Framework for Semi-Private Function Evaluation with Application to Secure Insurance Rate Calculation

ENCRYPTO CRYPTOGRAPHY AND PRIVACY ENGINEERING

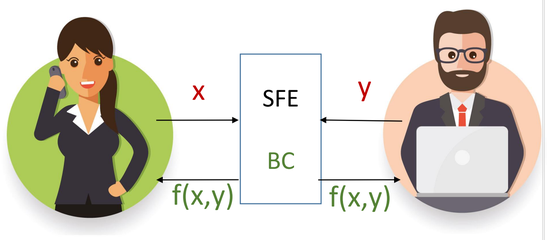
Daniel Günther, Ágnes Kiss, Lukas Scheidel, Thomas Schneider
Cryptography and Privacy Engineering Group (ENCRYPTO), TU Darmstadt, Germany
guenther@rangar.de, lukas.scheidel@web.de, {kiss, schneider}@encrypto.cs.tu-darmstadt.de

TECHNISCHE UNIVERSITÄT DARMSTADT

Semi-Private Function Evaluation (SPFE)

Secure Function Evaluation (SFE)

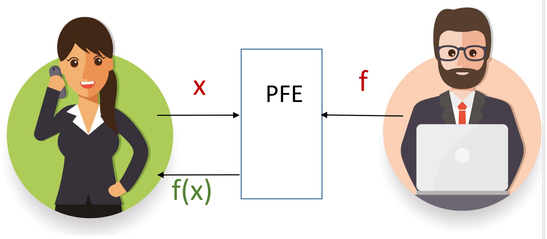
SFE [Yao86] allows 2 parties to jointly compute a public function f represented as Boolean Circuit (BC) on their private inputs x, y and obtain no information but f(x, y).



Alice and Bob learn f(x,y) without revealing x, y to each other.

Private Function Evaluation (PFE)

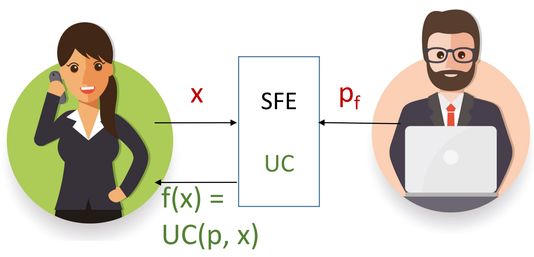
PFE allows Alice and Bob to jointly compute Bob‘s private function f on the private input x of Alice without revealing anything but the output



Alice learns f(x) without revealing x. f is hidden to Bob.

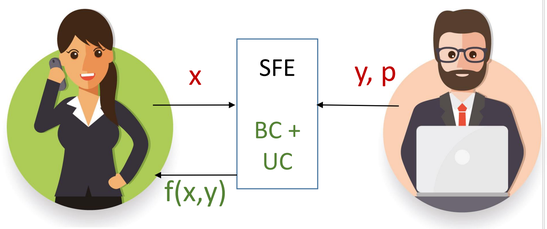
PFE can be reduced to SFE using a Universal Circuit (UC)

A Universal Circuit [Val76, KS16, GKS17] is a BC that can compute any Boolean function f of a given size n by specifying programming bits p, s.t. UC(p,x) = f(x).



Semi-Private Function Evaluation (SPFE)

In many applications not the whole function must be kept private. Dividing the function in sub-functions which are private and public leads to SPFE which results in a smaller total circuit.



Previous Works

[BK17]: N. Büscher, S. Katzenbeisser. Compilation for Secure Multi-party Computation. Springer
[DSZ15]: D. Demmler, T. Schneider, M. Zohner. ABY- A Framework for Efficient Mixed-Protocol Secure Two-Party Computation. In NDSS‘15.
[GKS17]: D. Günther, Á. Kiss, T. Schneider: More Efficient Universal Circuit Constructions. In ASIACRYPT‘17.
[KS16]: Á. Kiss, T. Schneider. Valiant‘s Universal Circuit is Practical. In EUROCRYPT‘16.
[Val76]: L.G. Valiant. Universal Circuits (Preliminary Report). In STOC‘76.
[Yao86]: A. C.-C. Yao. How to Generate and Exchange Secrets (Extended Abstract). In *FOCS‘86*.

Our SPFE Framework

- Our SPFE framework can be used on functions f that can be split into a set of sub-functions specified in the C programming language.
- Every sub-function must be declared either private or public.
- We use CBMC-GC [BK17] to compile each sub-function into a BC.
- Private sub-functions are further processed to UCs [GKS17] and the corresponding programming bits are deviated.
- BCs and UCs are merged using our merger to build one semi-private BC that can then be processed by the SFE framework ABY [DSZ15].

Applications of SPFE

Car Insurance

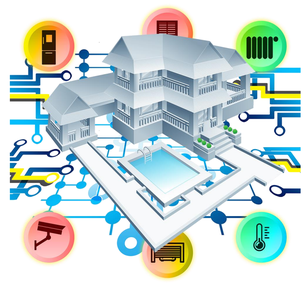
Car insurance companies can calculate different tariffs based on customer‘s private data

Smart Metering

Energy providers can calculate user-specific tariffs based on customer‘s secret data

Credit Worthiness Checking

Protect loanee‘s data and loaner‘s function to decide if the loanee can get a credit

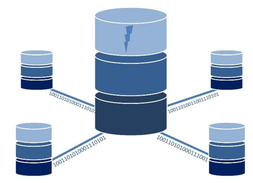


Confidential Information

Companies can check confidential information of customers without revealing any internal processes

Database Management System (DBMS)

Queries in a DBMS can be hidden and the processing function is kept private

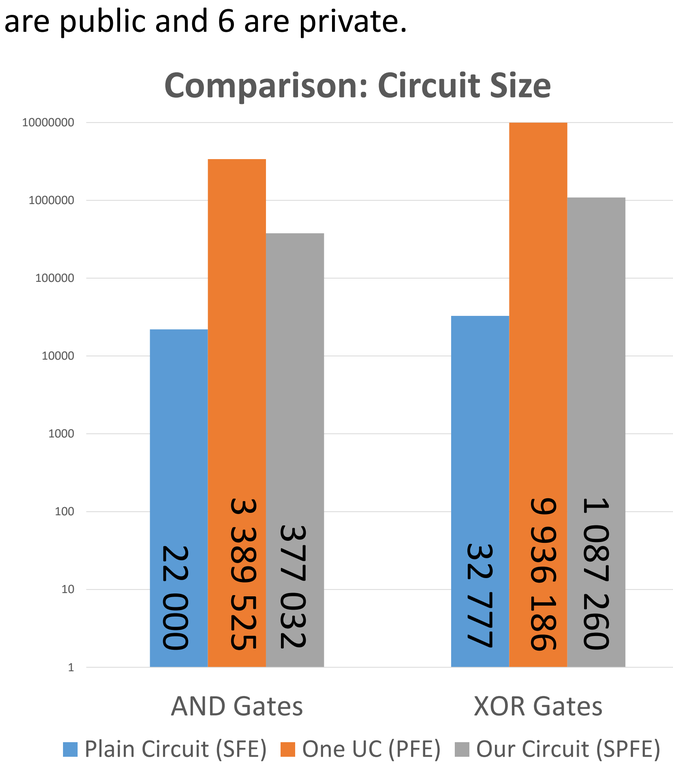


Benchmarks

Car Insurance Application

- Researched insurance rate calculation function to get f.
- Some information (e.g., higher prices for young people) is publicly known, i.e., this information does not have to be hidden in the function.
- Other information (e.g., how tariffs are surcharged for living in a specific location) has to be kept private in UCs.
- Our practice-oriented insurance rate calculation function is split into 15 sub-functions of which 9 are public and 6 are private.

SPFE circuit is up to 9x smaller than one large UC



SPFE is Practical

The resulting performance using Yao‘s garbled circuit protocol is **2,5 seconds** on a LAN and **17,5 MB**.

Architecture