RWTH AACHEN UNIVERSITY

Lecture "Privacy Enhancing Technologies for Data Science" (PETs4DS)
Winter Semester 2016/17
Informatik 5, Lehrstuhl Prof. Decker

Assignment 3

Date for exercise session : 20.12.2016

Information:

1. Participants are encouraged to present their solutions for the exercises.
2. In addition, solutions to the exercises will be presented at the exercise session and published in L2P.
3. Working on the exercises in groups is welcome.
4. The assignments of this lecture series are not graded.
5. Instead, participation in the exams depends on passing the presence exercise in January.

Execise 1: Secure Big-Data Analytics in the Cloud

(a) Shortly explain the idea of using fully homomorphic encryption (FHE) for secure cloud computing. Can you think of a reason why FHE is not a good solution for more than 2 parties?

(b) Think of possible attacks on fully homomorphic encryption and explain verifiable computation in this context.

(c) Explain which properties are provided by secure multiparty computation which are not provided by a combination of fully homomorphic encryption and verified computation.

Exercise 1 : Solution

(a) FHE is an approach for private computation where a party encrypts its data, and then hands the encrypted data to 2nd party for processing. After the 2nd party has processed the encrypted data, it returns the encrypted result to the first party. The first party then decrypts the result using the same key it used to encrypt the input data.
The main reason why FHE is not a good solution for more than 2 parties, is that every party needs to use the same key to encrypt / decrypt the data, so there has to be complete trust between the parties, and any number of adversaries can compromise the network.

(b) FHE is not secure against nodes which want to manipulate the result. Therefore FHE does not leak any private information, but the result of the computation can be made unusable and can lead to false inferences if e.g. decisions are made based on the result. VE can restore the integrity since a node can prove the correctness of the result of the computation which it performed on the input data.

(c) SMPC has a smaller overhead regarding the computational complexity in comparison with FHE and VC. In addition, SMPC eliminates the requirement for sharing private keys between the different parties which need to share the results.

Exercise 2: Secure Multiparty Computation (SMPC) for Exactly Three Parties

(a) Can SMPC be applied to untrusted clouds ? If yes, which circumstances have to be fulfilled ?

(b) Consider the Secure Addition Protocol for three parties from slide 42 of the lecture. Prove the correctness of the result, by showing that the sum of verification shares s_j is the same as the sum of the inputs x_i.

(c) Can the Secure Addition Protocol also be used for two parties? Explain shortly why or why not.

(d) Apply the Secure Multiplication Protocol from slide 49 for the Parties P_1 with input $a = 3$, P_2 with input $b = 15$ and P_3 with no input. The calculation should be done in $\mathbb{F}_{13}$.

(e) Consider P_3 helping P_1 and P_2 to do secure multiplication. Show that the Protocol Secure Multiplication is indeed insecure if he reveals his information to P_1 and P_2.

Exercise 2: Solution

(a) Yes, if a version of SMPC is used which protects against both HBC and malicious adversaries, and if it provides guarantees for both integrity and confidentiality. The SMPC protocols presented in the lecture provide guarantees for integrity and confidentiality, but they do not protect against active attacks by malicious adversaries. The CEPS protocol protects only against passive attacks by t HBC adversaries, for $t < \frac{n}{2}$. The basic SMPC protocols only protect against a single HBC adversary.

(b) Proof that $v = s_1 + s_2 + s_3 \bmod p$ is correct:

$$v = \sum_j s_j \bmod p = \sum_j \sum_i r_{i,j} \bmod p = \sum_i \sum_j r_{i,j} \bmod p = \sum_i x_i \bmod p$$

(c) The usage for two parties makes no sense, because the result can easily be reconstructed. Notice that this isn't a weakness of the protocol, but a property of the function.

(d)
- P_1 chooses $a_1 = 4$ and $a_2 = 5 \Rightarrow a_3 = 3 - 4 - 5 \bmod 13 = 7$
- P_2 chooses $b_1 = 9$ and $b_2 = 1 \Rightarrow b_3 = 15 - 9 - 1 \bmod 13 = 5$
- P_1 and P_2 distribute a_1, a_2, a_3 and b_1, b_2, b_3
- P_1 calculates $u_1 = a_2 * b_2 + a_2 * b_3 + a_3 * b_2 \bmod 13 = 5 * 1 + 1 * 7 + 5 * 5 \bmod 13 = 11$

- P_2 calculates $u_2 = 7 * 5 + 4 * 5 + 9 * 7 \bmod 13 = 1$
- P_3 calculates $u_3 = 4 * 9 + 4 * 1 + 5 * 9 \bmod 13 = 7$
- Use secure addition for $u_1 + u_2 + u_3 \bmod 13 = u$
- $u = 11 + 1 + 7 \bmod 13 = 6 = 3 * 15 \mod 13 = 3 * 2 \mod 13$

(e) P_3 sends all information to P_1 and P_2

$\Rightarrow$ P_1 now has the information of b_1, b_2 and b_3

$\Rightarrow$ P_1 can calculate $b_1 + b_2 + b_3 \bmod p = s \bmod p$

$\Rightarrow$ P_2 can calculate the secret of P_1 correspondingly

Exercise 3: Lagrange Interpolation and the Recombination Vector

Determine the recombination vector r for the following setting:

- Use $\mathbb{F}_7$ for the calculation.
- $t = 2$
- There are seven parties.
- Parties P_2, P_5 and P_7 are colluding.

Exercise 3 : Solution

Given the specified setting, we have $\mathbb{F}_7$ and $C = \{2, 5, 7\}$

$$\delta_2(0) = \frac{0-5}{2-5} * \frac{0-7}{2-7} = 0$$

$$\delta_5(0) = \frac{0-2}{5-2} * \frac{0-7}{5-7} = 0$$

$$\delta_7(0) = \frac{0-2}{7-2} * \frac{0-5}{7-5} = \frac{-2}{-5} * \frac{-5}{-2} = 1$$

$\Rightarrow\ r = (0, 0, 1)$

Bonus question: Why can these delta functions not be used for SMPC?

SMPC requires the prime p which is used for the finite field $\mathbb{F}_p$ to be bigger than the maximum number of players n, so $p > n$. So for 7 players, $p >= 11$ has to hold.

Exercise 4: SMPC Using the Circuit Evaluation with Passive Security (CEPS) Protocol

(a) Shortly explain the four steps of the CEPS-Protocol.

(b) What is the advantage of describing a function as an arithmetic circuit?

(c) Consider the Parties P_1, P_2, P_3 with the inputs $I_1 = 4, I_2 = 1$ and $I_3 = 9$. The function $f(x) = (I_1 + I_2) * (5 * I_3)$ should be calculated in $\mathbb{F}_{11}$ and $t = 1$.

- Draw the corresponding arithmetic circuit.
- Evaluate the circuit using the CEPS protocol. Also determine the result by using the output reconstruction step.

(d) How is the interim result stored after every evaluation step ?

(e) How would you need to change the evaluation of the circuit if there is an additional fourth party P_4, independent of the changes to $f(x)$?

(f) How many nodes can be untrusted when using CEPS?

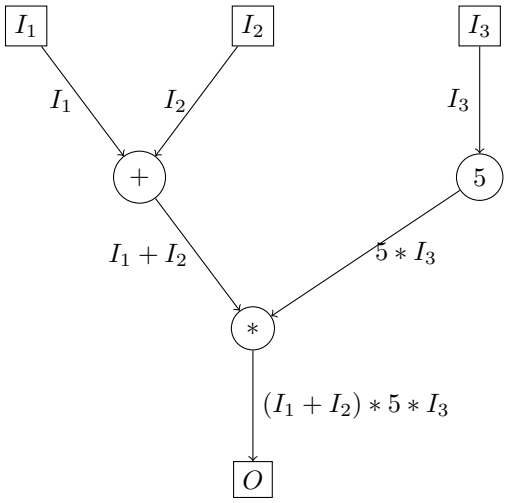
(g) Can you think of possible attacks on CEPS?

Exercise 4: Solution

(a) Chapter 4, Slides 82 - 85

(b) Any function can be represented by an arithmetic circuit. The representation through the three gate-types allow a simplified handling of complex functions.

(c) Arithmetic circuit :



- P_1 distributes 4 with $f_1(x) = 4 + x \Rightarrow [5, 6, 7]$
  P_2 distributes 1 with $f_2(x) = 1 + 2x \Rightarrow [3, 5, 7]$
  P_3 distributes 9 with $f_3(x) = 9 + x \Rightarrow [10, 0, 1]$
- $$\begin{array}{c|ccc} & I_1 & I_2 & I_3 \\ P_1 & 5 & 3 & 10 \\ P_2 & 6 & 5 & 0 \\ P_3 & 7 & 7 & 1 \end{array} \rightsquigarrow \begin{array}{c|cc} & I_1 + I_2 & I_3 \\ P_1 & 8 & 10 \\ P_2 & 0 & 0 \\ P_3 & 3 & 1 \end{array} \rightsquigarrow \begin{array}{c|cc} & I_1 + I_2 & 5 * I_3 \\ P_1 & 8 & 6 \\ P_2 & 0 & 0 \\ P_3 & 3 & 5 \end{array}$$

  $$\rightsquigarrow \begin{array}{c|c} & (I_1 + I_2) * 5 * I_3 \\ P_1 & 4 \\ P_2 & 0 \\ P_3 & 4 \end{array}$$

- P_1 distributes 4 with $g_1(x) = 4 + x \Rightarrow [5, 6, 7]$
  P_2 distributes 0 with $g_2(x) = 0 + 2x \Rightarrow [2, 4, 6]$
  P_3 distributes 4 with $g_3(x) = 4 + x \Rightarrow [5, 6, 7]$
- From $\mathbb{F}_{11}$ and $C = \{1, 2, 3\}$ follows that $r_1 = (3, 8, 1)$
- P_1 calculates $3 * 5 + 8 * 2 + 1 * 5 = 3$
  P_2 calculates $3 * 6 + 8 * 4 + 1 * 6 = 1$
  P_3 calculates $3 * 7 + 8 * 6 + 1 * 7 = 10$
- For the output reconstruction :
  - Each party sends their share of the final result securely to all other parties. We assume that P_1 wants to reconstruct the result using his share, and the share of P_2.
  - From $\mathbb{F}_{11}$ and $C = \{1, 2\}$ follows that $r_2 = (2, 10)$
  - $2 * 3 + 10 * 1 = 6 + 10 = 5$ which is the same as the result of evaluating the circuit in a centralised way: $O = 5 * 5 * 9 = 5$

(d) The interim result is stored in the y-intercept of the function shared by all parties.

(e) In order to evaluate a function with four parties, every party has to share their secrets four ways. There are no other changes to the evaluation necessary. In particular, the same recombination vectors can be used, as they are only based on the indices of the parties who combine their input to reconstruct a shared secret.

(f) The maximum number of untrusted nodes is the degree of the polynom t, because the result can be restored with at least $t + 1$ nodes.

(g) CEPS uses passive security. Therefore active attacks, like sending false information destroy, the integrity of the protocol.