Evidence for a posteriori security

Alexander Hicks
University College London
alexander.hicks@ucl.ac.uk

Steven J. Murdoch
University College London
s.murdoch@ucl.ac.uk

1. INTRODUCTION

Problems of security and privacy are often addressed by attempting to provide a priori secure systems through mechanisms that handle threats after the fact. Often, this is done by accounting for, and limiting, the flow and control of information across a system.

However, it is not always possible to apply this approach successfully. Some systems require flexibility to function as intended, particularly those that involve variable human input, which can sometimes be erroneous or malicious. But behaviour cannot be too limited without impeding the system and may be subject to vague regulations that cannot be codified algorithmically. Protocols can also fail for many reasons, through faults, malicious operations or design decisions. Determining actions to be correct, malicious or erroneous is hard, but must be accounted for when failures happen in situations involving multiple parties that are affected differently, whether negatively or positively, and liability must be assigned.

This requires accountability mechanisms, which serve to disincentivise malicious behaviour. In order to achieve this, an enabling mechanism is required: robust evidence that can be presented as proof of fact and used to enforce accountability.

The aim of this talk is to discuss the class of problems that require a posteriori evidence, the production and form of evidence and the security and privacy context it could be used in.

2. BACKGROUND: THE NEED FOR EVIDENCE

When things can go wrong, an important goal is to incentivise honest behaviour, or at least disincentivise dishonest behaviour. This requires a mechanism that enables rewards or punishments to be correctly assigned and delivered [2]. Generally, this requires a third party or some form of consensus who then need evidence in order to assign rewards or punishments. Evidence also features in formal models of accountability like the one proposed by Feigenbaum et al. [5], under the name *event traces*.

Examples of this exist in practice. Principles for designing systems to produce robust evidence have been formulated by Murdoch and Anderson in the case of payment systems [7]. Evidence of access to data requests, in the case of access to data by law enforcement is also proposed as a way of making the process more accountable, facilitating audits of such access requests [6]. The Bitcoin lighting network [8] (an off-chain payment system) relies on evidence, that is presentable to the network, to incentivise honest participation in the face a losing funds. Certified e-mails [1] and non repudiation protocols [9] also rely on evidence, and have been formally verified [3].

There are also examples where a definition of evidence would be beneficial. The new European General Data Protection Regulations (GDPR) [4] contains notions such as the right to explanation and right to erasure, but no notion of the evidence required for compliance, limiting the trust one may have in the system.

3. DISCUSSION

The talk will focus on the use of evidence as a mechanism to provide security and privacy when dealing with problems that cannot be addressed ex-ante, but instead require a posteriori solutions.

We consider general systems, divided into three distinct levels. First, he protocol level of the system that includes any machine level processes. Second, he human level of the system involves human input and operation of the system including decisional aspects. Third, the regulations, policies and other high level aspects that govern how and why the system is designed and used, including any requirements or constraints that are applied or aimed to be followed.

For large scale systems that involve many (protocol or human) components, failures can occur at many points of the system. At the protocol level, bugs, invalid inputs and (potentially) backdoors can cause failures or incompatibilities with other components of the system. At the individual level, failure can happen through misuse (intentional or not) of the system, for example by inputting the wrong data, or by executing the wrong computations. This goes beyond traditional security requirements that aim to make it impossible for ”bad” things to happen, as in these cases they cannot be accounted for without a priori limiting the functionality and use of the system. Regulations and policies then apply, by defining proper use in the context of the system as well as consequences (e.g., assigning liability) to misuse and responsibility to handle errors, whether they happen at the protocol or individual level.

The class of problems that can be addressed after the fact, through the use of evidence, differs from problems that can be addressed through a priori mechanisms. Rather than modifying the systems functionalities, the goal is rather to account for all the functionalities so that there is evidence of errors if they so happen. This of course links to ideas of auditability and accountability, in particular for systems that are not (and perhaps cannot) be fully transparent, in-

volve functionalities that must be unconstrained or involve usage policies that cannot be captured algorithmically e.g., human decision making.

Models.

We first discuss the family of threat models that are addressed by a posteriori security and evidence, rather than by a priori security methods. In particular, we look at how evidence, if it can be reliably and robustly produced, can provide solutions to these threat models, functioning as an enabling mechanism for disincentives when the system is properly aligned against malicious behaviour. Modelling the impact of evidence as an enabling mechanism on the alignment of the system is also part of this process.

Evidence production and constraints.

How can a system produce evidence? The goal should be to record the state of a system without affecting its functionality i.e., produce evidence without affecting other components of the system. Evidence could also be produce on a secondary system, as in the case of logs that record important properties of the system.

There are of course constraints. Evidence production must be such that it cannot be affected by the system it records evidence of, so that false evidence cannot be produced. The produced evidence must also be tamper evident, so that it cannot be unnoticeably altered after the fact. It must of course all necessary information, but in cases where private information is involved, it may be required to produce evidence that does not unconditionally reveal all information.

Forms and usage of evidence.

The next point of discussion is the form evidence might take, and its use. The form would depend on who the evidence is presented to. Generally speaking, public presentability and public verifiability must be considered. By presentability we mean that it is in some recognisable form i.e., it is possible to distinguish between different types and instances of the same type of evidence, so that one can be reassured that they are looking at the right evidence. By verifiability, we mean that the evidence is publicly verifiable i.e., it is the correct evidence for the use case, was produced correctly and remains tamper free. That the conclusions reached by evidence be verifiable is another stronger requirement that may be desired, for example in the cases of consensus protocols or automated forms of verification.

There are many possible use cases, but the use of evidence can be broken down into three levels. First, at the production level where it will be presented to parties that work on or with the system and have significant expertise. In that case, evidence is used to understand a failure internally so it may not need to take a publicly presentable form, or be publicly verifiable.

Second, at the level of an expert witness, or more generally when evidence is examined by an expert in a public setting. Although the public may not inspect the evidence themselves, trust in the conclusion of the expert is necessary. The expert may even find themselves accountable if their conclusion has an important impact.

Third, there is the level where a non-expert (e.g., a judge, or the public) evaluates the evidence. This requires high levels of presentability of information that may be technically complex, but this would make it easy for the public to trust the conclusion.

It is also important to consider evidence on a more technical level. Cryptography is the natural tool to use when security (e.g., tamper resistance) and privacy requirements exist. In particular, tools like digital signatures, hashes (and construction from hashes like Merkle trees), variants of Proof-of-X and verifiable computations can provide evidence of actions, tamper evidence or compliance. Privacy focused tools like zero-knowledge proofs, private computations and selective disclosure can, among other things, provide forms of privacy preserving evidence.

4. SUMMARY

The above involve a mix of Systems in the production of evidence, Cryptography in the form it takes and security or privacy guarantees it fulfills, and Game Theory in the role it plays in disincentivising malicious behaviour. A summary of talking points is as follows:

- **Models:** What threat models are addressed by evidence? How can the use of evidence be modelled?
- **Production of evidence:** How can evidence be produced so that it is robust and reveals information that cannot be tampered with? Can different pieces of evidence be used together so that they reveal different information to different parties, without leaking the wrong information to the wrong parties?
- **Forms of evidence:** What form should evidence take in the various cases it may be used?

5. REFERENCES

[1] M. Abadi and B. Blanchet. Computer-assisted verification of a protocol for certified email. In *International Static Analysis Symposium*, pages 316–335. Springer, 2003.

[2] S. Azouvi, A. Hicks, and S. J. Murdoch. Incentives in security protocols. 2018. *To appear in Proceedings of the Twenty-sixth International Workshop on Security Protocols.*

[3] G. Bella and L. C. Paulson. Accountability protocols: Formalized and verified. *ACM Transactions on Information and System Security (TISSEC)*, 9(2):138–161, 2006.

[4] European Parliament and Council. General Data Protection Regulations, 2016.

[5] J. Feigenbaum, A. D. Jaggard, and R. N. Wright. Towards a formal model of accountability. In *Proceedings of the 2011 New security paradigms workshop*, pages 45–56. ACM, 2011.

[6] A. Hicks, V. Mavroudis, , M. Al-Bassam, S. Meiklejohn, and S. J. Murdoch. VAMS: Verifiable auditing of access to confidential data. 2018. *Under submission.*

[7] S. J. Murdoch and R. Anderson. Security protocols and evidence: Where many payment systems fail. In *International Conference on Financial Cryptography and Data Security*, pages 21–32. Springer, 2014.

[8] J. Poon and T. Dryja. The Bitcoin lightning network: Scalable off-chain instant payments. *Technical Report (draft)*, 2015.

[9] J. Zhou and D. Gollman. A fair non-repudiation protocol. In *Security and Privacy, 1996. Proceedings., 1996 IEEE Symposium on*, pages 55–61. IEEE, 1996.