Poster: You Share Because We Care: Secure Allegation Escrow System

Nishat Koti
Indian Institute of Science
Bangalore, India
kotis@iisc.ac.in

Varsha Bhat Kukkala
Indian Institute of Science
Bangalore, India
varshak@iisc.ac.in

Arpita Patra
Indian Institute of Science
Bangalore, India
arpita@iisc.ac.in

***Abstract*—The rising issues of harassment, exploitation, corruption, and other forms of abuse have led victims to seek comfort by acting in unison against common perpetrators (e.g., #MeToo movement). One way to curb these issues is to install allegation escrow systems that allow victims to report such incidents. The escrows are responsible for identifying victims of a common perpetrator and taking the necessary action to bring justice to them. However, users hesitate to participate in these systems due to the fear of such sensitive reports being leaked to perpetrators, who may further misuse them. Thus, to increase trust in the system, cryptographic solutions are being designed to realize secure allegation escrow (SAE) systems.**

In the work of Arun et al. (NDSS'20), which presents the state-of-the-art solution, we identify attacks that can leak sensitive information and compromise victim privacy. We also report issues present in prior works that were left unidentified. To arrest all these breaches, we put forth an SAE system that prevents the identified attacks and retains the salient features from all prior works. The cryptographic technique of secure multi-party computation (MPC) serves as the primary underlying tool in designing our system. At the heart of our system lies a new duplicity check protocol and an improved matching protocol. We benchmark the proposed system with state-of-the-art MPC protocols and report the cost of processing an allegation, and showcase its practicality.

***Index Terms*—secure allegation escrows, multiparty computation**

I. Introduction

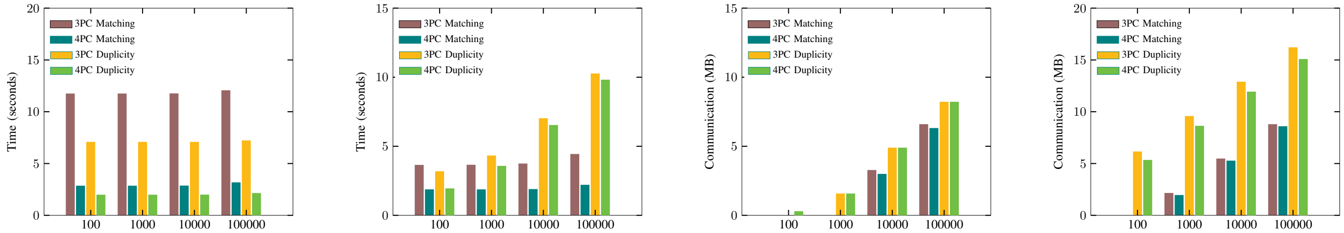
To deter crimes, institutions are mandated to appoint an organizational ombudsperson or a Chief Vigilance Officer (CVO) responsible for the prevention, detection, and punishment for malpractices. The victims are expected to report the inflicted crime to the CVO, which contains highly sensitive information. The profound harm that can be inflicted on victims if the CVO leaks this sensitive data to the perpetrator, which is likely when the latter is a person of influence, instills great fear in victims and prevents many from coming forward. Thus, such a system requires the victims to place enormous trust in the integrity of the CVO. Instead, a secure platform for reporting crimes is a more reliable solution. Thus, our work aims to design secure allegation escrow system that empowers victims to securely report allegations.

Desirable properties of secure allegation escrow: Victims often find it effective and comforting to come out as a group. A noteworthy example of this is Project Callisto [1]. To facilitate reporting and processing of such *collective* allegations, an allegation escrow system should have the following properties– (i) each victim must be able to independently file an allegation against a perpetrator, (ii) the system must be capable of matching allegations filed against a common perpetrator, (iii) these matched allegations should be revealed to the concerned authorities only once a predetermined condition for disclosure is met (e.g., Project Callisto requires at least two allegations against the same perpetrator before these can be revealed), (iv) the identity of the accuser, accused, and the details of the allegation must remain hidden until the allegation is revealed as a part of a collection. Additionally, instead of a centralized solution, it is desirable to have several independent escrows which collectively effectuate a secure allegation escrow (SAE) system with the above-mentioned properties and guarantee that *none* of the escrows can individually learn allegations on clear.

The condition for disclosure is one of the most crucial features of an SAE system. It defines the system's sensitivity towards handling an alleger's discomfort and is calibrated using a parameter called *reveal threshold*. The parameter captures the minimum size of the unison the alleger wishes to be a part of (excluding the alleger) when its allegation is revealed in clear to the concerned authorities. While Project Callisto [1] uses a globally-fixed public reveal threshold of one, the work of [2] extends support for a public reveal threshold of more than one. However, a system defined threshold may not cater to the needs of all the victims. The state of the art work of [3] recognizes this pressing requirement and allows an alleger the flexibility of deciding its reveal threshold, th, for its allegation. Here, a subset $\mathcal{S}$ of matching allegations can be revealed if and only if the threshold of each allegation in $\mathcal{S}$ is $< |\mathcal{S}|$ (size of $\mathcal{S}$). This is referred to as the *reveal criteria* of the set $\mathcal{S}$. Although [3] provides this key feature, it fails to do so while guaranteeing complete privacy to victims. Thus, we develop the *first* SAE system that offers not only a flexible user-defined threshold, but arrests privacy concerns in prior systems.

A. Our contributions

Attacks and drawbacks of prior systems: In [1], we identify various entities such as the LOC, database and communication server that form roots of trust, and describe attacks that breach privacy when these entities are compromised. We showcase how the invitation based system of Callisto is capable of



(a) Online Latency (b) Total Latency (c) Online Communication (d) Total Communication

Fig. 1: Comparison of matching and duplicity check protocol for varying number of allegations with $|\mathsf{P}| = 1/10 \cdot |\mathsf{A}|$ and maximum threshold set to 10 for 3PC, 4PC. Here, P is the number of already revealed perpetrators and A is the number of allegations in the system. Note that all plots are log-log plots with x-axis logarithmic in base 10 and y-axis logarithmic in base 2. We do not report values < 1 as their log is negative.

tracking its user activity, rendering it more as a trusted third party solution rather than a complete cryptographic one. In the work of [3], since the escrows learn allegation threshold and intermediate information such as match between allegations, we identify attacks that take advantage of such public information and render the system insecure.

Secure allegation escrow sytem: To address the privacy breaches, we design a secure allegation escrow system, while retaining the salient features from prior works. We are the first to provide a solution that keeps the threshold private too. The features provided by our system, in comparison to the prior works, appear in Table I. We prioritize user *privacy* over system *efficiency* since privacy is essential for an SAE system. A new replacement to the secure matching protocol that identifies a revealable set of matching allegations and the inclusion of a new duplicity check protocol that prevents users from filing duplicates lies at the heart of our system. We additionally provide features such as allegation modification and deletion, which were absent in [3].

Protocol	*Flexible* reveal threshold	*Private* reveal threshold	Duplicity complexity	Matching complexity
[1]	✘	✘	NA*	$\mathcal{O}(\mathsf{N})^{\dagger}$
[2]	✘	✘	$\mathcal{O}(\mathsf{N})$	$\mathcal{O}(\mathsf{N} \cdot q)$
[3]	✔	✘	✘	$\mathcal{O}(1)$
Ours	✔	✔	$\mathcal{O}(\mathsf{N})$	$\mathcal{O}(\mathsf{N} \cdot \mathsf{mxt})$

q: fixed reveal threshold, mxt: upper bound on flexible reveal threshold, N: number of allegations in the system.

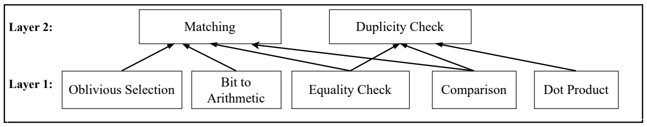
*Duplicity check is not applicable here. Filing a duplicate allegation, to prematurely reveal a genuine one, requires a threshold of at least 2 as opposed to 1 in Callisto.

†Due to missing details in Callisto, the complexity reported assumes requirement of a linear scan to identify matching allegations.

TABLE I: Comparison of SAE protocols

Modular approach: We resort to a modular approach to design protocols (see Fig. 2) by identifying MPC building blocks that would be required. We make black-box use of the MPC building blocks which not only allows to inherit the latter's security guarantees and efficiency, but also opens up the possibility of utilizing future advancements of MPC in a seamless way.

Benchmarks: We benchmark the complexity of our allegation processing over a WAN, instantiated using n1-standard-64 instances of Google Cloud, and report the overhead involved in the enhancement. We instantiate the MPC using state-of-the-art 3-party computation (3PC) and 4-party computation (4PC) frameworks of SWIFT [4] and Tetrad [5], respectively.



Primitives categorized into layers where higher one builds over lower ones, which implicitly build on Layer 0 - input sharing, reconstruction, addition, multiplication - provided by underlying MPC.

Fig. 2: Hierarchy of Primitives

The system comprises six phases–(i) initialization, (ii) user registration, (iii) allegation filing, (iv) duplicity check, (v) allegation matching and (vi) allegation revealing. Initialization is a one-time process, and hence does not add to the cost of keeping the system running. Cost for registering a user is given in Table II. Since multiple users can register simultaneously, we report the throughput (number of users registered in parallel per minute) which is 6364 for 3PC and 7182 for 4PC.

#Escrows	Online		Total	
	Latency (s)	Com (MB)	Latency (s)	Com (GB)
3	1.99	2.75	11.98	0.39
4	1.99	27.54	93.18	3.85

TABLE II: Communication and latency for registering a user in 3PC, 4PC.

We do not report the cost for allegation filing since it involves local operations. Duplicity check and allegation matching make up the compute-intensive phases and can only process one allegation at a time. These costs were not reported in [3] since it had a constant-time matching (and duplicity check was missing). Hence, our costs capture the overhead in comparison to [3] and is the price paid for obtaining full privacy. The results can be analyzed from Fig. 1.

References

[1] A. Rajan, L. Qin, D. W. Archer, D. Boneh, T. Lepoint, and M. Varia, "Callisto: A cryptographic approach to detecting serial perpetrators of sexual misconduct," in *Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies*, 2018.

[2] B. Kuykendall, H. Krawczyk, and T. Rabin, "Cryptography for# metoo," *PETS*, 2019.

[3] V. Arun, A. Kate, D. Garg, P. Druschel, and B. Bhattacharjee, "Finding safety in numbers with secure allegation escrows," in *NDSS*, 2020.

[4] N. Koti, M. Pancholi, A. Patra, and A. Suresh, "SWIFT: Super-fast and robust privacy-preserving machine learning," in *USENIX Security*, 2021, https://eprint.iacr.org/2020/592.

[5] N. Koti, A. Patra, R. Rachuri, and A. Suresh, "Tetrad: Actively secure 4pc for secure training and inference," *To Appear In NDSS*, 2022, https://ia.cr/2021/755.