Secure two-party computation in the preprocessing model

Supervisor: R. de Haan (CWI) - R.de.Haan@cwi.nl

Suppose that some parties want to jointly compute the output of a given function on their respective inputs, while keeping these inputs as private as possible. An example of this is obtaining statistical information from multiple secret databases without exchanging the data. An ideal solution would be modelled by a magic box that privately takes all the required inputs, computes and reveals the desired output to the parties and then forgets everything.

The whole point of the cryptologic area of secure multi-party computation is to securely *simulate* the behaviour of such a box. While this of course involves the use of encryption schemes, various dedicated techniques are additionally required.

An important class of solutions to the problem of secure multi-party computation, introduced by Beaver [1] in 1991, is based on dividing the computation into an *offline* and a subsequent *online* phase. The purpose of this is to push as much work as possible into the offline phase, which is independent of both the function to be computed and the inputs of the parties (except perhaps for some bounds on the complexity of the function).

Recently, Beaver's idea has been adapted to secure *two*-party computation [2]. This work has a particularly efficient online phase due to the use of authentication codes with "good" combinatorial structure as well as the use of linear error correcting codes "with a good square".

A possible bachelor project could address the following aspects of secure two-party computation:

1. How come combinatorial authentication codes with "good" arithmetic structure allow for a more efficient online phase?
2. How can error correcting codes "with a good square" be used to achieve even greater efficiency gains in the online phase [4]?
3. Approaches that lead to an efficient off-line phase, for instance based on
   (a) Fully homomorphic encryption
   (b) Semi-homomorphic encryption
   (c) Yao's garbled circuit
4. Applying these techniques to more specific problems, for instance secure recommendation systems [3].

References

[1] D. Beaver. Efficient Multiparty Protocols Using Circuit Randomization. In J. Feigenbaum, editor, *CRYPTO 1991*, volume 576 of *Lecture Notes in Computer Science*, pages 420–432. Springer, 1991.

[2] R. Bendlin, I. Damgård, C. Orlandi, and S. Zakarias. Semi-homomorphic Encryption and Multiparty Computation. In K.G. Paterson, editor, *EUROCRYPT 2011*, volume 6632 of *Lecture Notes in Computer Science*, pages 169–188. Springer, 2011.

[3] R. Cramer, R. de Haan, and T. Veugen. Recommendations in the malicious model, 2012. Work in progress together with TNO.

[4] I. Damgård and S. Zakarias. Constant-Overhead Secure Computation of Boolean Circuits using Preprocessing, 2012. Cryptology ePrint Archive: Report 2012/512.