APPLICATIONS AND PROTOCOLS

Some applications and protocols

- Internet Casino
- Commitment
- Shared coin flips
- Threshold cryptography
- Forward security
- Program obfuscation
- Zero-knowledge
- Certified e-mail
- Electronic voting
- Auctions

- Identity-based encryption
- Functional encryption
- Fully-homomorphic encryption
- Searchable encryption
- Oblivious transfer
- Garbling schemes
- Secure computation
- Group signatures
- Aggregate signatures

Internet Casino: Protocol G1

Player $\qquad$ Casino

$\xrightarrow{\$1,\, G}$

$T \stackrel{\$}{\leftarrow} \{1, 2, \ldots, 100\}$
if $G = T$ then $d \leftarrow \$200$
else $d \leftarrow \$0$

$\xleftarrow{T,\, d}$

Would you play?

Expected value of d is $\$200(\frac{1}{100}) = \$2 > \$1$ so probability theory says that the player will earn money by playing.

Problem: Casino can cheat

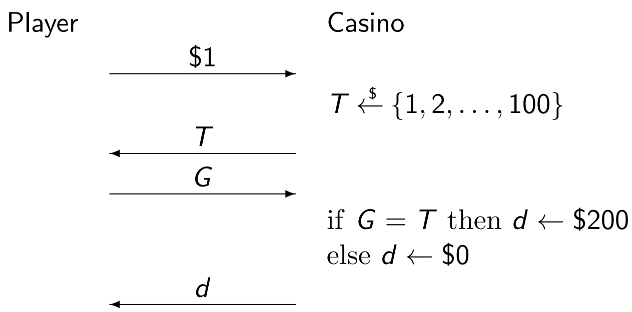
Player $\qquad$ Casino

$\xrightarrow{\$1,\, G}$

$T \stackrel{\$}{\leftarrow} \{1, 2, \ldots, 100\} \setminus \{G\}$
$d \leftarrow \$0$

$\xleftarrow{T,\, d}$

Internet Casino: Protocol G2



But now player can always win by setting $G = T$. No casino would do this!

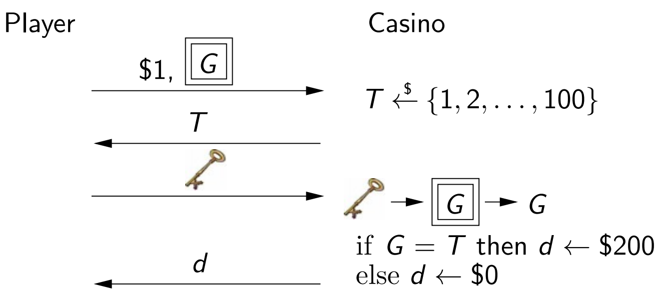
Internet Casino problem

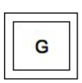
Player and Casino need to exchange G, T so that

- Casino cannot choose T as a function of G.
- Player cannot choose G as a function of T.

How do we resolve this Catch-22 situation?

"Internet" Casino: Protocol G3

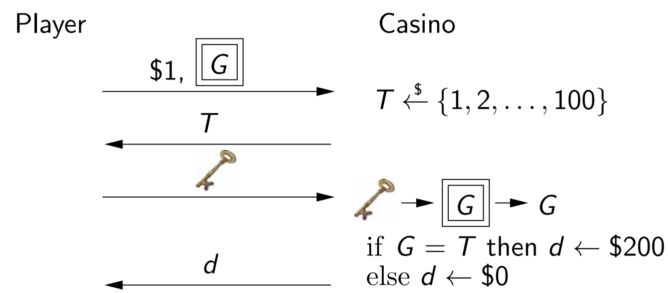




is a locked safe containing a piece of paper with G written on it.

 is a key to open the safe.

"Internet" Casino: Protocol G3



- Casino cannot choose T as a function of G because, without the key, it cannot see G.
- Player cannot choose G as a function of T because, by putting it in the safe, she is committed to it in the first move.

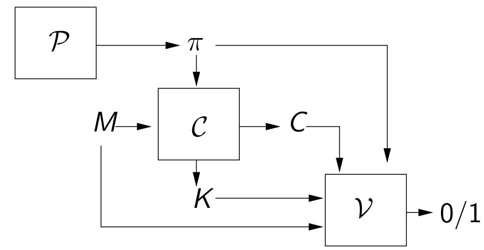
Internet Casino Protocol using cryptography

Player		Casino
$K \xleftarrow{\$} \{0,1\}^{128}$		
$C \leftarrow H(K\|G)$	$\xrightarrow{\quad C \quad}$	
		$T \xleftarrow{\$} \{1,2,\ldots,100\}$
	$\xleftarrow{\quad T \quad}$	
	$\xrightarrow{\quad G,K \quad}$	
		if $(H(K\|G) = C$ and $G = T)$ then $d \leftarrow \$99$ else $d \leftarrow \$0$
	$\xleftarrow{\quad d \quad}$	

Here H is a cryptographic hash function. More generally one can use a primitive called a *committment scheme*.

Commitment Schemes

A commitment scheme $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ is a triple of algorithms



Parameter generation algorithm P is run once by a trusted party to produce public parameters π.

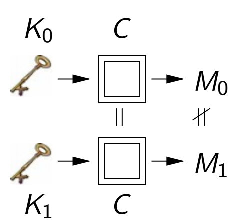
		In Internet Casino
M	Data being commited	G
C	Commital	$\boxed{G}$
K	Decommital key	

Security properties

- Hiding: A commital C generated via $(C,K) \xleftarrow{\$} \mathcal{C}(\pi, M)$ should not reveal information about M.

  $\boxed{\boxed{M}}$ $= C$ does not reveal M.

- Binding: It should be hard to find C, M_0, M_1, K_0, K_1 such that $M_0 \neq M_1$ but $\mathcal{V}(\pi, C, M_0, K_0) = \mathcal{V}(\pi, C, M_1, K_1) = 1$.



Internet Casino Protocol using a commitment scheme

Player		Casino
$(C,K) \xleftarrow{\$} \mathcal{C}(\pi, G)$	$\xrightarrow{\quad C \quad}$	
		$T \xleftarrow{\$} \{1,2,\ldots,100\}$
	$\xleftarrow{\quad T \quad}$	
	$\xrightarrow{\quad G,K \quad}$	
		if $(\mathcal{V}(\pi, C, G, K) = 1$ and $G = T)$ then $d \leftarrow \$99$ else $d \leftarrow \$0$
	$\xleftarrow{\quad d \quad}$	

Hiding Formally

Let $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ be a commitment scheme and A an adversary.

Game $\mathrm{HIDE}_{\mathcal{CS}}$	**procedure LR**(M_0, M_1)
procedure Initialize	$(C, K) \xleftarrow{\$} \mathcal{C}(\pi, M_b)$
$\pi \xleftarrow{\$} \mathcal{P}$; $b \xleftarrow{\$} \{0,1\}$	return C
return π	**procedure Finalize**(b')
	return $(b = b')$

The hiding-advantage of A is

$$\mathbf{Adv}^{\mathrm{hide}}_{\mathcal{CS}}(A) = 2 \cdot \Pr\left[\mathrm{HIDE}^A_{\mathcal{CS}} \Rightarrow \mathsf{true}\right] - 1.$$

Binding Formally

Let $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ be a commitment scheme and A an adversary.

Game $\mathrm{BIND}_{\mathcal{CS}}$	**procedure Finalize**(C, M_0, M_1, K_0, K_1)
procedure Initialize	$v_0 \leftarrow \mathcal{V}(\pi, C, M_0, K_0)$
$\pi \xleftarrow{\$} \mathcal{P}$	$v_1 \leftarrow \mathcal{V}(\pi, C, M_1, K_1)$
return π	return $(v_0 = v_1 = 1$ and $M_0 \neq M_1)$

The binding-advantage of A is

$$\mathbf{Adv}^{\mathrm{bind}}_{\mathcal{CS}}(A) = \Pr\left[\mathrm{BIND}^A_{\mathcal{CS}} \Rightarrow \mathsf{true}\right].$$

Commitment from symmetric encryption

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an IND-CPA-secure symmetric encryption scheme and let $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ be the commitment scheme where $\mathcal{P}$ returns $\pi = \varepsilon$ and

Alg $\mathcal{C}(\pi, M)$	**Alg** $\mathcal{V}(\pi, C, M, K)$
$K \xleftarrow{\$} \mathcal{K}$; $C \xleftarrow{\$} \mathcal{E}_K(M)$	if $\mathcal{D}_K(C) = M$ then return 1
return (C, K)	else return 0

Is this secure?

Commitment from symmetric encryption

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an IND-CPA-secure symmetric encryption scheme and let $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ be the commitment scheme where $\mathcal{P}$ returns $\pi = \varepsilon$ and

Alg $\mathcal{C}(\pi, M)$	**Alg** $\mathcal{V}(\pi, C, M, K)$
$K \xleftarrow{\$} \mathcal{K}$; $C \xleftarrow{\$} \mathcal{E}_K(M)$	if $\mathcal{D}_K(C) = M$ then return 1
return (C, K)	else return 0

Is this secure?

- Certainly hiding since $\mathcal{SE}$ is IND-CPA.

Commitment from symmetric encryption

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an IND-CPA-secure symmetric encryption scheme and let $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ be the commitment scheme where $\mathcal{P}$ returns $\pi = \varepsilon$ and

Alg $\mathcal{C}(\pi, M)$	**Alg** $\mathcal{V}(\pi, C, M, K)$
$K \xleftarrow{\$} \mathcal{K}$; $C \xleftarrow{\$} \mathcal{E}_K(M)$	if $\mathcal{D}_K(C) = M$ then return 1
return (C, K)	else return 0

Is this secure?

- Certainly hiding since $\mathcal{SE}$ is IND-CPA.
- But need not be binding: it may be possible to find C, M_0, M_1, K_0, K_1 such that $\mathcal{D}_{K_0}(C) = M_0$ and $\mathcal{D}_{K_1}(C) = M_1$.

  Exercise: Show such a binding-violating attack when $\mathcal{SE}$ is the CBC\$ scheme.

Surfacing randomness in asymmetric encryption

Let $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an asymmetric encryption scheme. Then $\mathcal{E}_{pk}(M; K)$ is the result of encrypting M with coins (randomness) set to K. Thus, the following processes return the same thing:

$C \xleftarrow{\$} \mathcal{E}_{pk}(M)$	$K \xleftarrow{\$} \mathrm{Coins}(pk)$; $C \leftarrow \mathcal{E}_{pk}(M; K)$
Return C	Return C

Here $\mathrm{Coins}(pk)$ is the space from which the randomness (coins) are drawn.

Example: With the SRSA scheme, $\mathrm{Coins}((N, e)) = \mathbf{Z}_N^*$ and

Alg $\mathcal{E}_{N,e}(M; K)$	**Alg** $\mathcal{E}_{N,e}(M)$
$C_a \leftarrow K^e \bmod N$	$K \xleftarrow{\$} \mathbf{Z}_N^*$
$C_s \leftarrow H(K) \oplus M$	$C_a \leftarrow K^e \bmod N$
return (C_a, C_s)	$C_s \leftarrow H(K) \oplus M$
	return (C_a, C_s)

Commitment from public key encryption

Let $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an IND-CPA-secure asymmetric encryption scheme and let $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ be the commitment scheme where

Alg $\mathcal{P}$	**Alg** $\mathcal{C}(pk, M)$	**Alg** $\mathcal{V}(pk, C, M, K)$
$(pk, sk) \xleftarrow{\$} \mathcal{K}$	$K \xleftarrow{\$} \mathrm{Coins}(pk)$	if $K \notin \mathrm{Coins}(pk)$ then return 0
$\pi \leftarrow pk$	$C \leftarrow \mathcal{E}_{pk}(M; K)$	if $\mathcal{E}_{pk}(M; K) = C$ then return 1
return π	return (C, K)	else return 0

Is this secure?

Commitment from public key encryption

Let $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an IND-CPA-secure asymmetric encryption scheme and let $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ be the commitment scheme where

Alg $\mathcal{P}$	**Alg** $\mathcal{C}(pk, M)$	**Alg** $\mathcal{V}(pk, C, M, K)$
$(pk, sk) \xleftarrow{\$} \mathcal{K}$	$K \xleftarrow{\$} \mathrm{Coins}(pk)$	if $K \notin \mathrm{Coins}(pk)$ then return 0
$\pi \leftarrow pk$	$C \leftarrow \mathcal{E}_{pk}(M; K)$	if $\mathcal{E}_{pk}(M; K) = C$ then return 1
return π	return (C, K)	else return 0

Is this secure?

- Certainly hiding since $\mathcal{AE}$ is IND-CPA.

Commitment from public key encryption

Let $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an IND-CPA-secure asymmetric encryption scheme and let $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ be the commitment scheme where

Alg $\mathcal{P}$	**Alg** $\mathcal{C}(pk, M)$	**Alg** $\mathcal{V}(pk, C, M, K)$
$(pk, sk) \xleftarrow{\$} \mathcal{K}$	$K \xleftarrow{\$} \mathrm{Coins}(pk)$	if $K \notin \mathrm{Coins}(pk)$ then return 0
$\pi \leftarrow pk$	$C \leftarrow \mathcal{E}_{pk}(M; K)$	if $\mathcal{E}_{pk}(M; K) = C$ then return 1
return π	return (C, K)	else return 0

Is this secure?

- Certainly hiding since $\mathcal{AE}$ is IND-CPA.
- Binding too since C has only one decryption relative to pk, namely $M = \mathcal{D}_{sk}(C)$.

Commitment from hashing

Let H be a hash function and $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ the commitment scheme where $\mathcal{P}$ returns $\pi = \varepsilon$ and

Alg $\mathcal{C}(\pi, M)$	**Alg** $\mathcal{V}(\pi, C, M, K)$
$C \leftarrow H(M)$; $K \leftarrow M$	return $(C = H(M)$ *and* $M = K)$
return (C, K)	

This is

Commitment from hashing

Let H be a hash function and $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ the commitment scheme where $\mathcal{P}$ returns $\pi = \varepsilon$ and

Alg $\mathcal{C}(\pi, M)$	**Alg** $\mathcal{V}(\pi, C, M, K)$
$C \leftarrow H(M)$; $K \leftarrow M$	return $(C = H(M)$ *and* $M = K)$
return (C, K)	

This is

- Binding if H is collision-resistant.

Commitment from hashing

Let H be a hash function and $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ the commitment scheme where $\mathcal{P}$ returns $\pi = \varepsilon$ and

Alg $\mathcal{C}(\pi, M)$	**Alg** $\mathcal{V}(\pi, C, M, K)$
$C \leftarrow H(M)$; $K \leftarrow M$	return $(C = H(M)$ *and* $M = K)$
return (C, K)	

This is

- Binding if H is collision-resistant.
- But not hiding. For example in the Internet Casino $M = G \in \{1, ..., 100\}$ so given $C = H(M)$ the casino can recover M via

  for $i = 1, ..., 100$ do
      if $H(i) = C$ then return i

Commitment from hashing

A better scheme is $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ where $\mathcal{P}$ returns $\pi = \varepsilon$ and

Alg $\mathcal{C}(\pi, M)$	**Alg** $\mathcal{V}^H(\pi, C, M, K)$
$K \xleftarrow{\$} \{0,1\}^{128}$	return $(H(K\|\|M) = C)$
$C \leftarrow H(K\|\|M)$	
return (C, K)	

Commitment schemes usage

Commitment schemes are very broadly and widely used across all kinds of protocol design and in particular to construct zero-knowledge proofs.

Flipping a common coin

- Alice and Bob are getting divorced
- They want to decide who keeps the Lexus
- They aggree to flip a coin, but
- Alice is in NY and Bob is in LA

Protocol CF1:

Alice		Bob
$c \xleftarrow{\$} \{0,1\}$	$\xrightarrow{\quad c \quad}$	

Flipping a common coin

- Alice and Bob are getting divorced
- They want to decide who keeps the Lexus
- They aggree to flip a coin, but
- Alice is in NY and Bob is in LA

Protocol CF1:

Alice		Bob
$c \xleftarrow{\$} \{0,1\}$	$\xrightarrow{\quad c \quad}$	

Bob is not too smart but he doesn't like it...

Can you help them out?

Protocol CF2

Let $\mathcal{CS} = (\mathcal{P}, \mathcal{C}, \mathcal{V})$ be a commitment scheme.

$$\begin{array}{rcl}
\text{Alice} & & \text{Bob} \\
a \xleftarrow{\$} \{0,1\} & & \\
(C,K) \xleftarrow{\$} \mathcal{C}(\pi, a) & \xrightarrow{\quad C \quad} & b \xleftarrow{\$} \{0,1\} \\
& \xleftarrow{\quad b \quad} & \\
& \xrightarrow{\quad a,K \quad} & \text{if } \mathcal{V}(\pi, C, a, K) = 1 \text{ then} \\
c \leftarrow a \oplus b & & \quad c \leftarrow a \oplus b \\
& & \text{else} \\
& & \quad c \leftarrow \perp
\end{array}$$

c is the common coin. Neither party can control it.

Protocol CF3: Concrete instantiation of CF2

$$\begin{array}{rcl}
\text{Alice} & & \text{Bob} \\
a \xleftarrow{\$} \{0,1\} \ ; \ K \xleftarrow{\$} \{0,1\}^{128} & & \\
C \leftarrow H(K \| a) & \xrightarrow{\quad C \quad} & b \xleftarrow{\$} \{0,1\} \\
& \xleftarrow{\quad b \quad} & \\
& \xrightarrow{\quad a,K \quad} & \text{if } H(K \| a) = C \text{ then} \\
c \leftarrow a \oplus b & & \quad c \leftarrow a \oplus b \\
& & \text{else} \\
& & \quad c \leftarrow \perp
\end{array}$$

c is the common coin. Neither party can control it. H is a cryptographic hash function.

Secure summation

Suppose we have n parties $1, ..., n$
Party i has an integer x_i
The parties want to know the value of

$$f(x_1, .., x_n) = x_1 + ... + x_n$$

Secure summation

Suppose we have n parties $1, ..., n$
Party i has an integer x_i
The parties want to know the value of

$$f(x_1, .., x_n) = x_1 + ... + x_n$$

Easy: Let

- Party i send x_i to party 1 $(2 \leq i \leq n)$
- Party 1 computes $f(x_1, .., x_n) = x_1 + ... + x_n$ and broadcasts it

Secure summation

Suppose we have n parties $1, \ldots, n$
Party i has an integer x_i
The parties want to know the value of

$$f(x_1, .., x_n) = x_1 + \ldots + x_n$$

Easy: Let

- Party i send x_i to party 1 $(2 \le i \le n)$
- Party 1 computes $f(x_1, .., x_n) = x_1 + \ldots + x_n$ and broadcasts it

What they don't like about this: Party 1 now knows everyone's values

Privacy constraint: Party i does not wish to reveal x_i

Secure summation

Party i has input x_i $(1 \le i \le n)$. The parties want to know $f(x_1, .., x_n) = x_1 + \ldots + x_n$ but do not want to reveal their inputs in the process.

Scenarios:

- x_i = score of student i on midterm exam
- x_i = salary of employee i
- $x_i \in \{0, 1\}$ = vote of voter i on proposition X on ballot

The model and goal

Parties i, j are connected via a secure channel $(1 \le i, j \le n)$.
Privacy and authenticity of messages sent over channel are guaranteed.

The parties will exchange messages to arrive at $f(x_1, \ldots, x_n)$.

If $i \neq j$ then, at the end of the protocol, party i should not know x_j.

For example you, as player i, enter x_i into some app on your cellphone which then communicates with the cellphones of the other parties. At the end, the sum shows up on your screen. Take your phone apart and examine all memory contents and you still will not discover x_j for $j \neq i$.

Setup for secure communication protocol

Let N be such that $x_1, \ldots, x_n \in Z_N = \{0, \ldots, N-1\}$.
Let $M = nN$.
Let S denote $x_1 + \ldots + x_n$.
We will compute $S \bmod M$, which is just S since

$$x_1 + \ldots + x_n \le n(N-1) < M$$

Protocol step 1: secret sharing

For $i = 1, \ldots, n$ party i

- Picks $x_{i,1}, \ldots, x_{i,n} \in Z_M$ at random subject to $x_{i,1} + \ldots + x_{i,n} \equiv x_i \pmod{M}$
- Sends $x_{i,j}$ to party j over secure channel $(1 \le j \le n)$

$$\begin{array}{cc} \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} \\ x_{3,1} & x_{3,2} & x_{3,3} & x_{3,4} \\ x_{4,1} & x_{4,2} & x_{4,3} & x_{4,4} \end{bmatrix} & \begin{array}{l} \to x_1 \\ \to x_2 \\ \to x_3 \\ \to x_4 \end{array} \end{array}$$

Observation: $x_{i,j}$ is a random number unrelated to x_i so party j has no information about x_i $(i \ne j)$

Protocol step 2,3: Column sums and conclusion

$$\begin{array}{cc} \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} \\ x_{3,1} & x_{3,2} & x_{3,3} & x_{3,4} \\ x_{4,1} & x_{4,2} & x_{4,3} & x_{4,4} \end{bmatrix} & \begin{array}{l} \to x_1 \\ \to x_2 \\ \to x_3 \\ \to x_4 \end{array} \\ \begin{array}{cccc} \downarrow & \downarrow & \downarrow & \downarrow \\ C_1 & C_2 & C_3 & C_4 \end{array} & \end{array}$$

For $j = 1, \ldots, n$ party j

- Computes $C_j = (x_{1,j} + x_{2,j} + \ldots + x_{n,j}) \bmod M$
- Sends C_j to party i $(1 \le i \le n)$

Observation: $S \equiv (C_1 + \cdots + C_n) \pmod{M}$.

So each party can compute $S \leftarrow (C_1 + \ldots + C_n) \bmod M$

Security of the protocol

$$\begin{array}{cc} \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} \\ x_{3,1} & x_{3,2} & x_{3,3} & x_{3,4} \\ x_{4,1} & x_{4,2} & x_{4,3} & x_{4,4} \end{bmatrix} & \begin{array}{l} \to x_1 \\ \to x_2 \\ \to x_3 \\ \to x_4 \end{array} \\ \begin{array}{cccc} \downarrow & \downarrow & \downarrow & \downarrow \\ c_1 & c_2 & c_3 & c_4 \end{array} & \end{array}$$

At end of protocol, party 1 knows (1) x_1, and the first-row entries of the matrix (2) the sum $S = x_1 + x_2 + x_3 + x_4$ (3) c_1, c_2, c_3, c_4 (4) the first column entries $x_{1,1}, x_{2,1}, x_{3,1}, x_{4,1}$.

Claims:

- Party 1 learn nothing about x_4
- Even if parties 1, 2 pool their information, they learn nothing about x_4
- ...

Secure summation project

Project: Analyze and prove secure the summation protocol: (1) Give a game based definition of privacy (2) Prove that the protocol meets it.

Secure Computation

Parties $1, \ldots, n$
Party i has private input x_i
They want to compute $f(x_1, \ldots, x_n)$

Fact: For any function f, there is a $n/2$ - private protocol to compute it.

A protocol is t-private if any t parties, getting together, cannot figure out anything about the input of the other parties other than implied by the value of $f(x_1, \ldots, x_n)$.

The protocol views f as a circuit (program) and computes it gate (instruction) by gate (instruction).

Enormous body of research.

Zero-Knowledge Proofs [GMR]

A zero-knowledge (ZK) proof allows you to

- Convince Bob your claim is true
- Without revealing anything beyond that

For example:

You claim to have	**Bob is**	**What is not revealed**
A solution to the homework problem	Another student	The solution
The password for this account	The server	The password
A proof that $\mathbf{P} \neq \mathbf{NP}$	The Clay Institute	The proof