Garbled Circuits (GC) is a protocol for implementing Secure Function Evaluation (SFE) which can evaluate any function that can be expressed as a Boolean circuit and obtain the result while keeping all parties’ inputs private.

We investigate, implement and evaluate a Secure Computation Infrastructure using FPGA Overlay architecture and leverage hardware acceleration to tackle the scalability and efficiency challenges inherent in garbled circuits.

Our implementation shows orders of magnitude improvement over a software package (ObliVM) for evaluating garbled circuits and demonstrates that the circuit being evaluated can change with almost no overhead.

### Garbled Circuits Algorithm Example

#### Users

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Random number a represents value i on wire a. Same applies to wire b and x.

Using cryptographic primitives, take random number as keys, encrypt the output key. Output: Four cyphertexts.

#### Garbler

- Encrypt a0, b0
- Encrypt a0, b1
- Encrypt a1, b0
- Encrypt a1, b1

#### Evaluator

- Input garbled value of f(x)
- Layer Extractor: a tool that we developed for extracting parallelism within each operation. Within each layer can be computed at the same time
- Output: Four cyphertexts.