From Selective-ID to Full-ID IBS without Random Oracles

Sanjit Chatterjee and Chethan Kamath

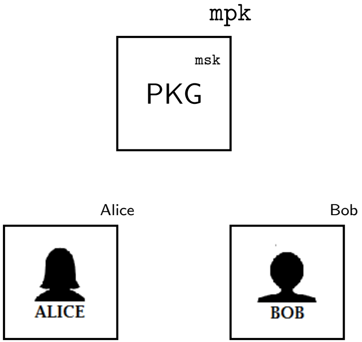
Indian Institute of Science, Bangalore

November 3, 2013

Table of contents

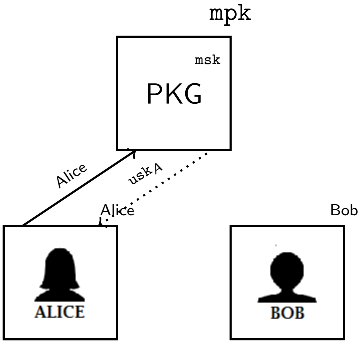
Identity-Based Cryptography

- Introduced by Shamir in 1984.
- Any *arbitrary* string, say e-mail address, can be used as public key.
- Certificate management can be avoided.
- A trusted *private key generator* (PKG) generates secret keys.



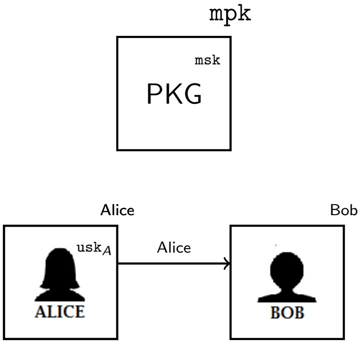
Identity-Based Cryptography

- Introduced by Shamir in 1984.
- Any *arbitrary* string, say e-mail address, can be used as public key.
- Certificate management can be avoided.
- A trusted *private key generator* (PKG) generates secret keys.



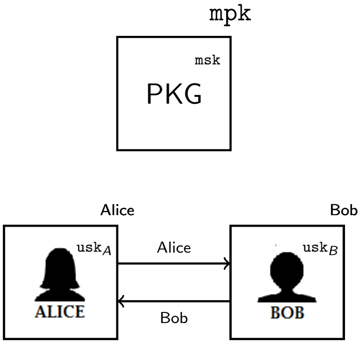
Identity-Based Cryptography

- Introduced by Shamir in 1984.
- Any *arbitrary* string, say e-mail address, can be used as public key.
- Certificate management can be avoided.
- A trusted *private key generator* (PKG) generates secret keys.



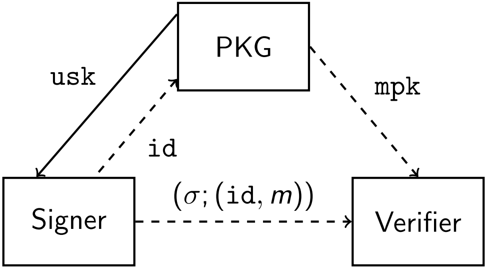
Identity-Based Cryptography

- Introduced by Shamir in 1984.
- Any *arbitrary* string, say e-mail address, can be used as public key.
- Certificate management can be avoided.
- A trusted *private key generator* (PKG) generates secret keys.



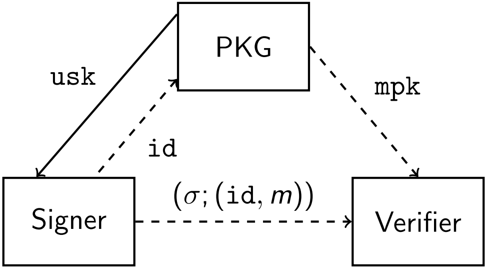
Identity-Based Signatures

- IBS is the concept of digital signatures *extended* to identity-based setting.



Identity-Based Signatures

- IBS is the concept of digital signatures *extended* to identity-based setting.



- Focus of the talk: construction of IBS schemes

FORMAL DEFINITIONS

Public-Key Signature

Consists of three PPT algorithms $\{\mathcal{K}, \mathcal{S}, \mathcal{V}\}$:

- **Key Generation**, $\mathcal{K}(\kappa)$
  - Used by the *signer* to generate the key-pair ($\mathtt{pk}$,$\mathtt{sk}$)
  - $\mathtt{pk}$ is published and the $\mathtt{sk}$ kept secret
- **Signing**, $\mathcal{S}_{\mathtt{sk}}(m)$
  - Used by the *signer* to generate signature on some message m
  - The secret key $\mathtt{sk}$ used for signing
- **Verification**, $\mathcal{V}_{\mathtt{pk}}(\sigma, m)$
  - Used by the *verifier* to validate a signature
  - Outputs 1 if σ is a valid signature on m; else, outputs 0

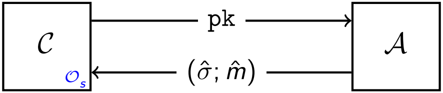
Identity-Based Signature

Consists of four PPT algorithms $\{\mathcal{G}, \mathcal{E}, \mathcal{S}, \mathcal{V}\}$:

- **Set-up**, $\mathcal{G}(\kappa)$
  - Used by *PKG* to generate the master key-pair ($\mathtt{mpk}$,$\mathtt{msk}$)
  - $\mathtt{mpk}$ is published and the $\mathtt{msk}$ kept secret
- **Key Extraction**, $\mathcal{E}_{\mathtt{msk}}(\mathtt{id})$
  - Used by *PKG* to generate the user secret key ($\mathtt{usk}$)
  - $\mathtt{usk}$ is then distributed through a secure channel
- **Signing**, $\mathcal{S}_{\mathtt{usk}}(\mathtt{id}, m)$
  - Used by the *signer* (with identity $\mathtt{id}$) to generate signature on some message m
  - The *user* secret key $\mathtt{usk}$ used for signing
- **Verification**, $\mathcal{V}_{\mathtt{mpk}}(\sigma, \mathtt{id}, m)$
  - Used by the *verifier* to validate a signature
  - Outputs 1 if σ is a valid signature on m by the user with identity $\mathtt{id}$; otherwise, outputs 0

STANDARD SECURITY MODELS

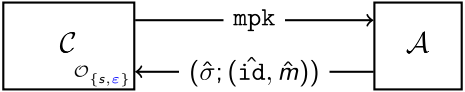
Security Model for PKS: EU-CMA



- Existential unforgeability under chosen-message attack
- $\mathcal{C}$ generates key-pair $(\mathtt{pk}, \mathtt{sk})$ and passes $\mathtt{pk}$ to $\mathcal{A}$.
- Signature Queries: Access to a signing oracle $\mathcal{O}_s$
- Forgery: $\mathcal{A}$ wins if $(\hat{\sigma}; \hat{m})$ is *valid* and *non-trivial*
- Adversary's advantage in the game $\mathsf{Adv}_{\mathcal{A}}^{\mathtt{EU-CMA}}(\kappa)$:

$$\Pr\left[1 \leftarrow \mathcal{V}_{\mathtt{pk}}(\hat{\sigma}; \hat{m}) \mid (\mathtt{sk}, \mathtt{pk}) \xleftarrow{\$} \mathcal{K}(\kappa); (\hat{\sigma}; \hat{m}) \xleftarrow{\$} \mathcal{A}^{\mathcal{O}_s}(\mathtt{pk})\right]$$

Security Model for IBS: $\mathtt{EU\text{-}ID\text{-}CMA}$



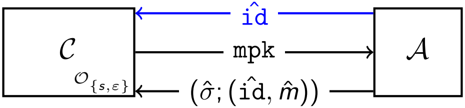
- Existential unforgeability with adaptive identity under chosen-message attack
- $\mathcal{C}$ generates key-pair $(\mathtt{mpk}, \mathtt{msk})$ and passes $\mathtt{mpk}$ to $\mathcal{A}$.
- Extract Queries, Signature Queries
- Forgery: $\mathcal{A}$ wins if $(\hat{\sigma}; (\hat{\mathtt{id}}, \hat{m}))$ is *valid* and *non-trivial*
- Adversary's advantage in the game $\mathsf{Adv}_{\mathcal{A}}^{\mathtt{EU-ID-CMA}}(\kappa)$:

$$\Pr\left[1 \leftarrow \mathcal{V}_{\mathtt{mpk}}(\hat{\sigma}; (\hat{\mathtt{id}}, \hat{m})) \mid (\mathtt{msk}, \mathtt{mpk}) \xleftarrow{\$} \mathcal{G}(\kappa); (\hat{\sigma}; (\hat{\mathtt{id}}, \hat{m})) \xleftarrow{\$} \mathcal{A}^{\mathcal{O}_{\{s,\varepsilon\}}}(\mathtt{mpk})\right]$$

THE SELECTIVE-IDENTITY MODEL

sID Model: Salient Features

- Introduced by Canetti *et al.*
- *Weaker* than the full model (EU-ID-CMA)
  - However, *easier* to design sID-secure protocols
- Adversary has to, beforehand, commit to the *target* identity
  - *Target* identity: the identity on which the adversary forges on
  - Adversary *cannot* extract query on the target identity



CONSTRUCTION OF IBS

Construction of IBS

- Considered *easier* task than IBE
- *Folklore* method: $\texttt{EU-ID-CMA}$-IBS $\equiv$ 2($\texttt{EU-CMA}$-PKS)
  - ($\texttt{EU-CMA}$-PKS) $\equiv$ ($\texttt{EU-GCMA}$-PKS)+($\texttt{CR}$-CHF)
  - Implies $\texttt{EU-ID-CMA}$-IBS $\equiv$ 2(($\texttt{EU-GCMA}$-PKS)+($\texttt{CR}$-CHF))

Construction of IBS

- Considered *easier* task than IBE
- *Folklore* method: $\texttt{EU-ID-CMA}$-IBS $\equiv$ 2($\texttt{EU-CMA}$-PKS)
    - ($\texttt{EU-CMA}$-PKS) $\equiv$ ($\texttt{EU-GCMA}$-PKS)+($\texttt{CR}$-CHF)
    - Implies $\texttt{EU-ID-CMA}$-IBS $\equiv$ 2(($\texttt{EU-GCMA}$-PKS)+($\texttt{CR}$-CHF))

- *From* $\texttt{sID}$ *Model*:
    - Random Oracle Model: guess the *index* of the target identity: polynomial degradation
    - Standard Model: guess the *target* identity itself: exponential degradation

Construction of IBS...

- Goal: construct ID-secure IBS from sID-secure IBS
  1. without random oracles
  2. with sub-exponential degradation (preferably, polynomial)

...Construction of IBS...

- Goal: construct $\texttt{ID}$-secure IBS from $\texttt{sID}$-secure IBS
  1. without random oracles
  2. with sub-exponential degradation (preferably, polynomial)

- Main result: $\texttt{EU-ID-CMA}$-IBS $\equiv$ ($\texttt{EU-sID-CMA}$-IBS)+($\texttt{EU-GCMA}$-PKS)+($\texttt{CR}$-CHF)
- Further: $\texttt{EU-ID-CMA}$-IBS $\equiv$ ($\texttt{EU-wID-CMA}$-IBS)+($\texttt{EU-GCMA}$-PKS)+($\texttt{CR}$-CHF)

THE TRANSFORMATION

Objects used

1. Chameleon Hash Function
2. `GCMA`-secure PKS

Chameleon Hash Function

- A family of randomised trapdoor hash functions
- Collision Resistant (CR)
- "Chameleon" property: anyone with trapdoor information can efficiently generate collisions

...Chameleon Hash Function...

Consists of three PPT $\{\mathcal{G}, \mathsf{h}, \mathsf{h}^{-1}\}$:

Key Generation, $\mathcal{G}(\kappa)$:

- Generates evaluation key `ek` and trapdoor key `td`

Hash Evaluation, $\mathsf{h}_{\mathtt{ek}}(m, r)$:

- A randomiser r used to evaluate the hash

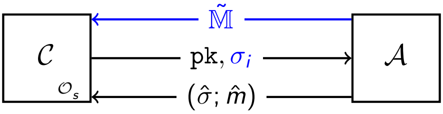
Collision Generation, $\mathsf{h}^{-1}_{\mathtt{td}}(m, r, m')$:

- Outputs randomiser r' such that (m, r) and (m', r') is a *collision*:

$$\mathsf{h}_{\mathtt{ek}}(m, r) = \mathsf{h}_{\mathtt{ek}}(m', r')$$

GCMA-secure PKS

- Adversary has to, beforehand, commit to a set of messages $\tilde{\mathbb{M}}$
  - The adversary can query with $\mathcal{O}_s$ on any message from $\tilde{\mathbb{M}}$
  - Adversary has to forge on a message *not in* $\tilde{\mathbb{M}}$



The Transformation

In a nutshell

- Takes as input:
  1. an EU-sID-CMA-secure IBS $\mathfrak{I}_s := \{\mathcal{G}_s, \mathcal{E}_s, \mathcal{S}_s, \mathcal{V}_s\}$
  2. a collision-resistant CHF $\mathfrak{H} := \{\mathcal{G}_h, \mathsf{h}, \mathsf{h}^{-1}\}$
  3. a GCMA-secure PKS $\mathfrak{P} := \{\mathcal{K}, \mathcal{S}_p, \mathcal{V}_p\}$
- Outputs an EU-ID-CMA-secure IBS $\mathfrak{I} := \{\mathcal{G}, \mathcal{E}, \mathcal{S}, \mathcal{V}\}$

The Transformation

In a nutshell

- Takes as input:
  1. an EU-sID-CMA-secure IBS $\mathfrak{I}_s := \{\mathcal{G}_s, \mathcal{E}_s, \mathcal{S}_s, \mathcal{V}_s\}$
  2. a collision-resistant CHF $\mathfrak{H} := \{\mathcal{G}_h, \mathsf{h}, \mathsf{h}^{-1}\}$
  3. a GCMA-secure PKS $\mathfrak{P} := \{\mathcal{K}, \mathcal{S}_p, \mathcal{V}_p\}$
- Outputs an EU-ID-CMA-secure IBS $\mathfrak{I} := \{\mathcal{G}, \mathcal{E}, \mathcal{S}, \mathcal{V}\}$

The idea:

- CHF used to map identities between $\mathfrak{I}$ and $\mathfrak{I}_s$
- PKS used to bind these identities

The Transformation...

Set-up, $\mathcal{G}(\kappa)$:

- Invoke $\mathcal{G}_s$, $\mathcal{K}$ and $\mathcal{G}_h$ to obtain $(\mathtt{msk}_s, \mathtt{mpk}_s)$, $(\mathtt{sk}, \mathtt{pk})$ and $(\mathtt{ek}, \mathtt{td})$
- Return $\mathtt{msk} := (\mathtt{msk}_s, \mathtt{sk})$ and $\mathtt{mpk} := (\mathtt{mpk}_s, \mathtt{pk}, \mathtt{ek})$

Key Extraction, $\mathcal{E}_{\mathtt{msk}}(\mathtt{id})$:

- Select a random r and compute $\mathtt{id}_s \leftarrow \mathsf{h}_{\mathtt{ek}}(\mathtt{id}, r)$
- Compute $\mathtt{usk}_s \xleftarrow{\$} \mathcal{E}_{s,\mathtt{msk}_s}(\mathtt{id}_s)$ and $\sigma_p \xleftarrow{\$} \mathcal{S}_{p,\mathtt{sk}}(\mathtt{id}_s)$
- Return $\mathtt{usk} := (\mathtt{usk}_s, r, \sigma_p)$

Signing, $\mathcal{S}_{\mathtt{usk}}(\mathtt{id}, m)$:

- Compute $\sigma_s \xleftarrow{\$} \mathcal{S}_{s,\mathtt{usk}_s}(\mathtt{id}_s, m)$
- Return $\sigma := (\sigma_s, r, \sigma_p)$ as the signature

Verification, $\mathcal{V}_{\mathtt{mpk}}(\sigma, \mathtt{id}, m)$:

- Return 1 only if σ_p and σ_s are valid signatures

SECURITY

Security Argument

Strategy:

- Adversaries classified into three: `type 1`, `type 2` and `type 3`
- `type 1`: break `sID`-security; `type 2` or `type 3`: break the binding

Adversary	Reduction	From	Degradation
`type 1`	$\mathcal{B}_s$	$\mathfrak{I}_s$	$O(q_s)$
`type 2`	$\mathcal{B}_p$	$\mathfrak{P}$	$O(1)$
`type 3`	$\mathcal{B}_h$	$\mathfrak{H}$	$O(1)$

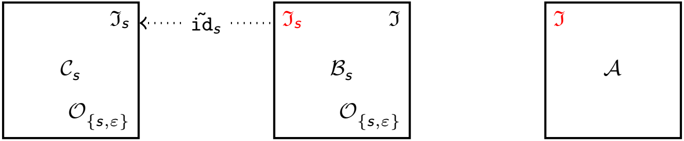
Table: q_s denotes the number of signature queries

Reduction $\mathcal{B}_s$

In a nutshell:

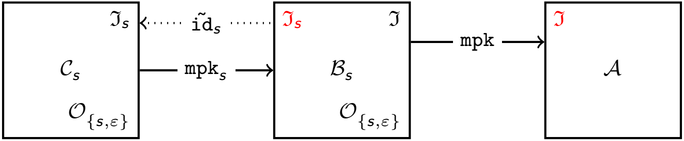
- Break $\texttt{sID}$-security – plug in challenge $\texttt{msk}_s$ in the IBS $\mathfrak{I}$
- $\texttt{type 1}$ adversary: target identity was queried to $\mathcal{O}_s$
- Strategy: guess the index of this *target* identity
  - Hence the $\mathrm{O}(q_s)$ degradation

...Reduction $\mathcal{B}_s$...



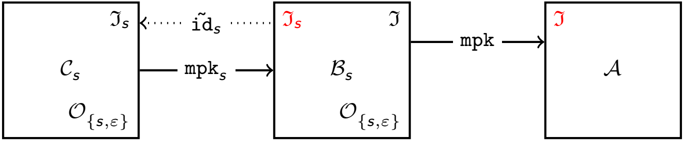
- Invoke $\mathcal{K}$ and $\mathcal{G}_h$ to obtain $(\mathtt{sk}, \mathtt{pk})$ and $(\mathtt{ek}, \mathtt{td})$
- Choose random $\mathtt{id}, r$ and commit $\tilde{\mathtt{id}} := \mathsf{h}_{ek}(\mathtt{id}, r)$ to $\mathcal{C}_s$ as the target identity; Make a guess $\tilde{\ell}$

...Reduction $\mathcal{B}_s$...



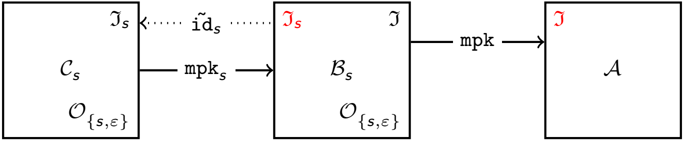
- Invoke $\mathcal{K}$ and $\mathcal{G}_h$ to obtain $(\mathtt{sk}, \mathtt{pk})$ and $(\mathtt{ek}, \mathtt{td})$
- Choose random $\mathtt{id}, r$ and commit $\tilde{\mathtt{id}} := \mathsf{h}_{ek}(\mathtt{id}, r)$ to $\mathcal{C}_s$ as the target identity; Make a guess $\tilde{\ell}$
- $\mathcal{C}_s$ releases $\mathtt{mpk}_s$ $\mathcal{B}_s$ passes $\mathtt{mpk} := (\mathtt{mpk}_s, \mathtt{pk}, \mathtt{ek})$ to $\mathcal{A}$;

...Reduction $\mathcal{B}_s$...



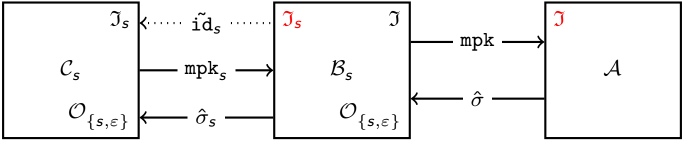
- Invoke $\mathcal{K}$ and $\mathcal{G}_h$ to obtain $(\mathtt{sk}, \mathtt{pk})$ and $(\mathtt{ek}, \mathtt{td})$
- Choose random $\mathtt{id}, r$ and commit $\tilde{\mathtt{id}} := \mathsf{h}_{ek}(\mathtt{id}, r)$ to $\mathcal{C}_s$ as the target identity; Make a guess $\tilde{\ell}$
- $\mathcal{C}_s$ releases $\mathtt{mpk}_s$ $\mathcal{B}_s$ passes $\mathtt{mpk} := (\mathtt{mpk}_s, \mathtt{pk}, \mathtt{ek})$ to $\mathcal{A}$;
- Extract Queries on $\mathtt{id}$:
  1. If query on the ℓ^{th} identity then abort (abort_1); else map $\mathtt{id}$ to a random $\mathtt{id}_s$
  2. Query oracle $\mathcal{O}_\varepsilon$ of $\mathcal{C}_s$ with $\tilde{\mathtt{id}}$

Reduction $\mathcal{B}_s$...



- Signature Queries on $(\mathtt{id}, m)$:
  1. If query on the $\tilde{\ell}^{\text{th}}$ identity then map $\mathtt{id}$ to $\tilde{\mathtt{id}}_s$ (using knowledge of trapdoor $\mathtt{td}$); else map to a random $\mathtt{id}_s$
  2. Query oracle $\mathcal{O}_s$ of $\mathcal{C}_s$ with $(\tilde{\mathtt{id}}, m)$

...Reduction $\mathcal{B}_s$...



- Signature Queries on $(\mathtt{id}, m)$:
  1. If query on the $\tilde{\ell}^{\text{th}}$ identity then map $\mathtt{id}$ to $\tilde{\mathtt{id}}_s$ (using knowledge of trapdoor $\mathtt{td}$); else map to a random $\mathtt{id}_s$
  2. Query oracle $\mathcal{O}_s$ of $\mathcal{C}_s$ with $(\tilde{\mathtt{id}}, m)$
- Forgery (σ, r, σ_p): If the forgery is on the ℓ^{th} identity, pass σ to $\mathcal{C}_s$; else abort (abort_2)

Analysis of $\mathcal{B}_s$

- Success probability governed by abort_1 and abort_2:

$$\mathsf{Adv}_{\mathcal{B}}^{\mathtt{EU-sID-CMA}}(\kappa) = \Pr\left[\neg \mathsf{abort}_1 \wedge \neg \mathsf{abort}_2\right] \times \mathsf{Adv}_{\mathcal{A}}^{\mathtt{EU-ID-CMA}}(\kappa)$$

- $\Pr\left[\neg \mathsf{abort}_2\right]$ is the *same* as that of guessing $\tilde{\ell}$

$$\Pr\left[\neg \mathsf{abort}_2\right] = 1/q_s$$

- $\Pr\left[\neg \mathsf{abort}_1 \mid \neg \mathsf{abort}_2\right] = 1$

Analysis of $\mathcal{B}_s$

- Success probability governed by abort_1 and abort_2:

$$\mathsf{Adv}_{\mathcal{B}}^{\mathtt{EU-sID-CMA}}(\kappa) = \Pr\left[\neg\mathsf{abort}_1 \wedge \neg\mathsf{abort}_2\right] \times \mathsf{Adv}_{\mathcal{A}}^{\mathtt{EU-ID-CMA}}(\kappa)$$

- $\Pr\left[\neg\mathsf{abort}_2\right]$ is the *same* as that of guessing $\tilde{\ell}$

$$\Pr\left[\neg\mathsf{abort}_2\right] = 1/q_s$$

- $\Pr\left[\neg\mathsf{abort}_1 \mid \neg\mathsf{abort}_2\right] = 1$
- Hence

$$\mathsf{Adv}_{\mathcal{B}}^{\mathtt{EU-sID-CMA}}(\kappa) = \mathsf{Adv}_{\mathcal{A}}^{\mathtt{EU-ID-CMA}}(\kappa)/q_s$$

TRANSFORMING FROM THE `wID` MODEL

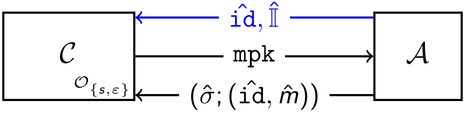
Transforming from the $\texttt{wID}$ Model

- $\texttt{wID}$: the weak selective-identity model
- Adversary has to, beforehand, commit to the *target* identity and a set of query identities
  - *Target* identity: the identity on which the adversary forges on
  - *Query* identities: the identities which it can query with $\mathcal{O}_{\{s,\varepsilon\}}$
  - Adversary *cannot* extract query on the target identity

$$
\begin{array}{ccc}
\boxed{\mathcal{C}\ \ \mathcal{O}_{\{s,\varepsilon\}}} & \xleftarrow{\hat{\texttt{id}},\ \hat{\mathbb{I}}} & \boxed{\mathcal{A}} \\
 & \xrightarrow{\texttt{mpk}} & \\
 & \xleftarrow{(\hat{\sigma};(\hat{\texttt{id}},\hat{m}))} &
\end{array}
$$

Transforming from the wID Model

- wID : the weak selective-identity model
- Adversary has to, beforehand, commit to the *target* identity and a set of query identities
  - *Target* identity: the identity on which the adversary forges on
  - *Query* identities: the identities which it can query with $\mathcal{O}_{\{s,\varepsilon\}}$
  - Adversary *cannot* extract query on the target identity



- A similar transformation *holds* for wID as well
  - EU-ID-CMA-IBS $\equiv$ (EU-wID-CMA-IBS)+(EU-GCMA-PKS)+(CR-CHF)

Conclusion and Future Work

- We discussed a generic transformation from `sID`/`wID` IBS to `ID` IBS
- Alternative *paradigm* for construction of IBS
- Linear degradation

Future Work

- Further *simplification* of the assumptions
- Transformation using *fewer* objects

THANK YOU!