Symbolic Abstractions for Quantum Protocol Verification

Lucca Hirschi[1] and David Basin[2]

[1]Inria & LORIA, France*
[2]ETH Zurich, Switzerland

Quantum protocols such as the BB84 Quantum Key Distribution (QKD) protocol [7] or its bit commitment variant [11] exchange qubits to achieve extremely strong security guarantees. Namely, quantum physics ensures that observing the exchanged qubits disturbs the observed data. This can be exploited by the parties involved to effectively detect the presence of an eavesdropper, which in turn can be leveraged to achieve information-theoretic security guarantees.

Many different quantum protocols, some already deployed and commercialized[1], essentially rely on the same mechanisms and can be seen as variants of the BB84 QKD protocol (*e.g.,* see [13]). For instance, the Quantum Bit-Commitment (QBC) variant of BB84 [11] was proposed in 1990 and was believed to be secure (see [12]) until it was shown to be flawed 7 years later [20]. The QKD BB84 protocol, however, is still believed to be secure and various pen-and-paper proofs of unconditional security exist [27]. However, such proofs often hold for the core protocol only and rarely account for other crucial components such as authentication or authorization, [26] being a notable exception. Furthermore, we stress that in such complex settings, where intruders have different capabilities that can be combined in infinitely many ways, manual proofs are highly error-prone. Finally, numerous variants of such protocols have been given or will be proposed in the future, making tedious and error-prone manual proofs impractical.

The proliferation of protocols and the complexity of their proofs call for formal and automated verification techniques that exhaustively explore all possible intruder behaviors. Automation usually comes at the cost of approximations and less precise security guarantees. In this paper, we explore this trade-off.

Automated Formal Verification. *Formal methods* offer mathematical frameworks to analyze security protocols. Two main approaches have emerged to provide mathematical foundations for this analysis, starting with the seminal works of [17, 18]: the *computational approach* and the *symbolic approach.*

The computational approach is based on the *standard model*: messages are modeled as bit strings, and agents and the attacker as probabilistic polynomial time Turing machines. Security goals are then defined using games played by the attacker and proofs are usually done via reductions (or hops) between successive games until reaching games expressing computational assumptions on cryptographic primitives. It is generally acknowledged that security proofs in this model offer strong security guarantees. However, a serious downside of this approach is that even for small protocols, the proofs are usually difficult, tedious, and error prone. Moreover, due to the high complexity of this model, automating such proofs is a difficult problem that is still in its infancy. More generally, computer-aided verification allows for only a low level of automation, even though considerable efforts have been put in developing verifiers such as CertiCrypt [4], EasyCrypt [3], FCF [8, 25], and F*[2].

In contrast to the computational approach, the *symbolic approach* is used when one is interested in analyzing in a reasonable time more complex protocols, rather than simple primitives

*This work was partially done while Lucca Hirschi was at ETH Zurich, Switzerland.

[1] See for example `www.idquantique.com`.

or core protocols. This model is more abstract and scales better. In particular, it makes strong assumptions on cryptographic primitives (*i.e.,* the perfect cryptography assumption) but fully models algebraic properties of these primitives as well as the protocol agents' interactions. Modeling security protocols using the symbolic approach allows one to benefit from machine support using established techniques, such as model-checking, resolution, and rewriting techniques. From the different lines of work in this area there have emerged verification tools (*e.g.,* TAMARIN [22], PROVERIF [10], DEEPSEC [14]) and large-scale formal analyses of real-life protocols (*e.g.,* TLS 1.3 [16, 9, 15], mobile telephony protocols [5]).

Hence a natural question is: *Can we use this successful line of work to analyze quantum protocols?* This paper proposes a first positive answer and motivates further research on this unexplored path.

State-of-the-Art. In the standard model, [28] proposes an extension of the pRHL logic (pRHL is used by EasyCrypt [3] and CertiCrypt [4]) to handle quantum protocols and post-quantum cryptography schemes. They provide a tool that produces machine-checked security proofs. While such techniques aim at establishing extremely strong security guarantees, they inherit the complexity of the computational model and its low level of automation. In this paper, we focus on automated verification techniques, even if this means providing weaker guarantees.

Other research explores the use of model checkers for probabilistic distributed programs. [24] models and analyzes the BB84 QKD protocol using the probabilistic model checker PRISM [19]. While such analyses can quantify the probability for the intruder to be detected or to learn the key, this approach does not consider a fully adversarial environment but rather considers a fixed, limited intruder behavior; namely a receive-resend behavior. Abstracting away probabilities, prior works have also used non-probabilistic model-checking tools for distributed programs. [23] verifies, in the presence of a fixed intercept-resend attacker, that the BB84 QKD protocol is trace equivalent to its specification. In contrast, [6] is interested in verifying safety properties in a non-adversarial environment, expressed in epistemic logic, using model-checking for multi-agents systems. Similarly, [1] proposes a verification technique for checking equivalence between quantum protocols. None of these works considers a fully adversarial environment and they all fall short of capturing other potential cryptographic components upon which the core quantum protocol is based (*e.g.,* authentication).

Symbolic models have been successfully used in the past to model classical security protocols in a fully adversarial setting. They cannot however be used off-the-shelf to analyze quantum protocols. The main features thereof that are not handled by classical techniques are: the intruder's quantum capabilities that are limited by quantum physics laws (*e.g.,* no-cloning, measurement) and the protocol logic conditioned by probabilities, and security parameters.

Our Contributions [21]. We formally define a novel extension of the classical Dolev-Yao intruder accounting for some quantum physics capabilities and extending the intruder's control to the quantum channels. Our attacker can read qubits, produce new qubits, and produce entangled qubits in order to perform EPR-attacks. However, his capabilities are restricted by standard Quantum Physics principles, for example the no-cloning theorem and the Heisenberg uncertainty principle. Hence, our framework accounts for a fully adversarial environment with regard to a rich class of intruder capabilities, as opposed to a fixed, trivial intruder strategy. However, due to known limitations of the symbolic model, our extension does not capture probabilistic attacks in their full generality. Still, our extension does capture *a class* of probabilistic attacks by allowing the attacker to guess a fix amount of the bits that are randomly chosen by honest parties.

We show how our quantum Dolev-Yao attacker can be embedded in some classical verification tools for cryptographic protocols such as TAMARIN, PROVERIF, and DEEPSEC using involved but generic encodings. We also show the practical relevance of our approach by presenting case studies. We analyze with TAMARIN key secrecy for one session of the BB84 QKD protocol for different threat models and automatically identify minimal assumptions in terms of message authenticity and the intruder's capabilities. We also automatically find several attacks, some of which were well-known and others that rely on minimal security assumptions that were not previously clearly identified. For instance, when sufficiently many verification bits are checked, we show that the authenticity of three specific classical messages out of four is a minimal requirement. Namely, we show attacks when this is violated for one of the three messages and provide a proof in our model otherwise. In particular, we automatically found that when verification bits are not necessarily authentic, an EPR attack completely breaks secrecy as the intruder can learn all the bits measured by Bob. We also model the BB84 QBC protocol and automatically re-discover the EPR-based binding attack that completely defeats the protocol purpose.

Since our framework is based on well-established frameworks and verifiers, it can handle complex cryptographic systems containing a quantum protocol at its core. This can theoretically be leveraged to assess the security of a whole security system as specified, or deployed, instead of a single quantum component in isolation.

Our Approach and its Abstractions. To achieve the above, we adopt the following abstractions and make the following modeling choices.

Based on meta-level probabilistic reasoning, we consider fixed bitstrings instead of random bitstrings. We thus deal with *possibilities* rather than with probabilities. However, even though the bitstrings under consideration are fixed, we explore all potential executions that are possible for such bitstrings. We believe that, by wisely choosing these bitstrings, we can capture a wide range of logical attacks. However, fixing these bitstrings naively would allow the intruder to perform bitstring-dependent attacks, which would be successful for the real system with only negligible probability. We thus make the data of the fixed bitstrings (i) initially secret, and (ii) partially guessable by the intruder. The amount of data that can be guessed in our model is defined through meta-level probabilistic reasoning. We also identify and mitigate an “intruder's knowledge propagation effect”: when the intruder correctly guesses a bit b picked by Alice, he automatically learns all other bits equal to *b for the fixed bitstrings* under consideration. This is taken care of by modeling all elements of the bitstrings by different terms and by adapting the equality and inequality relations accordingly. While we will miss out many probabilistic attacks this way, we believe this is analogous to the gap between the symbolic and the computational approach for classical cryptography. We recall that the goal here is to capture some *logical attacks* that can be performed with non-negligible probabilities. In short, we balance trade-offs differently: we provide weaker guarantees in exchange for automation.

We build on the symbolic model to model a quantum channel and a quantum intruder. We model qubits resulting from encoding of bits in orthonormal bases as an uninterpreted function over the bit and the base. Our viewpoint is that, analogous to classical channels in the standard symbolic model, quantum channels should be considered to be entirely under the intruder's control. We thus model a quantum channel where all outputs are given to the intruder and all inputs are chosen by the intruder. Contrary to the computational model that defines what the intruder *cannot do*, a symbolic model explicitly specifies what the intruder *can do*. We thus choose a fixed, yet rich, set of quantum intruder capabilities: our intruder can forward, transform, measure, and forge qubits. Those capabilities are restricted according to relevant physics laws. For example, a qubit is *consumed* (and cannot be reused) upon measurement or forwarding, measurement can yield random data (wrong basis) or the encoded bit (matching basis), and measurement by honest parties sometimes leaks data (*e.g.,* in EPR-attacks).

Finally, note that our framework is not complete (attacks may be missed) with regard to Nature. This is to be expected as we only model some intruder capabilities and we then abstract them away. More surprisingly, our framework does not provide sound falsifications (*i.e.,* no false attack) with regard to Nature either. This stems from our abstractions of random bitstrings, which might lead to attack scenarios that only happen with negligible probabilities in reality. While we took efforts to mitigate this, soundness of falsification does not currently hold and we do not see how it could with our current modeling choices. Hence, our abstractions related to quantum capabilities are attack preserving, while the ones abstracting away probabilities are not.

References

[1] E. Ardeshir-Larijani, S. J. Gay, and R. Nagarajan. Verification of concurrent quantum protocols by equivalence checking. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 500–514. Springer, 2014.

[2] G. Barthe, C. Fournet, B. Grégoire, P.-Y. Strub, N. Swamy, and S. Zanella-Béguelin. Probabilistic relational verification for cryptographic implementations. In *Proc. 41st Symposium on Principles of Programming Languages*, volume 49, pages 193–206. ACM, 2014.

[3] G. Barthe, B. Grégoire, S. Heraud, and S. Z. Béguelin. Computer-aided security proofs for the working cryptographer. In *Advances in Cryptology–CRYPTO 2011*, pages 71–90. Springer, 2011.

[4] G. Barthe, B. Grégoire, and S. Zanella Béguelin. Formal certification of code-based cryptographic proofs. *ACM SIGPLAN Notices*, 44(1):90–101, 2009.

[5] D. Basin, J. Dreier, L. Hirschi, S. Radomirović, R. Sasse, and V. Stettler. Formal Analysis of 5G Authentication. *ArXiv e-prints*, June 2018.

[6] F. Belardinelli, P. Gonzalez, and A. Lomuscio. Automated verification of quantum protocols using mcmas. *arXiv preprint arXiv:1207.1271*, 2012.

[7] H. Bennett Ch and G. Brassard. Quantum cryptography: public key distribution and coin tossing int. In *Conf. on Computers, Systems and Signal Processing (Bangalore, India, Dec. 1984)*, pages 175–9, 1984.

[8] L. Beringer, A. Petcher, Q. Y. Katherine, and A. W. Appel. Verified correctness and security of OpenSSL HMAC. In *24th USENIX Security Symposium*, pages 207–221, 2015.

[9] K. Bhargavan, B. Blanchet, and N. Kobeissi. Verified models and reference implementations for the tls 1.3 standard candidate. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 483–502, May 2017.

[10] B. Blanchet. Modeling and verifying security protocols with the applied pi calculus and ProVerif. *Foundations and Trends in Privacy and Security*, 1(1–2):1–135, Oct. 2016.

[11] G. Brassard and C. Crépeau. Quantum bit commitment and coin tossing protocols. In *Conference on the Theory and Application of Cryptography*, pages 49–61. Springer, 1990.

[12] G. Brassard, C. Crépeau, R. Jozsa, and D. Langlois. A quantum bit commitment scheme provably unbreakable by both parties. In *Foundations of Computer Science, 1993. Proceedings., 34th Annual Symposium on*, pages 362–371. IEEE, 1993.

[13] D. Bruss, G. Erdélyi, T. Meyer, T. Riege, and J. Rothe. Quantum cryptography: A survey. *ACM Computing Surveys (CSUR)*, 39(2):6, 2007.

[14] V. Cheval, S. Kremer, and I. Rakotonirina. Deepsec: Deciding equivalence properties in security protocols - theory and practice. In *Proceedings of the 39th IEEE Symposium on Security and Privacy (S&P'18)*, San Francisco, CA, USA, May 2018. IEEE Computer Society Press. Accepted for publication.

[15] C. Cremers, M. Horvat, J. Hoyland, S. Scott, and T. van der Merwe. A comprehensive symbolic analysis of TLS 1.3. In *ACM CCS 2017: Proceedings of the 24th ACM Conference on Computer and Communications Security, Dallas, USA, 2017.*, 2017. To appear.

[16] C. Cremers, M. Horvat, S. Scott, and T. van der Merwe. Automated analysis and verification of TLS 1.3: 0-RTT, resumption and delayed authentication. In *IEEE Symposium on Security and Privacy*, 2016.

[17] D. Dolev and A. Yao. On the security of public key protocols. *IEEE Transactions on information theory*, 29(2):198–208, 1983.

[18] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270 – 299, 1984.

[19] M. Kwiatkowska, G. Norman, and R. Segala. Automated verification of a randomized distributed consensus protocol using cadence smv and prism? In *International Conference on Computer Aided Verification*, pages 194–206. Springer, 2001.

[20] H.-K. Lo and H. F. Chau. Is quantum bit commitment really possible? *Physical Review Letters*, 78(17):3410, 1997.

[21] Lucca Hirschi. Quantum Dolev-Yao Attacker for and Verification of Quantum Protocols. *arXiv preprint arXiv:1904.04186*, 2019. URL: `https://arxiv.org/abs/1904.04186`.

[22] S. Meier, B. Schmidt, C. J. F. Cremers, and D. Basin. The TAMARIN Prover for the Symbolic Analysis of Security Protocols. In *CAV*, volume 8044 of *LNCS*, pages 696–701. Springer, 2013.

[23] R. Nagarajan and S. Gay. Formal verification of quantum protocols. *arXiv preprint quant-ph/0203086*, 2002.

[24] R. Nagarajan, N. Papanikolaou, G. Bowen, and S. Gay. An automated analysis of the security of quantum key distribution. *arXiv preprint cs/0502048*, 2005.

[25] A. Petcher and G. Morrisett. The foundational cryptography framework. In *Principles of Security and Trust*, pages 53–72. Springer, 2015.

[26] C. Portmann and R. Renner. Cryptographic security of quantum key distribution. *arXiv preprint arXiv:1409.3525*, 2014.

[27] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Dušek, N. Lütkenhaus, and M. Peev. The security of practical quantum key distribution. *Reviews of modern physics*, 81(3):1301, 2009.

[28] D. Unruh. Quantum relational hoare logic. *Proceedings of the ACM on Programming Languages (POPL)*, 2019.