# Script for Eurocrypt 2023 Talk

## Page 1

Thanks for the introduction. I am Hongrui, and I am happy to present our recent progress on improving the communication complexity of constant-round maliciously secure 2 party computation. This is a joint work with Xiao Wang, Kang Yang, and Yu Yu.

## Page 2

### View 1

To begin with, we recall the progress on the optimization of garbled circuits. Ever since the Yao’s proposal in the 80s, there has been continuous effort improving the efficiency of the garbling scheme, and that translates to the respective progress in constant-round semi-honest 2PC, which is essentially .

### View 2

A natural question is how to boost semi-honest GC-based 2PC into malicious security, where we allow arbitrary deviation from the protocol. The state-of-the-art technique is authenticated garbling and it’s proposed by Wang et al. in CCS2017. The basic idea is to authenticate the garbling process using message authentication code, or MAC. And to generate such MACs we will need an input-independent preprocessing protocol. The online phase is essentially semi-honest GC evaluation plus consistency checking.

### View 3

In the original WRK17, the preprocessing functionality is instantiated via an optimized TinyOT protocol while in a subsequent paper by the same group of authors an online protocol matching the communication complexity of semi-honest half-gates was proposed.

### View 4

Afterwards, generating OT-like correlations has been significantly improved by numerous pseudorandom correlation generators based on Learning Parity with Noise. In particular, they allow generating such triples using sublinear communication and linear computation. A natural idea is to try to optimize the preprocessing protocol for authenticated garbling using PCGs. Dittmer et al. studied that idea in Crypto 2022 and achieved bits per AND gate in the correlated OT hybrid model and bits per AND gate in the multiplication triple hybrid model. The improvement is significant compared to previous TinyOT-style protocols, but we can see there is still a gap between the semi-honest world and the malicious world.

### View 5

A natural question is can we bridge such a gap?

## Page 3

Here we list the main contributions of our work in this table. In this work we manage to achieve essentially the same one-way communication as semi-honest half-gates, and as for total communication we achieve about 34% improvement over previous state-of-the-art. Since one-way communication is more relevant in the full-duplex network, and hence the title “Actively Secure Half-Gates with Minimum Overhead under Duplex Networks”. Now let me try to convey to you the essential ideas of this work for the rest of this talk.

## Page 4

### View 1

Let’s start by recalling the garbling notations in the good-old semi-honest GC-based circuit evaluation protocol. In this scheme the garbler first prepares random mask and labels for each wire, subject to the constraint that the 0-label and 1-label on each wire XOR to a fixed random string. We use small lambda for wire masks and capital lambda for the masked wire values. The garbler first sends the garbled truth tables for each AND gate to the evaluator, which can be done even before the specific inputs are known.

### View 2

Then the garbler sends the labels and masked wire values to the evaluator, specifically since Bob’s inputs need to be protected, we need to use oblivious transfer for his input labels.

### View 3

Now the evaluator uses the input information to evaluate the circuit topologically. For each AND gate the garbler decrypts the table entry that corresponds to its masked input values, and gets the output wire label. Typically, the masked output wire value is acquired by extracting the least significant bit of the output label, but any other position suffices.

### View 4

Now that the evaluator gets the masked values for each wire, including the output wire, the garbler sends the wire masks for the output wires to the evaluator, allowing him to get the real outputs.

## Page 5

### View 1

In the above process, the garbler can generate the garbled truth table almost arbitrarily. Two particular attacks of interest are selective failure attack and inconsistent function attack. In selective failure attack a malicious garbler can corrupt some rows of the truth table, which once reached during evaluation, would lead to abort. And since the choice during evaluation is input-dependent, the evaluator can deduce the real wire value in case of an abort, violating privacy. As for inconsistent function attack the garbler can garble a completely different logic and the evaluator in the previous protocol has no way of checking it.

In authenticated garbling, the first problem is solved by sharing the wire masks, which disables the garbler’s ability to deduce real wire values in case of abort, making the abort event input-independent. The second attack is solved by essentially authenticating the masked wire values so that consistency can be enforced.

### View 2

By sharing the wire masks, the garbler no longer controls all the randomness during garbling and thus we need to generate some correlated randomness prior to garbling. This is captured in the functionality , which samples two global keys , wire masks and , product share   and  , as well as their authentications. The hat-values share the product of the two wire masks on each AND gate.

### View 3

And we denote the authentication using regular IT-MAC notations: M for MAC tag, K for MAC key.

### View 4

After preprocessing, the two parties can locally generate additive share of the garbled circuit. Alice can send its share to Bob who can then complete evaluation. One point to note is that the garbler’s IT-MAC key is also used as the free-XOR shift, so we require it to be as long as the computational security parameter, usually taken as 128 bits in practice.

### View 5

Bob’s authentication comes into play for consistency checking. In the WRK17 scheme, Alice prepares an additional garbled circuit, allowing Bob to acquire the MAC tag for the masked wire value which enforces correctness. Since Bob’s key is only used for MAC checking, it can be as short as the statistical parameter, usually taken as 40 bits.

## Page 6

### View 1

In the previous protocol Alice needs to send bits per AND gate. A subsequent work KRRW18 shows that this can be optimized to bits, matching semi-honest half-gates.

### View 2

As for consistency checking, the goal is to check the real wire values of AND gates are indeed of AND correlation. In some sense we may view IT-MAC as a commitment, and the observation of KRRW is that by making the masked wire values public, the whole equation can be evaluated using the linear homomorphism of IT-MAC, and thus the error terms can essentially be committed. So, by opening the random linear combination of the error terms consistency checking is essentially free.

## Page 7

### View 1

After authenticated garbling, our second ingredient is pseudorandom correlations generators and designated verifier zero knowledge on top of that

and adding IT-MAC checking respectively. The idea is that since the wire masks are shared, the garbler can no longer deduce the real wire value from the corrupted row index which means that abort probability is input-independent.