Two-Client and Multi-client Functional Encryption for Set Intersection and Variants

Tim van de Kamp  David Stritzl  Willem Jonker  Andreas Peter

ACISP 2019
Functional Encryption for Set Operations

- Privacy-preserving information sharing
- Two-client and multi-client constructions for various set operations
- Evaluation using a proof-of-concept implementation

\[ \text{evaluate } \bigcap_{i=1}^{n} S_i \]
Privacy-Preserving Information Sharing

Private Set Intersection

Computes a set operation using an interactive protocol
A participant learns the evaluation result

Functional Encryption for Set Operations
Computes a set operation using a non-interactive scheme
A third-party (the evaluator) learns the evaluation result

Use cases include:
privacy-preserving profiling
simple data mining
one-way data sharing
Privacy-Preserving Information Sharing

Private Set Intersection

Computes a set operation using an interactive protocol
A participant learns the evaluation result

Functional Encryption for Set Operations
Computes a set operation using a non-interactive scheme
A third-party (the evaluator) learns the evaluation result

Use cases include
privacy-preserving profiling
simple data mining
one-way data sharing
Privacy-Preserving Information Sharing

**Functional Encryption for Set Operations**

Computes a set operation using a **non-interactive scheme**

A third-party (the **evaluator**) learns the evaluation result

Use cases include

- privacy-preserving profiling
- simple data mining
- one-way data sharing
Multi-client Non-interactive Set Intersection Functionality

\[ f(S_1, S_2, \ldots, S_n) \]

- **FUNCTIONALITIES**
  - **intersection:** \( \bigcap S_i \)
  - **cardinality:** \( |\bigcap S_i| \)
  - **threshold:** \( |\bigcap S_i| > t \) \( \Rightarrow \bigcap S_i \) (also "with data transfer"

doesn't learn the individual clients' sets \( S_1, \ldots, S_n \)
cannot mix-and-match old and new inputs

collusion between the evaluator and client(s) does not reveal other clients' inputs
Multi-client Non-interactive Set Intersection

Functionality

\[ f(S_1, S_2, ..., S_n) \]

\[ f(S_1), f(S_2), ..., f(S_n) \]

FUNCTIONALITIES \( f \)

- intersection: \( \bigcap_i S_i \)
- cardinality: \( |\bigcap_i S_i| \)
- threshold: \( |\bigcap_i S_i| > t \Rightarrow \bigcap_i S_i \)

(also "with data transfer")

doesn't learn the individual clients' sets

cannot mix-and-match old and new inputs

collusion between the evaluator and client(s) does not reveal other clients' inputs
Multi-client Non-interactive Set Intersection

Security Requirements

\[ f(S_1, S_2, \ldots, S_n) \]

\[ (\text{ID}, S_1) \]

\[ (\text{ID}, S_2) \]

\[ (\text{ID}, S_n) \]

doesn’t learn the individual clients’ sets \( S_1, \ldots, S_n \)

\[ \bigcap_i S_i \]

FUNCTIONALITIES

\[ \text{cardinality:} \bigcap_i S_i \]

\[ \text{threshold:} \bigcap_i S_i \geq t \Rightarrow \bigcap_i S_i \] (also "with data transfer")

doesn’t learn the individual clients’ sets \( S_1, \ldots, S_n \)

doesn’t learn the individual clients’ sets \( S_1, \ldots, S_n \)
Multi-client Non-interactive Set Intersection
Security Requirements

\[ f(S_1, S_2, \ldots, S_n) \]

\[ \bigcap_i S_i \]

\[ \text{cardinality: } \bigcap_i S_i \]

\[ \text{threshold: } \bigcap_i S_i \geq t \rightarrow \bigcap_i S_i \] (also "with data transfer"

doesn't learn the individual clients' sets \( S_1, \ldots, S_n \)
cannot mix-and-match old and new inputs

collusion between the evaluator and client(s) does not reveal other clients' inputs
Multi-client Non-interactive Set Intersection

Security Requirements

\[ f(S_1, S_2, \ldots, S_n) \]

\( (ID, S_1) \)
\( (ID, S_2) \)
\( (ID, S_n) \)

\( S_1 \)
\( S_2 \)
\( S_n \)

\( 1 \)
\( 2 \)
\( n \)

collusion between the evaluator and client(s) does not reveal other clients’ inputs
Construction: Two-Client Set Intersection Cardinality

\[ |S_1 \cap S_2| = |\text{ct}_1 \cap \text{ct}_2| = \{ \phi_{\text{msk}}(\text{ID}, x_j) \mid x_j \in S_1 \} = \{ \phi_{\text{msk}}(\text{ID}, x_j) \mid x_j \in S_2 \} \]
Construction: Two-Client Set Intersection Cardinality

\[ |S_1 \cap S_2| = |ct_1 \cap ct_2| \]

- \( S_1 \)
  \[ ct_1 = \{ \phi_{msk}(ID, x_j) \mid x_j \in S_1 \} \]

- \( S_2 \)
  \[ ct_2 = \{ \phi_{msk}(ID, x_j) \mid x_j \in S_2 \} \]
Construction: Two-Client Set Intersection

\[ S_1 \cap S_2 = \{ \phi_{k_{ID,j}}^{-1}(c) \mid c \in ct_1 \cap ct_2 \} \]

\[ \begin{align*}
ct_1 &= \left\{ \left( \phi_{k_{ID,j}}(x_j) \right) \mid x_j \in S_1 \right\} \\
ct_2 &= \left\{ \left( \phi_{k_{ID,j}}(x_j) \right) \mid x_j \in S_2 \right\}
\end{align*} \]

\[ k_{ID,j} = \phi_{msk}(ID, x_j) \]
Construction: Two-Client Set Intersection

\[ S_1 \cap S_2 = \{ \phi_{k_{ID,j}}^{-1}(c) \mid c \in ct_1 \cap ct_2 \} \]

\[ k_{ID,j} = k_{ID,j}^{usk_1} \cdot k_{ID,j}^{usk_2} \]

\[ ct_1 = \{ (k_{ID,j}^{usk_1}, \phi_{k_{ID,j}}(x_j)) \mid x_j \in S_1 \} \]

\[ ct_2 = \{ (k_{ID,j}^{usk_2}, \phi_{k_{ID,j}}(x_j)) \mid x_j \in S_2 \} \]

\[ usk_1 + usk_2 = 1 \]

\[ k_{ID,j} = \phi_{msk}(ID, x_j) \]
Construction: Two-Client Set Intersection

$S_1 \cap S_2 = \left\{ \phi_{k_{ID},j}^{-1}(c) \mid c \in ct_1 \cap ct_2 \right\}$

$k_{ID,j} = k_{ID,j}^{usk_1} \cdot k_{ID,j}^{usk_2}$

$ct_1 = \left\{ (k_{ID,j}^{usk_1}, \phi_{k_{ID},j}(x_j)) \mid x_j \in S_1 \right\}$

$ct_2 = \left\{ (k_{ID,j}^{usk_2}, \phi_{k_{ID},j}(x_j)) \mid x_j \in S_2 \right\}$

 Doesn’t have to be $x_j \in S_1$; can be any **associated data**

$usk_1 + usk_2 = 1$

$k_{ID,j} = \phi_{msk}(ID, x_j)$
Intuition: Two-Client **Threshold** Set Intersection

$$S_1 \cap S_2 = \left\{ \phi_{k_{ID,j}}^{-1}(c) \mid c \in ct_1 \cap ct_2 \right\}$$

$$k_{ID,j} = k_{ID,j}^{usk_1} \cdot k_{ID,j}^{usk_2}$$

We also encrypt this value and require at least $t$ secret shares for decryption.

$$ct_1 = \left\{ (k_{ID,j}^{usk_1}, \phi_{k_{ID,j}}(x_j)) \mid x_j \in S_1 \right\}$$

$$ct_2 = \left\{ (k_{ID,j}^{usk_2}, \phi_{k_{ID,j}}(x_j)) \mid x_j \in S_2 \right\}$$

$$usk_1 + usk_2 = 1$$

$$k_{ID,j} = \phi_{msk}(ID, x_j)$$
Efficiency of the 2C-FE Constructions

![Graph showing the mean evaluation time (seconds) for different sizes of each client's set. The x-axis represents the size of each client's set, ranging from $10^1$ to $10^5$. The y-axis represents the mean evaluation time in seconds, ranging from $10^{-6}$ to $10^0$. The data points are connected by a line, indicating a linear relationship between the size of the set and the mean evaluation time. The graph is labeled with "CA" for clarity.]
Efficiency of the 2C-FE Constructions

Size of each client's set vs. Mean evaluation time (seconds)

- CA
- SI
Efficiency of the 2C-FE Constructions

![Graph showing the efficiency of the 2C-FE Constructions](image-url)

- CA
- SI
- Th-CA
- Th-SI

Size of each client’s set - Mean evaluation time (seconds)
Construction: Multi-client Set Intersection Cardinality

\[
\text{count } \prod_{i=1}^{n} H(ID, x_j)^{usk_i} \neq 1
\]

\[
\sum_{i=1}^{n} usk_i = 0
\]
Efficiency of the MC-FE Construction

Theoretical
Polynomial in the number of set elements per client:
\[ \mathcal{O}(\prod_i |S_i|) \]

Practice

![Graph showing mean evaluation time versus size of each client’s set for different CA values](#)
**Improved Set Intersection Cardinality Scheme**

**Intuition**

1. Compute the set intersection $\bigcap_i S_i$ “in the encrypted domain”;
2. For some client $i'$, determine how many set elements $e_j \in S_{i'}$ are in the encrypted set intersection, i.e.,

$$\left| \left\{ e_j \mid e_j \in \bigcap_{i=1}^n S_i, e_j \in S_{i'} \right\} \right|.$$
Improved Set Intersection Cardinality Scheme

Intuition

1. Compute the set intersection $\bigcap_i S_i$ “in the encrypted domain”;
2. For some client $i'$, determine how many set elements $e_j \in S_{i'}$ are in the encrypted set intersection, i.e.,

$$\left| \left\{ e_j \mid e_j \in \bigcap_{i=1}^n S_i, e_j \in S_{i'} \right\} \right|.$$  

“Tools”

- Bloom filters $\rightarrow$ to represent sets in a single data structure
- Homomorphic encryption $\rightarrow$ to compute in the encrypted domain
- Functional encryption $\rightarrow$ to determine whether an element is in a set
Preliminaries: Bloom filters

Set Intersection

\[
\begin{array}{cccccccc}
\text{bs[1]} & \text{bs[2]} & \text{bs[3]} & \text{bs[4]} & \text{bs[5]} & \text{bs[6]} & \text{bs[7]} & \text{bs[8]} & \text{bs[9]} \\
\hline
S_1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
\cap & \wedge & \\
S_2 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\
\hline
S_1 \cap S_2 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
\end{array}
\]
### Construction (simplified)

#### Set Intersection using Secret Sharing

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_{1,1}$</td>
<td>$S_{1,2}$</td>
<td>$r_{1,3}$</td>
<td>$S_{1,4}$</td>
<td>$S_{1,5}$</td>
<td>$S_{1,6}$</td>
<td>$r_{1,7}$</td>
<td>$r_{1,8}$</td>
<td>$r_{1,9}$</td>
<td></td>
</tr>
</tbody>
</table>

+ 

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_{2,1}$</td>
<td>$r_{2,2}$</td>
<td>$r_{2,3}$</td>
<td>$S_{2,4}$</td>
<td>$r_{2,5}$</td>
<td>$S_{2,6}$</td>
<td>$r_{2,7}$</td>
<td>$r_{2,8}$</td>
<td>$S_{2,9}$</td>
<td></td>
</tr>
</tbody>
</table>

= 

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tilde{r}_1$</td>
<td>$\tilde{r}_2$</td>
<td>$\tilde{r}_3$</td>
<td>1</td>
<td>$\tilde{r}_5$</td>
<td>1</td>
<td>$\tilde{r}_7$</td>
<td>$\tilde{r}_8$</td>
<td>$\tilde{r}_9$</td>
<td></td>
</tr>
</tbody>
</table>

$\text{Encrypt}(usk_i, ID, S_i)$

$H(ID, \ell)$

$r_i, \ell$ if $bs[\ell] = 0$

$s_i, \ell$ if $bs[\ell] = 1$

$\text{Evaluate}(ct_1, \ldots, ct_n)$

$H(ID, \ell) = s_0, \ell \cdot \pi_{n=1} H(ID, \ell) s_i, \ell$
**Construction (simplified)**

**Set Intersection using Secret Sharing**

$$\text{Enc}(S_1) = r_1,1 \ S_1,2 \ r_1,3 \ S_1,4 \ S_1,5 \ S_1,6 \ r_1,7 \ r_1,8 \ r_1,9$$

$$\text{Enc}(S_2) = r_2,1 \ r_2,2 \ r_2,3 \ S_2,4 \ r_2,5 \ S_2,6 \ r_2,7 \ r_2,8 \ S_2,9$$

$$\text{Enc}(S_1 \cap S_2) = \tilde{r}_1 \ \tilde{r}_2 \ \tilde{r}_3 \ 1 \ \tilde{r}_5 \ 1 \ \tilde{r}_7 \ \tilde{r}_8 \ \tilde{r}_9$$

**Encrypt**(usk$_i$, ID, $S_i$)

$$H(\text{ID}, \ell)^{r_{i,\ell}} \text{ if } bs[\ell] = 0;$$

$$H(\text{ID}, \ell)^{s_{i,\ell}} \text{ if } bs[\ell] = 1$$
**Construction (simplified)**

**Set Intersection using Secret Sharing**

\[
\text{Enc}(S_1) = r_{1,1} | S_{1,2} | r_{1,3} | S_{1,4} | S_{1,5} | S_{1,6} | r_{1,7} | r_{1,8} | r_{1,9}
\]

\[
\text{Enc}(S_2) = r_{2,1} | r_{2,2} | r_{2,3} | S_{2,4} | r_{2,5} | S_{2,6} | r_{2,7} | r_{2,8} | S_{2,9}
\]

\[
\text{Enc}(S_1 \cap S_2) = \tilde{r}_1 | \tilde{r}_2 | \tilde{r}_3 | 1 | \tilde{r}_5 | 1 | \tilde{r}_7 | \tilde{r}_8 | \tilde{r}_9
\]

**Encrypt**(usk\(_i\), ID, \(S_i\))

- \(H(\text{ID}, \ell)^{r_{i,\ell}}\) if \(bs[\ell] = 0\);
- \(H(\text{ID}, \ell)^{s_{i,\ell}}\) if \(bs[\ell] = 1\)

**Evaluate**(ct\(_1, ..., \), ct\(_n\))

\[H(\text{ID}, \ell)^{s_{0,\ell}} \cdot \left(\prod_{i=1}^{n} H(\text{ID}, \ell)^{s_{i,\ell}}\right)\]
Construction (simplified)

Set Intersection using Secret Sharing

```
Enc(S_1)
Enc(S_2)
| r_1  | r_2  | r_3  | 1    | r_5  | 1    | r_7  | ₋_8  | ₋_9  |
```

Actual construction is more involved:

- element testing uses
  \[ (H(ID, ℓ)^s_{0,ℓ} g^{t r}) \cdot \prod_{i=1}^{n} H(ID, ℓ)^s_{i,ℓ} = (g^r)^t \]
- using Shamir secret sharing instead of additive secret sharing

Encrypt(usk_i, ID, S_i)

- \( H(ID, ℓ)^{r_{i,ℓ}} \) if \( bs[ℓ] = 0 \);  
- \( H(ID, ℓ)^{s_{i,ℓ}} \) if \( bs[ℓ] = 1 \)

Evaluate(ct_1, \ldots, ct_n)

\[ H(ID, ℓ)^{s_{0,ℓ}} \cdot \left( \prod_{i=1}^{n} H(ID, ℓ)^{s_{i,ℓ}} \right) \]
Efficiency of the MC-FE Construction

Theoretical
Polynomial in the number of set elements per client:
\[ O(x^2) \]

Practice

![Graph showing the mean evaluation time (seconds) vs. size of each client's set for different values of CA. The graph indicates that the mean evaluation time increases with the size of the set and differs slightly between CA 5 and CA 3.](graph.png)
Efficiency of the MC-FE Construction

**Theoretical**
Polynomial in the number of set elements per client:

\[ O(x^2) \]

**Practice**

![Graph showing the mean evaluation time (seconds) vs. size of each client's set for different set sizes and constructions CA n = 5, CA n = 3, CA-BF n = 5, CA-BF n = 3.]
Summary

- **Non-interactive** privacy-preserving information sharing
- Efficient two-client constructions for various set operations
- Theoretical constructions for various multi-client set operations

Interested?

Implementation: https://github.com/CRIPTIM/nipsi

Contact: t.r.vandekamp@utwente.nl
Summary

- **Non-interactive** privacy-preserving information sharing
- Efficient two-client constructions for various set operations
- Theoretical constructions for various multi-client set operations

Interested?

Implementation: https://github.com/CRIPTIM/nipsi
Contact: t.r.vandekamp@utwente.nl