Don't Eject the Impostor
Fast Three-Party Computation With a Known Cheater

ENCRYPTO Cryptography and Privacy Engineering

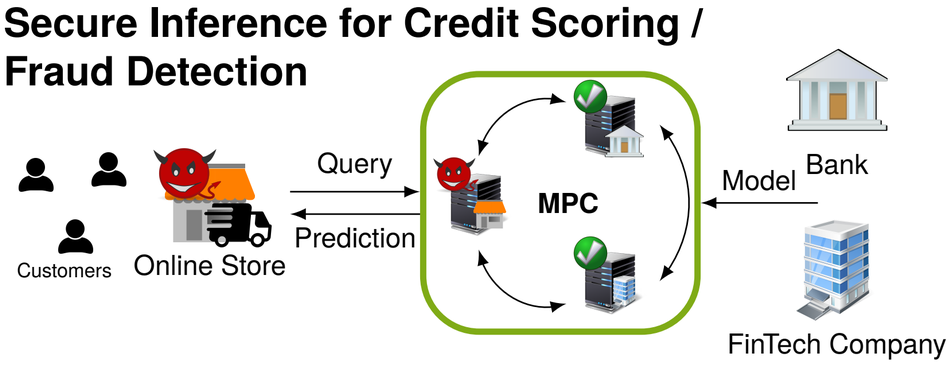
Andreas Brüggemann, Oliver Schick, Thomas Schneider, Ajith Suresh, Hossein Yalame

Technische Universität Darmstadt

TII Technology Innovation Institute

Motivating Example

Secure Inference for Credit Scoring / Fraud Detection



Lower trust in online store ⇒ might behave maliciously😈

- Semi-honest security model (✅) ⇒ insufficient
- Malicious security model (😈) ⇒ inefficient

⇒ **Mixed model (✅ + 😈) fits best**

Are symmetric security models really a good fit for any real-world application with diverse parties?

Setting

- Three parties (**3PC**)
- Parties do not collude ⇒ **honest majority**
- Only one **fixed and known party susceptible to malicious corruption**, others semi-honest

Extension of prior works for an asymmetric 2PC setting:

- Yao's garbled circuits [Yao86] when using maliciously secure OT: semi-honest garbler, malicious evaluator
- **client-malicious setting** [LMS+21; CGO+22]: malicious client, semi-honest server
- malicious server, semi-honest client [DWL+23]

Our Contributions

- **Auxiliator & Socium**, 3PC protocols with asymmetric security model
- **Streamlined online** phase between two evaluators
- Multiple options for **efficient preprocessing**
- **ML-friendly**, but multi-purpose design
- **Open-source** implementation

⇒ **Bridging the gap between symmetric semi-honest and malicious security model**

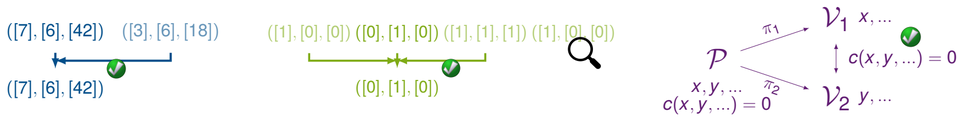
Detection and Prevention of Cheating

Majority of verification done during preprocessing using

- Triple Sacrificing (ring version) [CDE+18]
  Including novel **Matrix Triple Sacrificing**
- Cut-and-Choose (binary) [ABF+17]
- Distributed ZKPs [BBC+19; BGI+19]

adapted to our asymmetric setting; requiring either low computation or low communication overhead.

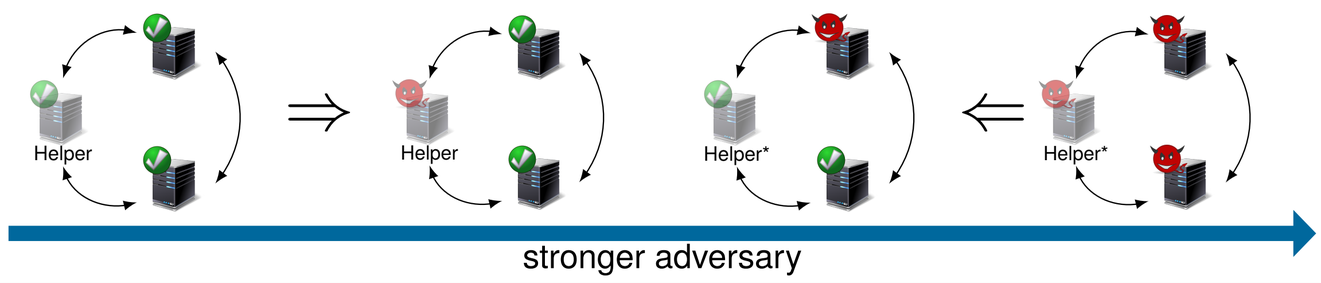
Outsource verification to semi-honest parties!



The World Between Astra & Swift

Auxiliator and Socium are based on the 3PC protocols Astra and Swift for the symmetric semi-honest respectively malicious security model.

Astra [CCP+19] **Auxiliator (new)** **Socium (new)** Swift [KPP+21]



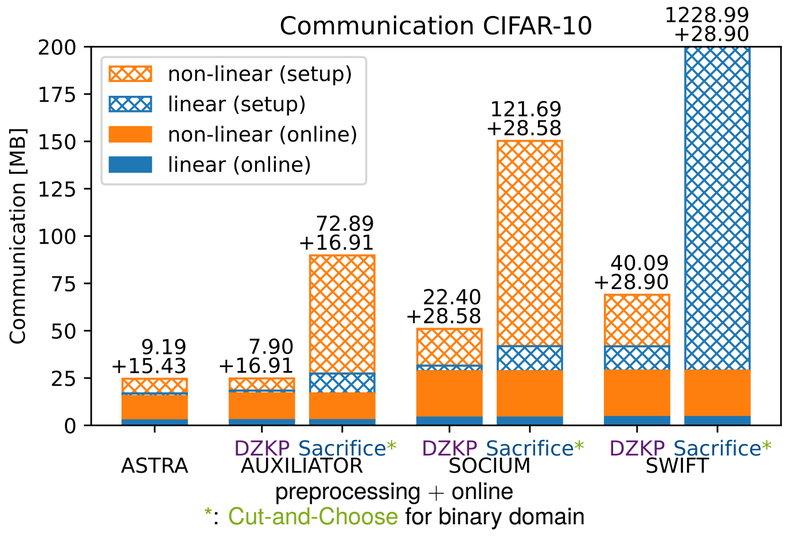
Communication per Operation

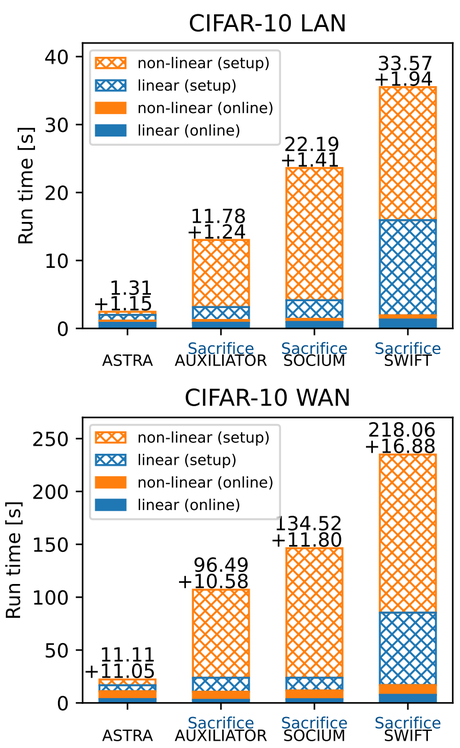
Operation / Verification	$\mathbf{a} \cdot \mathbf{b} = \mathbf{c}$ Sacrifice	$\mathbf{a} \cdot \mathbf{b} = \mathbf{c}$ DZKP	$\square \cdot \square = \square$ Sacrifice	$\square \cdot \square = \square$ DZKP	$\mathbf{x} \wedge \mathbf{y} = \mathbf{z}$ Cut-and-Choose	$\mathbf{x} \wedge \mathbf{y} = \mathbf{z}$ DZKP
Preprocessing						
Astra [CCP+19]	$1 \times \mathbb{Z}_\ell$		$\square$		1 bit	
Auxiliator (**this work**)	$4 \times \mathbb{Z}_{\ell+\sigma}$	$1 \times \mathbb{Z}_\ell$	$2\square + 2\square$	$\square$	11 bits	1 bit
Socium (**this work**)	$5 \times \mathbb{Z}_{\ell+\sigma}$	$2 \times \mathbb{Z}_\ell$	$3\square + 2\square$	$2\square$	12 bits	2 bits
Swift [KPP+21]	$9 \times \mathbb{Z}_{\ell+\sigma}$	$3 \times \mathbb{Z}_\ell$	$6\square + 3\square$	$3\square$	21 bits	3 bits
Online						
Astra/Auxiliator	$2 \times \mathbb{Z}_\ell$		$2\square$		2 bits	
Swift/Socium	$3 \times \mathbb{Z}_\ell$		$3\square$		3 bits	

ℓ: ring size σ: statistical security parameter

Performance for Secure Inference

We benchmarked Astra, Auxiliator, Socium & Swift for secure inference using the 7-layer CNN from MiniONN [LJL+17] trained on the CIFAR-10 dataset.





What is the advantage of extending semi-honest 2PC by a malicious helper? (setup only, same online phase)

Layers	ABY2.0 [PSS+21] original	ABY2.0 [PSS+21] improved OT	Auxiliator (this work) Sacrifice*	Auxiliator (this work) DZKP
Linear	178.97 GB	60.59 GB	10.46 MB	1.48 MB
Non-linear	524.25 MB	23.04 MB	62.43 MB	6.42 MB
Total	179.50 GB	60.61 GB	72.89 MB	7.90 MB

What is the advantage of extending the client-malicious setting by a semi-honest helper?

Layers	SIMC [CGO+22]	Socium (this work) Sacrifice*	Socium (this work) DZKP
Linear	140 MB	20.01 MB	7.08 MB
Non-linear	2.24 GB	148.33 MB	43.90 MB
Total	2.38 GB	168.34 MB	50.98 MB

References

[ABF+17] T. Araki, A. Barak, J. Furukawa, T. Lichter, Y. Lindell, A. Nof, K. Ohara, A. Watzman, and O. Weinstein. "Optimized Honest-Majority MPC for Malicious Adversaries - Breaking the 1 Billion-Gate Per Second Barrier". In: *IEEE S&P*. 2017.

[BBC+19] D. Boneh, E. Boyle, H. Corrigan-Gibbs, N. Gilboa, and Y. Ishai. "Zero-Knowledge Proofs on Secret-Shared Data via Fully Linear PCPs". In: *CRYPTO*. 2019.

[BGI+19] E. Boyle, N. Gilboa, Y. Ishai, and A. Nof. "Practical Fully Secure Three-Party Computation via Sublinear Distributed Zero-Knowledge Proofs". In: *CCS*. 2019.

[CCP+19] H. Chaudhari, A. Choudhury, A. Patra, and A. Suresh. "ASTRA: High Throughput 3PC over Rings with Application to Secure Prediction". In: *CCSW@CCS*. 2019.

[CDE+18] R. Cramer, I. Damgård, D. Escudero, P. Scholl, and C. Xing. "SPD $\mathbb{Z}_{2^k}$: Efficient MPC mod 2^k for Dishonest Majority". In: *CRYPTO*. 2018.

[CGO+22] N. Chandran, D. Gupta, S. L. B. Obbattu, and A. Shah. "SIMC: ML Inference Secure Against Malicious Clients at Semi-Honest Cost". In: *USENIX Security*. 2022.

[DWL+23] C. Dong, J. Weng, J.-N. Liu, Y. Zhang, Y. Tong, A. Yang, Y. Cheng, and S. Hu. "Fusion: Efficient and Secure Inference Resilient to Malicious Servers". In: *NDSS*. 2023.

[KPP+21] N. Koti, M. Pancholi, A. Patra, and A. Suresh. "SWIFT: Super-fast and Robust Privacy-Preserving Machine Learning". In: *USENIX Security*. 2021.

[LJL+17] J. Liu, M. Juuti, Y. Lu, and N. Asokan. "Oblivious Neural Network Predictions via MiniONN Transformations". In: *CCS*. 2017.

[LMS+21] R. Lehmkuhl, P. Mishra, A. Srinivasan, and R. A. Popa. "Muse: Secure Inference Resilient to Malicious Clients". In: *USENIX Security*. 2021.

[PSS+21] A. Patra, T. Schneider, A. Suresh, and H. Yalame. "ABY2.0: Improved Mixed-Protocol Secure Two-Party Computation". In: *USENIX Security*. 2021.

[Yao86] A. C.-C. Yao. "How to Generate and Exchange Secrets (Extended Abstract)". In: *FOCS*. 1986.

DFG Deutsche Forschungsgemeinschaft

European Research Council Established by the European Commission

CROSSING

Privacy and Trust for Mobile Users

Paper https://ia.cr/2023/1744

Code https://encrypto.de/code/MOTION-FD

Website https://encrypto.de