Formal Methods for Socio-technical Security
(Formal and Automated Analysis of Security Ceremonies)

Luca Viganò(✉)

Department of Informatics, King's College London, London, UK
luca.vigano@kcl.ac.uk

Abstract. Software engineers and analysts traditionally focus on cyber systems as technical systems, which are built only from software processes, communication protocols, crypto algorithms, etc. They often neglect, or choose not, to consider the human user as a component of the system's security as they lack the expertise to fully understand human factors and how they affect security. However, humans should not be designed out of the security loop. Instead, we must deal with security assurance as a true socio-technical problem rather than a mere technical one, and consider cyber systems as socio-technical systems with people at their hearts. The main goal of this short paper, which accompanies my keynote talk at the 24th International Conference on Coordination Models and Languages (COORDINATION 2022), is to advocate the use of formal methods to establish the security of socio-technical systems, and to discuss some of the most promising approaches, including those that I have helped develop.

1 Introduction

A recent study by IBM revealed that 95% of cyber-attacks are due to human error [35]. This is not surprising as, in a landscape where the security threats and attacks are in continuous evolution and high-value private information can be lost or manipulated, there is an increasing number of cyber systems (for communication, commerce, business, voting, industrial processes, critical infrastructures, etc.) whose security depends intrinsically on human users.[1] However, software engineers and analysts traditionally focus on cyber systems as *technical systems*, which are built only from software processes, communication protocols, crypto algorithms, etc. They often neglect, or choose not, to consider the human user as a component of the system's security as they lack the expertise to fully understand human factors and how they affect security. Humans should not be

[1] A *cyber system* is a system of interlinked computers forming part of cyberspace. More specifically, a cyber system is any combination of facilities, equipment, personnel, procedures, and communications integrated to provides cyber services. Information and communication technology (ICT) systems and cyber-physical systems (CPS) are examples of cyber systems. See, e.g., [49] for some useful definitions and discussions.

M. H. ter Beek and M. Sirjani (Eds.): COORDINATION 2022, LNCS 13271, pp. 3–14, 2022.
https://doi.org/10.1007/978-3-031-08143-9_1

designed out of the security loop [29]: we must deal with security assurance as a true socio-technical problem rather than a mere technical one, and consider cyber systems as *socio-technical systems (STSs)* with people at their hearts.

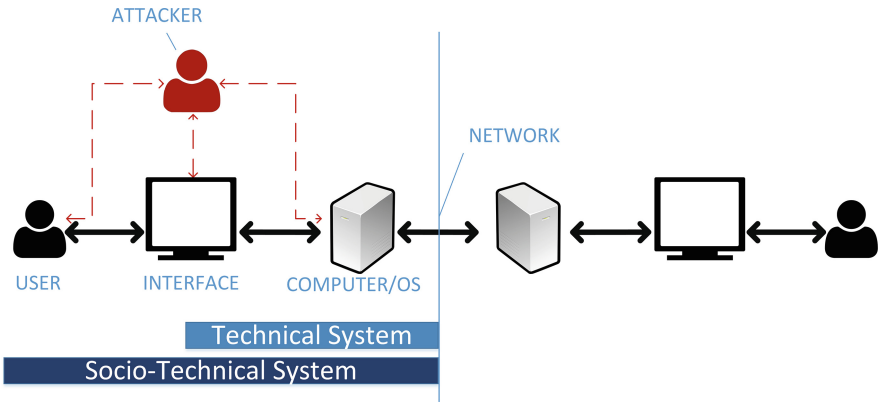

Fig. 1. Socio-technical system vs. technical system

Figure 1 shows the differences between a technical system and an STS: the technical system consists of a machine that communicates over a network with another machine, whereas the STS extends outwards, including user interfaces and actual users. The attacker can interfere in any of the communicating entities (i.e., user, user-interface, computer/OS). In fact, the security requirements of an STS don't simply derive from the system's technical layers, such as those of the OSI model, but also from the non-technical layers surrounding them. Here, humans often follow peculiar paths of practice due to diverse societal/personal reasons and to physical/social contexts wherein humans liaise with the technology. These paths may differ from those written on the system's user manual, or get consolidated out of day-to-day experience because no manual exists. We must seek to better understand how the two components of an STS, the *technical component* and the *social component* (e.g., user interaction processes and user behavior), interoperate to achieve (or not!) overall security. This requires extending the technical analysis/testing approaches with a mature understanding of human behavior. We need to develop appropriate *formal methods* that are up to this task.

Experience over the last 30+ years has namely shown that the design of systems for cyber security is highly error-prone and that conventional analysis techniques based on informal/semi-formal arguments and/or standard testing approaches fail to discover a large number of attacks. Formal methods for modeling and analysis aim at guaranteeing that a system satisfies certain properties of interest. This is typically achieved by developing a logico-mathematical model of the system (i.e., its structure and behavior) and of the desired properties, and

an analysis algorithm that checks whether the model of the system satisfies the properties.

The main goal of this short paper, which accompanies my keynote talk at the 24th International Conference on Coordination Models and Languages (COORDINATION 2022), is to advocate the use of formal methods to establish the security of socio-technical systems, and to discuss some of the most promising approaches. Please, note that, despite the general-sounding title, this paper is by no means a comprehensive and up to date survey of the state of the art in socio-technical security (such a survey would be most welcome, though, so I hope that others will be able to provide one soon). On the contrary, I will focus mainly on security ceremonies as a specific example of socio-technical systems and on the approaches that I helped develop as specific examples of formal and automated methods.

2 From Technical to Socio-technical Security

Traditionally, security has been established technically, by means of implementations of crypto algorithms, security protocols, intrusion detection systems, firewalls, dedicated hardware, etc. [15], but there has always been awareness of the human risk of *mistakes* (i.e., the user's failure to do what she intends to do), *slips* (i.e., momentary lapses that see the user take unintended actions), or *noncompliance* (i.e., the user's possibly deliberate failure to do what the system intended of the user). Until recently, this risk was mitigated by means of user manuals, but manuals have disappeared today when the technology is a computer or a mobile device exposing browsers and apps. Modern users are bound, for instance, to access their on-line bank account through a browser (but using also their smartphone in those cases in which multi-factor authentication is foreseen) without having studied, on a manual and beforehand, what to do. This contributed to inspiring research in *usable security* [30] to assess the ease with which users can learn the behavior that they are expected to take while operating security-sensitive technology. However, humans are complicated and nothing guarantees that, even if they learned how to operate a technology, either from a manual or through its use, they will comply with what they learned. Reasons include cognitive biases, fallacies, ignorance, distraction, laziness, curiosity of different uses, insufficient awareness of the security sensitivity of their behavior, etc.

Much effort has been devoted to the technical analysis of the security of cyber systems. As a concrete example, consider security protocols. A *security protocol*, sometimes also called *cryptographic protocol*, is essentially a communication protocol (an agreed sequence of actions performed by two or more communicating entities in order to accomplish some mutually desirable goals) that makes use of cryptographic techniques, allowing the communicating entities to satisfy one or more security properties (such as authentication, confidentiality of data or integrity of data).

Several formal methods and automated tools (e.g., [2,3,5,8,16,21–23,28,40,42]) have been developed to analyze security protocols to check whether they do

indeed satisfy the security properties they were designed for. These approaches rely on symbolically modeling the agents involved in the protocol along with an attacker who is trying to subvert the protocol's security. There are thus two types of agents:

- *honest agents*, who behave only according to what the protocol specifies (encrypting, decrypting and sending and receiving messages as specified by the protocol), and
- the *attacker*, a dishonest agent who can behave as he wishes, including following the protocol steps.

The attacker is often modeled using the *Dolev-Yao attacker model* [26], which allows him to send, read, encrypt and decrypt any message as long as he possesses the corresponding cryptographic keys.[2] In other words, cryptography is assumed to be perfect.[3]

Testing approaches have also been put forward to analyze security protocol implementations rather than their specifications (e.g., [25,33,44,53]). Even though these approaches are often limited in the strength of the protocols that can be considered, model-based testing approaches in which the formal analysis of a protocol specification is used to generate test cases for the protocol code have proven to be quite successful.

In contrast to formal analysis and testing of security protocols, for which a plethora of mature approaches and tools exist, socio-technical security is a discipline still in its childhood, with no widely recognized methodologies or comprehensive tools mature enough to take into full account human behavioral and cognitive aspects in their relation with "machine" security, and thus reason with the breadth and depth that is required by real-life STSs.

3 Formal and Automated Analysis of Security Ceremonies

Most of the research efforts on formal methods for socio-technical security have focused on security ceremonies as concrete, relevant, and timely examples of STSs (e.g., [6,7,9,10,12,19,20,24,31,36,38,39,45–47,50,51]).

The term *ceremony* was coined by Jesse Walker [37] to describe the interaction between a user and computing devices. The use of the term in the area of information/cyber security is due to Ellison [27]: a *security ceremony* expands a

[2] One can show that in the presence of such a powerful attacker it is enough to consider only one attacker [4], whereas in other scenarios (e.g., where movement of the agents or where different devices are considered), one might need to model attackers that have different capabilities and collaborate to carry out an attack.

[3] In addition to symbolic approaches, there are also a number of cryptographically-faithful approaches in which the perfect cryptography assumption is relaxed and the properties of the employed crypto algorithms are considered explicitly, e.g., [17,48]. I will not consider them here as the focus is on the human users of the protocols and ceremonies.

security protocol with everything that is considered out-of-band to it. More precisely: "Ceremonies include all protocols, as well as all applications with a user interface, all workflow and all provisioning scenarios" [27]. Therefore, the innovative stance of security ceremonies is to include human nodes alongside computer nodes, with communication links that comprise user interfaces, human-to-human communication and transfers of physical objects that carry data.

Security ceremonies are essentially "rituals" with finely orchestrated actions being carried out in a prescribed order by the agents involved in the ceremony. Works in the socio-technical security field typically use the term security ceremony to refer to an extension of a security protocol in which human agents and software-based agents exchange encrypted messages to achieve certain goals. Other kinds of ceremonies are key-signing ceremonies such as those required for DNSSEC [34], which are somewhat more akin to "a public or religious occasion that includes a series of formal or traditional actions" (which is one of the meanings of "ceremony" according to the Oxford English Dictionary). Another meaning is related to the secure key generation process that constitutes the initialisation phase of the wallet infrastructure and private keys in the realm of crypto-currencies [32]. In this paper, I take the socio-technical security view and focus on the first meaning.

As technology progresses in any area, human beings are increasingly surrounded by, and immersed in, such security ceremonies during their everyday lives. They carry out security tasks that occur through a virtually infinite range of scenarios interposing people's: (i) professional activities, such as logging into their employer's computer systems using two-factor authentication, (ii) business or leisure activities, such as taking a flight which involves getting through airport security, and (iii) chores, such as paying for their shopping with a debit card. As a concrete example, consider the sequence diagram shown in Fig. 2, in which a human user carries out a two-factor authentication security ceremony by interacting with an interface to exchange messages with a device and a database.

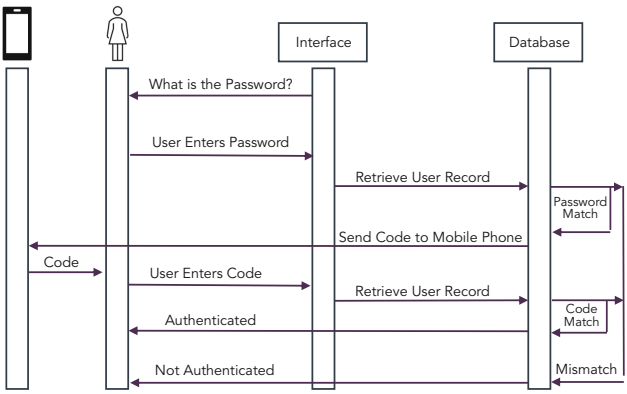

Fig. 2. A sequence diagram of a two-factor authentication security ceremony

Although many of the works cited at the beginning of this section are quite preliminary studies, and a mature and systematic approach is still missing, some formal methods and automated tools have been successfully extended to analyze a number of real-life security ceremonies. For instance, Bella and Coles-Kemp [9] defined a layered model of socio-technical protocols between a user persona and a computer interface, and used the Isabelle theorem prover to analyze a ceremony that allows Internet users to register with a service provider; Basin et al. [6,7] provided a formal account on human error in front of basic authentication properties and described how to analyze security ceremonies such as MP-Auth using the Tamarin tool; Giustolisi [31] used Tamarin to analyze the security of mobile tickets used in public transport ceremonies in Denmark; Martimiano and Martina [38] showed how a popular security ceremony for remote file sharing could be made fail-safe assuming a weaker threat model than normally considered in formal analysis and compensating for that with usability; Bella, Giustolisi and Lenzini carried out a socio-technical formal analysis of TLS certificate validation in modern browsers [10] and provided a novel protocol for secure exams that they formally analyzed using the tool ProVerif [11]; Bella, Giustolisi and Schürmann [12] used Tamarin to analyze two deposit-return systems currently deployed in Denmark and a variant that they designed to strengthen them.

As I mentioned above, the vast majority of the approaches for the formal analysis of security protocols adopt the Dolev-Yao attacker model, and most of the works on the formal analysis of security ceremonies extend this model to capture different socio-technical aspects. However, Sempreboni, Bella, Giustolisi and I remarked in [51] that in the case of security ceremonies such an attacker provides an inherent "flattening" that likely makes one miss relevant threat scenarios, and thus we advocated that for security ceremonies we need an approach that provides a birds-eye view, an "overview" that allows one to consider what are the different threats and where they lie, with the ultimate aim of finding novel attacks. Our main contribution in [51] is the systematic definition of an encompassing method to build the *full threat model chart* for security ceremonies from which one can conveniently reify the threat models of interest for the ceremony under consideration. To demonstrate the relevance of the chart, we formalized this threat model using Tamarin and analyzed three real-life ceremonies that had already been considered, albeit at different levels of detail and analysis, in the literature: MP-Auth [7], Opera Mini [47], and the Danish Mobilpendlerkort ceremony [31]. The full threat model chart suggested some interesting threats that hadn't been investigated although they are well worth of scrutiny. In particular, we found out that the Danish Mobilpendlerkort ceremony is vulnerable to the combination of an attacking third party and a malicious phone of the ticket holder. The threat model that leads to this vulnerability had not been considered before and arose thanks to our charting method.

One of most promising approaches for the formal and automated analysis of security ceremonies is the *mutation-based approach* that Sempreboni and I proposed in [52], which allows security analysts to model possible mistakes by human users as mutations with respect to the behavior that the ceremony origi-

nally specified for such users. In security ceremonies humans are (and should be considered to be) first-class actors, so it is not enough to take the "black&white" view of security protocol analysis, in which there is a Dolev-Yao attacker (the black agent) against a set of honest agents (the white agents). It is not enough to model human users as "honest processes" or as attackers, because they are neither. Modeling a person's behavior is not simple and requires formalizing the human "shades of gray". It requires modeling the way humans interact with the protocols, their behavior and the mistakes they may make, independent of attacks and, in fact, possibly independent of the presence of an attacker. In other words, when considering the mistakes that human users might make when interacting in a security ceremony, attacks may occur even without the presence of an attacker.

In [52] and the journal version that we are currently working on, we formalize four main human mutations of a ceremony:

- skipping one or more of the actions that the ceremony expects the human user to carry out (such as sending or receiving a message),
- adding an action (e.g., sending a message twice),
- replacing a message with another one (e.g., forgetting to include some information in a message and thus sending a shorter message than expected, or sending an altogether different message),
- neglecting to adhere to one or more inner behaviors expected by the ceremony (e.g., neglecting to carry out a check on the contents of a message).

Given a specification of a ceremony and its goals, our approach generates a mutated specification that models possible behaviors of a human user, along with mutations in the behavior of the other agents of the ceremony to match the human-induced mutations. This allows for the analysis of the original ceremony specification and its possible mutations, which may include the way in which the ceremony has actually been implemented. We have automated our approach by implementing a tool called *X-Men*, which builds on top of Tamarin.[4] As a proof of concept, we have applied our approach to three real-life case studies, uncovering a number of concrete vulnerabilities.

Similar to what happens for the formal analysis of security protocols, if the tool terminates[5] and verifies the model under consideration, then we can provide a security guarantee. Our approach allows us to provide such guarantee not only for the original ceremony but also for its mutations for which Tamarin's analysis terminates with a proof.

To some extent, this applies also when X-Men's/Tamarin's analysis of a security ceremony times out without a proof or without having discovered an attack, since, if the timeout is large enough, it can provide some degree of guarantee that an attack is unlikely. Still, an attack might be possible: even if a tool is

[4] The name X-Men was chosen to suggest that it considers human mutations.

[5] The analysis of security protocols in the presence of an active attacker is an undecidable problem, so protocol analysis tools might not terminate, unless one introduces some restrictions and bounds to force the analysis to complete.

not able to find an attack in, say, 10 min, nothing guarantees that the attack would not have been discovered if only one had allowed the tool to run for a couple more minutes. To tame the complexity of the search for an attack in case of security ceremonies and their mutations, we will consider adapting *compositionality* results like those of [1,41,43], which identify conditions that allow one to split a complex composed protocol into its subprotocols that can be analyzed independently with the guarantee that also their composition is secure.

If the analysis of a security protocol model instead terminates with the discovery of an attack, typically the attack trace can be used to distill a fix to the protocol specification; one would also usually wish to check whether the attack on the model also applies for the concrete implementation (assuming that it is available), so the attack trace can also be used to devise test cases for the implementation (see, e.g., [33,44,53]). The same applies in the case of our mutated ceremonies and we plan to extend X-Men to generate test cases similar to mutation-based testing [18,25].

Given the presence of human users, one can also aim to use the attack trace to distill recommendations and guidelines for the users of the ceremony so that they interact with it in a way that does not endanger security. The rules of [7] that restrict how the human can deviate from the protocol specification are a good example for such guidelines. In future work, we aim to investigate if and how recommendations and guidelines could be generated (semi-)automatically from the analysis of security ceremonies and their mutations (similar to the generation of test cases), and how they could be communicated to human users in an effective way. To that end, we plan to exploit also our works on how to provide security explanations to laypersons [54–58] and on how to beautify security ceremonies [13,14] and thus make their secure use more appealing to human users.

Acknowledgments. Thanks to Giampaolo Bella, Rosario Giustolisi, Jacques Ophoff, Karen Renaud, Diego Sempreboni for their invaluable contributions to our joint works on socio-technical security. Thanks also to Lynne Coventry and Gabriele Lenzini for many interesting discussions. I acknowledge funding from the UKRI Trustworthy Autonomous Systems Hub (EP/V00784X/1).

References

1. Almousa, O., Mödersheim, S., Modesti, P., Viganò, L.: Typing and compositionality for security protocols: a generalization to the geometric fragment. In: Pernul, G., Ryan, P.Y.A., Weippl, E. (eds.) ESORICS 2015. LNCS, vol. 9327, pp. 209–229. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24177-7_11
2. Armando, A., et al.: The AVANTSSAR platform for the automated validation of trust and security of service-oriented architectures. In: Flanagan, C., König, B. (eds.) TACAS 2012. LNCS, vol. 7214, pp. 267–282. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28756-5_19

3. Armando, A., et al.: The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications. In: Etessami, K., Rajamani, S.K. (eds.) CAV 2005. LNCS, vol. 3576, pp. 281–285. Springer, Heidelberg (2005). https://doi.org/10.1007/11513988_27
4. Basin, D.A., Caleiro, C., Ramos, J., Viganò, L.: Distributed temporal logic for the analysis of security protocol models. Theor. Comput. Sci. **412**(31), 4007–4043 (2011). https://doi.org/10.1016/j.tcs.2011.04.006
5. Basin, D.A., Cremers, C., Meadows, C.: Model checking security protocols. In: Handbook of Model Checking, pp. 727–762. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-10575-8_22
6. Basin, D.A., Radomirović, S., Schläpfer, M.: A complete characterization of secure human-server communication. In: Proceedings of the 28th IEEE Computer Security Foundations Symposium (CSF 2015), pp. 199–213. IEEE (2015). https://doi.org/10.1109/CSF.2015.21
7. Basin, D.A., Radomirović, S., Schmid, L.: Modeling human errors in security protocols. In: Proceedings of the 29th IEEE Computer Security Foundations Symposium (CSF 2016), pp. 325–340. IEEE (2016). https://doi.org/10.1109/CSF.2016.30
8. Bella, G.: Formal Correctness of Security Protocols. Springer, Berlin (2007). https://doi.org/10.1007/978-3-540-68136-6
9. Bella, G., Coles-Kemp, L.: Layered analysis of security ceremonies. In: Gritzalis, D., Furnell, S., Theoharidou, M. (eds.) SEC 2012. IAICT, vol. 376, pp. 273–286. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-30436-1_23
10. Bella, G., Giustolisi, R., Lenzini, G.: Socio-technical formal analysis of TLS certificate validation in modern browsers. In: Castella-Roca, J. et al. (ed.), Proceedings of the 11th International Conference on Privacy, Security and Trust (PST 2013), pp. 309–316. IEEE Press (2013). https://doi.org/10.1109/PST.2013.6596067
11. Bella, G., Giustolisi, R., Lenzini, G.: Invalid certificates in modern browsers: a socio-technical analysis. J. Comput. Secur. **26**(4), 509–541 (2015). https://doi.org/10.3233/JCS-16891
12. Bella, G., Giustolisi, R., Schürmann, C.: Modelling human threats in security ceremonies. J. Comput. Secur. (2022, to appear)
13. Bella, G., Renaud, K., Sempreboni, D., Viganò, L.: An investigation into the "beautification" of security ceremonies. In: Proceedings of the 16th International Conference on Security and Cryptography, pp. 125–136. Scitepress Digital Library (2019). https://doi.org/10.5220/0007921501250136
14. Bella, G., Viganò, L.: Security is beautiful. In: Christianson, B., Švenda, P., Matyáš, V., Malcolm, J., Stajano, F., Anderson, J. (eds.) Security Protocols 2015. LNCS, vol. 9379, pp. 247–250. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-26096-9_25
15. Bishop, M.: Computer Security: Art and Science. 2d edition, Addison-Wesley Professional, Boston (2019)
16. Blanchet, B.: An efficient cryptographic protocol verifier based on prolog rules. In: Proeedings of the IEEE Computer Society Foundations Workshop (CSFW 2001), IEEE CS Press (2001). https://doi.org/10.1109/CSFW.2001.930138
17. Blanchet, B.: A computationally sound mechanized prover for security protocols. In: Proceedings of the IEEE Symposium on Security and Privacy, pp. 140–154. IEEE (2006). https://doi.org/10.1109/SP.2006.1
18. Büchler, M., Oudinet, J., Pretschner, A.: Security mutants for property-based testing. In: Gogolla, M., Wolff, B. (eds.) TAP 2011. LNCS, vol. 6706, pp. 69–77. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21768-5_6

19. Carlos, M.C., Martina, J.E., Price, G., Custódio, R.F.: A proposed framework for analysing security ceremonies. In: Proceedings of the International Conference on Security and Cryptography - Volume 1: SECRYPT (ICETE 2012), pp. 440–445. INSTICC, Scitepress Digital Library (2012). https://doi.org/10.5220/0004129704400445
20. Carlos, M.C., Martina, J.E., Price, G., Custódio, R.F.: An updated threat model for security ceremonies. In: Proceedings of SAC 2013, pp. 1836–1845. ACM (2013). https://doi.org/10.1145/2480362.2480705
21. Cortier, V., Kremer, S.: Formal models and techniques for analyzing security protocols: a tutorial. Found. Trends Program. Lang. **1**(3), 151–267 (2014). https://doi.org/10.1561/2500000001
22. Cortier, V., Kremer, S.: Formal models for analyzing security protocols: some lecture notes. In: Dependable Software Systems Engineering, volume 45 of NATO Science for Peace and Security Series – D: Information and Communication Security, pp. 33–58. IOS Press (2016). https://doi.org/10.3233/978-1-61499-627-9-33
23. Cremers, C.J.F.: The Scyther tool: verification, falsification, and analysis of security protocols. In: Gupta, A., Malik, S. (eds.) CAV 2008. LNCS, vol. 5123, pp. 414–418. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-70545-1_38
24. Curzon, P., Rukšėnas, R., Blandford, A.: An approach to formal verification of human-computer interaction. Formal Aspects Comput. **19**(4), 513–550 (2007). https://doi.org/10.1007/s00165-007-0035-6
25. Dadeau, F., Héam, P.-C., Kheddam, R., Maatoug, G., Rusinowitch, M.: Model-based mutation testing from security protocols in HLPSL. Softw. Test. Verification Reliab. **25**(5–7), 684–711 (2015). https://doi.org/10.1002/stvr.1531
26. Dolev, D., Yao, A.: On the security of public-key protocols. IEEE Trans. Inf. Theor. **29**(2), 198–208 (1983). https://doi.org/10.1109/TIT.1983.1056650
27. Ellison, C.M.: Ceremony design and analysis. IACR Cryptology ePrint Archive **399**, 1–17 (2007)
28. Escobar, S., Meadows, C., Meseguer, J.: Maude-NPA: cryptographic protocol analysis modulo equational properties. In: Aldini, A., Barthe, G., Gorrieri, R. (eds.) FOSAD 2007-2009. LNCS, vol. 5705, pp. 1–50. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03829-7_1
29. Flechais, I., Riegelsberger, J., Sasse, M.A.: Divide and conquer: the role of trust and assurance in the design of secure socio-technical systems. In: Proceedings of the 2005 Workshop on New Security Paradigms (NSPW), pp. 33–41. ACM (2005). https://doi.org/10.1145/1146269.1146280
30. Garfinkel, S., Lipford, H.R.: Usable Security: History, Themes, and Challenges. Morgan & Claypool, San Rafael (2014)
31. Giustolisi, R.: Free rides in Denmark: lessons from improperly generated mobile transport tickets. In: Lipmaa, H., Mitrokotsa, A., Matulevičius, R. (eds.) NordSec 2017. LNCS, vol. 10674, pp. 159–174. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70290-2_10
32. CVA Cybersecurity Working Group. Trusted Key Ceremony Guidelines – Guidelines for Generating Digital Asset Secrets (2020). https://d1c2gz5q23tkk0.cloudfront.net/assets/uploads/2956620/asset/CVA_Trusted_Key_Ceremony_Guidelines.pdf?1594131972
33. Pierre-Cyrille, H., Dadeau, F., Kheddam, R., Maatoug, G., Rusinowitch, M.: A model-based testing approach for security protocols. In: Proceedings of the 2016 IEEE International Conference on Computational Science and Engineering (CSE)

and IEEE International Conference on Embedded and Ubiquitous Computing (EUC) and 15th International Symposium on Distributed Computing and Applications for Business Engineering (DCABES), pp. 553–556. IEEE (2016). https://doi.org/10.1109/CSE-EUC-DCABES.2016.240
34. Internet Assigned Numbers Authority (IANA). Key Signing Ceremonies (2022). https://www.iana.org/dnssec/ceremonies
35. IBM Global Technology Services (Managed Security Services). IBM Security Services 2014 Cyber Security Intelligence Index (2014)
36. Johansen, C., Jøsang, A.: Probabilistic modelling of humans in security ceremonies. In: Garcia-Alfaro, J., et al. (eds.) DPM/QASA/SETOP -2014. LNCS, vol. 8872, pp. 277–292. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-17016-9_18
37. Lortz, V.B., Walker, J.R., Hegde, S.S., Kulkarni, A.A., Tai, T.Y.C.: Device introduction and access control framework, Google Patents US Patent 8,146,142 (2012)
38. Martimiano, T., Martina, J.E.: Daemones non operantur nisi per artem. In: Matyáš, V., Švenda, P., Stajano, F., Christianson, B., Anderson, J. (eds.) Security Protocols 2018. LNCS, vol. 11286, pp. 96–105. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-03251-7_11
39. Martimiano, T., Martina, J.E., Olembo, M.M., Carlos, M.C.: Modelling user devices in security ceremonies. In: Proceedings of the Workshop on Socio-Technical Aspects in Security and Trust, pp. 16–23 (2014). https://doi.org/10.1109/STAST.2014.11
40. Meier, S., Schmidt, B., Cremers, C., Basin, D.: The TAMARIN prover for the symbolic analysis of security protocols. In: Sharygina, N., Veith, H. (eds.) CAV 2013. LNCS, vol. 8044, pp. 696–701. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39799-8_48
41. Mödersheim, S., Viganò, L.: Secure pseudonymous channels. In: Backes, M., Ning, P. (eds.) ESORICS 2009. LNCS, vol. 5789, pp. 337–354. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04444-1_21
42. Mödersheim, S., Viganò, L.: The open-source fixed-point model checker for symbolic analysis of security protocols. In: Aldini, A., Barthe, G., Gorrieri, R. (eds.) FOSAD 2007-2009. LNCS, vol. 5705, pp. 166–194. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03829-7_6
43. Mödersheim, S., Viganò, L.: Sufficient conditions for vertical composition of security protocols. In: Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security (ASIA CCS 2014), pp. 435–446. ACM (2014). https://doi.org/10.1145/2590296.2590330
44. Peroli, M., De Meo, F., Viganò, L., Guardini, D.: MobSTer: a model-based security testing framework for web applications. Softw. Test. Verification Reliab. **28**(8), e1685 (2018). https://doi.org/10.1002/stvr.1685
45. Probst, C.W., Kammüller, F., Hansen, R.R.: Formal modelling and analysis of socio-technical systems. In: Probst, C.W., Hankin, C., Hansen, R.R. (eds.) Semantics, Logics, and Calculi. LNCS, vol. 9560, pp. 54–73. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-27810-0_3
46. Radke, K., Boyd, C.: Security proofs for protocols involving humans. Comput. J. **60**(4), 527–540 (2017). https://doi.org/10.1093/comjnl/bxw066
47. Radke, K., Boyd, C., Gonzalez Nieto, J., Brereton, M.: Ceremony analysis: strengths and weaknesses. In: Camenisch, J., Fischer-Hübner, S., Murayama, Y., Portmann, A., Rieder, C. (eds.) SEC 2011. IAICT, vol. 354, pp. 104–115. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21424-0_9

48. Ramsdell, J.D.: Cryptographic protocol analysis and compilation using CPSA and Roletran. In: Dougherty, D., Meseguer, J., Mödersheim, S.A., Rowe, P. (eds.) Protocols, Strands, and Logic. LNCS, vol. 13066, pp. 355–369. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-91631-2_20
49. Refsdal, A., Solhaug, B., Stølen, K.: Cyber-risk management. In: Cyber-Risk Management. SCS, pp. 33–47. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23570-7_5
50. Rukšėnas, R., Curzon, P., Blandford, A.: Modelling and analysing cognitive causes of security breaches. Innov. Syst. Softw. Eng. **4**, 143–160 (2008). https://doi.org/10.1007/s11334-008-0050-7
51. Sempreboni, D., Bella, G., Giustolisi, R., Viganò, L.: What are the threats? (Charting the threat models of security ceremonies). In: Proceedings of the 16th International Joint Conference on e-Business and Telecommunications, ICETE – Volume 2: SECRYPT, pp. 161–172. Scitepress Digital Library (2019). https://doi.org/10.5220/0007924901610172
52. Sempreboni, D., Viganò, L.: X-Men: a mutation-based approach for the formal analysis of security ceremonies. In: Proceedings of the 5th IEEE European Symposium on Security and Privacy (EuroS&P), pp. 87–104. IEEE (2020). https://doi.org/10.1109/EuroSP48549.2020.00014
53. Viganò, L.: The SPaCIoS Project: Secure Provision and Consumption in the Internet of Services. In: Proceedings of the IEEE Sixth International Conference on Software Testing, Verification and Validation (ICST), pp. 497–498. IEEE (2013). https://doi.org/10.1109/ICST.2013.75
54. Viganò, L.: Explaining cybersecurity with films and the arts. In: Imagine Math 7, pp. 297–309. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-42653-8_18
55. Viganò, L.: Nicolas Cage is the center of the cybersecurity universe. In: Ardito, C., et al. (eds.) INTERACT 2021. LNCS, vol. 12932, pp. 14–33. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-85623-6_3
56. Viganò, L.: Don't tell me the cybersecurity moon is shining... (Cybersecurity show and tell). In: Emmer, M. (ed.), Imagine Math 8. Springer, Cham (to appear)
57. Viganò, L., Magazzeni, D.: Explainable security. CoRR (2018). http://arxiv.org/abs/1807.04178
58. Viganò, L., Magazzeni, D.: Explainable security. In: IEEE European Symposium on Security and Privacy Workshops, EuroS&P Workshops 2020, pp. 293–300. IEEE (2020). A preliminary version appeared as [58]. https://doi.org/10.1109/EuroSPW51379.2020.00045