Identity-Based Identification Schemes

Guomin Yang

Centre for Computer and Information Security Research
School of Computing and Information Technology
University of Wollongong
1. Identification Schemes
   - Standard Identification Schemes
   - Identity Based Identification Schemes
   - From IBI to IBS

2. IBI Framework 1
   - SI to IBI Transformation

3. IBI Framework 2
   - IBI with Passive Security
   - IBI with Active and Concurrent Security
Standard Identification (SI) Scheme

Prover (pk, sk)  Verifier (pk)

P₁  ← V₁  →

P₂  ← V₂  →

···

Pₙ  ← Vₙ  →

Accept/Reject
Standard Identification (SI) Scheme

Prover \((pk, sk)\) \hspace{1cm} \text{Verifier} \((pk)\)

\[ P_1 \rightarrow V_1 \]
\[ \rightarrow P_2 \]
\[ \rightarrow V_2 \]
\[ \rightarrow \ldots \]
\[ \rightarrow P_n \]

Accept/Reject

1. **Passive Attack (imp-pa)**
   1. **Stage 1:** the attacker obtains communication transcripts between the prover and an honest verifier
   2. **Stage 2:** the attacker tries to impersonate the prover
Standard Identification (SI) Scheme

Prover (pk, sk) → Verifier (pk)

P1 → V1

P2 → V2

... → Pn

Accept/Reject

Passive Attack (imp-pa)

1. Stage 1: the attacker obtains communication transcripts between the prover and an honest verifier
2. Stage 2: the attacker tries to impersonate the prover

Active Attack (imp-aa)

1. Stage 1: the attacker communicates with the prover as a verifier, one session at a time
2. Stage 2: the attacker tries to impersonate the prover

Concurrent Attack (imp-ca) similar to Active Attack, but the attacker can have concurrent sessions with the prover in Stage 1
### Standard Identification (SI) Scheme

<table>
<thead>
<tr>
<th>Prover (pk, sk)</th>
<th>Verifier (pk)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>V1</td>
</tr>
<tr>
<td>V1</td>
<td>P2</td>
</tr>
<tr>
<td>P2</td>
<td>V2</td>
</tr>
<tr>
<td>…</td>
<td>…</td>
</tr>
<tr>
<td>Pn</td>
<td>Accept/Reject</td>
</tr>
</tbody>
</table>

1. **Passive Attack (imp-pa)**
   - Stage 1: the attacker obtains communication transcripts between the prover and an honest verifier
   - Stage 2: the attacker tries to impersonate the prover

2. **Active Attack (imp-aa)**
   - Stage 1: the attacker communicates with the prover as a verifier, one session at a time
   - Stage 2: the attacker tries to impersonate the prover

3. **Concurrent Attack (imp-ca)** similar to Active Attack, but the attacker can have concurrent sessions with the prover in Stage 1
Identity Based Identification (IBI) Schemes

\[ \text{PKG (mpk, msk)} \]

\[ \text{usk}_{ID} = \text{UKGen}(msk, ID) \]

\[ \text{Prover (usk}_{ID}) \]

\[ \text{Verifier (mpk, ID)} \]

\[ P_1 \rightarrow V_1 \]

\[ P_2 \rightarrow V_2 \]

\[ \ldots \]

\[ P_n \rightarrow \text{Accept/Reject} \]
Canonical IBI

PKG (mpk, msk)  

$usk_{ID} = UKGen(msk, ID)$

User (ID)

$usk_{ID}$

The identification protocol has three moves

Prover ($usk_{ID}$)  

Verifier (mpk, ID)

Cmt  

Ch  

Rsp  

Accept/Reject
Canonical IBI to IBS

Algorithm $\text{Sign}(usk_{ID}, M)$
1. $(\text{Cmt}, St) \leftarrow P(usk_{ID})$
2. $\text{Ch} = H(ID, \text{Cmt}, M)$
3. $\text{Rsp} \leftarrow P(St, \text{Ch})$
4. Return $\sigma = (\text{Cmt}, \text{Rsp})$

Algorithm $\text{Ver}(ID, M, \sigma)$
1. Parse $\sigma$ as $(\text{Cmt}, \text{Rsp})$
2. $\text{Ch} = H(ID, \text{Cmt}, M)$
3. Return $V(ID, \text{Cmt}||\text{Ch}||\text{Rsp})$

Theorem (BNN’04)

If the canonical IBI scheme is secure under passive attacks, then the IBS scheme is existentially unforgeable under adaptive chosen message attacks in the random oracle model.
A Binary Relation $\mathcal{R} = \{(x_1, y_1), (x_2, y_1), \cdots, (x_m, y_n)\}$

Notations:
- $\langle \mathcal{R} \rangle$: description of $\mathcal{R}$
- $\mathcal{R}^{-1}(y) = \{x : (x, y) \in \mathcal{R}\}$

Trapdoor Samplable Relation:
- Samplable: easy to sample uniformly distributed $(x, y) \in \mathcal{R}$
- Regular: for any $y \neq y'$, $|\mathcal{R}^{-1}(y)| = |\mathcal{R}^{-1}(y')|$
- Invertible: $\exists$ a trapdoor $t$ such that $\text{Inv}(t, y)$ outputs $x \in \mathcal{R}^{-1}(y)$
SI to IBI Transformation

**Definition**

A standard identification scheme SI is said to be **convertible** if its key space \{\( (pk, sk) \)\} can be described by a trapdoor samplable relation \( R \).
SI to IBI Transformation

**Definition**
A standard identification scheme SI is said to be *convertible* if its key space \{<(pk, sk)>\} can be described by a trapdoor samplable relation \(R\).

**The Transformation:**

- **Algorithm MKGen(1^k)**
  1. Generate a relation \(R\) with trapdoor \(t\)
  2. \(mpk = \langle R\rangle, msk = t\)
  3. Return \((mpk, msk)\)

- **Algorithm UKGen(msk, I)**
  1. Set \(y = H(I)\)
  2. Compute \(x \leftarrow \text{Inv}(t, y); usk[I] = x\)
  3. Return \(usk[I]\)

Prover: run the SI prover algorithm with secret key \(usk[I]\)
Verifier: run the SI verifier algorithm with public key \(H(I)\)

**Theorem**
*The IBI is id-imp-atk secure if the SI is imp-atk secure for atk \(\in \{pa, aa, ca\}\).*
Example: the Guillou–Quisquater SI scheme (Crypto’88)

Algorithm Kg(1^k)
(N, e, d) ← K_{rsa}(1^k)
\( x \leftarrow \mathbb{Z}_N^* \)
\( X = x^e \mod N \)
\( pk = ((N, e), X) \)
\( sk = ((N, e), x) \)
Return (pk, sk)

GQ is imp-pa secure under the RSA assumption.
Example: the Guillou–Quisquater SI scheme (Crypto’88)

Algorithm Kg(1^k)
\((N, e, d) \leftarrow K_{rsa}(1^k)\)
\(x \leftarrow \mathbb{Z}_N^*\)
\(X = x^e \mod N\)
\(pk = ((N, e), X)\)
\(sk = ((N, e), x)\)
Return \((pk, sk)\)

\[P\]
\[V\]
\[y \leftarrow \mathbb{Z}_N^*\]
\[Y \leftarrow y^e \mod N\]
\[Y \xrightarrow{c} \mathbb{Z}_{2m(k)}\]
\[c \xleftarrow{\mathcal{R}} \mathbb{Z}_{2m(k)}\]
\[z = x^c y \mod N\]
\[z \xrightarrow{c} \mathbb{Z}_N^*\]
If \(z^e \equiv X^c Y \mod N\)
then acc else rej

GQ is imp-pa secure under the RSA assumption.

\[R = \{(x, X) \in \mathbb{Z}_N^* \times \mathbb{Z}_N^* : X = x^e \mod N\}\]
\[\langle R \rangle = (N, e), \ \text{Trapdoor: } (N, d)\]

Regular: \(R\) is a one-to-one mapping
Example: the transformed IBI scheme from GQ-SI

\[
\begin{align*}
\text{MKGen}(1^k) & \quad (N, e, d) \leftarrow K_{\text{rsa}}(1^k) \\
msk & \leftarrow (N, d) \\
mpk & \leftarrow (N, e) \\
\text{UKGen}(msk, I) & \\
X & \leftarrow H(I) \\
x & = X^d \mod N \\
\text{Return } usk[I] = x
\end{align*}
\]
A Binary Relation $R = \{(x_1, y_1), (x_2, y_1), \cdots, (x_m, y_n)\}$.

**Trapdoor Weak-one-more Relation (TWR):**

- **Invertible:** $\exists$ a trapdoor $t$ such that $\text{Inv}(t, y)$ outputs $x \in R^{-1}(y)$
- **Weak-one-more secure:** given

  $$\{y_1, y_2, \cdots, y_k\} \text{ and } \{x_1, x_2, \cdots, x_{k-1}\}$$

such that $(x_j, y_j) \in R$ for $1 \leq j \leq k - 1$, difficult to find $x_k$. 
TWR Instantiations

1. RSA \((N, e)\)

\[
\mathbb{R}^{RSA} = \{(x, y) \in \mathbb{Z}_N^* : x^e = y \mod N\}
\]

with \(t = (N, d)\)

Remark: the third one is not samplable
TWR Instantiations

1. RSA \((N, e)\)

\[
\mathcal{R}^{RSA} = \{(x, y) \in \mathbb{Z}_N^* : x^e = y \mod N\}
\]

with \(t = (N, d)\)

2. CDH in a bilinear group \((\mathbb{G}_1, \mathbb{G}_2, q, P, S, e)\)

\[
\mathcal{R}^{CDH} = \{(x, y) \in \mathbb{G}_1^2 : e(P, x) = e(S, y)\}
\]

with \(t = \log_PS\)
TWR Instantiations

1. RSA $(N, e)$

\[ R^{RSA} = \{(x, y) \in \mathbb{Z}_N^* : x^e = y \mod N\} \]

with \( t = (N, d) \)

2. CDH in a bilinear group $(G_1, G_2, q, P, S, e)$

\[ R^{CDH} = \{(x, y) \in G_1^2 : e(P, x) = e(S, y)\} \]

with \( t = \log_P S \)

3. Digital signature scheme $SIG$ (UF-KMA)

\[ R^{SIG} = \{(\sigma, m) : \mathcal{V}(pk, m, \sigma) = 1\} \]

with \( t = sk \)

Remark: the third one is not samplable
1 Honest Verifier Zero Knowledge. There exists a simulation algorithm that can simulate a protocol transcript without knowing the prover’s secret key $x$.

2 Special Soundness. Given two transcripts $(Cmt, Ch_1, Rsp_1)$ and $(Cmt, Ch_2, Rsp_2)$ where $Ch_1 \neq Ch_2$, we can extract the prover’s secret key $x$. 
TWR + HVZK-PoK = IBI with imp-pa security

Algorithm MKGen(1^k)

1. Generate a TWR relation \( R \) with trapdoor \( t \)
2. \( mpk = \langle R \rangle, msk = t \)
3. Return \((mpk, msk)\)

Algorithm UKGen(msk, I)

1. Set \( y = H(I) \)
2. Compute \( x \leftarrow \text{Inv}(t, y); \) \( usk[I] = x \)
3. Return \( usk[I] \)

Prover: run the HVZK-PoK prover algorithm with secret key \( usk[I] \)
Verifier: run the HVZK-PoK verifier algorithm with public key \( H(I) \)
TWR $+$ HVZK-PoK example

TWR: a bilinear group $(\mathbb{G}_1, \mathbb{G}_2, q, P, e)$ and $Q \in \mathbb{G}_1$

\[
\mathcal{R} = \{(x, y) : x \in \mathbb{G}_1; \ y \in \mathbb{Z}_q; \ e(yP + Q, x) = e(P, P)\}
\]

with trapdoor $t = \log_P Q$.

- Invertible: Given $y \in \mathbb{Z}_q$, compute $x = \frac{1}{y+t}P$.
- Weak-one-more secure: under the assumption that the K-CAA problem is hard
  “Given $(P, Q, h_1, \cdots, h_K, \frac{1}{h_1+t}P, \cdots, \frac{1}{h_K+t}P)$, compute $(h, \frac{1}{h+t}P)$ such that $h \notin \{h_1, h_2, \ldots, h_K\}$.”

- This relation is NOT samplable!
TWR + HVZK-PoK = IBI with imp-pa security

An HVZK-PoK for the TWR

1. $Cmt := r(yP + Q)$ where $r \leftarrow \mathbb{Z}_q$
2. $Ch := c$ where $c \leftarrow \mathbb{Z}_q$
3. $Rsp := rcP + x$

Verification:

$$e(Rsp, yP + Q) = e(P, P)e(P, Cmt)^c.$$
A Binary Relation $R = \{(x_1, y_1), (x_2, y_1), \cdots, (x_m, y_n)\}$.

**Trapdoor Strong-one-more Relation (TWR):**

- Invertible: $\exists$ a trapdoor $t$ such that $\text{Inv}(t, y)$ outputs $x \in R^{-1}(y)$
- Strong-one-more secure: given

  $$\{y_1, y_2, \cdots, y_k\} \text{ and } \{x_1, x_2, \cdots, x_k\}$$

such that $(x_j, y_j) \in R$ for $1 \leq j \leq k$, difficult to find $x'_i \neq x_i \wedge (x'_i, y_i) \in R$ for any $i$. 
TSR Instantiations

1. RSA: $(N, e, g)$ where $g \leftarrow \mathbb{Z}_N^*$. 

   $\mathbb{R}^{RSA} = \{((x_1, x_2), Y) \in (\mathbb{Z}_e \times \mathbb{Z}_N^*) \times \mathbb{Z}_N^* : g^{-x_1}x_2^{-e} \equiv Y \mod N\}$

   with $t = (N, d)$.

2. Digital signature scheme $SIG$ (SUF-KMA)

   $\mathbb{R}^{SIG} = \{(\sigma, m) : \mathcal{V}(pk, m, \sigma) = 1\}$

   with $t = sk$. 
Witness Indistinguishable Proof

Let \((x, y) \in R\) and \((x', y) \in R\) for a many-to-one TSR \(R\).

**Witness Indistinguishability (WI):** \((\text{Cmt}, \text{Ch}, \text{Rsp})\) and \((\text{Cmt}', \text{Ch}', \text{Rsp}')\) are computationally indistinguishable.

**Fact:** WI is preserved under concurrent protocol execution (Feige-Shamir STOC’90)
TSR + WI-PoK = IBI with imp-aa/ca security

Construction: same as TWR + HVZK-PoK

Proof idea: If \( \mathcal{A} \) can break the concurrent security of the IBI scheme, \( \mathcal{B} \) can solve the Strong-one-more problem.

- \( \mathcal{B} \) simulates Stage 1 using \( x \).
- If \( \mathcal{A} \) can impersonate the prover in Stage 2, \( \mathcal{B} \) can extract \( \hat{x} \) from \( \mathcal{A} \) such that \( (\hat{x}, y) \in R \) because of the Proof ok Knowledge property.
- Since Stage 1 is witness indistinguishable, with probability at least 1/2, \( \hat{x} \neq x \).
TSR + WI-PoK example (Okamoto-IBI Crypto’92)

RSA based TSR:

$$R^{RSA} = \{((x_1, x_2), Y) \in (\mathbb{Z}_e \times \mathbb{Z}_N^*) \times \mathbb{Z}_N^* : g^{-x_1} x_2^{-e} \equiv Y \mod N\}$$

WI-PoK:

\[
\begin{align*}
&P(x_1, x_2) \\
&y_1 \overset{R}{\leftarrow} \mathbb{Z}_e; y_2 \overset{R}{\leftarrow} \mathbb{Z}_N^* \\
&Cmt \leftarrow g^{y_1} y_2^e \mod N \\
&z_1 \leftarrow y_1 + cx_1 \mod e \\
&z_2 \leftarrow g^{\left\lfloor \frac{(y_1 + cx_1)}{e} \right\rfloor} y_2 x_2^c \mod N \\
&Rsp = (z_1, z_2) \\
&\text{Cmt} \leftarrow g^{z_1 z_2^e Y^c} \mod N
\end{align*}
\]