Concurrent Non-Malleable Witness Indistinguishability

Rafail Ostrovsky (UCLA, USA)
Giuseppe Persiano (Univ. Salerno – ITALY)
Ivan Visconti (Univ. Salerno – ITALY)
Outline

- Concurrent ZK, NMZK, Witness Indist.
- Non-Malleable Witness Indistinguishability
- Cnst-Rnd Concurrent NMWI in the plain model
- Cnst-Rnd Concurrent NMZK in the BPK Model
- UC with preprocessing
Interactive Proof System

\[ x \in L \]

Prover P \hspace{1cm} a \hspace{1cm} b \hspace{1cm} z \hspace{1cm} Verifier V

Los Angeles, nov 15 2006
Interactive Proof System

$x \in L$

Properties:

Completeness: if $x \in L$ then $V$ outputs 1
Soundness: if NOT($x \in L$) then $V$ outputs 0
Interactive Zero-Knowledge Proofs

Zero Knowledge:

\[ x \in L \]
Interactive Proof of Knowledge

\[ x \in L \]

Witness Extraction:

\[ x \in L \]
Outline

- Concurrent ZK, NMZK, Witness Indist.
- Non-Malleable Witness Indistinguishability
- Cnst-Rnd Concurrent NMWI in the plain model
- Cnst-Rnd Concurrent NMZK in the BPK Model
- UC with preprocessing
Man-in-the-Middle (MiM) Attack

$x \in L$

$r, w$

$x' \in L$
Concurrent MiM Attack

\[ x \in \mathbb{L} \]

\[ x' \in \mathbb{L} \]
Concurrent NMZK

\[ x \in L \quad x' \in L \]

\[ y': (x', y') \in R_L \]
Concurrent NMZK

\[ x \in L \]

\[ x' \in L \]
Outline

- Concurrent (ZK, NMZK), Witness Indist.
- Non-Malleable Witness Indistinguishability
- Const-Rnd Concurrent NMWI in the plain model
- Const-Rnd Concurrent NMZK in the BPK Model
- UC with preprocessing
Witness Indistinguishable Proofs

$x \in L$

Witness Indistinguishability:
For all $x \in L$, for all pair $(y,y')$ of valid witnesses for $x \in L$

$\text{View}^*_V(P(y),x,y,y') \approx \text{View}^*_V(P(y'),x,y,y')$ where ZK implies WI
Witness Indistinguishability

ZK implies WI
but WI helps for the design of ZK protocols
(e.g., FLS-paradigm):
  Non-Black-Box ZK
  NIZK in the SRS model [FLS90,DDOPS01]

can we use a notion of WI secure against MiM
attacks for the design of CNMZK protocols?
Outline

- Concurrent (ZK, NMZK), Witness Indist.
- Non-Malleable Witness Indistinguishability
- Cnst-Rnd Concurrent NMWI in the plain model
- Cnst-Rnd Concurrent NMZK in the BPK Model
- UC with preprocessing
Witness Encoded in a Proof

- we focus on commit-and-prove arguments where in the first message the prover commits to the witness by using a statistically binding (therefore we consider computational indistinguishability) commitment scheme (this message is the “witness encoded in the proof”) and then proves that the committed message is an NP-witness for $x \in L$

- the goal of the MiM is to relate the witnesses encoded in the proofs he gives with the witnesses encoded in the proofs he receives
Concurrent MiM Attack

$x \in L$

$x' \in L$

$w$
CNMWI, very informally

**CNM Witness Indistinguishability:**

“the distribution of the witnesses encoded in the proofs given by the man-in-the-middle is independent of the distribution of the witnesses encoded in the proofs given by the prover”
CNMWI, informally

**CNM Witness Indistinguishability:**

let $\text{mim}_{<x>}(<w>)$ the random variable that the describes the witnesses encoded in the proofs given by the mim when receiving proofs for $<x>$ from P with encoded witnesses $<w>$

CNMWI requires that the following distributions are comput. indistinguishable

$\{\text{mim}_{<x>}(<w>)\}, \{\text{mim}_{<x>}(<w'>)\}$
CNMZK vs CNMWI

CNMWI

\( x \in L \)

\( w_0, w_1 \)

CNMZK

\( x'^{'} \in L \)

\( y'^{'}: (x'^{'}, y'^{'} \in R_L \)

Los Angeles, nov 15 2006
CNMWI+ (informal)

CNMWI+ following the Simulation paradigm:

“for any PPT adversary A that in a MiM attack proves statements \(<x>\) to a honest verifier with proofs that encode witnesses \(<w>\), there exists a ppt S that by accessing to A proves statements \(<x>\) to a honest verifier with proofs that encode witnesses \(<w>\)”

this definition implies both the previous def. of CNMWI and that of CNMZK
CNM Commitments [PR05]

CNM Commitments:

“for any PPT adversary A that in a MiM attack commits to messages \( <w> \), there exists a PPT S that by accessing to A outputs commitments to messages \( <w> \)”

Can CNM commitment schemes help for designing CNMWI argument systems?
Outline

- Concurrent ZK, NMZK, Witness Indist.
- Non-Malleable Witness Indistinguishability
- Const-Rnd Concurrent NMWI in the plain model
- Const-Rnd Concurrent NMZK in the BPK Model
- UC with preprocessing
Constant Round CNMWI

$P \rightarrow V$ send a commitment of the witness $w$
$P \rightarrow V$ use the one-left many-right statistical concurrent non-malleable ZK argument of knowledge of [PR05a] for proving that $w$ is a witness for $x \in L$

Remark: this protocol is a PoK and it is only a cosmetic variation of the one by [PR05b] for concurrent non-malleable commitments
Outline

- Concurrent ZK, NMZK, Witness Indist.
- Non-Malleable Witness Indistinguishability
- \textit{Cnst-Rnd} Concurrent NMWI in the plain model
- \textit{Cnst-Rnd} Concurrent NMZK in the BPK Model
- UC with preprocessing
The Bare Public-Key (BPK) model (CGGM00)

- In a key-registration stage:
  - Each verifier (non-interactively) posts her public key on a public file, common to all parties.
  - There is no bound on the power of the adversary that therefore can control the entire resulting file.

- In the proof stage:
  - The same public file is part of the common input in all proofs and the verifiers can use their private keys.

- BPK is a weaker version of the (PKI) model since
  - public keys do NOT need to be certified during the key-registration phase.
BPK model: the Key-Registration Stage

\[
\text{sk}_j \rightarrow \text{pk}_j
\]
BPK model: first attack of the MiM

sk_j
V_j

(pk_j)
aux

pk_j
pk_i
BPK model: the **Proof Stage**

\[ x \in L \]

\[ x' \in L \]

\[ pk_j \]

\[ pk_i \]

\[ w \]

\[ sk_j \]
CNMZK in the BPK model

\[ \gamma_{j_0} = f(s_{k_{j_0}}), \gamma_{i_1} = f(s_{k_{j_1}}) \]

\[ x \in L \]

\[ \text{CNMWIPoK } s_{k_{j_0}} \lor s_{k_{j_1}} \]

\[ \text{CNMWIPoK } x \in L \lor s_{k_{j_0}} \lor s_{k_{j_1}} \]
Man-in-the-Middle Attack

\[ y_{j0} = f(sk_{j0}), \ y_{i1} = f(sk_{j1}) \]

\[ \begin{array}{c|c}
    Y_{j0} & Y_{j1} \\
    \hline
    Y^*_{j0} & Y^*_{j1} \\
\end{array} \]

\[ x \in L \quad x' \in L \]

\[ \begin{array}{c}
    sk^*_{j0} \lor sk^*_{j1} \\
    x \in L \lor sk^*_{j0} \lor sk^*_{j1} \\
\end{array} \]

\[ \begin{array}{c}
    sk_{j0} \lor sk_{j1} \\
    x' \in L \lor sk_{j0} \lor sk_{j1} \\
\end{array} \]

Los Angeles, nov 15 2006
Simulator for the MiM

\( \gamma_{j0} = f(\text{sk}_{j0}), \gamma_{i1} = f(\text{sk}_{j1}) \)

\( \approx \quad \text{Sim of} \quad \approx \quad \text{CGGM00} \)

\( x \in L \quad \Rightarrow \quad \text{sk}_{jb} \)

\( x' \in L \quad \Rightarrow \quad \text{sk}_{j0} \lor \text{sk}_{j1} \)

\( x \in L \lor \text{sk}^*_{j0} \lor \text{sk}^*_{j1} \)

\( x' \in L \lor \text{sk}_{j0} \lor \text{sk}_{j1} \)
Concurrent NMZK

\[ x \in L \quad y' : (x', y') \in R_L \]
Simulator for the MiM

\[ \gamma_{j0} = f(sk_{j0}), \gamma_{j1} = f(sk_{j1}) \]

\[ y_{j0} = f(sk_{j0}), y_{j1} = f(sk_{j1}) \]

\[ x \in L \]

\[ sk_{j0} \lor sk_{j1} \]

\[ x' \in L \]

\[ sk_{j0} \lor sk_{j1} \]

\[ x \in L \lor sk_{j0} \lor sk_{j1} \]

\[ x' \in L \lor sk_{j0} \lor sk_{j1} \]
Concurrent NMZK

\[ x \in L \quad \Rightarrow \quad x' \in L \]

get \( w \in \{ y', sk_{j0}, sk_{j0} \} \)

if \( w = y' \) \( \checkmark \)

else if \( w = sk_{j(1-b)} \) \( \checkmark \)

else if \( w = sk_{jb} \) ??
Simulator for the MiM

\[ x \in L \]

\[ x' \in L \]

\[ sk_{j0} \lor sk_{j1} \]

\[ sk_{j0} \lor sk_{j1} \]

\[ sk^*_{j0} \lor sk^*_{j1} \]

\[ sk^*_{j0} \lor sk^*_{j1} \]

\[ sk_{jb} \]

Los Angeles, nov 15 2006
Simulator for the MiM

\[ x \in L \]

\[ x' \in L \]

\[ \text{sk}^*_{j0} \lor \text{sk}^*_{j1} \]

\[ \text{sk}_{j0} \lor \text{sk}_{j1} \]

\[ \text{sk}^*_{j(b)} \]

\[ \text{sk}_{jb} \]
The MiM for CNMZK in BPK is reduced to a MiM for CNMWI in the plain model

\[
\begin{align*}
  x & \in L \\
  x' & \in L \\
  sk^*_{j0} \lor sk^*_{j1} & \\
  sk_{j0} \lor sk_{j1} & \\
  x \in L \lor sk^*_{j0} \lor sk^*_{j1} & \\
  x' \in L \lor sk_{j0} \lor sk_{j1} & \\
  sk_{jb} & \\
  sk_{jb}, sk^*_{j(b)} & \\
  sk_{j0} \lor sk_{j1} & \\
  sk^*_{j0} \lor sk^*_{j1} & \\
  x \in L \lor sk^*_{j0} \lor sk^*_{j1} & \\
  x' \in L \lor sk_{j0} \lor sk_{j1} & \\
  sk_{jb} & \\
\end{align*}
\]
Reducing the MiM to a MiM for CNMWI

\[ \text{sk}_{j0} \lor \text{sk}_{j1} \]
\[ x \in L \lor \text{sk}^*_{j0} \lor \text{sk}^*_{j1} \]

\[ \text{sk}_{j0}, \text{sk}^*_{j(0)} \]

\[ \text{sk}_{j1} \lor \text{sk}_{j1} \]
\[ x \in L \lor \text{sk}^*_{j0} \lor \text{sk}^*_{j1} \]

\[ \text{sk}^*_{j0} \lor \text{sk}^*_{j1} \]
\[ x' \in L \lor \text{sk}_{j0} \lor \text{sk}_{j1} \]

\[ \text{sk}_{j0} \]

\[ \text{sk}_{j1}, \text{sk}^*_{j(1)} \]

\[ \text{sk}^*_{j0} \lor \text{sk}^*_{j1} \]
\[ x' \in L \lor \text{sk}^*_{j0} \lor \text{sk}^*_{j1} \]

\[ \text{sk}_{j1} \]
## Comparison with previous CNMZK

<table>
<thead>
<tr>
<th>Paper</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDOPS 01</td>
<td>Shared Random String</td>
</tr>
<tr>
<td>PS 04 / BS 05</td>
<td>Relaxed Security</td>
</tr>
<tr>
<td>KLP 05</td>
<td>Timing Assumption</td>
</tr>
<tr>
<td>PRS 06</td>
<td>Plain (polylog rounds)</td>
</tr>
<tr>
<td>This work</td>
<td>Bare Public Key</td>
</tr>
</tbody>
</table>
Outline

- Concurrent ZK, NMZK, Witness Indist.
- Non-Malleable Witness Indistinguishability
- Cnst-Rnd Concurrent NMWI in the plain model
- Cnst-Rnd Concurrent NMZK in the BPK Model
- UC with preprocessing
UC [Can01+CLOS02+BCNP04]

- [CLOS02] UC for any functionality can be reduced to realizing $F_{\text{mcom}}$ (multi-instance commitment functionality)
- [BCNP04] $F_{\text{mcom}}$ can be reduced to realizing $F_{\text{kr}}$ (key registration funct.)
Key Registration Funct. [BCNPO04]

- $F_{kr}$ requires that the functionality can see each private key and guarantees that
  - each party has a well formed public key
  - the public keys of the honest parties are safe (private keys are not known by the adversary)
Key Registration Funct. [BCNP04]

- $F_{kr}$ is realized in BCNP04
- Assuming the existence of trusted third parties
  - With any $F_{crs}$
  - With a PKI-like registration service where the key authority generates public keys and gives the public keys to parties
  - With a PKI-like registration service where parties generate keys but have to send both the public and secret keys to the authority
  - With semi-trusted authorities
- Assuming isolated stand-alone executions
  - Each party generates a public key and gives a ZKPoK of the secret key to a trusted authority
UC with Preprocessing

- key-stage preprocessing (non-interactive):
  - run the key-stage of the CNMZK protocol in the BPK model; each party generates and posts also the additional public key PK used in BCNPO4

- key-knowledge preprocessing (interactive):
  - each party interested in running protocols with other parties, runs the proof stage of the CNMZK protocol in the BPK model, proving knowledge of the secret key SK
## Comparison with previous results

<table>
<thead>
<tr>
<th>Paper</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLOS 02</td>
<td>Common Reference String</td>
</tr>
<tr>
<td>BCNP 04</td>
<td>TTP or Isolated ZKPoK</td>
</tr>
<tr>
<td>PS 04 / BS 05</td>
<td>Relaxed Security</td>
</tr>
<tr>
<td>KLP 05</td>
<td>Timing Assumption</td>
</tr>
<tr>
<td><strong>This work</strong></td>
<td><strong>Preprocessing (2 stages)</strong></td>
</tr>
</tbody>
</table>
the prover

the verifier

the simulator

the extractor

the man-in-the-middle

Thanks!