Turbospeedz: Double Your Online SPDZ!

A Note On SPDZ with Function Dependent Preprocessing

Aner Ben-Efraim  Ariel University & Ben-Gurion University
Secure Multiparty Computation

- Idea: parties compute a function of their inputs, revealing only the output, even if some of the parties are corrupt.
  - Examples: online auction, tender, elections, cloud computing...
- Some desirable properties:

<table>
<thead>
<tr>
<th>Correctness</th>
<th>Privacy</th>
<th>Independence of Inputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>$2+2=5$</td>
<td>![Privacy Image]</td>
<td>![Independence Image]</td>
</tr>
<tr>
<td>Fairness</td>
<td>Guaranteed Output Delivery</td>
<td>Efficient Efficiency</td>
</tr>
<tr>
<td>![Fairness Image]</td>
<td>![Guaranteed Image]</td>
<td>![Efficiency Image]</td>
</tr>
</tbody>
</table>
The “Offline-Online” Model

“Offline” Phase
- Performed before the inputs are known (preprocessing)
- Expensive computations
- Relatively slow

“Online” Phase
- Performed after inputs are known
- Cheap computations
- Should be very fast

Long preparation time

“Real-time” secure computation
“Offline” Preprocessing
Function Dependent vs. Function Independent

**Function Independent**
- Does not require knowledge of the actual function computed
- Requires knowledge on number of AND/multiplication gates used
- **SPDZ offline is function independent**
  - Offline requires #multiplications

**Function Dependent**
- Requires knowledge of the actual function computed
- **Turbospeedz offline is function dependent**
  - Offline requires circuit
SPDZ

- SPDZ introduced by Smart, Pastero, Daamgard, and Zacharias [SPDZ12]
  - Offline generates “multiplication triples”
    • $(\langle a \rangle, \langle b \rangle, \langle c \rangle)$, where $c = ab$
    • $\langle \cdot \rangle$-share – share + share of MAC

- Fastest SPDZ online protocol by Daamgard et al. [DKLPSS13]
  - Multiplication gates require partially opening 2 $\langle \cdot \rangle$-shared values
    • Main online communication cost
    • Also affects computation
Our Result: TurboSPEEDz Online

- New online protocol
  - Multiplication gates require partially opening only $1! \cdot \langle \cdot \rangle$-shared value
    - Main online communication cost is halved!
    - Also reduces some of the computation
  - In some scenarios, online throughput expected to almost double!
Caveats

1. We add a function dependent preprocessing protocol on top of SPDZ offline

   Conclusion:
   - Relatively cheap, comparable to SPDZ online
   - Online is more important and the function is known at the offline,
     if online is more important

2. We require an additional \( \langle \cdot \rangle \)-shared random value per multiplication gate
   - Created at (function independent) offline
   - Relatively cheap compared to generating triples

3. We drastically reduce online computation, while slightly increasing offline & total time
Turbospeedz: Double Your Online SPDZ!
A Note On SPDZ with Function Dependent Preprocessing

Soon on eprint

Aner Ben-Efraim  Ariel University & Ben-Gurion University