Preface to Yehuda Lindell's PhD Thesis

Oded Goldreich
Department of Computer Science and Applied Mathematics
Weizmann Institute of Science
Rehovot, ISRAEL.
oded@wisdom.weizmann.ac.il

June 22, 2003

The modern society is quite preoccupied with various statistics like the average, median and deviation of various attributes (e.g., salary) of it members. On the other hand, individuals often wish to keep their own attributes secret (although they are interested in the above statistics). Furthermore, on top of being suspicious of other people, individuals are growing to be suspicious of all (the society's) establishments and are unwilling to trust the latter with their secrets. Under these circumstances it is not clear whether there is a way for the members of the society to obtain various statistics (regarding all secrets) without revealing their individual secrets to other people.

The above question is a special case of a general problem. We are talking about computing some (predetermined) function of inputs that are scattered among different parties, without having these parties reveal their individual inputs. The mutually suspicious parties have to employ some distributed protocol in order to compute the function value, without leaking any other information regarding their inputs to one another. Furthermore, in some settings, some of the parties may deviate from the protocol, and it is desired that such malfunctioning will not be of any advantage to them. At best, we would like to "emulate" a trusted party (which collects the inputs from the parties, computes the corresponding outputs, and hand them to the corresponding parties), and do so in a distributed setting in which no trusted parties exist. This, in a nutshell, is what secure cryptographic protocols are all about.

General results concerning secure two-party and multi-party computations were first announced in the mid 1980's.[1] In a nutshell, assuming the existence of trapdoor permutations, these results assert that one can construct protocols for securely computing any desirable multi-party functionality. These results either require a majority of honest players or allow dishonest players to suspend the execution (while being detected as bad). Subsequent "unconditional" results in the "private channel model" require a 2/3-majority of honest player.[2]

[1] See O. Goldreich, S. Micali and A. Wigderson, "Proofs that Yield Nothing but their Validity or All Languages in NP Have Zero-Knowledge Proof Systems" (*Journal of the ACM*, Vol. 38, No. 1, pages 691–729, 1991; and preliminary version in *27th IEEE Symposium on Foundations of Computer Science*, 1986); A.C. Yao, "How to Generate and Exchange Secrets" (in *27th IEEE Symposium on Foundations of Computer Science*, pages 162–167, 1986); and O. Goldreich, S. Micali and A. Wigderson, "How to Play any Mental Game – A Completeness Theorem for Protocols with Honest Majority" (in *19th ACM Symposium on the Theory of Computing*, pages 218–229, 1987). For details see O. Goldreich, *Secure Multi-Party Computation* (available from `http://theory.lcs.mit.edu/~oded/gmw.html`).

[2] See M. Ben-Or, S. Goldwasser and A. Wigderson, "Completeness Theorems for Non-Cryptographic Fault-Tolerant Distributed Computation" (in *20th ACM Symposium on the Theory of Computing*, pages 1–10, 1988); and D. Chaum, C. Crépeau and I. Damgård, "Multi-party unconditionally Secure Protocols" (in *20th ACM Symposium on the Theory of Computing*, pages 11–19, 1988).

The aforementioned results were discovered at a time in which *intensive* electronic multi-party interactions seemed a remote possibility. So it seems fair to say that, while generating considerable interest within the theory community, these results generated little interest in the applied cryptography community. But times have changed: Intensive electronic multi-party interactions seems almost a reality, and the entire cryptographic community seems very much interested in a variety of natural problems which arise from such a reality. This has triggered a serious reconsideration of the definitions and results established in the 1980's. The current work makes an important contribution to this fundamental project.

One important research direction initiated and developed in recent years is the study of the preservation of security under concurrent executions of protocols. This research is aimed at extending the treatment developed in the 1980's, which only refers to stand-alone executions. Needless to say, in most settings, one would like security to hold not only when a single execution takes place but rather even if multiple executions are taking place concurrently. Furthermore, whereas we do not want to require honest parties to coordinate their actions in the various executions (which may not even be possible), the adversary may coordinate its actions in the various executions. Thus, this research direction is of great theoretical and practical importance. Two of the three technical chapters of the current work present important contributions to this research direction:

1. Chapter 4 deals with the notion of environmental-security (a.k.a "Universal Composability" or UC-security) put forward by Canetti.[3] In particular, environmental-security implies security under concurrent executions of protocols. It was known how to construct environmentally-secure multi-party computations *with honest majority*, and that this is impossible (in the bare model) when the honest parties are not in majority (and most importantly in the two-party case).

   The current work shows that augmenting the model with a "random reference string" yields a model in which environmentally-secure *two-party* computations are possible. More generally, environmentally-secure multi-party computations are possible in this model even without a honest majority. It should be noted that the availability of a "random reference string" is a reasonable assumption that has been used in several related contexts.

2. Chapter 2 deals with the composability of Byzantine Agreement protocols. It is well-known that (in the plain model) Byzantine Agreement requires a special majority (of two-thirds) of honest players. To obtain higher resilience, one typically uses a public-key infrastructure, yielding so-called *authenticated* Byzantine Agreement.

   The current work shows that authenticated Byzantine Agreement is not preserved under concurrent composition, unless distinct IDs can be externally assigned (in a secure manner) to individual executions. Thus, in contrast to previous beliefs, authenticated Byzantine Agreement can be safely used *only* in contexts where each protocol invocation is assigned a distinct ID.

In addition, this work studies the phenomenon that Byzantine Agreement (or a postulated broadcast channel) is used in all prior work regarding secure multi-party computation. This question of whether or not this phenomenon is inherent to secure multi-party computation arises even in the stand-alone context. In Chapter 3 it is shown that if the definition of secure multi-party computation is slightly relaxed (in a way that does not imply Byzantine Agreement as a special

[3]See R. Canetti, "Universally Composable Security: A New Paradigm for Cryptographic Protocols" (in *42nd IEEE Symposium on Foundations of Computer Science*, pages 136–145, 2001).

case) then one can obtain secure protocols without using Byzantine Agreement (or postulated broadcast channels). Avoiding the use of the Byzantine Agreement has several advantages for the stand-alone model as well as for restricted types of protocol composition.

The focus of the current work is on the *general* study of secure multi-party computation (rather than on protocols for solving specific problems), which is natural in the context of the theoretical treatment of the subject matter. We wish to highlight the importance of this theoretical study to practice. Firstly, this study clarifies fundamental issues regarding security in a multi-party environment. Secondly, it draws the lines between what is possible in principle and what is not. Thirdly, it develops general techniques for design of secure protocols. And last, sometimes, it may even yield schemes (or modules) that may be incorporated in practical systems. Thus, we believe that the current work is both of theoretical and practical importance.