend in cybersecurity is to disincentivize potential attacks by using deterrence-based mechanisms that make stakeholders accountable for their actions. However, using such mechanisms requires knowledge of the underlying technology, which is not accessible to all people.
This poster looks at ways to extend decision systems with *user-accountable mechanisms* enabling users to verify correct executions and provide dispute resolution capabilities by combining cryptographic techniques for human senses with advanced cryptographic protocols. If successful, this line of work will provide novel ways to secure decision systems by creating disincentivizing mechanisms that are accessible to any human user.

1 Motivation

Currently, decision systems require user expertise in the auditing technology. It is an open question on how to make auditing accessible to everyone. End-to-end verifiable voting schemes, which allow voters to check that the outcome of an election is correct, have no user-friendly mechanisms for dispute resolution in case of incorrect tallying, hence they do not provide adequate attack deterrence [8]. Current systems for procurement, grant management, and competitive examinations heavily rely on trusted parties that run core parts of the system as black boxes at the price of a lack of transparency [6]. We challenge such design in favour of a trust-no-one and user-accountable design. Moreover, general approaches for algorithmic accountability have been recently proposed in the context of AI, machine learning, and secure multi-party computation (MPC) [7], but the verification procedures require relevant expertise in the underlying technology to accomplish auditing.

The verification procedures for accountability should essentially be a human task. State-of-the-art technologies only provide guarantees for machines, while leaving out human users. This undermines public confidence in the system's reliability and in the end affects negatively the trustworthiness of decision systems. We thus need to design accountability mechanisms that can be used by humans.

We observe that existing cryptographic techniques for human senses, such as visual cryptography and hash visualization, can achieve classic security goals such as confidentiality and authentication. We aim at exploring whether a combination of cryptographic techniques for human senses with state-of-the-art cryptographic protocols enables user-accountable mechanisms in decision systems. We deem the following combinations to be of particular relevance:

Cryptographic techniques for human senses		**Protocols**
Visual cryptography [9]		Zero-knowledge proofs
Audio cryptography [3]	×	Oblivious transfer schemes
Lound-and-clear [5]		Identity-based encryption
Hash visualization [10]		Homomorphic schemes
EyeDecrypt [4]		

Building user-accountable mechanisms for decision systems will give: (i) a concrete method to design accountable decision systems, in the age of pervasive security attacks aiming at breaking down public trust; (ii) a novel security paradigm that revisits the established view of "users being the weakest link *in* security" in to "users as the needed link *for* security"; (iii) new research directions in cybersecurity aimed at reconciling the mathematical guarantees of cryptography with the public confidence in democratic and meritocratic processes.

2 Approach

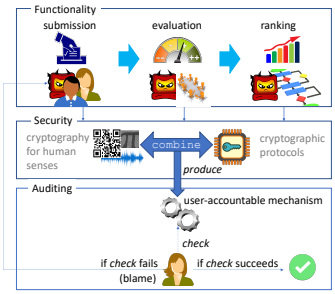

Fig. 1: Design approach for user-accountable mechanism in decision systems.

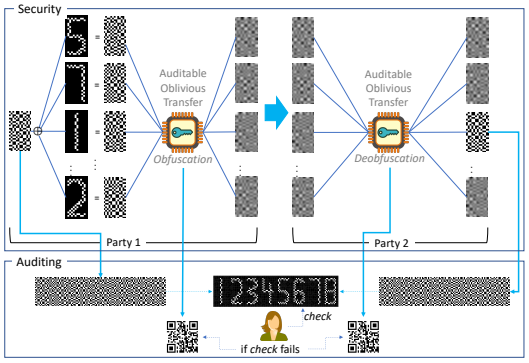

Fig. 2: A user-accountable mechanism for verifying the correct generation of anonymous identifiers.

Figure 1 presents our approach to design accountable mechanisms for decision systems. We consider potential misbehaving parties willing to attack one or more functionalities in a decision system. We identify the needed security guarantees and choose accordingly cryptographic protocol and technique for human sense. We want to allow a user to detect the failure of a functionality requirement and to find the misbehaving parties who caused that failure. We model a user-accountable mechanism by intertwining the chosen protocol and technique in order to provide evidence that can be used by humans. Below we show how one can construct a user-accountable mechanism by combining visual cryptography and an oblivious transfer scheme.

2.1 User-Accountable Anonymous Identifiers

One of the core functionalities of a decision system is to guarantee anonymous submissions. For example, in voting, this is equivalent to ballot privacy. The goal is to generate identifiers to anonymise submissions *and* to make their construction user-accountable in case of a dispute. In a trustless environment, anonymous identifiers can be normally built using secret sharing. We describe how one can design a user-accountable mechanism for anonymous identifiers by combining oblivious transfer schemes, visual cryptography, digital signatures, and QR codes (cf. Figure 2). We note that the combination of digital signatures and oblivious transfer can provide auditability. In a two-party setting, an auditable oblivious transfer enables one party to generate a secret random permutation of the set of possible identifiers, and the other party to obliviously select one random element of the permutation. Parties can send each other encrypted audits, which can be encoded as QR codes. Also, each party encodes their secrets using visual cryptography, and only when the secrets are brought together the identifier is determined.

More specifically, to encode a single secret visual character c, one party generates a random visual crypto image share α, prints it, and generates a complementary set of visual crypto shares $\beta_1, \ldots, \beta_k$ for each of the possible $c_1, \ldots, c_k$ characters, such that $\beta_i \leftarrow_{\pi_\mathcal{R}} (\alpha \oplus c_i)$, with $\pi_\mathcal{R}$ being a random permutation of the list of characters. The same party receives from the other party a Pedersen commitment $y = g^x h^\gamma$ on an unique index $\gamma \in_\mathcal{R} [1, k]$, randomly permutes the order of the set of visual crypto shares, and sends to the other party an obfuscated version of the set $\omega_1, \ldots, \omega_k$ using Tzeng's oblivious transfer scheme [12], such that $\omega_i = \langle a_i, b_i \rangle \leftarrow \langle g^{r_i}, \beta_i \left(\frac{y}{h}\right)^{r_i} \rangle$ together with signatures on to the other party's commitment. The selected obfuscation can be signed and printed as a QR code for auditing purposes. Then, the other party deobfuscates a random element of the set based on its commitment $\beta = \frac{b_\gamma}{(a_\gamma)^x}$ and prints the deobfuscated share as well as the signature on its commitment as QR code. Notably, none of the parties learns which share has been printed by the other party, hence the secret character is revealed to the parties only when the shares are brought together. The procedure can be iterated so that a secret identifier can be built from a sequence of secret characters.

This example shows that the human user can visually verify whether any of the parties misbehaved since no intelligible identifier would be determined if any of the parties misprints their visual shares. Thanks to the digital signatures encoded in the QR code, the user can also blame which parties misprinted their share, making them accountable for the generation of anonymous identifiers.

3 Challenges and Potential

One of the main challenges is that a single user-accountable mechanism might not fit all situations. We can address this by investigating several alternative combinations of protocols and techniques, and by developing specialised mechanisms for specific systems. Having defined a way to design user-accountable mechanisms and to add them to decision systems, we will first implement our user-accountable mechanisms in a mock voting system. Then, we will test whether our user-accountable mechanisms can be integrated into existing secure decision systems such as Helios [1], Prêt à Voter [11], and Confichair [2]. This will convince us whether our approach can scale up to any other decision systems.

Also, privacy and accountability are intuitively two contrasting requirements: accountability demands for more evidence to accomplish the verification procedures aiming at increase confidence in the decision outcome; privacy demands for minimising such evidence. For example, a voting system should provide high confidence in the result of the election even for voters who do not necessarily trust the voting authority. On the other hand, failing to provide vote privacy opens to effective manipulation of voters and to control the outcome of the election. While it is challenging designing mechanisms that maximise both accountability and privacy, one can explore mechanisms that allow one to set an appropriate trade-off between privacy and accountability.

Providing a practical design for user-accountable mechanisms enables people to audit the system and fosters public trust in accepting computer assistance in decision systems. This has the potential not only to pave the way for an exciting research agenda in developing a new generation of decision systems, but it can provide new directions in securing distributed systems and MPC. Also, the field of AI and machine learning, whose current efforts are aimed at helping the machines to understand and interpret humans, can eventually benefit from this line of work, which is about helping humans to interpret the machines.

4 Conclusion and Open Questions

This work explores ways for building verification mechanisms for accountability in decision systems by combining existing cryptographic techniques for human senses with state-of-the-art cryptographic protocols. The ultimate goal of this line of work is to enable human users to directly execute the verification mechanism themselves without the need of relying on trusted computer parties. This poster also provides a practical example on how one can implement such a mechanism.

Although our example works for generating verifiable anonymous identifiers for submissions, it may not work to address any requirement of decision systems. Still, with this work, we challenge the established position of minimising people involvement in cybersecurity by building methods that enable people to understand *when* the underlying technology works, rather than *how* it works.

This line of work can provide a practical way to build and strengthen public confidence in decision systems technology. The need for user-accountable mechanisms is concrete and actual, as it could be readily used in today's challenges, such as providing credible elections. This can also inspire addressing future challenges, such as allowing any user to audit AI-powered decision systems.

Some open questions can be easily drafted. An obvious one comes from observing that some decision systems are very different from each other, and a user-accountable mechanism may not be used in, e.g., both voting and procurement systems. In fact, there are obvious differences in the requirements among categories and also within a single category. One can address this question by focusing on the core functionalities and related requirements, which exist among all decision systems. We believe that even a limited number of user-accountable mechanisms provide some guarantees to the users and deters attacks, unlike current decision systems.

Finally, we observe that security researchers have not looked at this before for some reason. The cryptographic community focuses on making highly secure and efficient algorithms for machines while usable security community focuses on how to make secure technologies more human-centric. The two communities do not collaborate often. This line of work also aims to bridge the gap.

References

1. Adida, B.: Helios: Web-based open-audit voting. USENIX (2008)
2. Arapinis, M., Bursuc, S., Ryan, M.: Privacy Supporting Cloud Computing: ConfiChair, a Case Study. POST (2012)
3. Desmedt, Y., Hou, S., Quisquater, J.J.: Audio and optical cryptography. In: Ohta, K., Pei, D. (eds.) ASIACRYPT (1998)
4. Forte, A.G., Garay, J.A., Jim, T., Vahlis, Y.: Eyedecrypt — private interactions in plain sight. In: Abdalla, M., De Prisco, R. (eds.) SCN (2014)
5. Goodrich, M.T., Sirivianos, M., Solis, J., Tsudik, G., Uzun, E.: Loud and clear: Human-verifiable authentication based on audio. In: In ICDCS (2005)
6. Kanav, S., Lammich, P., Popescu, A.: A conference management system with verified document confidentiality. In: Biere, A., Bloem, R. (eds.) CAV (2014)
7. Kroll Joshua A.: Accountable Algorithms. Ph.D. thesis, Princeton (2015)
8. Küsters, R., Truderung, T., Vogt, A.: Accountability: definition and relationship to verifiability. In: CCS (2010)
9. Naor, M., Shamir, A.: Visual cryptography. In: EUROCRYPT (1995)
10. Perrig, A., Song, D.: Hash visualization: a new technique to improve real-world security. In: Int. Work. Techniques and E-Commerce (1999)
11. Ryan, P.Y.A., Schneider, S.A.: Prêt à voter with re-encryption mixes. ESORICS (2006)
12. Tzeng, W.G.: Efficient 1-out-of-n Oblivious Transfer Schemes with Universally Usable Parameters. IEEE Trans. on Computers (2004)