CFEM workshop: Theory and Practice of Multiparty Computation
4 to 8 June, 2012
Aarhus University, Denmark

Speaker: Thomas Schneider, TU Darmstadt

Title: Secure Set Intersection with Untrusted Hardware Tokens

Secure set intersection protocols are the core building block for a manifold of privacy-preserving applications. In their basic form, secure protocols for set intersection follow from general feasibility results for secure two-party computation. However, efficiency requirements for practical deployment have recently incited efforts to design dedicated protocols which are significantly more efficient.

In a recent work, Hazay and Lindell (ACM CCS 2008) introduced the idea of using trusted hardware tokens for the set intersection problem, devising protocols which improve over previous (in the standard model of two-party computation) protocols in terms of efficiency and secure composition. Their protocol uses only a linear number of symmetric-key computations and the amount of data stored in the token does not depend on the sizes of the sets. The security proof of the protocol is in the universal composability model and is based on the strong assumption that the token is trusted by both parties. In this paper we revisit the idea and model of hardware-based secure set intersection, and in particular consider a setting where tokens are not necessarily trusted by both participants. More precisely, we allow one party to use multiple hardware tokens from different manufacturers which are not trusted by the other party. Security for the one is guaranteed as long as the other party is not able to compromise all tokens, e.g., through side channel attacks, firmware trapdoors or malicious hardware. Our protocols are very efficient and achieve the same level of security as those by Hazay and Lindell for trusted tokens. For untrusted tokens, our protocols ensure privacy against malicious adversaries, and correctness facing covert adversaries.

Joint work with Marc Fischlin, Benny Pinkas, Ahmad-Reza Sadeghi, and Ivan Visconti.