Cryptographic Protocols
Spring 2017
Part 7

Commit State

Given: \( f(x, y) = f_{00} + f_{10}x + f_{01}y + \ldots + f_{tt}x^ty^t \in \mathbb{F}[x, y] \)

Fact 1: \( f_{00}(x) := f(x, y) \) is a one-dimensional polynomial of degree \( t \).

Proof: 
\[
\begin{align*}
\text{Fact 1: } f_{00}(x) &:= f(x, y) \\
&= (f_{00} + f_{01}y + \ldots + f_{0t}y^t) \\
&+ (f_{10} + f_{11}y^2 + \ldots + f_{1t}y^{t+1})x \\
&+ \ldots \\
&+ (f_{t0} + f_{t1}y^{2t} + \ldots + f_{tt}y)^t \\
\end{align*}
\]

Proof: 

Open Protocol

1. Open
\( D \) broadcasts \( g(x) \).

2. Check consistency
\( P_i \) accuses dealer if \( g(\alpha_i) \neq s_i \), transmission

3. Compute secret
\[
\begin{align*}
\text{If } \leq t \text{ accusations: } s &= g(0) \\
\text{If } > t \text{ accusations: } \text{disqualify dealer.}
\end{align*}
\]

Proof:

Generic Commitment Multiplication Protocol

0. Starting point: \( D \) is committed to \( a, b, c \) by \( a_1, b_1, \) and \( c_1 \).

1. CSP of \( a, b \) with degree \( t \)
\[
\Rightarrow f(x), g(x)
\]

2. CSP of \( c \) with degree \( 2t \)
use \( h(x) = f(x)g(x) \)

3. Checks
\[
\forall P_i: d_i = a_i b_i \text{ broadcast accusation bit.}
\]

On accusation: Open \( a_1, b_1, d_1 \), check \( a_1 b_1 = d_1 \).

Facts about two-dimensional polynomials (1/3)

Given: \( f(x, y) = f_{00} + f_{10}x + f_{01}y + \ldots + f_{tt}x^ty^t \in \mathbb{F}[x, y] \)

Fact 1: \( f_{00}(x) := f(x, y) \) is a one-dimensional polynomial of degree \( t \).

Proof: 
\[
\begin{align*}
\text{Fact 1: } f_{00}(x) &:= f(x, y) \\
&= (f_{00} + f_{01}y + \ldots + f_{0t}y^t) \\
&+ (f_{10} + f_{11}y^2 + \ldots + f_{1t}y^{t+1})x \\
&+ \ldots \\
&+ (f_{t0} + f_{t1}y^{2t} + \ldots + f_{tt}y)^t \\
\end{align*}
\]

Facts about two-dimensional polynomials (2/3)

Given: \( f(x, y) = f_{00} + f_{10}x + f_{01}y + \ldots + f_{tt}x^ty^t \in \mathbb{F}[x, y] \)

Fact 2: Let \( X = \{x_1, \ldots, x_{t+1}\} \) and \( Y = \{y_1, \ldots, y_{t+1}\} \). Then \( f(x, y) \) is uniquely defined by \( W := \{(x_i, y_j) \mid (x_i, y_j) \in X \times Y\} \).

Proof (existence): \( \exists 1 \text{ such } f(x, y) \) by Lagrange-Interpolation

Find \( L_{ij}(x; y) \) with \[
\begin{align*}
L_{ij}(x, y) &= 1 \\
L_{ij}(x', y) &= 0 \text{ for } (x', y) \neq (i, j)
\end{align*}
\]

\[
L_{ij}(x, y) := \prod_{i' \neq i}^{t+1} \frac{x - x_{i'}}{x_i - x_{i'}} \prod_{j' \neq j}^{t+1} \frac{y - y_{j'}}{y_j - y_{j'}}
\]

and define
\[
f(x, y) := \sum_{i,j=1}^{t+1} L_{ij}(x, y)z_{ij}.
\]
Facts about two-dimensional polynomials (3/3)

Given: \( f(x, y) = f_{00} + f_{10}x + f_{01}y + f_{11}xy + \ldots + f_{tt}x^ty^t \in \mathbb{F}[x, y] \)

Fact 2: Let \( X = \{x_1, \ldots, x_{t+1}\} \) and \( Y = \{y_1, \ldots, y_{t+1}\} \). Then \( f(x, y) \) is uniquely defined by \( W := \{(x_i, y_j, z_{ij}) | (x_i, y_j) \in X \times Y\} \).

Proof (uniqueness): \( \exists \exists 1 \) such \( f(x, y) \)

1. Let \( f_1(x, y) \) and \( f_2(x, y) \) degree-\( t \)-polynomials through \( W \).
2. \( f_D(x, y) := f_1(x, y) - f_2(x, y) \) is a degree-\( t \)-polynomial.
3. \( \forall (x_i, y_j) \in X \times Y : f_D(x_i, y_j) = 0 \).
4. \( \forall y_j \in Y : f_{y_j}(x) := f_D(x, y_j) \) is a polynomial of degree \( t \) (Fact 1).
5. \( \forall y_j \in Y : f_{y_j}(x) = f_{y_j}(x_1) = f_{y_j}(x_2) = \ldots = f_{y_j}(x_{t+1}) = 0 \); thus \( f_{y_j} \equiv 0 \).
6. Thus: \( \forall (x_i, y_j) \in \mathbb{F} \times Y : f_D(x_i, y_j) = 0 \).
7. \( \forall x \in \mathbb{F} : f_D(y) := f_D(x, y) \) is a polynomial of degree \( t \) (Fact 1).
8. \( \forall x \in \mathbb{F} : f_D(y_1) = f_D(y_2) = \ldots = f_D(y_{t+1}) = 0 \); thus \( f_D \equiv 0 \).
9. Thus: \( \forall (x, y) \in \mathbb{F} \times \mathbb{F} : f_D(x, y) = 0 \); thus \( f_D \equiv 0 \).

Commit Protocol

1. Distribution
   \( D \) selects random polynomial
   \[ f(x, y) = \sum_{i=0}^{t} \sum_{j=0}^{t} f_{ij} x^i y^j, \] with \( f_{00} = s \),
   and sends \( h_1(\cdot) = f(x, \alpha_i), h_2(\cdot) = f(\alpha_i, y) \) to \( P_i \).
2. Consistency checks
   \( \forall P_i, P_j : P_i \) sends \( k_i(\alpha_j) \) to \( P_j \), \( P_j \) complains if \( k_i(\alpha_j) \neq h_j(\alpha_j) \).
   \( D \) broadcasts \( f(\alpha_i, \alpha_j) \).
3. Accusation
   \( \forall P_i : \) if \( P_i \) has received contradicting values from \( D \): accuse \( D \).
   \( D \) broadcasts \( h_1(x) \) and \( h_2(y) \).
   Repeat until no further accusation.
4. Compute share
   If > \( t \) accusations: disqualify dealer.
   If \( \leq t \) accusations: \( s_i = k_i(0) \).

MPC Operations: Overview

<table>
<thead>
<tr>
<th></th>
<th>Passive</th>
<th>Active</th>
<th>Crypt.</th>
<th>Inf.-Theo.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>Shamir sharing</td>
<td>+commit/CTP</td>
<td>normal</td>
<td>distributed</td>
</tr>
<tr>
<td>Add</td>
<td>linearity</td>
<td>+homomorph</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mult.</td>
<td>mult.prot. ([n/2])</td>
<td>+CMP/CTP</td>
<td></td>
<td>generic ([n/3])</td>
</tr>
<tr>
<td>Output</td>
<td>interpolate</td>
<td>+CTP</td>
<td></td>
<td>([n/3])</td>
</tr>
</tbody>
</table>