E0 312: Foundations of Secure Computation

Arpita Patra
Department of Computer Science and Automation
Indian Institute of Science, India
arpita@csa.iisc.ernet.in

1. Name of the Course. Foundations of Secure Computation

2. Course Level and No. of Credits. 300 / 3:1

3. Instructor. Arpita Patra

4. Time of offering. August-December 2016.

5. Motivation and Objectives of the Course. The fantabulous journey of Secure Computation had originated with the seminal work of Andew Chi-Chih Yao published in Foundation of Computer Science (FOCS) 1982. The idea of secure computation is so groundbreaking that Yao was bestowed with the prestigious Turing Award in 2000.

Many compelling applications involve computations that require sensitive data from two or more entities. Consider the following example. The Earth is orbited by nearly 7000 man-made satellites and more than 21000 orbital debris larger than 10 centimeters. The growing number of satellites and space debris orbiting the planet is increasing the danger of collisions. This is not a hypothetical scenario. There are many such reported collisions. Most recently in 2009, two communication satellites belonging to the US and Russia collided in orbit. Given how expensive the satellites are, in terms of replacing the satellite, the host countries want to avoid collision. A collision can only be predicted if the detailed orbit information of the satellites are known. However, the detailed location of each satellite can be a closely guarded secret data; it can even be a national secret. So what is needed is a way to determine whether two satellites are about to clash with each other based on the detailed locations of the satellites, but *without* the need of disclosing the locations of the satellites. The problem of *secure computation* models such applications that make simultaneous demands for the privacy and usability of sensitive data. Informally, the problem of secure computation is defined as follows: We have a set of n distrusting parties $\{P_1, \ldots, P_n\}$, each with its own private input $x_1, \ldots, x_n$. They want to compute some publicly known function f on their inputs without disclosing their inputs. The distrust among the parties is formalized by an adversary that may corrupt some of the parties. Being a powerful abstraction, the problem of secure computation is known as the "holy-grail" problem of cryptography. Yet, as far as my knowledge is concerned with, we are yet to see a formal course on this topic in India. This course, a first of its kind in India, promises to offer a comprehensive understanding on this topic. It will unfold the evolution of this topic since 1982 to till date and teach the fundamental and intricate results in this area including some of the groundbreaking works done by 2012 Turing Award winners Shafi Goldwasser and Silvio Micali from MIT. It is my sincere hope that some of the students will aspire to become experts on this area after completing this course.

6. Syllabus. Indistinguishability, real-ideal world and simulation-based security notions; Secret Sharing, Verifiable Secret Sharing, Oblivious Transfer, Circuit Garbling and function encoding, Commitment Scheme, Zero-knowledge Proof, Threshold Cryptography, Encryptions, Broadcast & Byzantine

Agreement, Coin-tossing protocol, Theoretical and practical protocols for secure computation in various models.

7. References. A part of the course will be covered from the following books

1. "Efficient Two-part Protocols- Techniques and Constructions" by Carmit Hazay and Yehuda Lindell. Springer.
2. "Secure Multiparty Computation and Secret Sharing - An Information Theoretic Approach" by Ronald Cramer, Ivan Damgaard and Jesper Buus Nielsen. Cambridge Press.

The rest will be taken from research reports.

8. Prerequisites. Mathematical maturity will be assumed.