

Advanced Topics on Privacy Enhancing Technologies

CS523

Privacy-preserving Crypto I Exercises

Exercise 1

MPC and Arithmetic Sharing

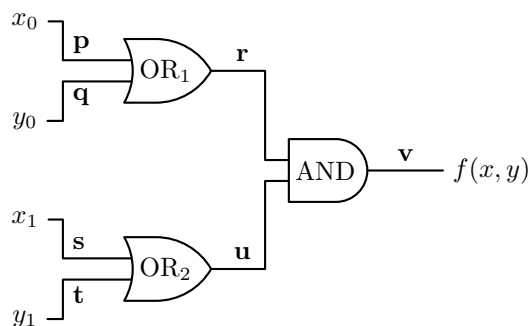
Compute by hand a MPC protocol using arithmetic secret sharing in a specific ring. You will assume Beaver triplets were generated by a trusted third party.

Select a modular ring to work with. Compute *by hand* circuits 1 and 7 from the SMC project.

Exercise 2

Garbled Circuits

The goal of this exercise is to gain some familiarity with Yao's garbled circuits. We will use Yao's scheme to allow two parties A and B with respective 2-bit inputs $x = x_0x_1$ and $y = y_0y_1$ to compute the function $f(x, y) = (x_0 \vee y_0) \wedge (x_1 \vee y_1)$. We use a simple encryption scheme based on the one-time pad over 4 bits. For a 4-bit key k and 4-bit plain text x , we have $E_k(x) = x \oplus k$.



The above figure represents the circuit of f . A 's input bits (x_0 and x_1) are on the input wires $\mathbf{p}$ and $\mathbf{s}$. B 's input bits (y_0 and y_1) are on the input wires $\mathbf{q}$ and $\mathbf{t}$.

a) You take the place of party A who is in charge of generating the garbled circuit. For the gate OR_1 , you generate the following random keys:

Input wire $\mathbf{p}$: $k_{\mathbf{p}}^0 = 1010$ $k_{\mathbf{p}}^1 = 1011$

Input wire $\mathbf{q}$: $k_{\mathbf{q}}^0 = 0101$ $k_{\mathbf{q}}^1 = 1111$

Output wire $\mathbf{r}$: $k_{\mathbf{r}}^0 = 1001$ $k_{\mathbf{r}}^1 = 1100$

The keys $k_{\mathbf{p}}^i$ are *garbled values* for the two possible values of bit x_0 . The keys $k_{\mathbf{q}}^i$ are garbled

values for the two possible values of bit y_0 . Finally, keys $k_{\mathbf{r}}^i$ are garbled values for the two possible output values of the gate.

Complete the garbled gate OR_1 below.

x_0	y_0	encryption	result
0	0	$k_{\mathbf{p}}^0 \oplus k_{\mathbf{q}}^0 \oplus k_{\mathbf{r}}^0$	0110
0	1		
1	0		
1	1		

b) You now take the place of party B to evaluate the circuit. From A , you receive all the garbled gates (OR_1 , OR_2 and AND). You also receive A 's garbled inputs $k_{\mathbf{p}} = 1011$ and $k_{\mathbf{s}} = 0001$. We assume that B 's input is $y_0 = 0, y_1 = 0$. After running an oblivious transfer protocol with A , you obtain the garbled inputs $k_{\mathbf{q}} = 0101$ and $k_{\mathbf{t}} = 1101$.

At this point, B can't know the values of x_0 and x_1 because the garbled inputs $k_{\mathbf{p}}$ and $k_{\mathbf{s}}$ are random. The security of the oblivious transfer guarantees that A does not know y_0 and y_1 .

The encrypted values for the garbled gate OR_2 are:

0111
1110
 0010
 1000

You know that exactly one of these values is the encryption of the garbled output $k_{\mathbf{u}}$ under keys $k_{\mathbf{s}}$ and $k_{\mathbf{t}}$. The problem is that you have no idea which one to decrypt! Yao's protocol actually requires the encryption scheme to be *checkable*, meaning that it should be easy for B to check whether a given cipher was indeed encrypted with a given key. The one time pad does not have this property, so to make things easier for you, we indicate (in bold) which entry in the garbled gate you should decrypt.

The garbled gate AND (that takes as inputs $k_{\mathbf{r}}$ and $k_{\mathbf{u}}$) is:

1110
 0010
 0101
 0110

- What is the value of the garbled output $k_{\mathbf{u}}$ of OR_2 ?
- The garbled output of the gate OR_1 is $k_{\mathbf{r}} = 1100$. What is the garbled output of the circuit?
- Party A now reveals that the garbled outputs of the circuit are $k_{\mathbf{v}}^0 = 1111$ and $k_{\mathbf{v}}^1 = 0000$. What is the value of $f(x, y)$? What can B infer about x from $f(x, y)$? Could B also have inferred this information if B 's input was something else than $y_0 = y_1 = 0$?

Solutions to the Exercises

Solution 1

Solution coded in the project.

Solution 2

	x_0	y_0	encryption	result
	0	0	$k_{\mathbf{p}}^0 \oplus k_{\mathbf{q}}^0 \oplus k_{\mathbf{r}}^0$	0110
a)	0	1	$k_{\mathbf{p}}^0 \oplus k_{\mathbf{q}}^1 \oplus k_{\mathbf{r}}^1$	1001
	1	0	$k_{\mathbf{p}}^1 \oplus k_{\mathbf{q}}^0 \oplus k_{\mathbf{r}}^1$	0010
	1	1	$k_{\mathbf{p}}^1 \oplus k_{\mathbf{q}}^1 \oplus k_{\mathbf{r}}^1$	1000

Only the ‘result’ column is sent to party B . The rows of the table should first be randomly permuted to prevent party B from being able to infer the value of the garbled output from its position in the table.

b)

- Decrypting 1110 under keys $k_{\mathbf{s}}$ and $k_{\mathbf{t}}$ gives $1110 \oplus 0001 \oplus 1101 = 0010 = k_{\mathbf{u}}$.
- Decrypting 1110 under keys $k_{\mathbf{r}}$ and $k_{\mathbf{u}}$ gives $1110 \oplus 1100 \oplus 0010 = 0000 = k_{\mathbf{v}}$.
- The garbled output of f is 0000, so $f(x, y) = 1$. It thus must be the case that $x_0 = x_1 = 1$. For any other input from B , at least one of the inputs bits from A would remain secret. The security of Yao’s protocol guarantees that A and B learn *nothing more* about each others’ input, than what they can infer directly from the output!