途中計算情報の漏洩に対する認証鍵交換プロトコルの安全性考察

2011/9/16

米山 一樹

NTT Information Sharing Platform Laboratories

Aim of this work

- Security classification of previous (Diffie-Hellman type) AKE schemes in the **seCK model**

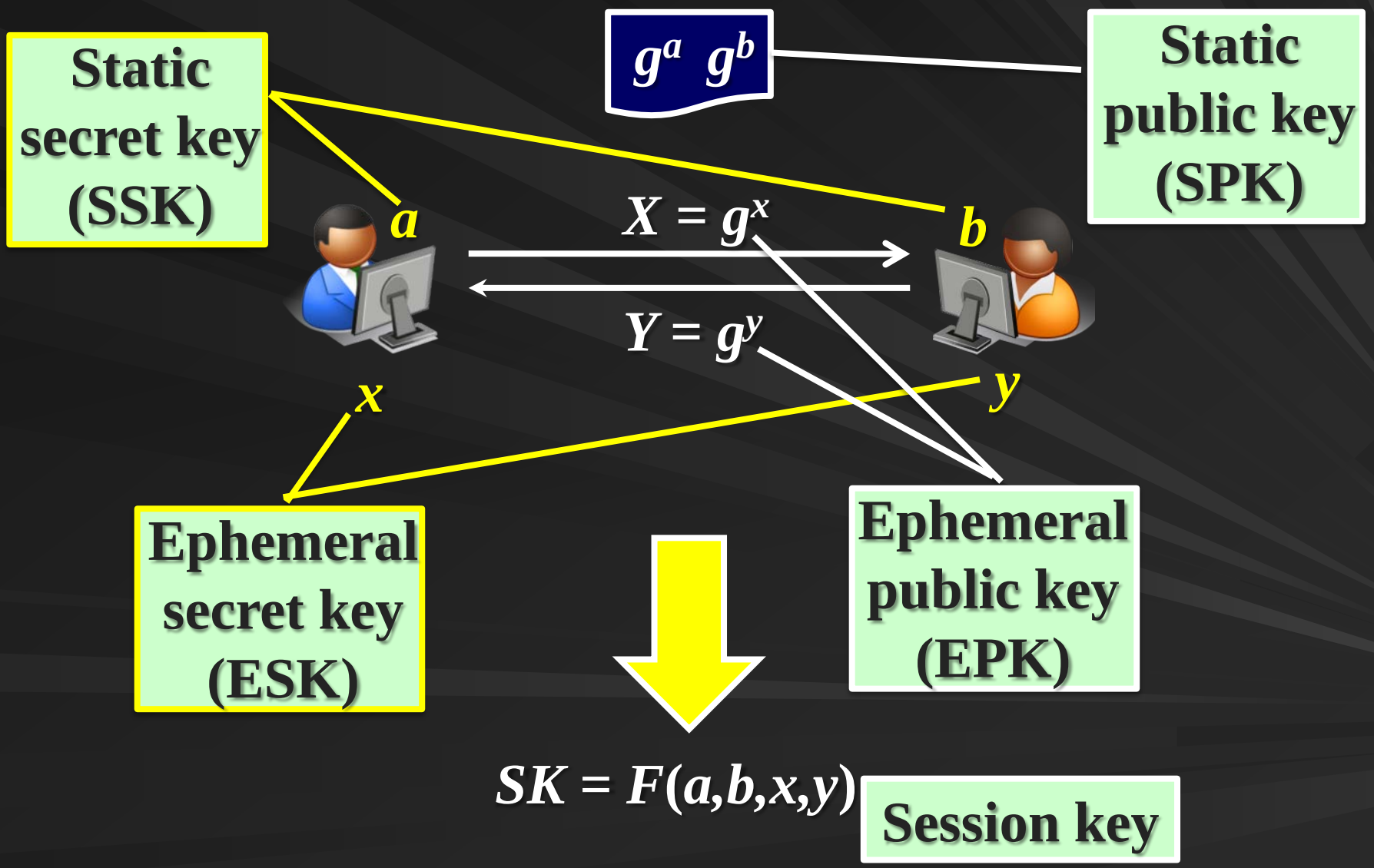
seCK model : security against leakage of intermediate computation results

- Security reconsideration of SMQV protocol
  - SMQV **was** proved to be secure in the seCK model.

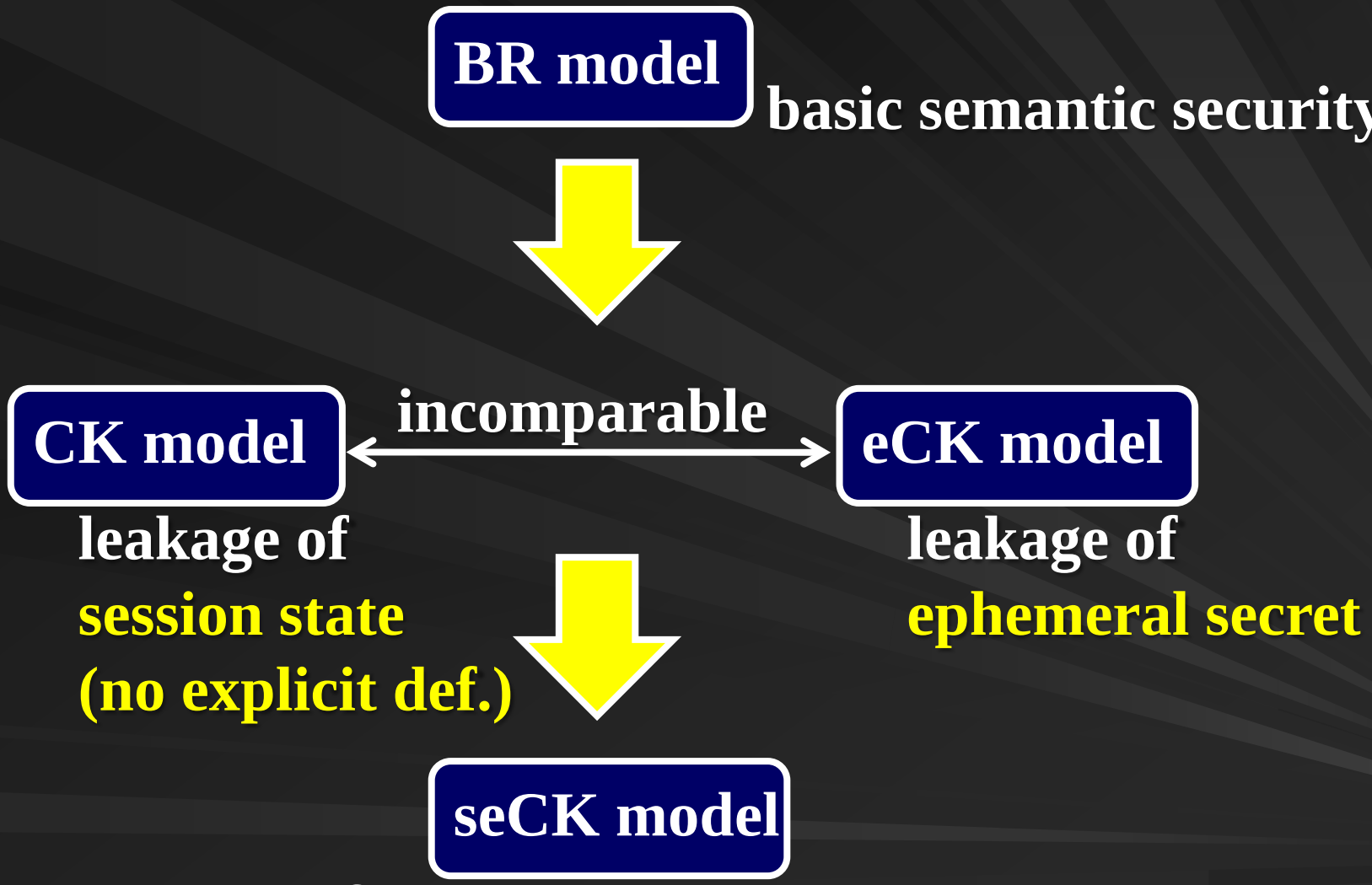
Outline

- Introduction
  - DH-type AKE
  - Security models
  - SMQV protocol
- SMQV revisited
  - Original proof strategy for ICR reveal
  - Proof error
- Classification

(DH-type) authenticated key exchange

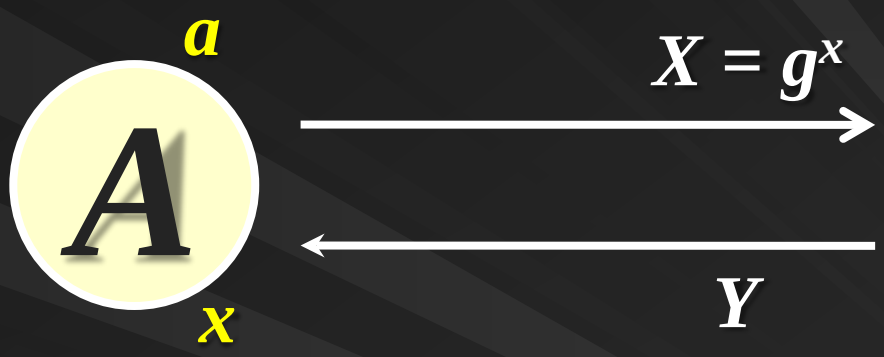


Security models for AKE



HMQV protocol

$K_A=g^a$ K_B



$D=H_1(X,B)$, $E=H_1(Y,A)$

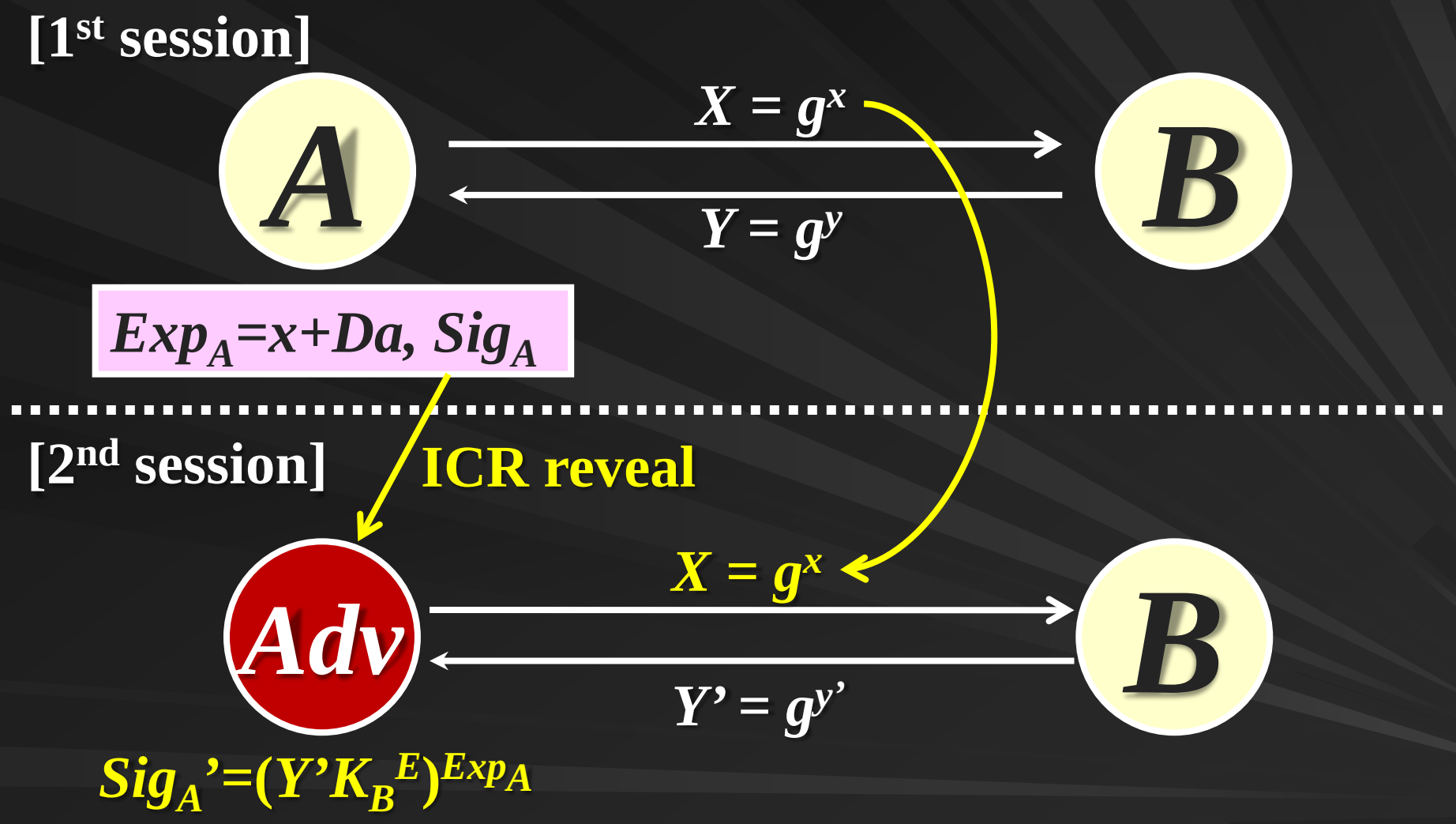
$Exp_A=x+Da$

$Sig_A=(YK_B^E)^{Exp_A}$

Inter. comp. results (ICR)

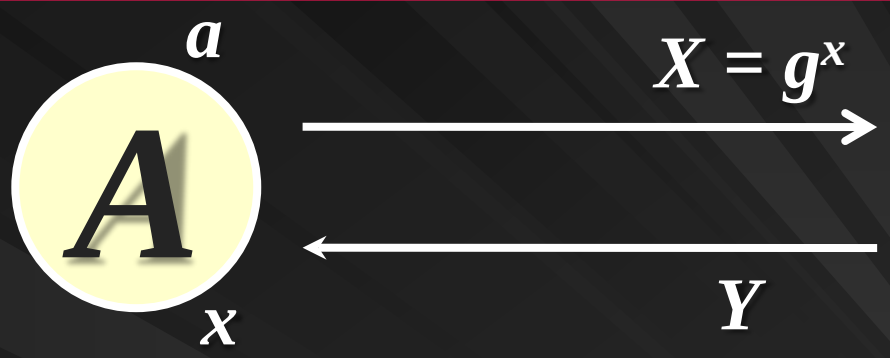
$SK=H_2(Sig_A)$

Attack (atk 1) to HMQV with ICR [SEVB10]



Adv **successfully impersonates** *A* **to** *B*

SMQV protocol (resilient to atk 1)



HMQV	SMQV
$D=H_1(X,B)$, $E=H_1(Y,A)$	$D=H_1(X,Y,A,B)$, $E=H_1(Y,X,A,B)$

binding *D* and *E* by EPK of peer

***Exp*$_A$ is not reusable since *D* is unique for a session.**

HMQV	SMQV
$Exp_A=x+Da$	$Exp_A=Dx+a$
$Sig_A=(YK_B^E)^{Exp_A}$	$Sig_A=(Y^EK_B)^{Exp_A}$

Our motivation

- seCK model is considerably strong model.
  - But, only SMQV is proved to be secure.

Is there seCK secure protocol other than SMQV?

Are there explicit vulnerabilities in ‘insecure’ schemes?

This work

- Security classification of previous (Diffie-Hellman type) AKE in the **seCK model**.
  - **SMQV** was stated to be secure... but, **proof is flawed**!
  - There is **no** known secure scheme!

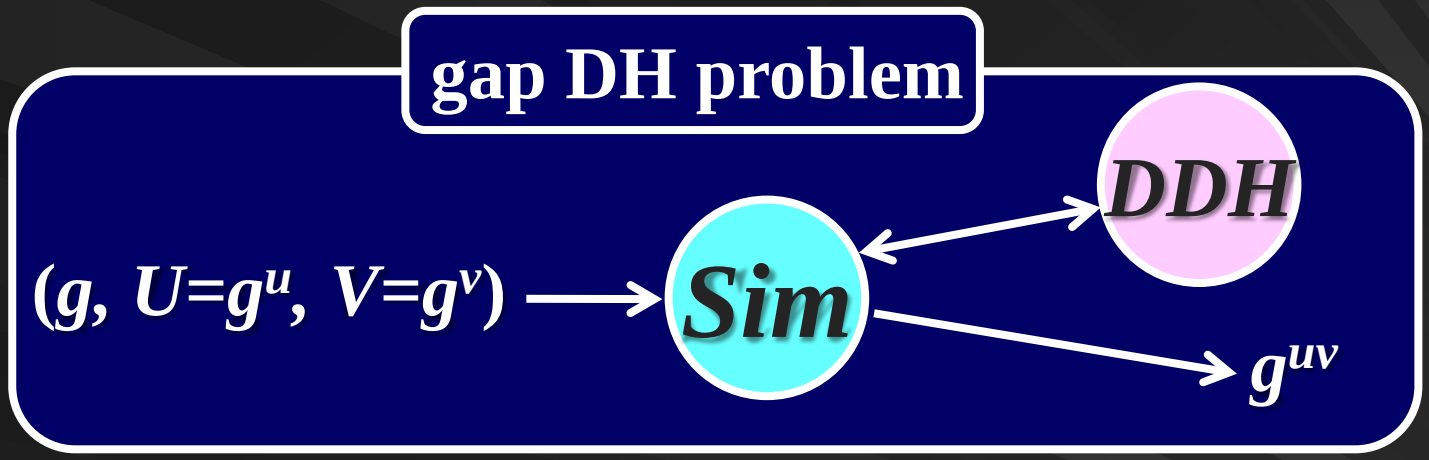
Secure	Hard to prove	Insecure	Total break
none	SMQV FHMQV NAXOS	MQV HMQV Kim-Fujioka-Ustaoglu KEA+	CMQV UP Fujioka-Suzuki Okamoto Moriyama-Okamoto

Outline

- Introduction
  - DH-type AKE
  - Security models
- SMQV revisited
  - Original proof strategy for ICR reveal
  - Proof error
- Classification

Strategy of original proof

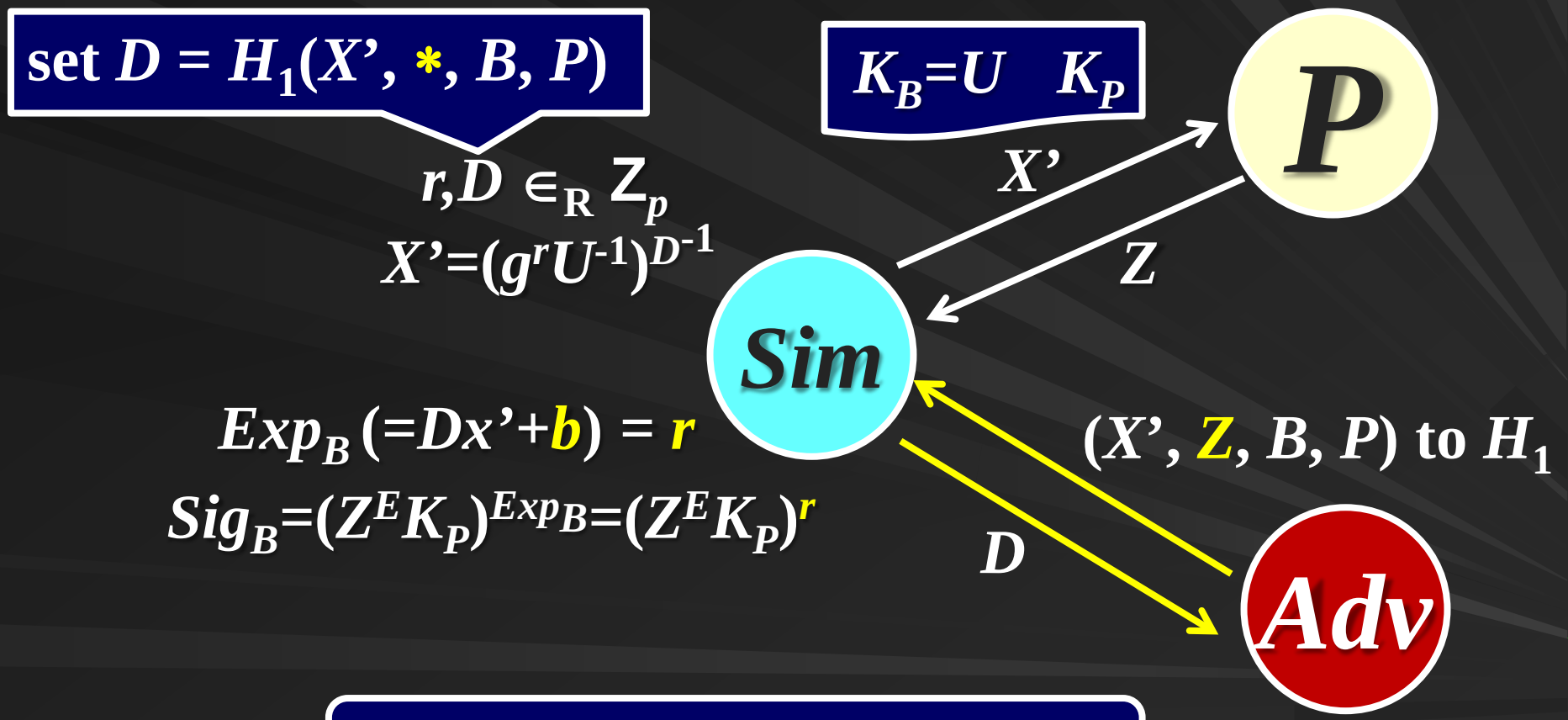
- Giving reduction to the **gap DH assumption** in the random oracle model.



- Most subtle point is to simulate the case (event E) that U and V are embedded to X and K_B.

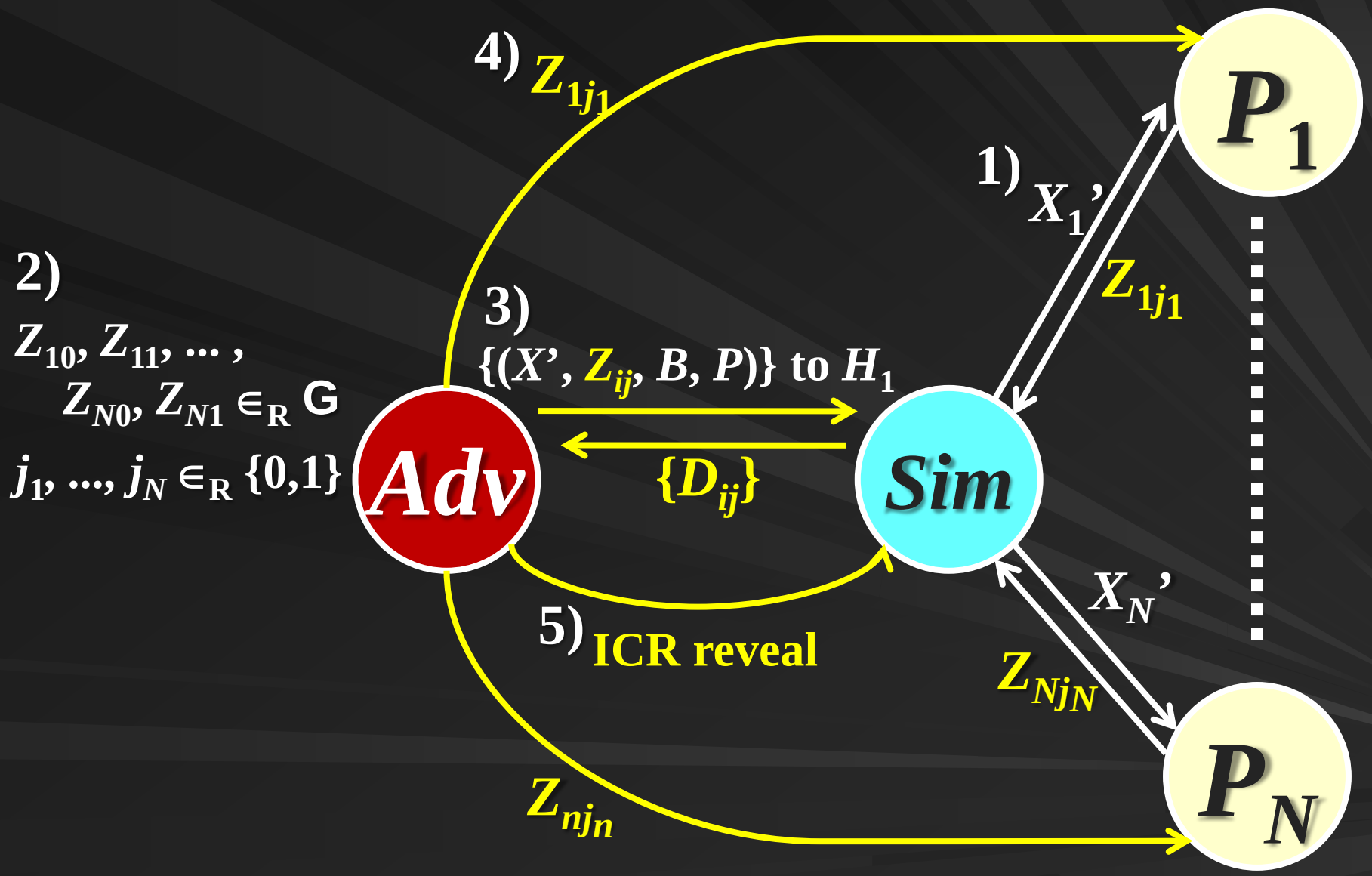
Simulation of event *E*

- ***Sim*** **must simulate ICR of *B* without knowing *b*.**



Seemingly it works correctly

An attack scenario (atk 2)



Possible strategy to atk 2

- ***Sim*** **must fix** D_i **for** $X_i = (g^{r_i}U^{-1})^{D_i^{-1}}$ **before knowing** Z_{ij_i}**.**
  - **But,** ***Sim*** **cannot know whether** D_i **should be set as** $H_1(X', Z_{i0}, B, P)$ **or** $H_1(X', Z_{i1}, B, P)$**.**
  - **So,** ***Sim*** **must guess** j_i **for all** ***i.***

- **If one of** N **guesses is failed, the simulation is failed.**
  - **Pr[*Sim* succeeds]** $\leq 1/2^N$

negligible!

SMQV is not proved to be secure in the seCK model

Outline

- Introduction
  - DH-type AKE
  - Security models
  - SMQV protocol
- SMQV revisited
  - Original proof strategy for ICR reveal
  - Proof error
- Classification

Classifying security levels

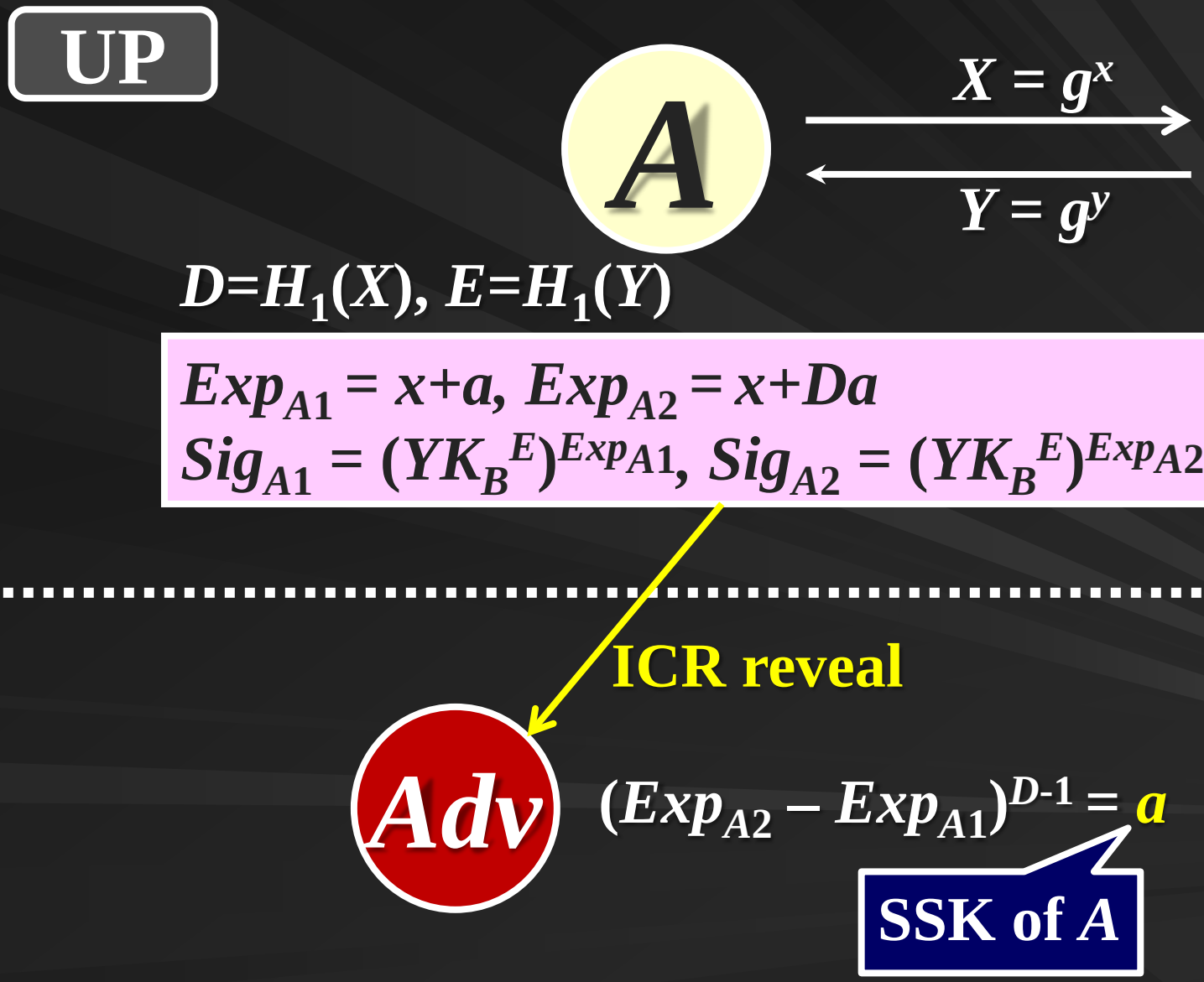
- ‘Secure’: **provable** in the seCK model
- ‘Hard to prove’: way to prove is **unknown** as SMQV (**no explicit attack**)
- ‘Insecure’: existence of explicit attack to **break session key security**
- ‘Total break’: existence of explicit attack to **reveal SSK**

Classification result

- There is no ‘secure’ scheme.
- Some schemes fall into ‘total break’.

Secure	Hard to prove	Insecure	Total break
none	SMQV FHMQV NAXOS	MQV HMQV Kim-Fujioka-Ustaoglu KEA+	CMQV UP Fujioka-Suzuki Okamoto Moriyama-Okamoto

Revealing SSK of UP with ICR



Conclusion

- Unfortunately, we have **no (DH-type) protocol** which is secure in the seCK model.
  - We guess that **two-move and implicitly authenticated** protocol is hard or impossible to prove.
  - **Explicit authenticated or three-move** protocol may be possible.

- Be careful with **multiple sessions**.
  - Frequently, adversaries can do complex attack scenarios with information of multiple sessions.

フォーマルメソッドの導入に向けて

- 今回は**人力**で証明ミスと攻撃を発見した.
  - **自動化**できたら嬉しい（FAIS研究会的には）

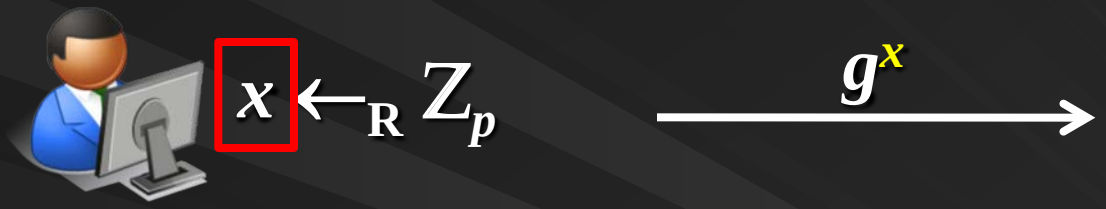
- １つの方向性：**Scyther**の利用
  - Cas Cremers作成の解析フレームワーク・ツール
  - ICRやESKの漏洩をモデル化可能
  - ∃記号モデルでの攻撃 ⇒ ∃計算論モデルでの攻撃

Thank you!

Ephemeral Key Leakage

- Ephemeral secret key (ESK)
  - Temporary and session-specific randomness

    e.g.) Diffie-Hellman (DH) key exchange



- Reveal of ephemeral secret key

poor pseudo-random generator

physical attacks (e.g., cold boot attacks)

Two implementation modes