Outsourcing Secure Two-Party Computation as a Black Box

Henry Carter[1], Benjamin Mood[2], Patrick Traynor[2], Kevin Butler[2]

[1]School of Computer Science, College of Computing, Georgia Institute of Technology

[2]SENSEI Center, Computer & Information Science & Engineering Department, University of Florida

Secure multiparty computation (SMC) offers a technique to preserve functionality and data privacy in mobile applications. Current protocols that make this costly cryptographic construction feasible on mobile devices securely outsource the bulk of the computation to a cloud provider. However, these outsourcing techniques are built on specific secure computation assumptions and tools, and applying new SMC ideas to the outsourced setting requires the protocols to be completely rebuilt and proven secure. In this work, we develop a generic technique for lifting any secure two-party computation protocol into an outsourced two-party SMC protocol. By augmenting the function being evaluated with auxiliary consistency checks and input values, we can create an outsourced protocol with low overhead cost. Our implementation and evaluation show that in the best case, our outsourcing additions execute within the confidence intervals of two servers running the same computation, and consume approximately the same bandwidth. In addition, the mobile device itself uses minimal bandwidth over a single round of communication. This work demonstrates that efficient outsourcing is possible with any underlying SMC scheme, and provides an outsourcing protocol that is efficient and directly applicable to current and future SMC techniques.

Mobile Privacy

- Mobile apps use a significant amount of private user data.
- App servers are not necessarily well-secured or trustworthy when given this information.
- Secure Multiparty Computation (SMC) allows apps to process encrypted data, but is computationally expensive.

Secure Multiparty Computation

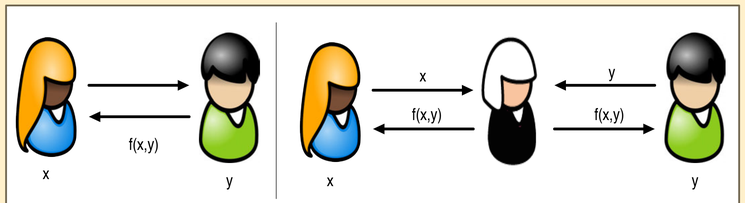

Fig. 1. Real vs. Ideal world execution

- The goal of SMC is to allow mutually distrustful parties to jointly compute a result.
- Security is defined by input privacy and output correctness, and is proven through the real/ideal simulation paradigm.
- Techniques for performing SMC include garbled circuits, secret sharing, homomorphic encryption, and others.
- The optimal technique differs based on function representation, available bandwidth, and number of parties.

Black Box Construction

- Outsourcing to the Cloud offers a way to run costly protocols between mobile devices and app servers.
- Previous protocols build on fixed SMC techniques and require significant re-engineering to update.

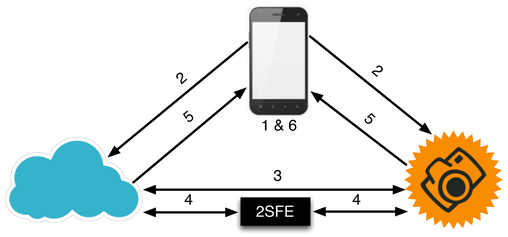

Fig. 2. The Black Box Protocol

- Our protocol outsources any two-party SMC protocol.
- The mobile device performs minimal input preparation operations, then hands off computation to the app server and Cloud.

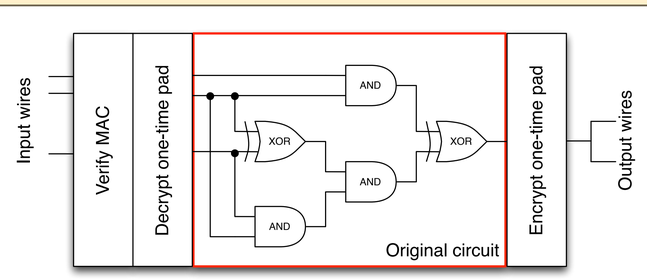

Fig. 3. The Augmented Circuit

- Rather than adding consistency checks to the protocol to ensure correctness, we add them to the evaluated circuit.

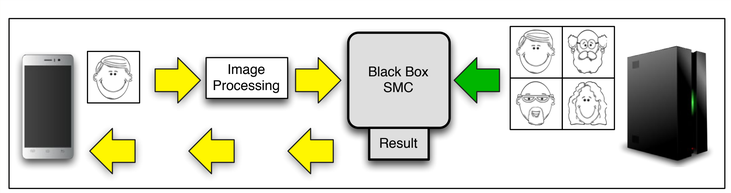

Fig. 4. An example facial recognition application

- We show the practicality of our protocol in an example application: secure facial recognition.
- The mobile can compare a picture to a database of faces without learning the full database contents. Also, the database does not learn the contents of the query.

Performance

Program Name	SS13 Total	BB Total	Increase	SS13 Non-XOR	BB Non-XOR	Increase
Dijkstra10	259,232	456,326	1.8x	118,357	179,641	1.5x
Dijkstra20	1,653,542	1,949,820	1.2x	757,197	849,445	1.1x
Dijkstra50	22,109,732	22,605,018	1.0x	10,170,407	10,324,317	1.0x
MatrixMult3x3	424,748	1,020,196	2.4x	161,237	345,417	2.1x
MatrixMult5x5	1,968,452	3,360,956	1.7x	746,977	1,176,981	1.6x
MatrixMult8x8	8,069,506	11,354,394	1.4x	3,060,802	4,075,082	1.3x
MatrixMult16x16	64,570,969	77,423,481	1.2x	24,494,338	28,458,635	1.2x
RSA128	116,083,727	116,463,648	1.0x	41,082,205	41,208,553	1.0x

Fig. 5. Test circuit gate counts and overhead

- Our experiments use a garbled circuit SMC scheme and measure the overhead incurred from outsourcing.
- For large circuits, the added gate count becomes minimal.

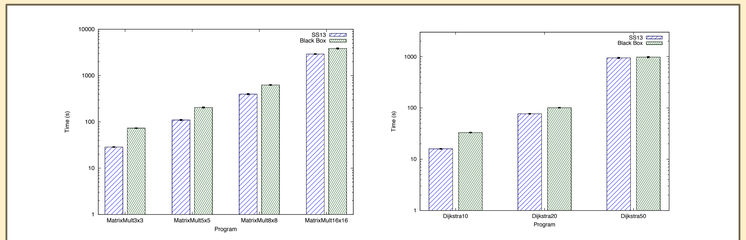

Fig. 6. Execution time for multiple test applications

- For large circuits, the outsourced protocol and two-party protocol run in approximately the same time.
- This overhead will be further reduced as better SMC protocols and MAC protocols develop.

Conclusions & Future Work

- Mobile applications use private data. SMC allows this data to remain encrypted, but requires a high computational cost.
- We demonstrate a protocol for outsourcing *any* two-party SMC protocol for efficient use on a mobile device.
- Our future work will explore ways to relax security in exchange for practical efficiency and use-cases.

This material is based on research sponsored by DARPA under agreement number FA8750-11-2-0211

For further information, contact Henry Carter (*carterh@gatech.edu*).