The 29th Australasian Conference on Information Security and Privacy

Pairing-Free ID-Based Signatures as Secure as Discrete Logarithm in AGM

Jia-Chng (Jason) Loh, Fuchun Guo, Willy Susilo

Institute of Cybersecurity and Cryptology,
University of Wollongong, Australia

iC²
Institute of Cybersecurity and Cryptology

UNIVERSITY OF WOLLONGONG AUSTRALIA

OUTLINE

P

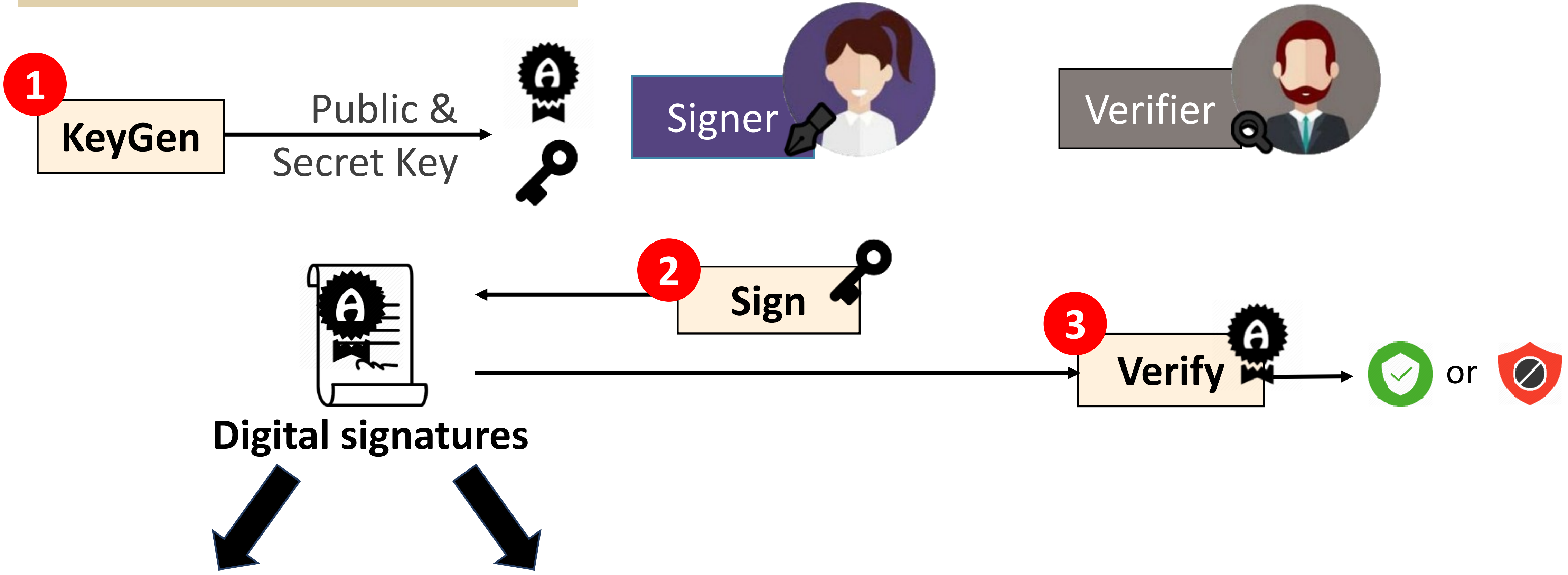
ONE

01

Introduction

Digital Signatures

To sign a digital document…



Computed mathematically

Provable secure (forging is **computationally hard**)

Identity-Based Signatures

In practice, digital signatures require the Public Key Infrastructure (PKI)

Identity (ID)-based Signatures (IBS) – [Shamir84]

- Users' identity *ID* serves as the public key
- E.g. email address and ID number

How to Prove?

To prove the security of a scheme…

Pre-defined **1** **Security Model (Game)**

Adversary $\mathcal{A}$ against the scheme

$\epsilon_{\mathcal{A}}$

Reduction

Reduction/ Security Loss **3**

$L = \epsilon_{\mathcal{A}}/\epsilon_{\mathcal{B}}$

$\epsilon_{\mathcal{B}}$

Simulator $\mathcal{B}$ for **Hard Problems** **2**

"Ideal security" in cyclic group setting

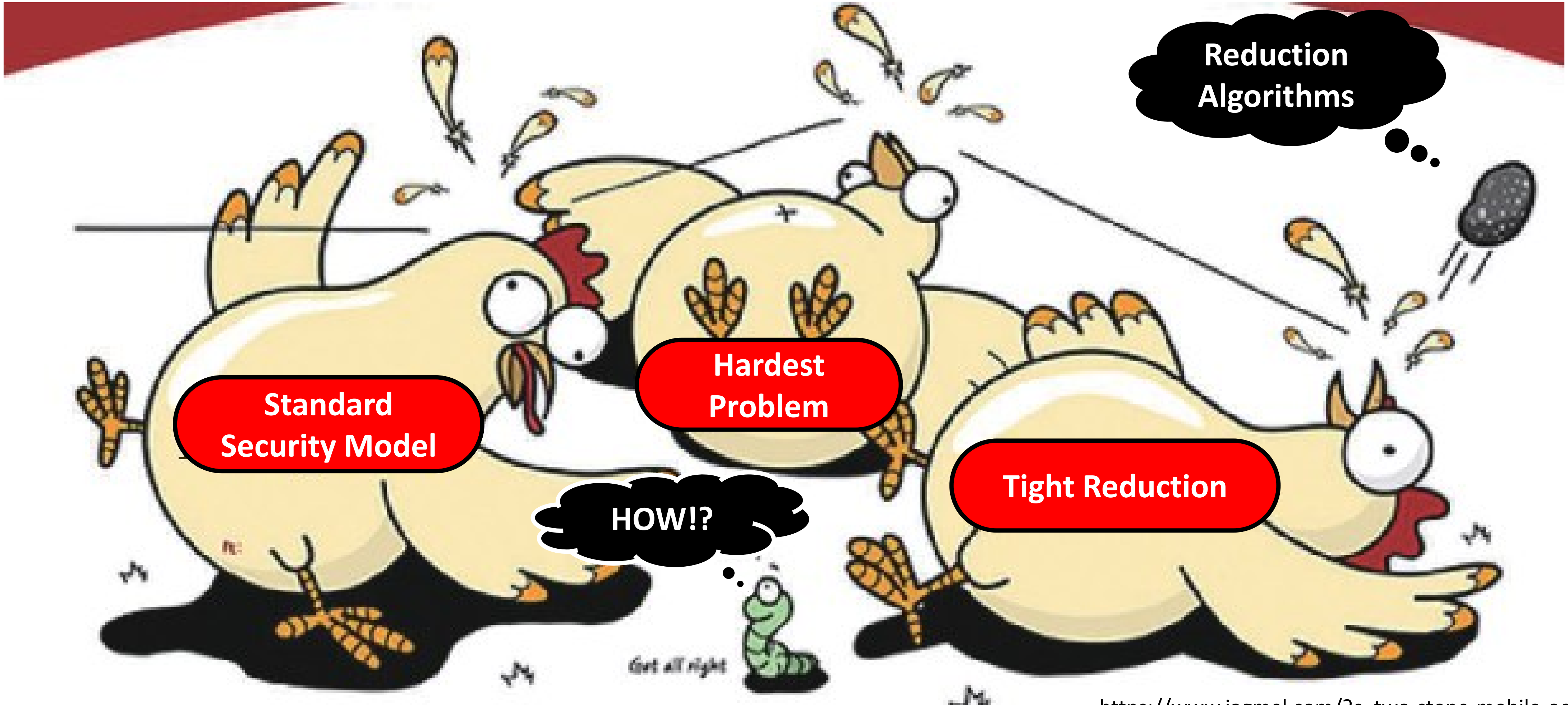
1. Standard security model: EUF-CMA
2. Hardest problem: Discrete logarithm (DL)
3. Tight reduction: Loss factor is $O(1)$

Better theoretical result

Efficient construction

Optimal parameter size

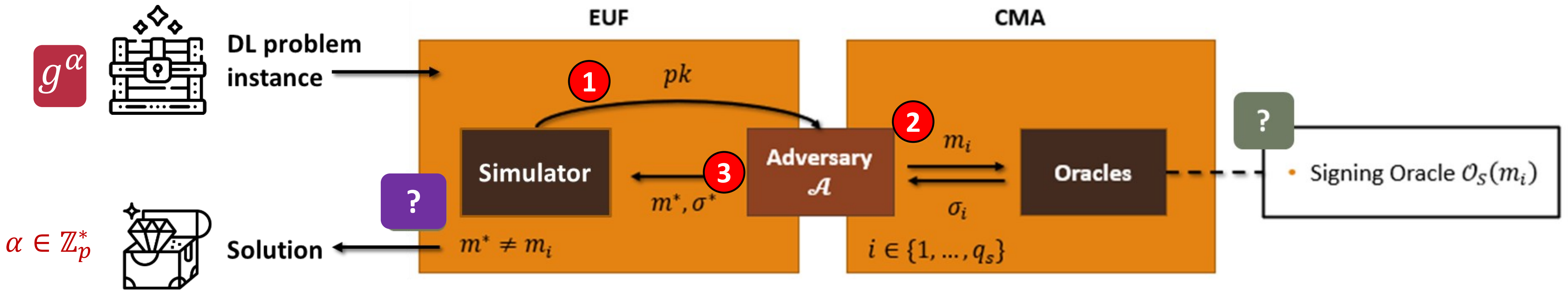
One Stone Three Birds?



https://www.jagmel.com/?c=two-stone-mobile-oo-43XMRJI4

How to Achieve: Ideal Security

Constructing and proving signature schemes with ideal security is challenging



Can they both succeed with overwhelming success probability?

?

?

Yes → **Reduction is tight**

No →

a. Modify scheme constructions

b. Prove under weaker models or easier problems

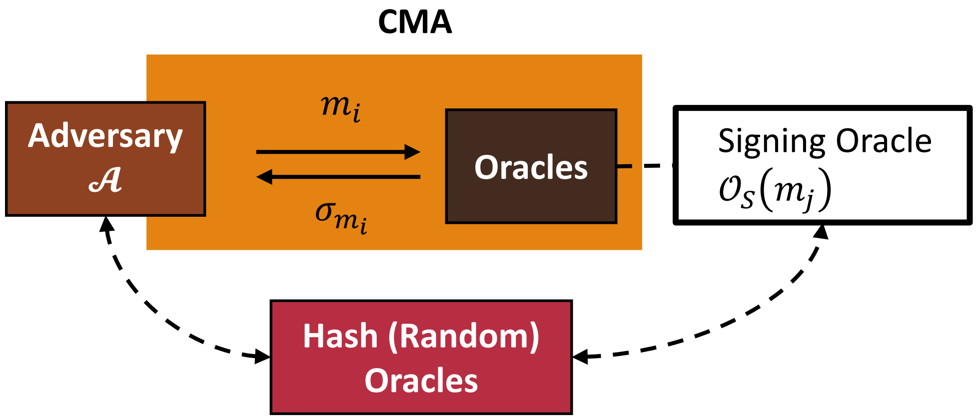
c. Resort "**Idealized Models**"

The Rescuer: Idealized Models

Enabling proofs based on certain idealizations

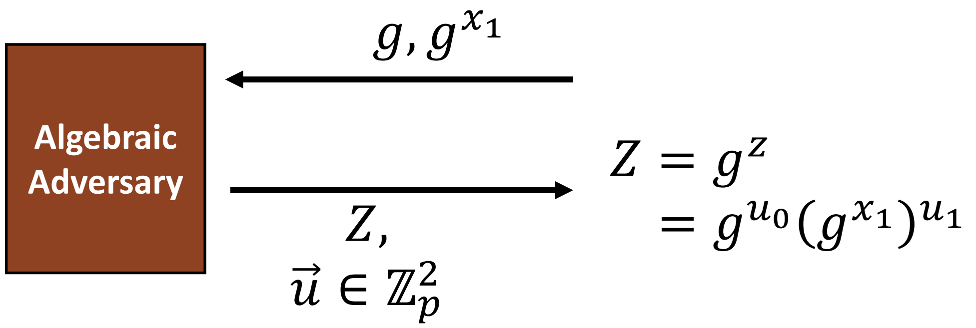
Random Oracle Model (ROM) – [BR93]

The hash functions is mediated (random).



Algebraic Group Model (AGM) – [FKL18]

The adversary's computation is algebraic over group.
Let $g \in \mathbb{G}, x \in \mathbb{Z}_p$



$$z = u_0 + x_1 u_1$$

TWO

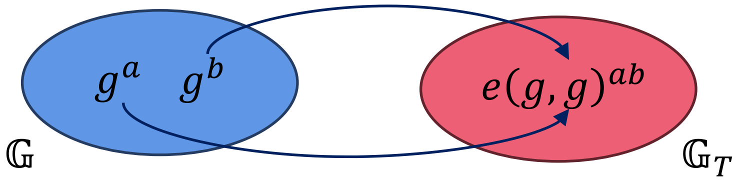
02

Research Motivations & Gaps

Why Pairing-Free IBS?

Pairing-Free Schemes

Pairing:



- Without bilinear pairing property
- Better efficiency
- Lighter computational complexity

Prior **candidates** for **resource-constrained** applications

Some Notable Results under ID-Based EUF-CMA

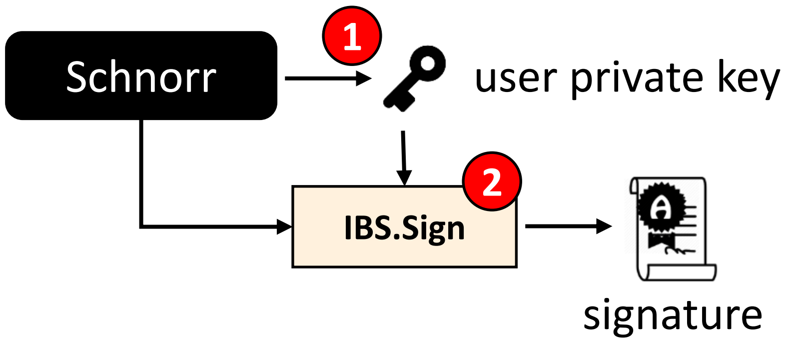
	Pairing-Free	Hardness Assumption	Tight Reduction	Standard Model (SM)/ ROM/ AGM
ChCh-IBS [CH03]	✗	CDH	✗	ROM
BBMQ-IBS [BBMQ05]	✗	q-SDH	✗	ROM
Waters-IBS [PS06]	✗	CDH	✗	SM
BBG-IBS [KN09]	✗	mCDH	✗	SM
BNN-IBS [BNN09], Beth-IBS [Beth88], Schnorr-like IBS [GG09]	✓	DL	✗	ROM
FH-IBS* [FH17,18]	✓	DDH	✓	ROM
BLS-IBS [LGSY23]	✗	DL	✓	AGM + ROM
This work	✓	DL	✓	AGM + ROM

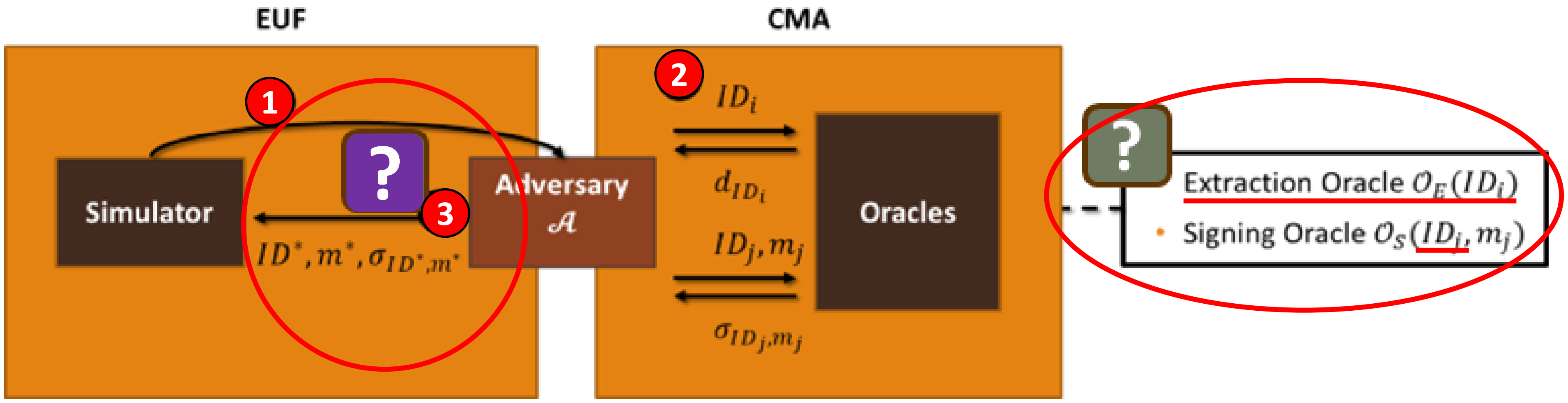
* A slightly different EUF-CMA model: simulator may return different user private key may for each query

Gaps to "Ideal Security": From Most Efficient Pairing-free IBS

Schnorr-like IBS by Galindo-Garcia @ AfricaCrypt'09

- Most efficient – based on Schnorr's signatures
- Proven under DL assumption – Loose reduction in ROM
- Extra caution: Chosen-identity-and-message attacks





Institute of Cybersecurity and Cryptology

UNIVERSITY OF WOLLONGONG AUSTRALIA

P

THREE

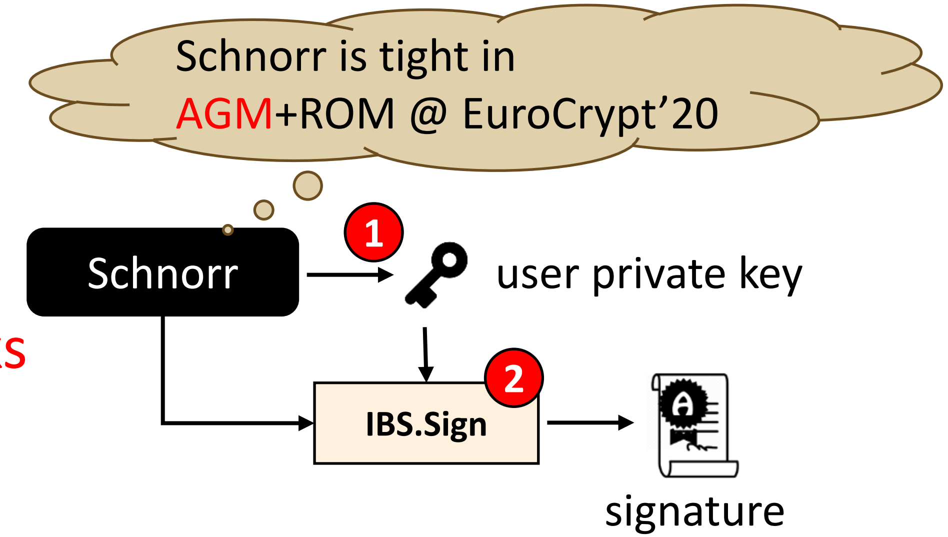
03

Challenge & Contribution

Challenge Encountered in Schnorr-like IBS

Galindo-Garcia's (GG) IBS @ AfricaCrypt'09

- Extra caution: Chosen-identity-and-message attacks
- Whether AGM+ROM helps?



iC²
Institute of Cybersecurity and Cryptology

UNIVERSITY OF WOLLONGONG AUSTRALIA

Challenge Encountered in Schnorr-like IBS

Schnorr is tight in AGM+ROM @ EuroCrypt'20

Galindo-Garcia's (GG) IBS @ AfricaCrypt'09

- Extra caution: Chosen-identity-and-message attacks
- Whether AGM+ROM helps? ➡ ☹

Schnorr → 1 user private key → 2 IBS.Sign → signature

Oracles

Signing Oracle $\mathcal{O}_S(ID', m_j) \rightarrow \sigma_{ID', m_j}$

Extraction Oracle $\mathcal{O}_E(ID') \nrightarrow d_{ID'}$

Adversary $\mathcal{A}$ → $ID^* = ID'$, m^*, σ_{ID^*, m^*} → **Simulator** →

Can it simulate any key $d_{ID'}$? ✗

Can it reduce any forgery? ✗

Couldn't solve both with existing known techniques (even with AGM)

I don't believe...

Next... How Schnorr achieves ideal security in AGM + ROM

iC² Institute of Cybersecurity and Cryptology

UNIVERSITY OF WOLLONGONG AUSTRALIA

Schnorr's Signatures

Schnorr's Signatures

Users' perspective

Parameters

$g \in \mathbb{G}$

$H(\cdot,\cdot) \to h_i \in \mathbb{Z}_p^*$

KeyGen → Public and secret key pair

$pk = Z = g^z$

$sk = z \in \mathbb{Z}_p^*$

Message m_i → **Sign** → Signature σ_{m_i}

$$\sigma_{m_i} \begin{cases} R_i = g^{r_i} \\ y_i = r_i + z \cdot h_i \end{cases}$$

such that $r_i \in \mathbb{Z}_p^*$ and $h_i = H(m_i, R_i)$

Message and Signature pair (m_i, σ_{m_i}) → **Verify** → ✓ or ✗

Check if:

$$g^{y_i} \stackrel{?}{=} R_i \cdot Z^{H(m_i, R_i)}$$

Schnorr Simulation (DL Problem)

Schnorr's Signatures

$\left\{ \begin{array}{l} g \\ g^{\alpha} \end{array} \right.$

Simulator's perspective

Parameters

$g \in \mathbb{G}$

$H(\cdot,\cdot) \rightarrow h_i \in \mathbb{Z}_p^*$

Random Oracle

Hash list

Setup

Public and secret key pair

$pk = Z = g^{\alpha}$

$\cancel{sk = z \in \mathbb{Z}_p^*}$

Message m_i

$\mathcal{O}_{Sign}$

Signature

$\sigma_{m_i} \left\{ \begin{array}{l} R_i = g^{s_i}(g^{\alpha})^{-h_i} \\ y_i = s_i \end{array} \right.$

select $s_i, h_i \in \mathbb{Z}_p^*$ and

program $h_i = H(m_i, R_i)$

Message and Signature pair (m_i, σ_{m_i})

Verify

or

Check if:

$$g^{y_i} \stackrel{?}{=} R_i \cdot Z^{H(m_i,R_i)}$$

$$g^{s_i} \stackrel{?}{=} \left(g^{s_i} g^{-\alpha h_i}\right) \cdot g^{\alpha h_i}$$

Reduction of Schnorr in AGM + ROM

Algebraic Adversary

$R^* = g^{u_0} Z^{u_1} \prod_{i=1}^{q_s} (R_i)^{u_{2,i}}$

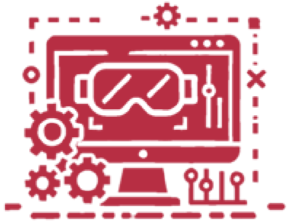
$\vec{u} = \left(u_0, u_1, u_{2,1} \dots, u_{2,q}\right)$

Request:

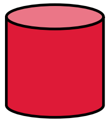
1 $H(m^*, R^*),\ \ \vec{u}$

2 h^*

Simulator



Hash list



Random Oracle

Set $H(m^*, R^*) = h^* \in \mathbb{Z}_p^*$

$R^* = g^{y^*} \cdot Z^{-H(m^*,R^*)}$

3 $\sigma_{m^*} = (R^*, y^*),\ \ \vec{u}$

Check if:

$g^{y^*} \stackrel{?}{=} R^* \cdot Z^{H(m^*,R^*)}$

Reduction of Schnorr in AGM + ROM

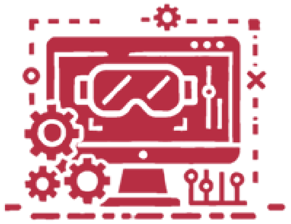
Algebraic Adversary

g^{α}

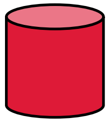
$R_i = g^{s_i}(g^{\alpha})^{-h_i}$

Request:

Simulator



Hash list



$$R^* = g^{u_0} Z^{u_1} \prod_{i=1}^{q_s} (R_i)^{u_{2,i}}$$

$$\vec{u} = (u_0, u_1, u_{2,1} \dots, u_{2,q})$$

1 $H(m^*, R^*), \ \vec{u}$

Random Oracle

2 h^*

Set $H(m^*, R^*) = h^* \in \mathbb{Z}_p^*$

$$R^* = g^{y^*} \cdot Z^{-H(m^*, R^*)}$$

3 $\sigma_{m^*} = (R^*, y^*), \ \vec{u}$

Check if:

$$g^{y^*} \stackrel{?}{=} R^* \cdot Z^{H(m^*, R^*)}$$

$$y^* - \alpha h^* = u_0 + \alpha u_1 = \sum_{i=1}^{q_s} (s_i - \alpha h_i)\, u_{2,i}$$

$$\alpha = \frac{y^* - u_0 - \sum_{i=1}^{q_s} s_i u_{2,i}}{u_1 + h^* - \sum_{i=1}^{q_s} h_i u_{2,i}}$$

The denominator is non-zero

Schnorr-like IBS

Users' perspective

Parameters

$$g,\ Z = g^z, H_1(\cdot,\cdot) \to h_{ID_i} \in \mathbb{Z}_p^*\ ,$$

$$H_2(\cdot,\cdot,\cdot) \to h_{ID_i,m_i} \in \mathbb{Z}_p^*$$

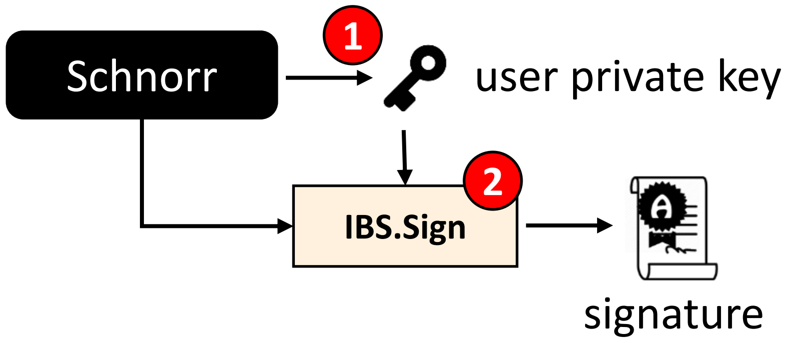
User ID_i private key $d_{ID_i}\begin{cases} R_i = g^{r_i} \\ y_i = r_i + z \cdot H_1(ID_i, R_i) \end{cases}$

Signature on (ID_i, m_i) $\sigma_{ID_i,m_i}\begin{cases} R_i \\ A_i = g^{a_i} \\ s_i = a_i + y_i \cdot H_2(ID_i, m_i, A_i) \end{cases}$

Check if: or

$$g^{s_i} \stackrel{?}{=} A_i\left(R_i \cdot Z^{H_1(ID_i,R_i)}\right)^{H_2(ID_i,m_i,A_i)}$$

Construction in High Level



- Concatenation of Schnorr's signatures
- User private key: $Schorr.Sign(ID, z) \to d_{ID}$
- Signature: $Schorr.Sign\ (m, d_{ID}) \to \sigma_{ID,m}$

Schnorr-like IBS Simulations (DL Problem)

Users' perspective

Parameters

$g,\ Z = g^z, H_1(\cdot,\cdot) \to h_{ID_i} \in \mathbb{Z}_p^*$,

$H_2(\cdot,\cdot,\cdot) \to h_{ID_i,m_i} \in \mathbb{Z}_p^*$

User ID_i private key $d_{ID_i} \begin{cases} R_i = g^{r_i} \\ y_i = r_i + z \cdot H_1(ID_i, R_i) \end{cases}$

Signature on (ID_i, m_i) $\sigma_{ID_i,m_i} \begin{cases} R_i \\ A_i = g^{a_i} \\ s_i = a_i + y_i \cdot H_2(ID_i, m_i, A_i) \end{cases}$

Check if: ✓ or ⊘

$$g^{s_i} \overset{?}{=} A_i\left(R_i \cdot Z^{H_1(ID_i,R_i)}\right)^{H_2(ID_i,m_i,A_i)}$$

Simulator's perspective

$\begin{cases} g \\ g^{\alpha} \end{cases}$

Simulator aborts in query phase

$Z = g^{\alpha}$

$A_i = g^{a_i'}\left(g^{r'} g^{\alpha h_{ID'}}\right)^{-h_{ID',m_i}}$

Suppose $\mathcal{O}_S(ID', m_i)$ was queried.

Private key $\mathcal{O}_E(ID') \nrightarrow d_{ID'}$ is not simulatable

Simulator cannot solve for DL problem

$Z = g^{\alpha}$

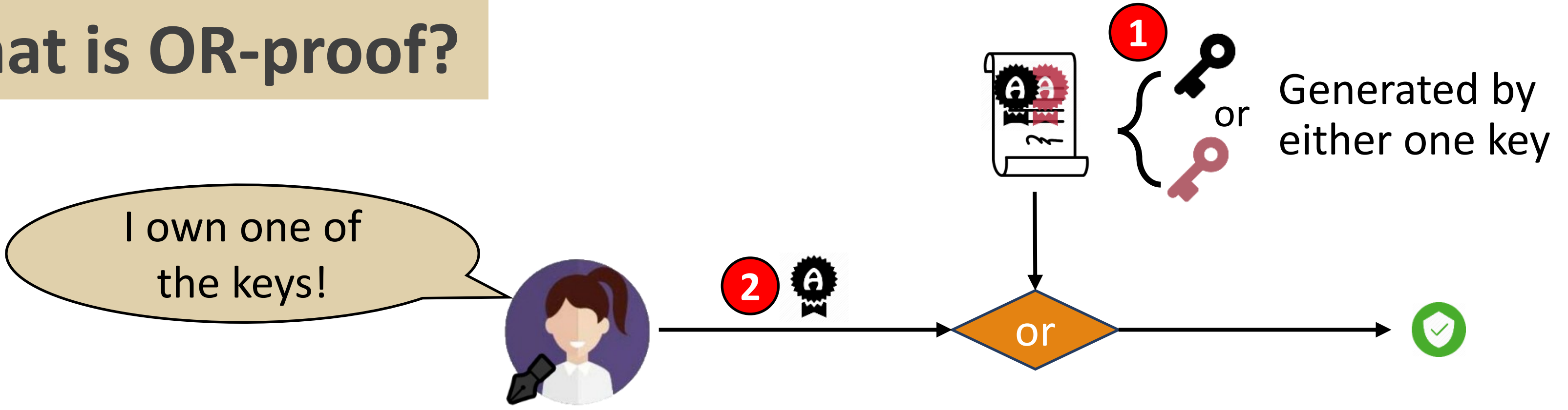
$R' = g^{r'}(g^{\alpha})^{-h_{ID'}}$

Forgery $\sigma_{ID^*,m^*} = (R^*, A^*, s^*)$ is non-reducible as g^{α} vanishes by setting $R^* = R'$.

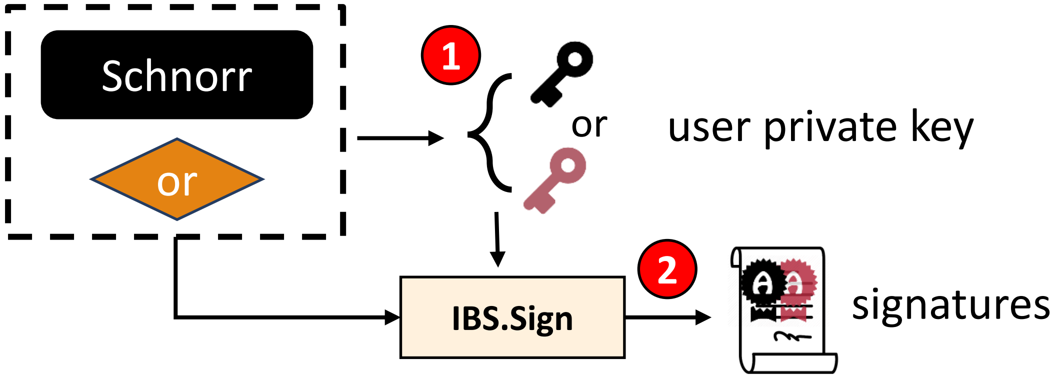
In AGM, representation $\vec{u}$ cannot help.

Solution: OR-Proof Technique

What is OR-proof?



We obtain a new pairing-free IBS scheme…



I believe now.
Skip please

Next, the proof…

Ideal Security?

iC² Institute of Cybersecurity and Cryptology

UNIVERSITY OF WOLLONGONG AUSTRALIA

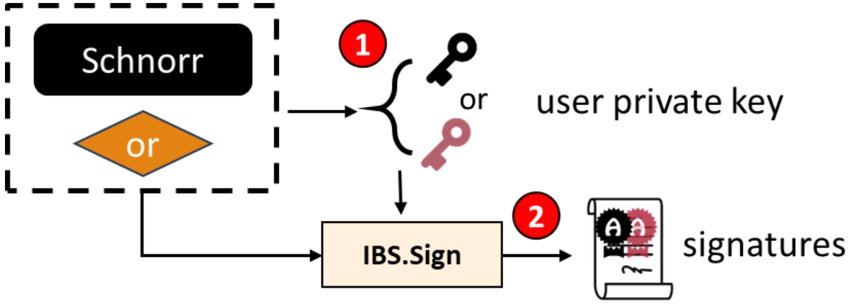
The Proposed Pairing-free IBS

Parameters

$g,\ Z = g^z, H_1(\cdot,\cdot,\cdot) \rightarrow h_{ID_i} \in \mathbb{Z}_p^*$,

$H_2(\cdot,\cdot,\cdot,\cdot) \rightarrow h_{ID_i,m_i} \in \mathbb{Z}_p^*$

Users' perspective



User ID_i private key d_{ID_i}

$$d_{ID_i}\begin{cases} R_0 = g^{r_0}, \quad R_1 = g^{r_1}, \\ b \in \{0,1\}, \quad y = r_b + z \cdot H_1(ID, R_0, R_1) \end{cases}$$

Signature on (ID_i, m_i) σ_{ID_i,m_i}

$$\sigma_{ID_i,m_i}\begin{cases} R_0, & R_1, \\ A_0 = g^{a'_0}\left(R_0 \cdot Z^{h_{ID}}\right)^{b\cdot(-c_{1-b})}, & A_1 = g^{a'_1}\left(R_1 \cdot Z^{h_{ID}}\right)^{(1-b)\cdot(-c_{1-b})}, \\ s_{1-b} = a'_{1-b}, & s_b = a'_b + y \cdot c_b, \\ c_{1-b}, & c_b = H_2(ID, m, A_0, A_1) - c_{1-b} \end{cases}$$

erm......

Check if: or

For $i \in \{0,1\}$, 1 $g^{s_i} \stackrel{?}{=} A_i\left(R_i \cdot Z^{H_1(ID_i,R_0,R_1)}\right)^{H_2(ID,m,A_0,A_1)}$

2 $H_2(ID, m, A_0, A_1) \stackrel{?}{=} c_0 + c_1$

Next, the proof...

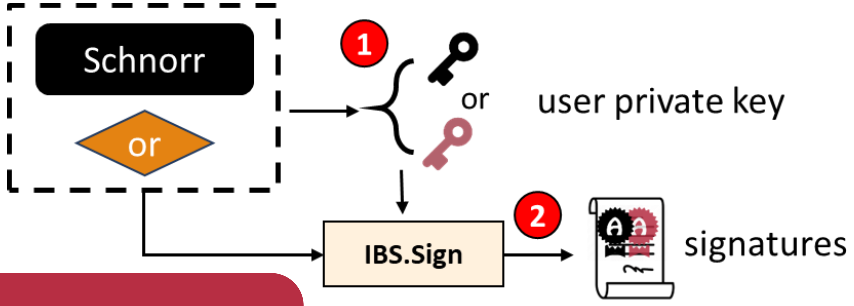
Ideal Security?

The Proposed Pairing-free IBS

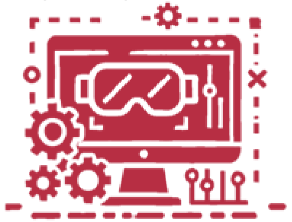
Parameters

g, $Z = g^z$ $H_1(\cdot,\cdot,\cdot) \rightarrow h_{ID_i} \in \mathbb{Z}_p^*$,

$H_2(\cdot,\cdot,\cdot,\cdot) \rightarrow h_{ID_i,m_i} \in \mathbb{Z}_p^*$



Simulator's perspective



Random Oracle

Hash list

User ID_i private key d_{ID_i} $\begin{cases} R_0 = g^{r_0}, \\ b = 1 \end{cases}$

$R_1 = g^{r_1}(Z)^{-H_1(ID,R_0,R_1)}$,

$y = r_1$

Signature on (ID_i, m_i) σ_{ID_i,m_i}

$$\begin{cases} R_0, & R_1, \\ A_0 = g^{a_0}\left(g^{r_1} Z^{h_{ID}}\right)^{-h_{ID,m}}, & A_1 = g^{a_1}, \\ s_0 = a_0 & s_b = a_b' + y \cdot c_b, \\ c_{1-b}, & c_b = H_2(ID, m, A_0, A_1) - c_{1-b} \end{cases}$$

oh……

Check if: or

For $i \in \{0,1\}$, 1 $g^{s_i} \stackrel{?}{=} A_i\left(R_i \cdot Z^{H_1(ID_i,R_0,R_1)}\right)^{H_2(ID,m,A_0,A_1)}$

2 $H_2(ID, m, A_0, A_1) \stackrel{?}{=} c_0 + c_1$

Next, the proof…

Ideal Security?

iC² Institute of Cybersecurity and Cryptology

UNIVERSITY OF WOLLONGONG AUSTRALIA

Security Proof (in high level)

We propose a simulation…

- Simulate any user private key
- Reduce any forgery with ½ chance
- The concrete analysis is done in AGM + ROM

or → IBS.Sign → signatures

Oracles

1. Extraction Oracle $\mathcal{O}_E(ID') \rightarrow d_{ID'}$
2. Signing Oracle $\mathcal{O}_S(ID', m_j) \rightarrow \sigma_{ID',m_j}$

g^{α}

or

One key is embedded with problem instance

One key is simulatable

Algebraic Adversary $\mathcal{A}_{alg}$ → $ID^* = ID'$, m^*, σ_{ID^*,m^*}, $\vec{u}$ → **Simulator** →

½ chance

or

Solution $\alpha \in \mathbb{Z}_p^*$

We defer the full proof…

P

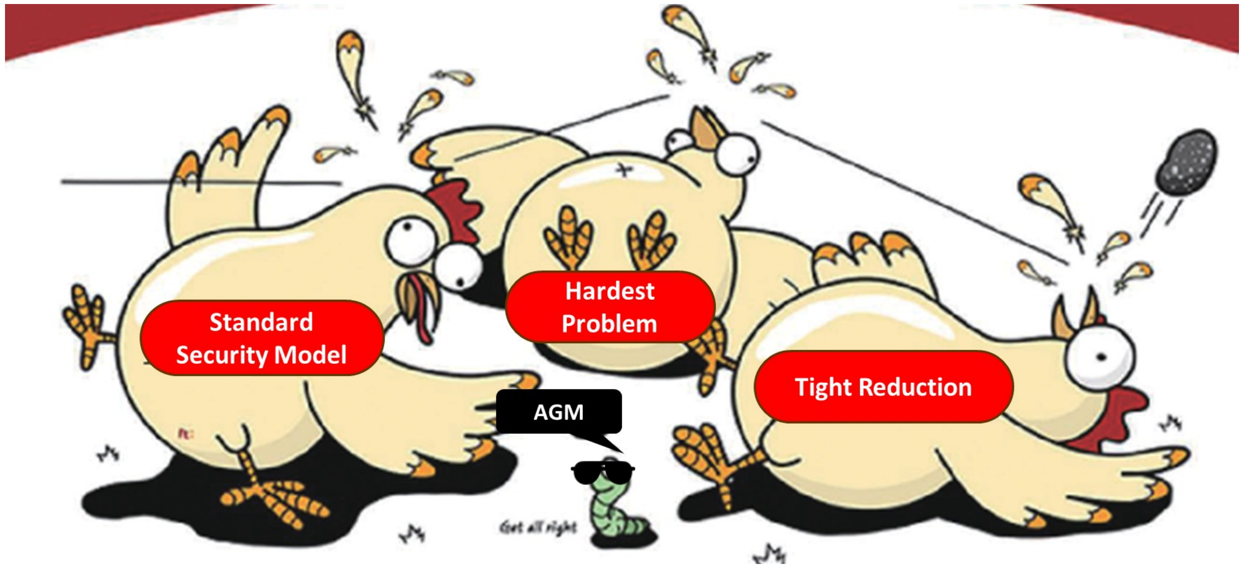
FOUR

04

Conclusion

Summary & Future Works

Summary



- Discussed challenge in Schnorr-like IBS
- A new pairing-free IBS scheme: Thanks to OR-proof technique
- Achieved "ideal security" in AGM + ROM
- Reduction loss is 2

Future Works

"Idealized" Slingshot

- Minimize the signature size, as our signature size: $4\ \mathbb{G} + 4\ \mathbb{Z}_p^*$?
- Can we omit ROM? Pairing-free in AGM only (under DL assumption + tight + standard security model)

The 29th Australasian Conference on Information Security and Privacy

iC²
Institute of Cybersecurity and Cryptology

UNIVERSITY OF WOLLONGONG AUSTRALIA

Thank You