On Higher-Order Cryptography

Ugo Dal Lago
Università di Bologna & INRIA Sophia Anitpolis

In modern cryptography, computation is necessarily randomised, and being able to restrict the adversary's time complexity is crucial: most modern cryptographic schemes are *insecure* against computationally unbounded adversaries. Noticeably, higher-order constructions are often considered in cryptography, in particular when modelling active adversaries, which have access to oracles for the underlying encryption, decryption, or authentication functions, and can thus naturally be seen as second-order functions. Another example of useful cryptographic constructions which can be spelled out at different orders are *pseudorandom* primitives. Indeed, pseudorandomness can be formulated on (families of) strings, giving rise to so-called pseudorandom *generators*, but also on (families of) first-order functions on strings, giving rise to so-called pseudorandom *functions*.

But how about making cryptographic schemes themselves higher-order? We will give an answer to this question, by first describing *why* higher-order cryptography is interesting as an object of study, then showing *how* the concept of probabilistic polynomial time algorithm can be generalised so as to encompass algorithms of order strictly higher than two, and finally proving some positive and negative results about the existence of higher-order cryptographic primitives, namely authentication schemes and pseudorandom functions.

Noticeably, the notion of a feasible functional we consider turns out to be considerably different from the one(s) in the literature. On the one hand, the functions we are interested in analysing are *randomised*. On the other hand, the notion of feasibility cryptography relies on bounds runtimes based *on the security parameter*, a global numerical value which controls the complexity of all the involved parties.

This is based on a joint work with Boaz Barak (from Harvard University) and Raphaëlle Crubillé (from LORIA, Nancy).