Cryptographic Computation Outsourcing  
(Délégaion de calculs cryptographiques)

Encadrant : Damien Vergnaud  
Localisation : Département d’informatique - École normale supérieure

Cryptography is a crucial and ubiquitous component of information security. It permits to deal with basic computer security needs, related to e.g. confidentiality, privacy, integrity or authentication, but also more unconventional ones.

In 2013, the former US Vice President Dick Cheney revealed that his doctors disabled his pacemaker’s wireless capabilities to thwart possible assassination attempts. Current pacemakers and implantable cardiac defibrillators have some means to wirelessly communicate with external programmer device. The programmer device can collect patient data and adapt the therapy of the patient. Furthermore, it can generate fibrillations in test mode. The communication is not protected and it was shown that it is possible to get patient data, change the therapies of the patient, and even to induce fibrillations. If such a device is computationally incapable of carrying out cryptographic algorithms, the consequences can thus be dramatic. It is thus necessary to develop methods that provide security and privacy in this setting.

Numerous works studied the question of how a computationally limited device may outsource a resource-consuming computation to another, potentially malicious, device that is computationally much more powerful. Secure computation outsourcing is usually performed by leveraging algebraic homomorphic properties and can be used at various scales: a SIM card can delegate a sensitive computation to a mobile phone, a mobile phone can delegate to a laptop computer, and a laptop can delegate to a powerful server. First, one needs to ensure that this malicious device does not learn anything about what it is actually computing (secrecy) and sometimes one also needs to detect any failures (verifiability). Even if the problem of outsourcing cryptographic operations is not new, it has known a revival of interest in the last ten years with the development of mobile technologies.

The core operation of the majority of public-key cryptosystems is group exponentiation, i.e., computing $u^a$ in a group $G$ (multiplicatively written) from a group element $u$ and an exponent $a$. In 2005, Hohenberger and Lysyanskaya [HL05] provided a formal security definition for secure outsourcing and showed how to securely outsource group exponentiation to two, possibly dishonest, servers that are physically separated (and do not communicate). Several protocols were subsequently proposed in the two-server setting as well as in the more realistic one-server setting. Recently, Chevalier, Laguillaumie and Vergnaud analyzed in [CLV16] new and existing protocols outsourcing group (multi-)exponentiation to a single untrusted helper. This work gives rise to many interesting open problems. On a theoretical level, one goal of the internship is to study the relationship between computational efficiency and memory usage when implementing protocols for delegation of group exponentiation. On a more practical level, another goal is to consider groups equipped with efficient endomorphisms as proposed by Gallant, Lambert and Vanstone [GLV01] to perform efficient group exponentiation.

Pairings (or bilinear maps) were introduced in cryptography in 2000 by Joux [Jou00] and Boneh-Franklin [BF01]. Pairings proved to be an amazingly flexible and useful tool for the construction of cryptosystems with unique features (e.g. efficient identity-based cryptography or short signatures) but pairing computation is a lot more resource-consuming that group exponentiation. In 2005, Girault and Lefranc [GL05] introduced the first secure pairing delegation protocol via the
notion of *Server-Aided Verification*, which consists in speeding up the verification step of an authentication/signature scheme. During the internship, we will also study how to securely outsource pairing computation. Recently, Guillevic and Vergnaud [GV14] proposed two efficient protocols for pairing delegation using a single server. Their approach is unfortunately bandwidth consuming and another goal is to design new protocols and to give lower bounds on the communication and computational efficiencies of secure pairing delegation protocols. New techniques are needed since the number-theoretic techniques from [CLV16] do not seem to apply.

References


