Transputation: Transport Framework for Secure Computation

Markus Brandt\textsuperscript{1,2}, Claudio Orlandi\textsuperscript{2}, Kris Shrishak\textsuperscript{1}, and Haya Shulman\textsuperscript{1,3}
\textsuperscript{1}TU Darmstadt \textsuperscript{2}Aarhus University \textsuperscript{3}Fraunhofer SIT

Secure two-party computation (2PC) allows two mutually distrustful parties to jointly compute any function on their private inputs without revealing anything but the output. In the last decade the performances of 2PC protocols based on garbled circuits have greatly improved and, thanks also to hardware support for cryptographic operations, it is now widely believed that the remaining main bottleneck for 2PC is the bandwidth [ALSZ13, ZRE15].

Although there has been work on reducing the communication complexity and limiting the amount of data that needs to be exchanged between the two parties, no research has yet been undertaken on the networking layer of 2PC. Furthermore, current evaluations of 2PC disregard the issues faced by practical deployment of 2PC: many implementations are not evaluated in real life setups where network conditions, packet losses, latency, other traffic and other processes can affect performances negatively [Hal18, Kre17].

In this work, we explore two main issues in the performance of Secure Two-Party Computation (2PC): (1) interaction of 2PC with the transport layer and (2) evaluation of 2PC implementations.

**Transport layer.** Although significantly improved, the performance of 2PC is still prohibitive for practical systems. Contrary to the common belief that bandwidth is the remaining bottleneck for 2PC implementation, we show that the network is under-utilised due to the use of standard TCP sockets. Nevertheless, using other sockets is a nontrivial task: the developers of secure computation need to integrate them into the operating systems, which is challenging even for systems experts. To resolve this issue and break the efficiency barrier of 2PC, we design and develop a framework, we call Transputation, which automates the integration of transport layer sockets into 2PC implementations. The goal of Transputation is to enable developers of 2PC protocols to easily identify and use the optimal transport layer protocol for the given computation task and network conditions and hence to improve performance of secure computation.

We integrated selected transport layer protocols into Transputation and evaluated the performance for a number of computational tasks. As a highlight, even a general purpose transport layer protocol, such as UDT, improves the run-time of 2PC over TCP on EU-Australia connection for circuits with \(> 10^6\) Boolean gates by a factor of 8.

**Evaluations of 2PC.** Evaluations of 2PC implementations do not reflect performance in real networks since they are typically done on simulated environments and even more often on a single host. To address this issue, we provide a testbed platform for evaluation of 2PC implementations in real life settings on the Internet.

References


