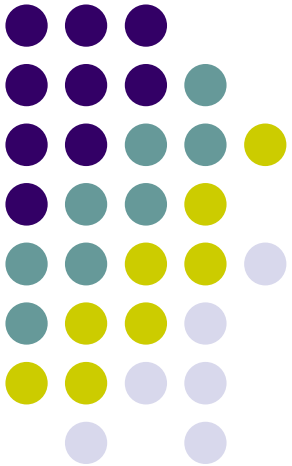
Public Key Encryption and Security against CCA

Takahiro Matsuda (AIST)

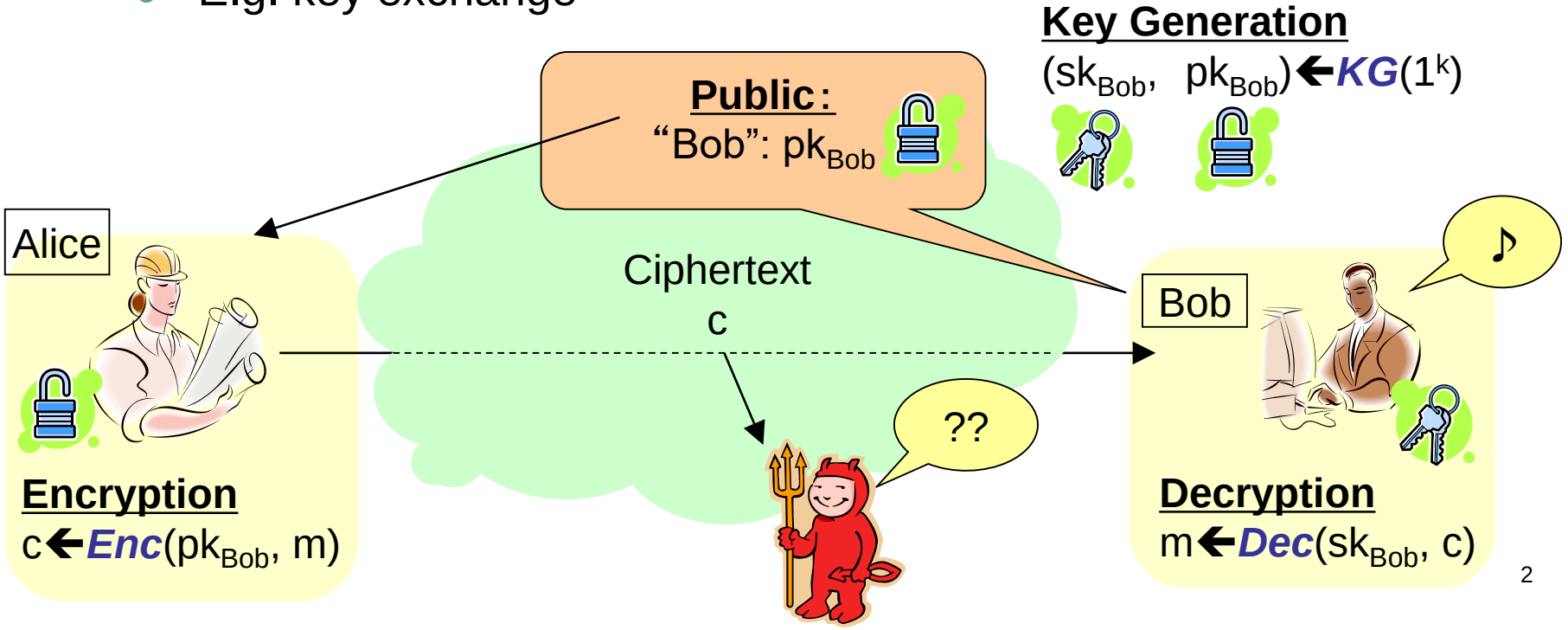


※ This slide is a slightly revised version of
the slide file used in the tutorial talk in INDOCRYPT 2018.
(References can be found in the end)

Public Key Encryption (PKE)

[DH76]

- A fundamental tool for secure communication
  - w/o shared secret information in advance
- Used as building blocks for many higher-level protocols
  - E.g. key exchange



Syntax of PKE

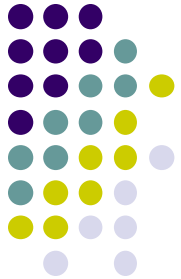
- PKE consists of 3 algorithms
  - ***KG*** ➔ (*pk*, *sk*)
  - ***Enc***(*pk*, *m*) ➔ *C*
  - ***Dec***(*sk*, *C*) ➔ *m* / ⊥ (invalid symbol)

Correctness:

∀(*pk, sk*)🡐 ***KG***:

It holds that ***Dec***(*sk*, ***Enc***(*pk*, *m*)) = *m*

Basic Security Notions of PKE

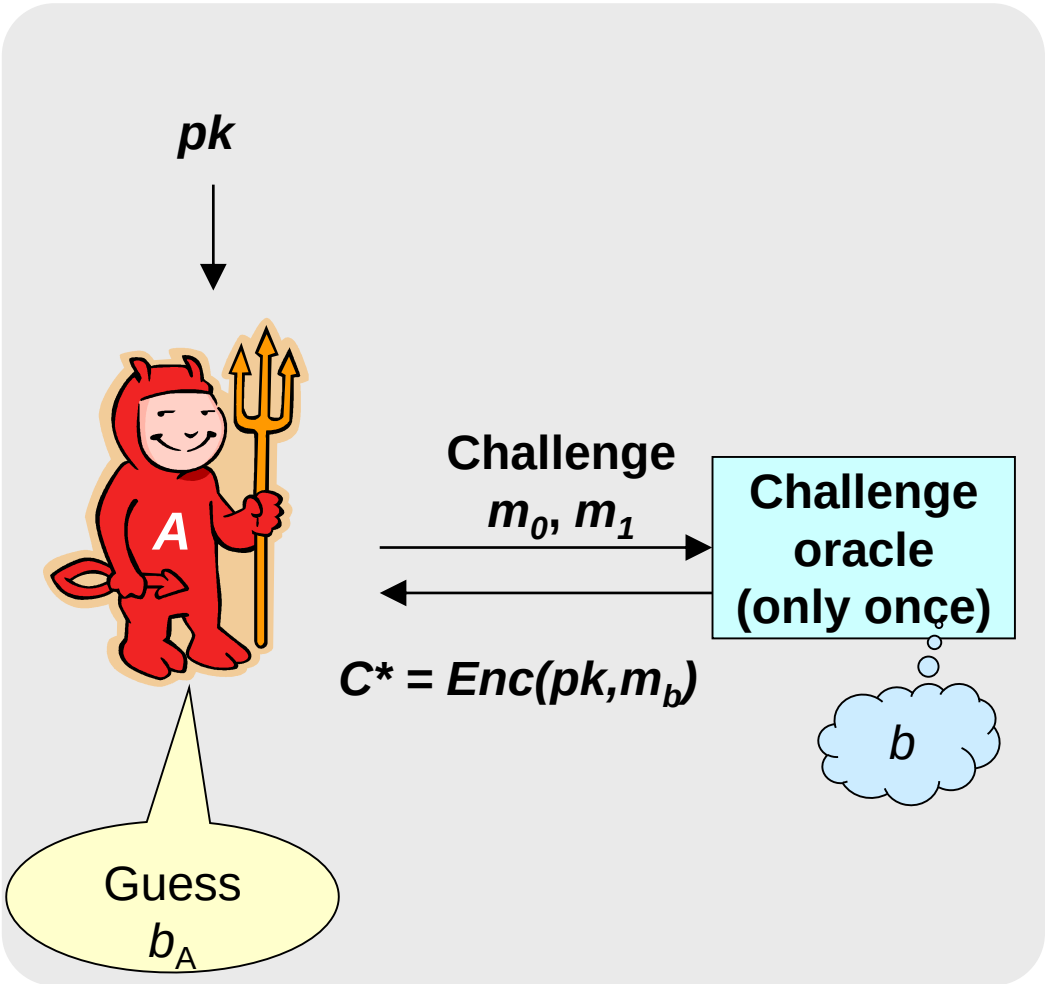


- Indistinguishability under Chosen Plaintext Attack (IND-CPA-security, or just CPA-security) [GM84]
  - No information leaks from a ciphertext
  - Captures security against passive adversaries

- IND under Chosen Ciphertext Attack (IND-CCA2-security, or just CCA-security) [NY90], [RS91]
  - No information leaks even if an “active” adversary can make “decryption queries”

IND-CPA Security

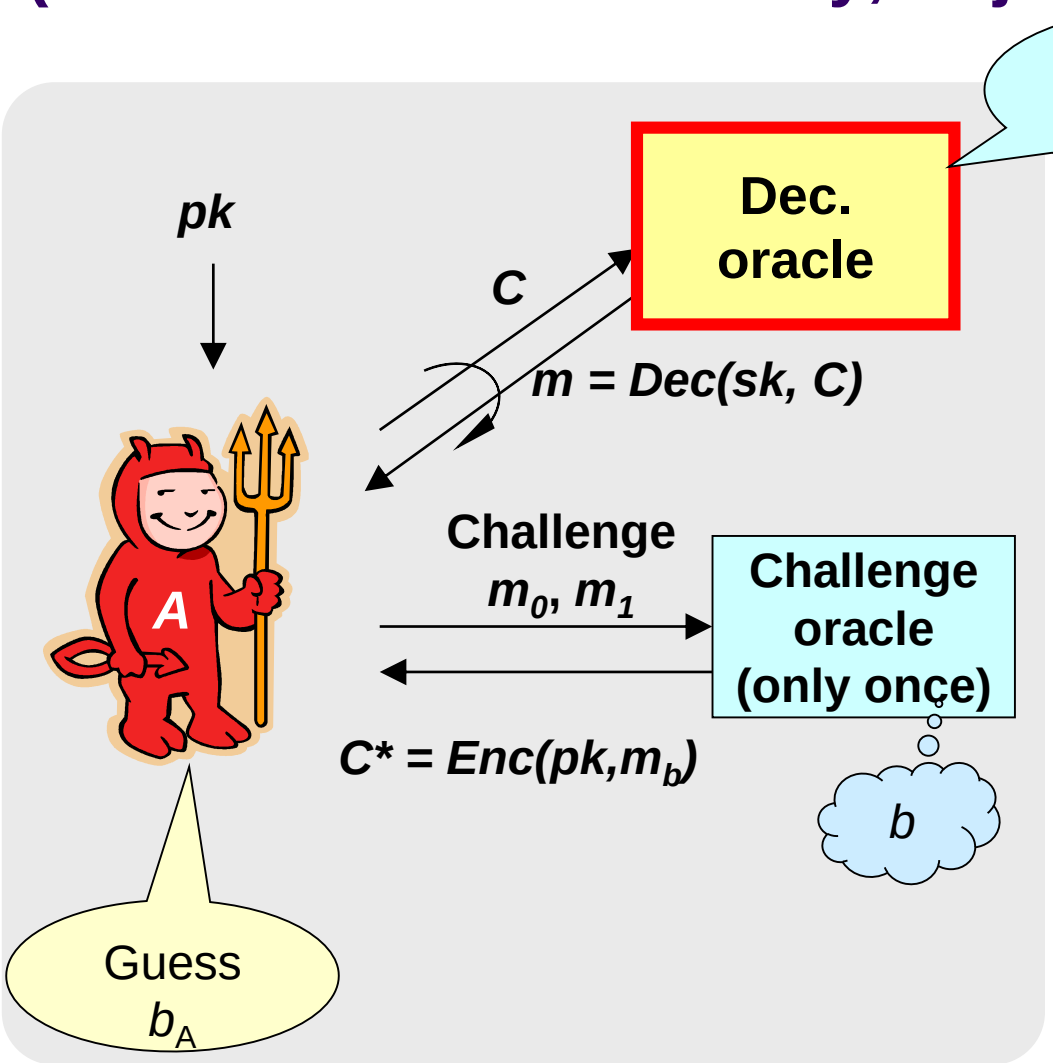
(Or just CPA security)



•PKE is **IND-CPA** secure if for any PPT adversary ***A,*** $|\Pr[b_A = b] - 1/2| = neg.$

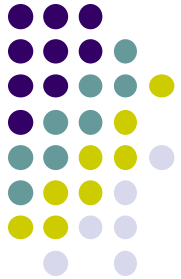
IND-CCA Security

(a.k.a. IND-CCA2 Security, or just CCA security)



•PKE is **IND-CCA** secure if for any PPT adversary ***A,*** $|\Pr[b_A = b] - 1/2| = \text{neg}$.

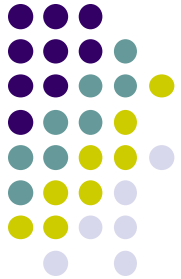
Why CCA Security so Important?



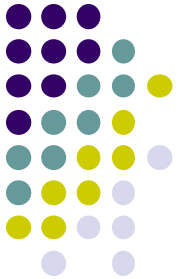
- CCA itself might not occur in practice, however…
- **CCA-like attacks occur in practice**
  - Bleichenbacher's attack on PKCS#1 v.1.5 [Ble98,BFK+12]
- **CCA security implies many important & useful notions**
  - Non-malleability [DDN91,BS99,BS06,PSV07]
  - Universal composability [Can01,CKN03]
  - …
- **Easy to formulate and work with**
  - Avoid complicated and error-prone security proofs

➔ Nowadays, de-facto standard security for PKE!!

This Tutorial



- Part 1:
  - Review of classical techniques for constructing CCA secure PKE and their security proofs
- Part 2:
  - Brief survey of recent topics on CCA secure PKE
    - and open problems



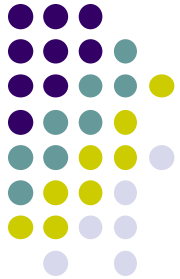
Part 1:
Classical Constructions of CCA Secure PKE

Part 1 Outline

- Naor-Yung construction
- KEM & Hybrid Encryption
- Hash proof systems
- Fujisaki-Okamoto construction

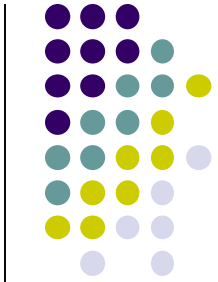
- We will see how each construction looks like, and how its security is proved
  - If you are a starter of public-key cryptography research, I highly recommend writing down every detail of the formal proofs of these constructions by yourself !!

Naor-Yung @ STOC'90



- Introduced the notion of CCA security
- Original paper [NY90] showed the first **IND-CCA1** secure construction
- Later several works extend it to achieve full **IND-CCA2**
  - Dolev-Dwork-Naor @ STOC'91 [DDN91]
  - Sahai @ STOC'99 [Sah99]
  - De Santis et al. @ C'01 [DDOPS'01]
  - Lindell @ EC'03 [Lin03]

Naor-Yung Construction Overview



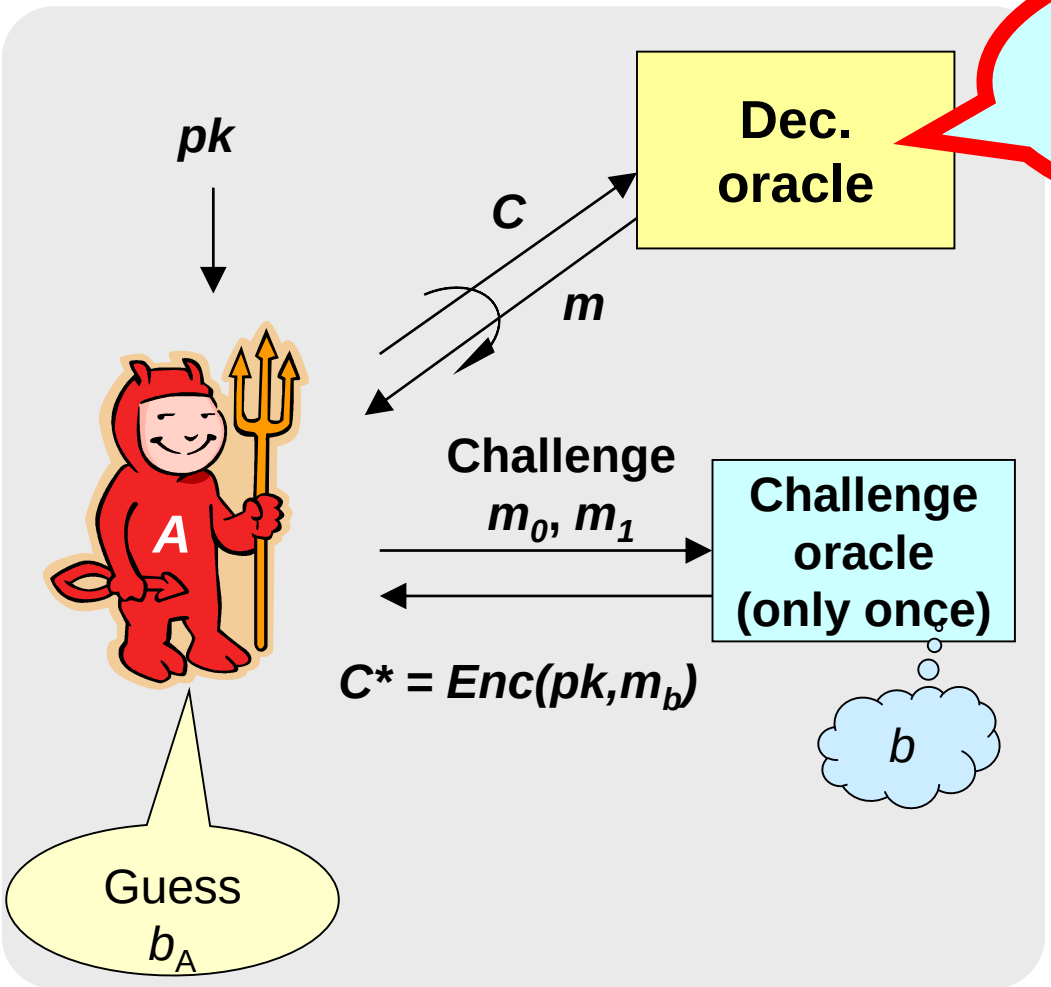
- [NY90]

CPA PKE + **NIZK proof** (in CRS model) → **CCA1 PKE**

- [Sah99, Lin03]

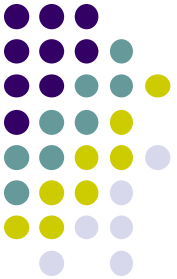
CPA PKE + **NIZK proof** (in CRS model) **with simulation soundness** → **CCA2 PKE**

IND-CCA1 Security



•PKE is **IND-CCA1** secure if for any PPT adversary ***A,*** $|\Pr[b_A = b] - 1/2| = neg$.

Preliminary for NIZK Proof: NP Language



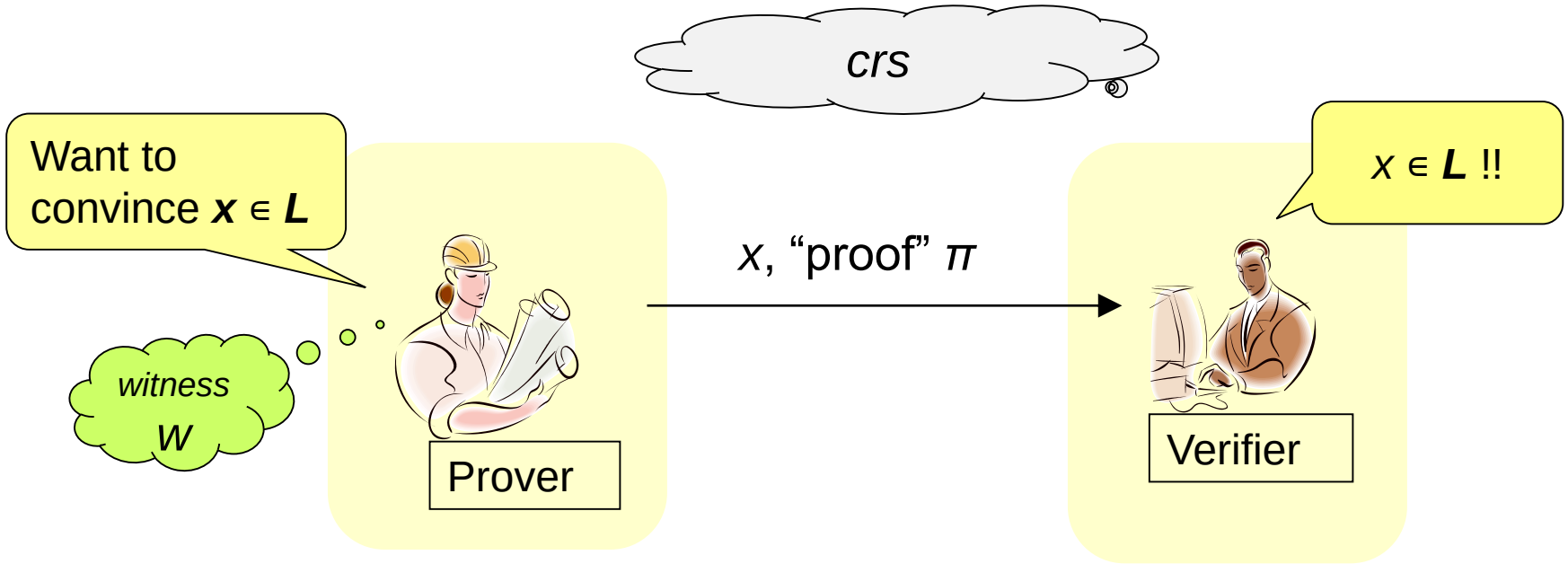
- Informal definition:
  - ***L*** is an **NP** language
    ⟷ Given a witness *w* for the fact of $x \in \boldsymbol{L}$,
    one can check it in poly-time in |*x*|

- A bit for formal definition:
  - ***L*** is an **NP** language
    ⟷ ∃ poly *p*, *q* and PPT algorithm $\boldsymbol{R_L}$:
    - ∀(*x*, *w*): $\boldsymbol{R_L}(x, w)$ runs in at most *p*(|x|) steps
    - $\forall x \in \boldsymbol{L},\ \exists\ w \in \{0,1\}^{q(|x|)} : \boldsymbol{R_L}(x, w) = 1$
    - $\forall x \notin \boldsymbol{L},\ \forall\ w \in \{0,1\}^{q(|x|)} : \boldsymbol{R_L}(x, w) = 0$

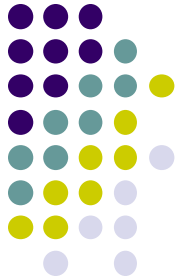
Non-interactive Proof System

in the common reference string (CRS) model

- Prover who knows a witness *w* for $x \in \boldsymbol{L}$, can convince Verifier of the validity of $x \in \boldsymbol{L}$, by just one message
- CRS model allows system-wide public parameter
  - Necessary to consider zero-knowledge property for NP



Non-interactive Proof System for NP Language L (in the CRS Model)



- Algorithms:

CRS generation	crs	$\leftarrow$ ***CRSGen***(1^k)
Prover's algorithm	π	$\leftarrow$ ***Prove***(crs, x, w)
Verifier's algorithm	1 or 0	$\leftarrow$ ***Ver***(crs, x, π)

$(x, w) \in R_L$

Correctness: $\forall (x, w) \in \boldsymbol{R_L}, \forall crs \leftarrow$ **CRSG**$(1^{|x|})$:
It holds that ***Ver***$(crs, x,$ ***Prove***$(crs, x, w)) = 1$

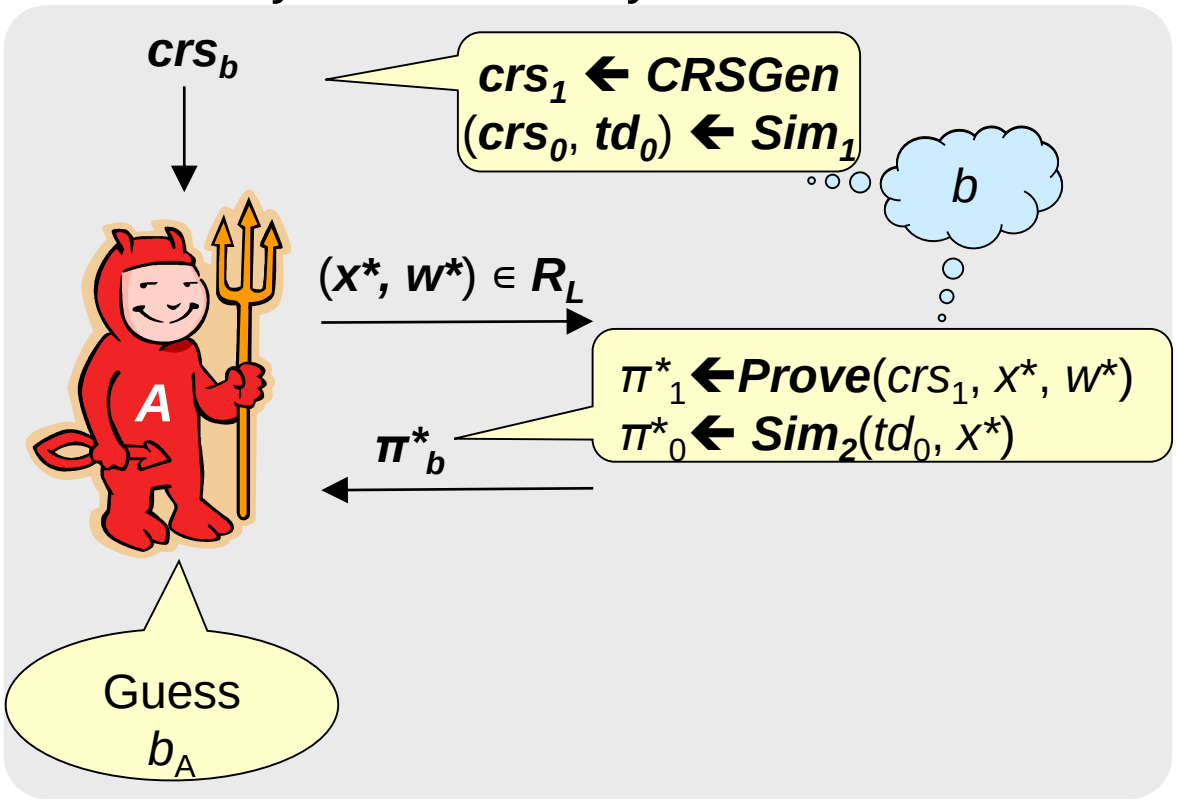
- For defining "Zero-knowledge property", it is convenient to consider the following algorithms as part of the syntax

Simulation of CRS	(crs, td)	$\leftarrow$ $\boldsymbol{Sim_1}(1^k)$
Simulation of Proof	π	$\leftarrow$ $\boldsymbol{Sim_2}(td, x)$

A "fake" proof can be generated even for a "false" statement $x \notin \boldsymbol{L}$

Security Requirements of NIZK Proof System (1/2)

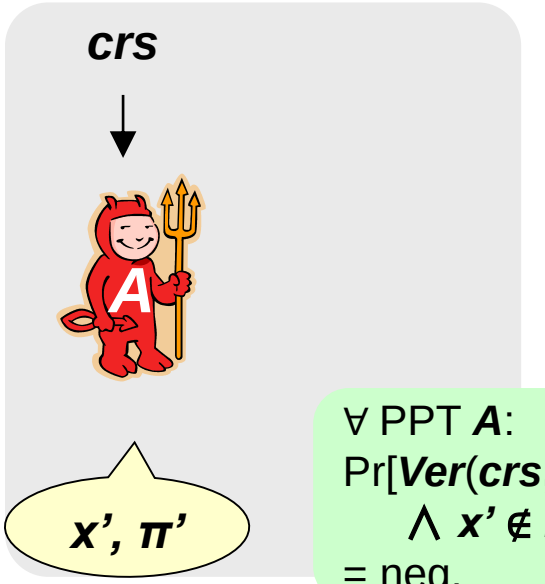
- **Zero Knowledge (ZK) Property**
  - Proof π does not leak information of witness w beyond the validity of a statement x



•For any PPT adversary ***A,*** $|\Pr[b_A = b] - 1/2| = neg.$

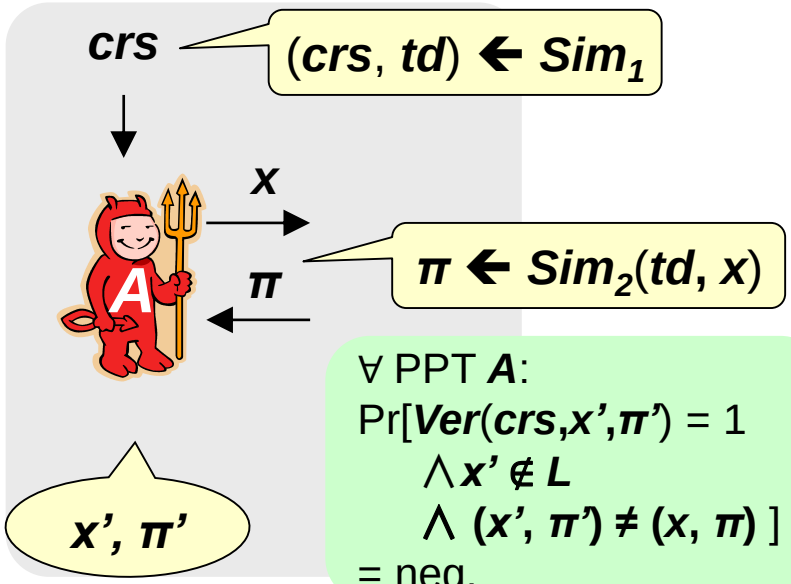
Security Requirements of NIZK Proof System (2/2)

- **Soundness**
  - Adversary can't find a valid proof for a false statement, i.e. (x', π') s.t.
    (1) ***Ver***(crs, x', π') = 1
    (2) $x' \notin \boldsymbol{L}$



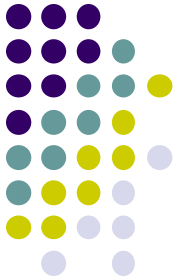
∀ PPT ***A***:
Pr[***Ver***(***crs***,***x'***,***π'***) = 1
∧ ***x'*** ∉ ***L***]
= neg.

- **(One-time) Simulation Soundness**
  - Adversary can't find a proof for a false statement, ***even under simulated CRS and after observing one simulated proof***



∀ PPT ***A***:
Pr[***Ver***(***crs***,***x'***,***π'***) = 1
∧***x'*** ∉ ***L***
∧ (***x'***, ***π'***) ≠ (***x***, ***π***)]
= neg.

Naor-Yung Construction



- Building Blocks:
  - PKE $\boldsymbol{\Pi_1} = (\boldsymbol{KG_1}, \boldsymbol{Enc_1}, \boldsymbol{Dec_1})$
  - PKE $\boldsymbol{\Pi_2} = (\boldsymbol{KG_2}, \boldsymbol{Enc_2}, \boldsymbol{Dec_2})$
  - NIZK $\boldsymbol{P} = (\boldsymbol{CRSG}, \boldsymbol{Prove}, \boldsymbol{Ver}, \boldsymbol{Sim_1}, \boldsymbol{Sim_2})$ for the following NP language $\boldsymbol{L_{eq}}$

Note: Π_1 and Π_2 can be the same PKE

$$\boldsymbol{L_{eq}} = \{ (pk_1, pk_2, c_1, c_2) \mid \exists (m, r_1, r_2) \text{ s.t. } c_i = \boldsymbol{Enc_i}(pk_i, m; r_i)_{(i = 1,2)} \}$$

Guarantees c_1 and c_2 encrypt the same m

- NY construction: $\boldsymbol{\Pi_{NY}} = (\boldsymbol{KG_{NY}}, \boldsymbol{Enc_{NY}}, \boldsymbol{Dec_{NY}})$

- $\boldsymbol{KG_{NY}}$:
  1. $(pk_i, sk_i) \leftarrow \boldsymbol{KG_i}$ for $i = 1,2$
  2. $crs \leftarrow \boldsymbol{CRSGen}$
  3. $PK \leftarrow (pk_1, pk_2, crs)$
  4. $SK \leftarrow (sk_1, sk_2)$
  5. Return (PK, SK)

- $\boldsymbol{Enc_{NY}}(PK, m)$
  1. $r_1, r_2 \leftarrow$ random
  2. $c_i \leftarrow \boldsymbol{Enc_i}(pk_i, m; r_i)$ for $i = 1,2$
  3. $x \leftarrow (pk_1, pk_2, c_1, c_2)$
  4. $w \leftarrow (m, r_1, r_2)$
  5. $\pi \leftarrow \boldsymbol{Prove}(crs, x, w)$
  6. Return $C \leftarrow (c_1, c_2, \pi)$

- $\boldsymbol{Dec_{NY}}(SK, C)$
  1. $(c_1, c_2, \pi) \leftarrow C$
  2. $x \leftarrow (pk_1, pk_2, c_1, c_2)$
  3. If $\boldsymbol{Ver}(crs, x, \pi) = 0$ then return $\perp$
  4. Return $m \leftarrow \boldsymbol{Dec_1}(sk_1, c_1)$

Security of NY Construction

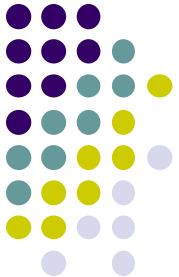
- Thm. [NY90]:
  Assume
  - PKE Π_1 and Π_2 are **IND-CPA**
  - P is **NIZK** for L_{eq}

  ➔ NY construction Π_{NY} is **IND-CCA1**

- Thm. [Sah90,Lin03]:
  Assume
  - PKE Π_1 and Π_2 are **IND-CPA**
  - P is **NIZK** for L_{eq} and satisfies **(one-time) simulation soundness**

  ➔ NY construction Π_{NY} is **IND-CCA2**

Intuition for CCA Security Proof of NY Construction



- Basic strategy: Use **CPA security** of underlying PKE $\boldsymbol{\pi_1}$ and $\boldsymbol{\pi_2}$ to show that challenge ciphertext $C^* = (c^*_1, c^*_2, \pi^*)$ doesn't leak m_b
  1. π^* is dependent on m_b and randomness r^*_1 and r^*_2 used in c^*_1 and c^*_2 But due to ***ZK*** of ***P***, *crs* and π^* can be replaced *with* simulated ones *without using* m_b *and* r^*_1, r^*_2
  2. Due to **CPA security** of $\boldsymbol{\Pi_2}$, m_b can be erased from c^*_2
  3. Due to ***simulation soundness*** of ***P***, after checking the validity of π, we almost always have $\boldsymbol{Dec_1}(sk_1, c_1) = \boldsymbol{Dec_2}(sk_2, c_2)$
     ➔ sk_2 can be used to answer dec. queries, instead of sk_1
  4. Now m_b is contained only in c^*_1, but it is protected by **CPA security** of $\boldsymbol{\Pi_1}$

➔ m_b is leaked from nowhere ☺

Game-Hopping-Style Proof [Sho04], [BR06]

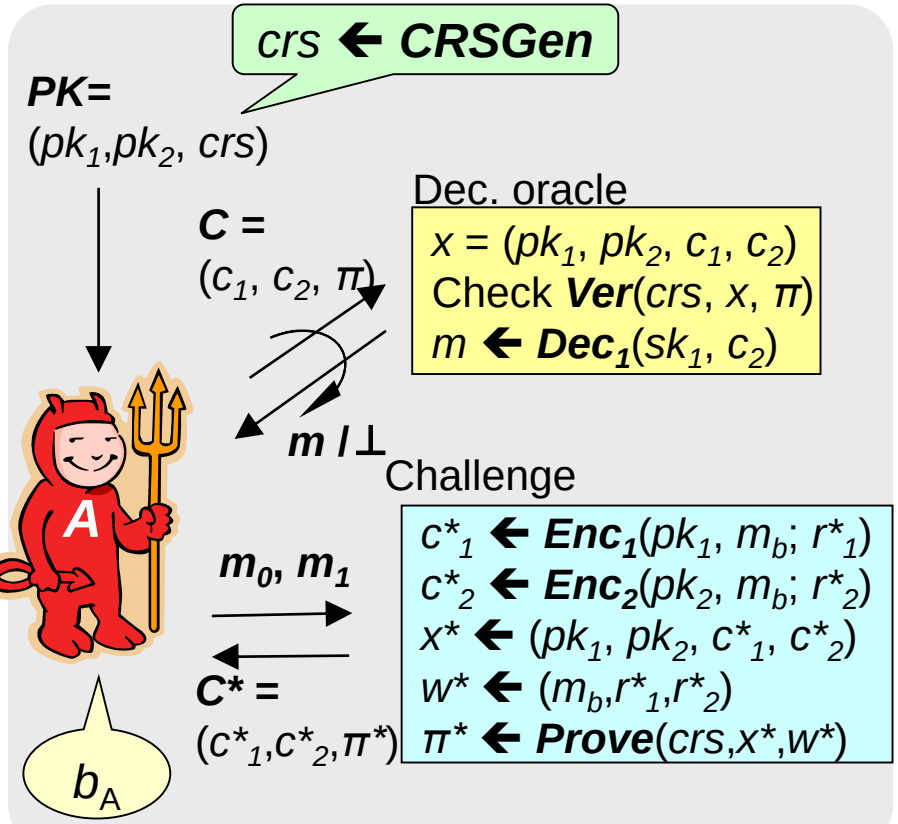
- A methodology for proving security of a crypto primitive based on multiple assumptions
  - In the case of this tutorial:
  1. Define **Game 1** = Original CCA security game
  2. Define several "related" security games (**Games 2, 3, 4, …**)
  3. Then, show that
     - Pr[***A*** wins **Game 1**] ≈ Pr[***A*** wins **Game 2**],
       Pr[***A*** wins **Game 2**] ≈ Pr[***A*** wins **Game 3**], etc., and
     - Pr[***A*** wins **Final Game**] ≈ 1/2

Each step is a usual reduction-style proof, or uses an info.-theoretic argument

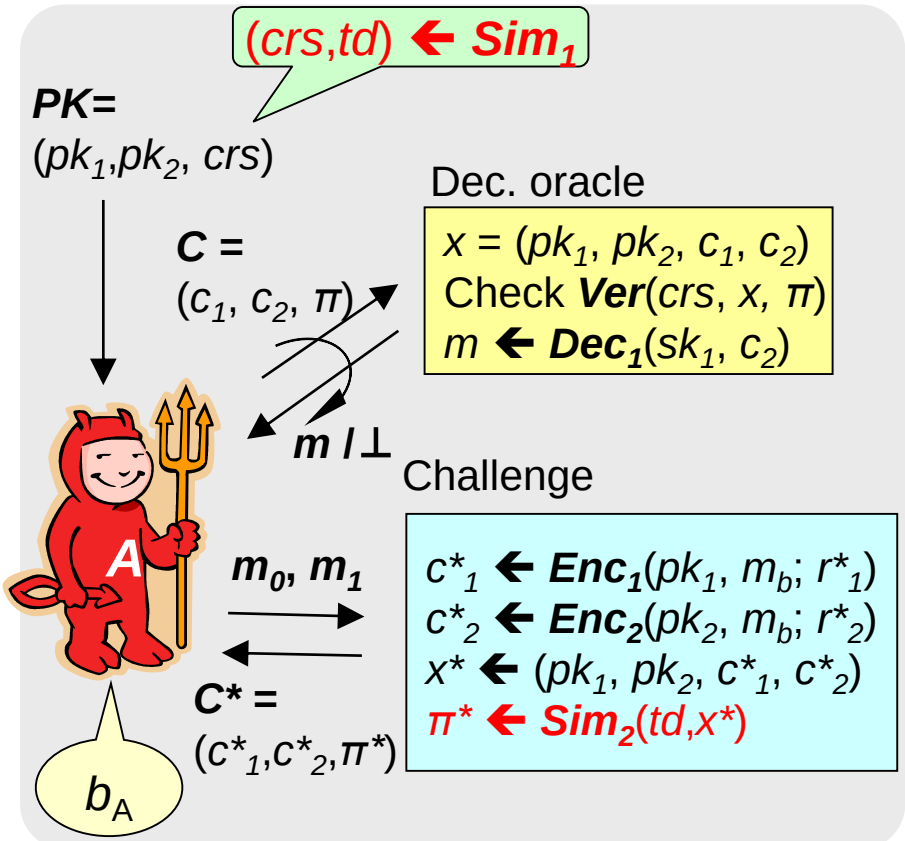
  - Then, the security proof is completed because
    A's CCA advantage = |Pr[***A*** wins **Game 1**] – 1/2|
    ≈ |Pr[***A*** wins **Final Game**] – 1/2| = neg.

Security Proof of NY (1/5)

- **Game 1** (CCA game)



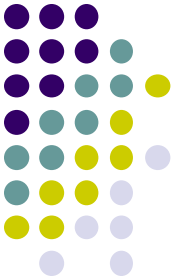
- **Game 2**



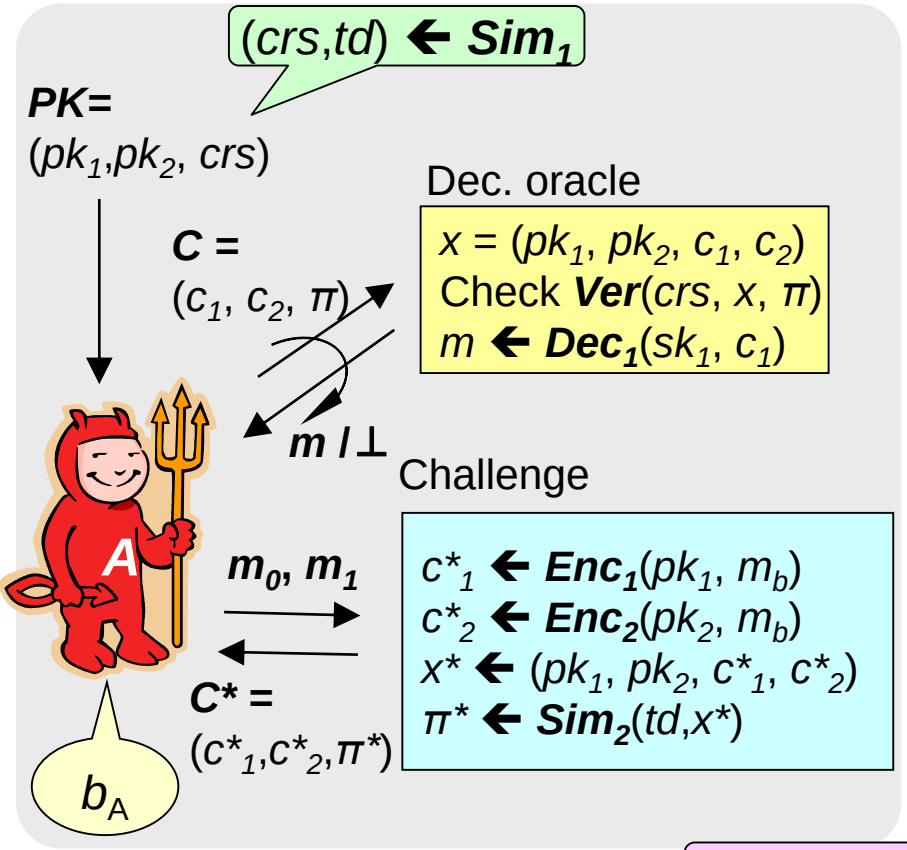
Due to ZK property of NIZK *P*

Pr[$b_A = b$ in **Game 1**] ≈ Pr[$b_A = b$ in **Game 2**]

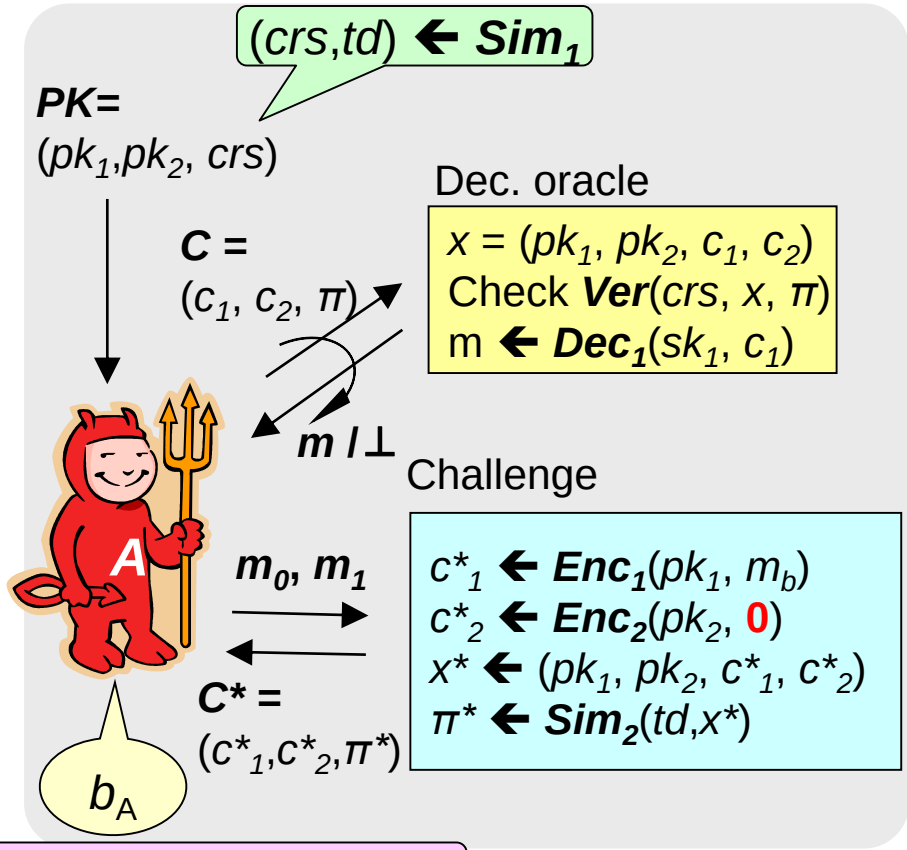
Security Proof of NY (2/5)



- **Game 2**



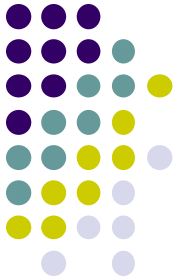
- **Game 3**



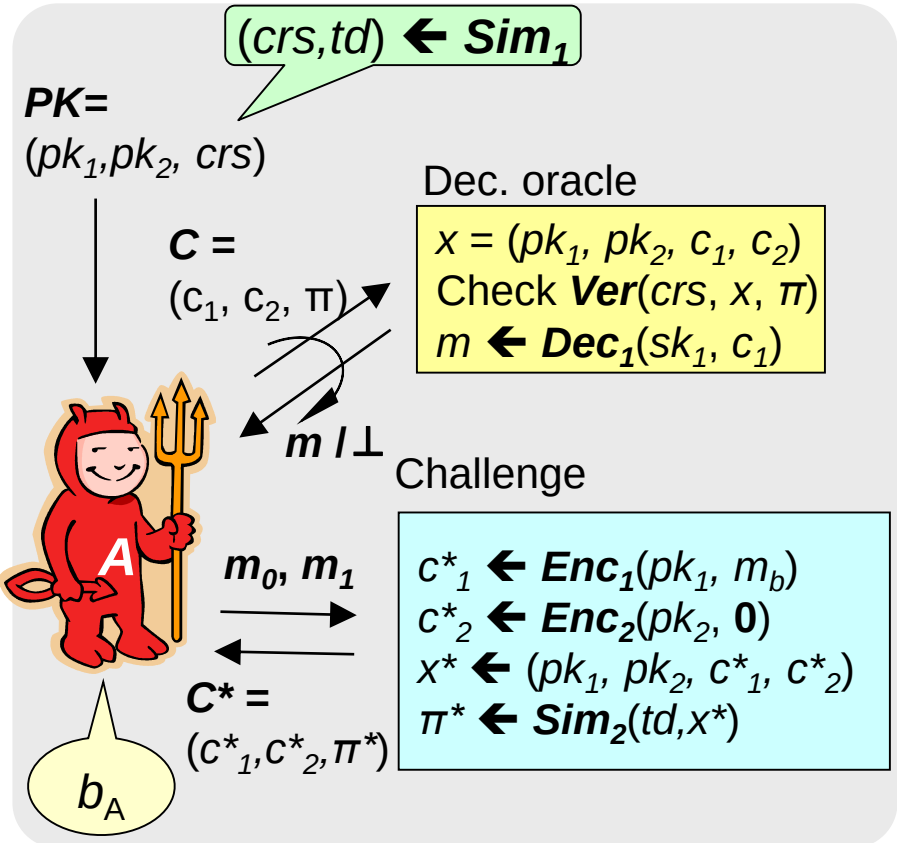
Due to CPA security of PKE π_2

$\Pr[b_A = b \text{ in } \textbf{Game 2}] \approx \Pr[b_A = b \text{ in } \textbf{Game 3}]$

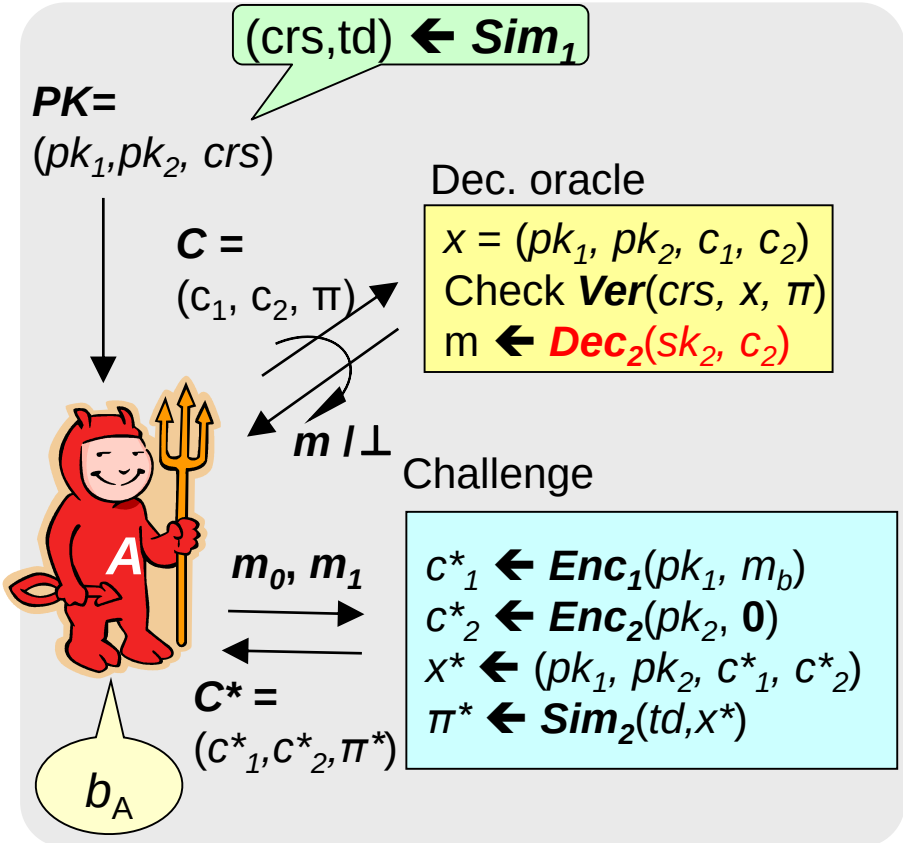
Security Proof of NY (3/5)



- **Game 3**



- **Game 4**

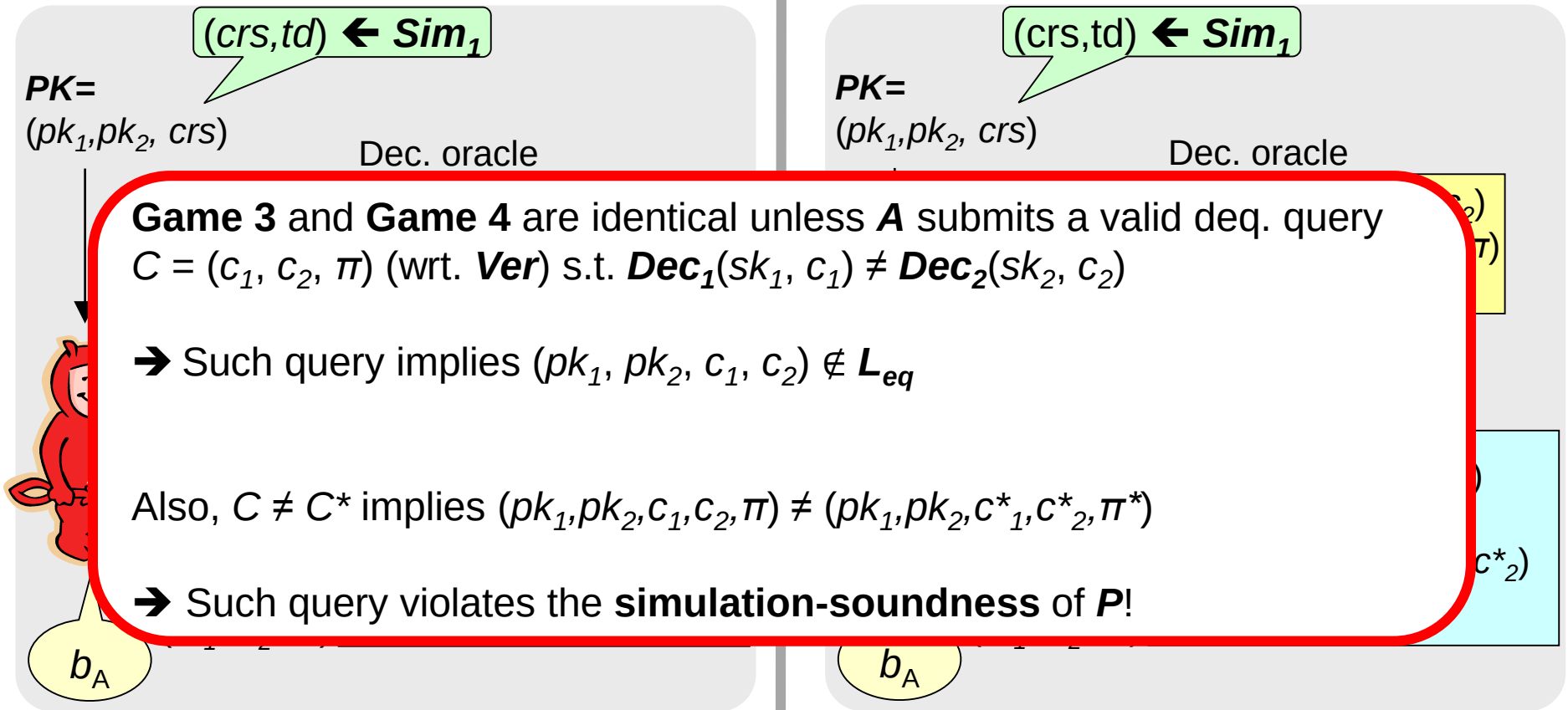


Pr[$b_A = b$ in **Game 3**]

Pr[$b_A = b$ in **Game 4**]

Security Proof of NY (3/5)

- **Game 3**
- **Game 4**



Game 3 and **Game 4** are identical unless ***A*** submits a valid deq. query $C = (c_1, c_2, \pi)$ (wrt. ***Ver***) s.t. $\boldsymbol{Dec_1}(sk_1, c_1) \neq \boldsymbol{Dec_2}(sk_2, c_2)$

➔ Such query implies $(pk_1, pk_2, c_1, c_2) \notin \boldsymbol{L_{eq}}$

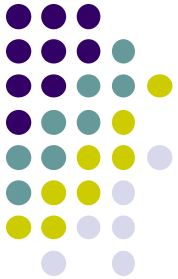
Also, $C \neq C^*$ implies $(pk_1,pk_2,c_1,c_2,\pi) \neq (pk_1,pk_2,c^*_1,c^*_2,\pi^*)$

➔ Such query violates the **simulation-soundness** of ***P***!

Due to Sim.-soundness of NIZK *P*

$\Pr[b_A = b \text{ in } \textbf{Game 3}] \approx \Pr[b_A = b \text{ in } \textbf{Game 4}]$

Security Proof of NY (4/5)



- **Game 4**

$(crs,td) \leftarrow \boldsymbol{Sim_1}$

$\boldsymbol{PK}=$
(pk_1,pk_2, crs)

$\boldsymbol{C} =$
(c_1, c_2, π)

Dec. oracle
$x = (pk_1, pk_2, c_1, c_2)$
Check $\boldsymbol{Ver}(crs, x, \pi)$
$m \leftarrow \boldsymbol{Dec_2}(sk_2, c_2)$

$\boldsymbol{m} / \perp$

Challenge
$c^*_1 \leftarrow \boldsymbol{Enc_1}(pk_1, m_b)$
$c^*_2 \leftarrow \boldsymbol{Enc_2}(pk_2, \mathbf{0})$
$x^* \leftarrow (pk_1, pk_2, c^*_1, c^*_2)$
$\pi^* \leftarrow \boldsymbol{Sim_2}(td,x^*)$

$\boldsymbol{m_0, m_1}$

$\boldsymbol{C^*} =$
(c^*_1,c^*_2,π^*)

A

b_A

- **Game 5**

$(crs,td) \leftarrow \boldsymbol{Sim_1}$

$\boldsymbol{PK}=$
(pk_1,pk_2, crs)

$\boldsymbol{C} =$
(c_1, c_2, π)

Dec. oracle
$x = (pk_1, pk_2, c_1, c_2)$
Check $\boldsymbol{Ver}(crs, x, \pi)$
$m \leftarrow \boldsymbol{Dec_2}(sk_2, c_2)$

$\boldsymbol{m} / \perp$

Challenge
$c^*_1 \leftarrow \boldsymbol{Enc_1}(pk_1, \mathbf{0})$
$c^*_2 \leftarrow \boldsymbol{Enc_2}(pk_2, \mathbf{0})$
$x^* \leftarrow (pk_1, pk_2, c^*_1, c^*_2)$
$\pi^* \leftarrow \boldsymbol{Sim_2}(td,x^*)$

$\boldsymbol{m_0, m_1}$

$\boldsymbol{C^*} =$
(c^*_1,c^*_2,π^*)

A

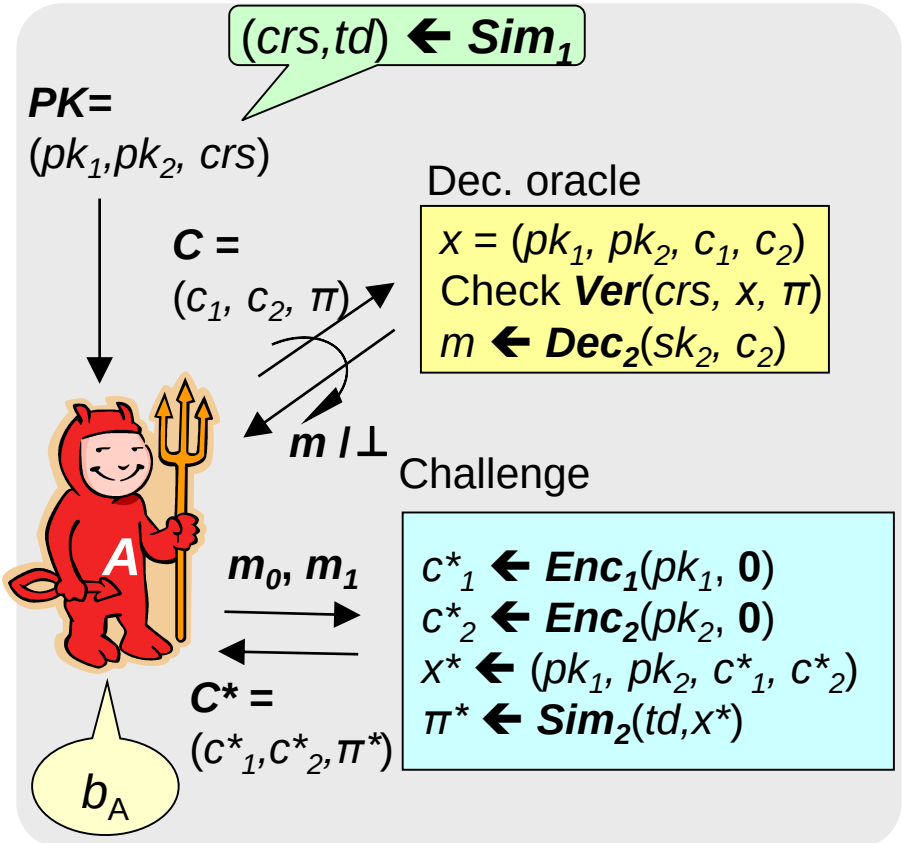
b_A

Due to CPA security of PKE $\boldsymbol{\pi_1}$

$\Pr[b_A = b \text{ in } \mathbf{Game\ 4}] \approx \Pr[b_A = b \text{ in } \mathbf{Game\ 5}]$

Security Proof of NY (5/5)

- **Game 5**

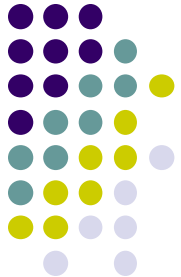


- In **Game 5**, no info. on m_b appears in ***A***'s view
  ➔ Pr[b_A = b in **Game 5**] = 1/2

- In summary,
  Pr[b_A = b in **Game 1**]
  ≈ … ≈
  Pr[b_A = b in **Game 5**] = 1/2

➔ ***A***'s CCA advantage =
|Pr[b_A = b in **Game 1**] − 1/2| = neg.

Some Remarks



- NY is versatile and is used to achieve many types of CCA security
  - KDM security, selective-opening security, leakage resilience, etc.

- NIZK proofs for any NP language (with adaptive soundness)
  - Based on (enhanced version of)
    Trapdoor permutation [Gol01], [Gol04], [Gol11], [GR13], [CL18]
  - Based on Bilinear maps [GOS06] etc.

- Transformation of any NIZK w/o sim.-soundness
  into one with (one-time) sim.-soundness [Lin03]

- CPA PKE + NIZK w/o simulation soundness [DDN91]
  - Construction is much more complicated than NY construction

Part 1 Outline

- Naor-Yung construction
- KEM & Hybrid Encryption
- Hash proof systems
- Fujisaki-Okamoto construction

Hybrid Encryption and KEM

- Typically, PKE is order of magnitude slower than symmetric key primitives, and not suitable for encrypting long messages

- **Hybrid encryption** is a simple yet quite practical method
- Idea: Split the encryption process into “pk-part” and “sk- part”
  - PK-encrypt a (fixed-length) session-key *K*
  - SK-encrypt a (potentially long) message *m* using *K* as a key

- The “pk-part” of hybrid encryption is formalized as **key encapsulation mechanism (KEM)**
  - Formalized in [Sho00]

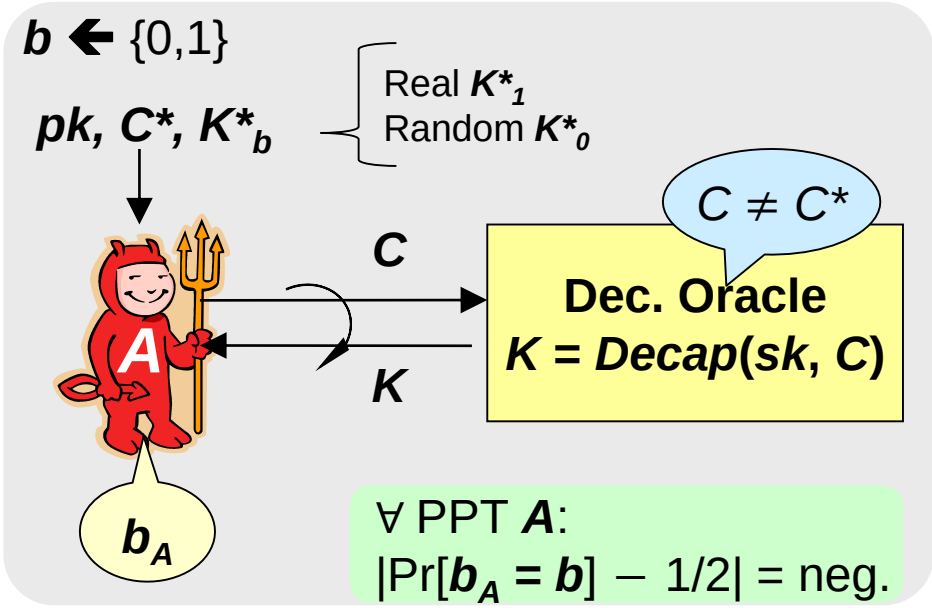
Also called DEM (data encapsulation mechanism) in the context of hybrid encryption

Key Encapsulation Mechanism (KEM)

= “public-key” part of hybrid encryption

- Syntax
  - Key Generation:
    $(pk, sk) \leftarrow KKG(1^k)$
  - Encapsulation:
    $(C, K) \leftarrow Encap(pk)$
    K: a session-key used by SKE
  - Decapsulation:
    K or $\perp \leftarrow Decap(sk, C)$

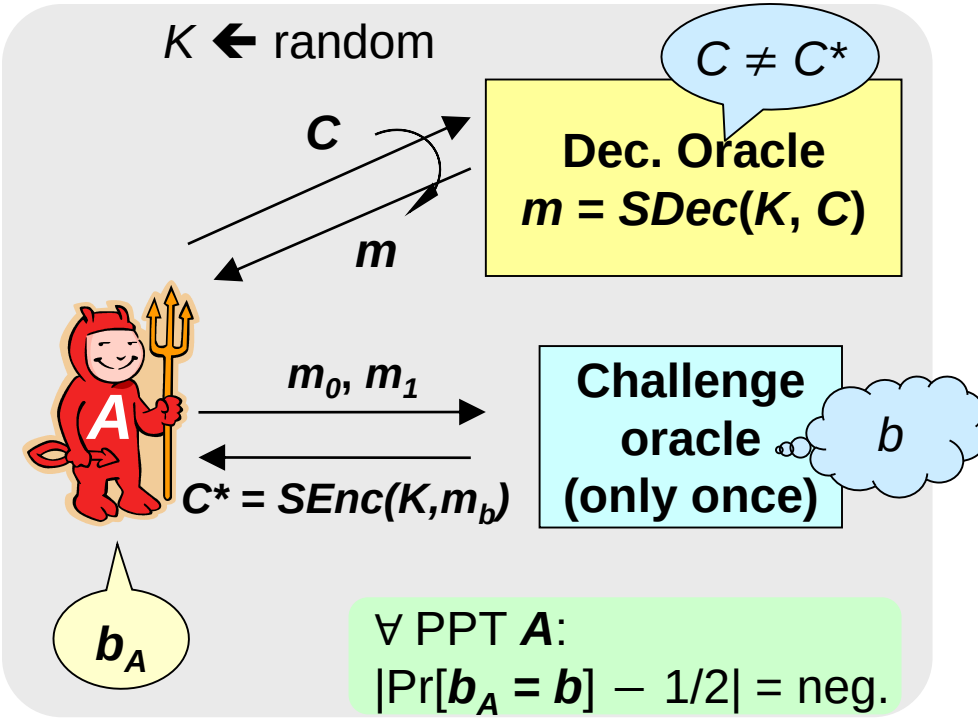
- **IND-CCA** Security:



Symmetric Key Encryption (SKE)

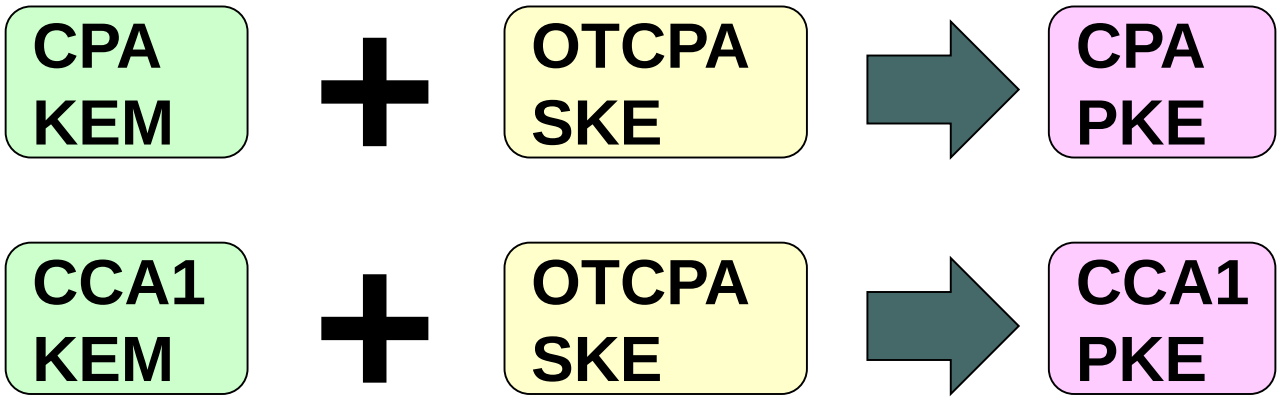
- Syntax
  - Encryption:
    $C \leftarrow SEnc(K, m)$
  - Decryption:
    m or $\perp \leftarrow SDec(K, C)$

- **IND-OTCCA** Security:



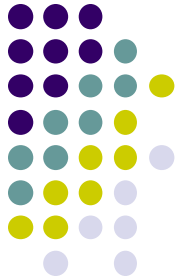
Useful Composition Results

- [HHK10]



- [CS03]

Hybrid Encryption



- Building Blocks:
  - KEM $\boldsymbol{\Gamma} = (\boldsymbol{KKG}, \boldsymbol{Encap}, \boldsymbol{Decap})$
  - SKE $\boldsymbol{E} = (\boldsymbol{SEnc}, \boldsymbol{SDec})$

- Hybrid PKE construction: $\boldsymbol{\Pi} = (\boldsymbol{KG}, \boldsymbol{Enc}, \boldsymbol{Dec})$

- ***KG***:

1. $(pk, sk) \leftarrow \boldsymbol{KKG}$
2. Return (pk, sk)

- ***Enc***(pk, m)

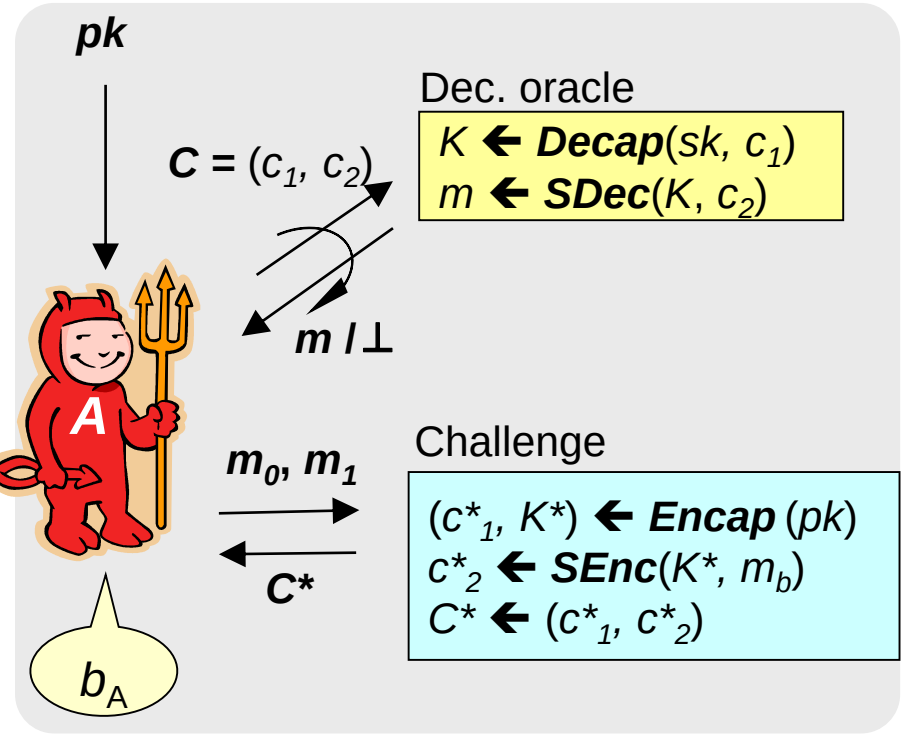
1. $(c_1, K) \leftarrow \boldsymbol{Encap}(pk)$
2. $c_2 \leftarrow \boldsymbol{SEnc}(K, m)$
3. Return $C \leftarrow (c_1, c_2)$

- ***Dec***(sk, C)

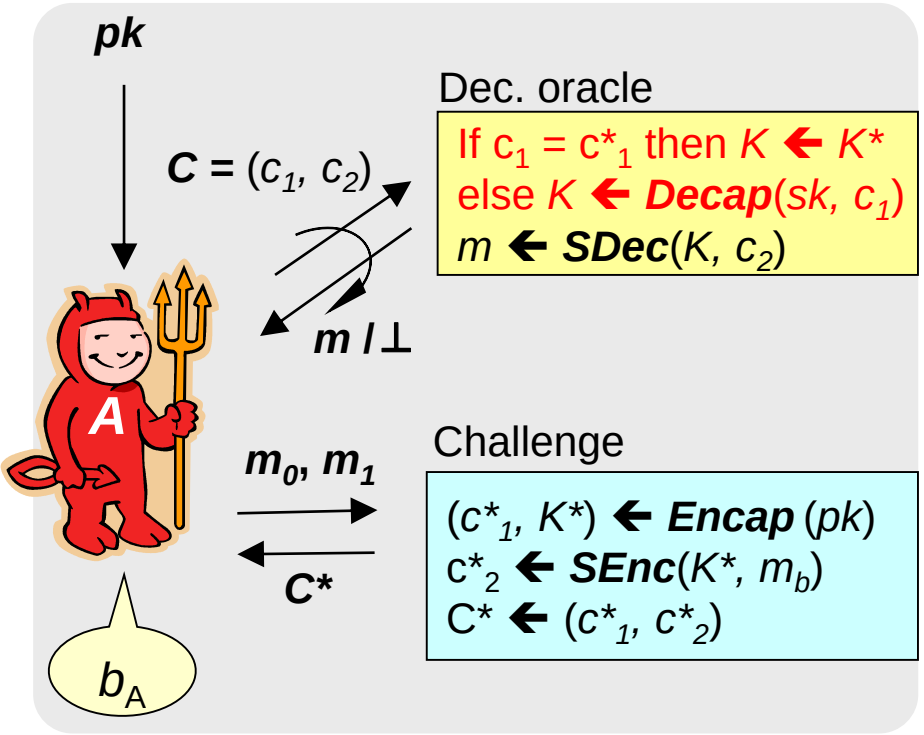
1. $(c_1, c_2) \leftarrow C$
2. $K \leftarrow \boldsymbol{Decap}(sk, c_1)$
3. If $K = \perp$ then return $\perp$
4. Return $m \leftarrow \boldsymbol{SDec}(K, c_2)$

Proof of CCAKEM + CCASKE = CCAPKE (1/4)

- **Game 1** (CCA game)



- **Game 2**

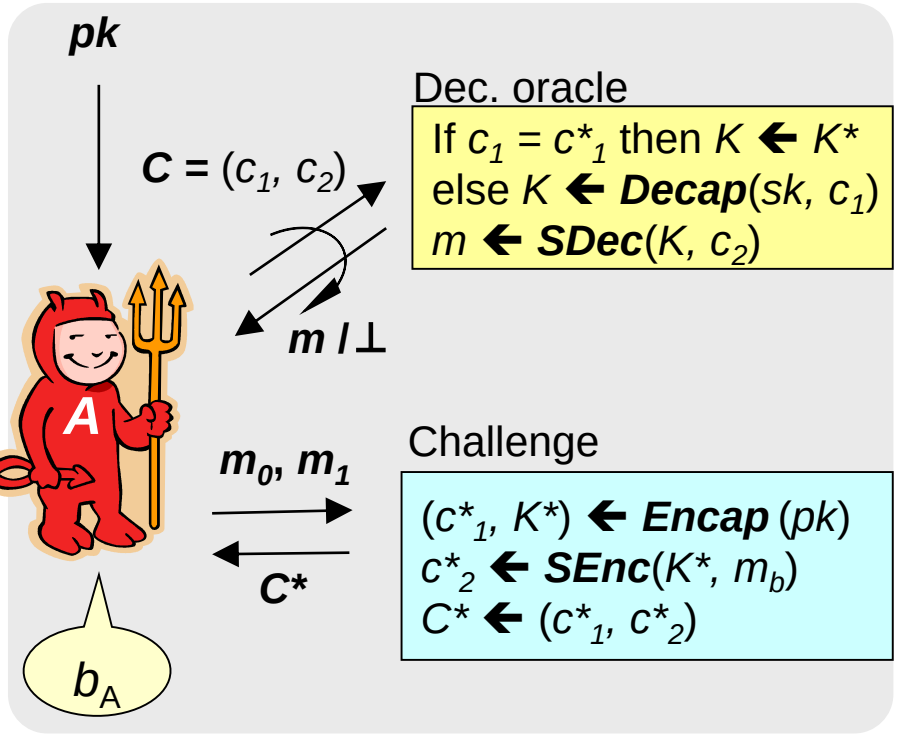


No difference in the behavior of Dec. oracle
(If $c_1 = c^*_1$, then ***Decap***$(sk, c_1) = K^*$)

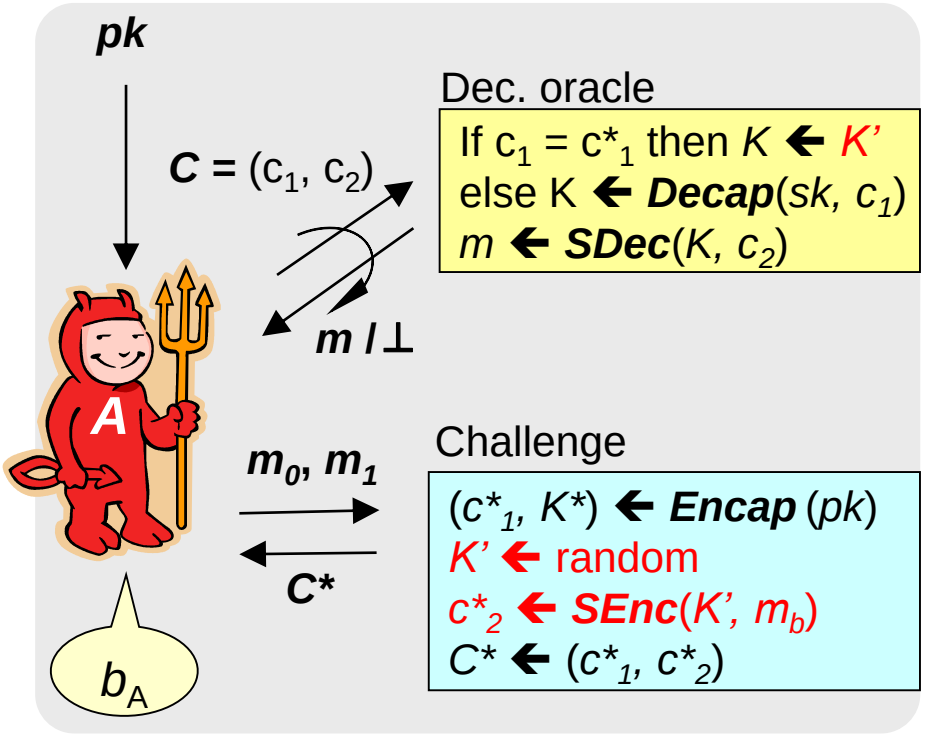
$\Pr[b_A = b$ in **Game 1**$]$ = $\Pr[b_A = b$ in **Game 2**$]$

Proof of CCAKEM + CCASKE = CCAPKE (2/4)

- **Game 2**



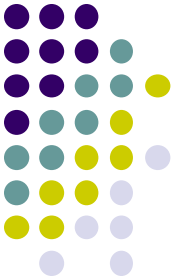
- **Game 3**



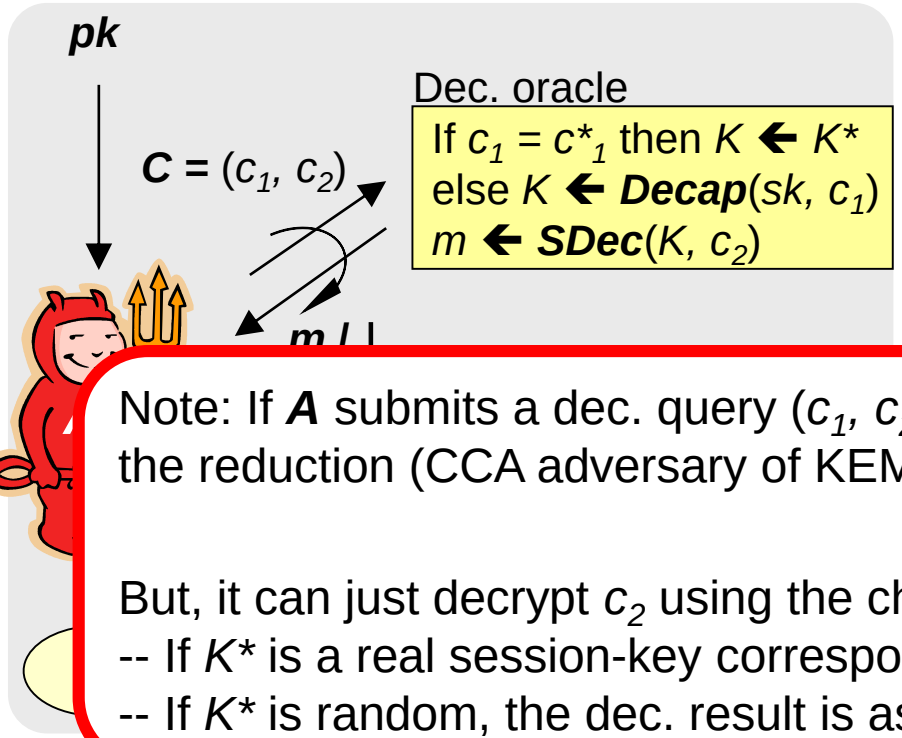
Pr[$b_A = b$ in **Game 2**]

Pr[$b_A = b$ in **Game 3**]

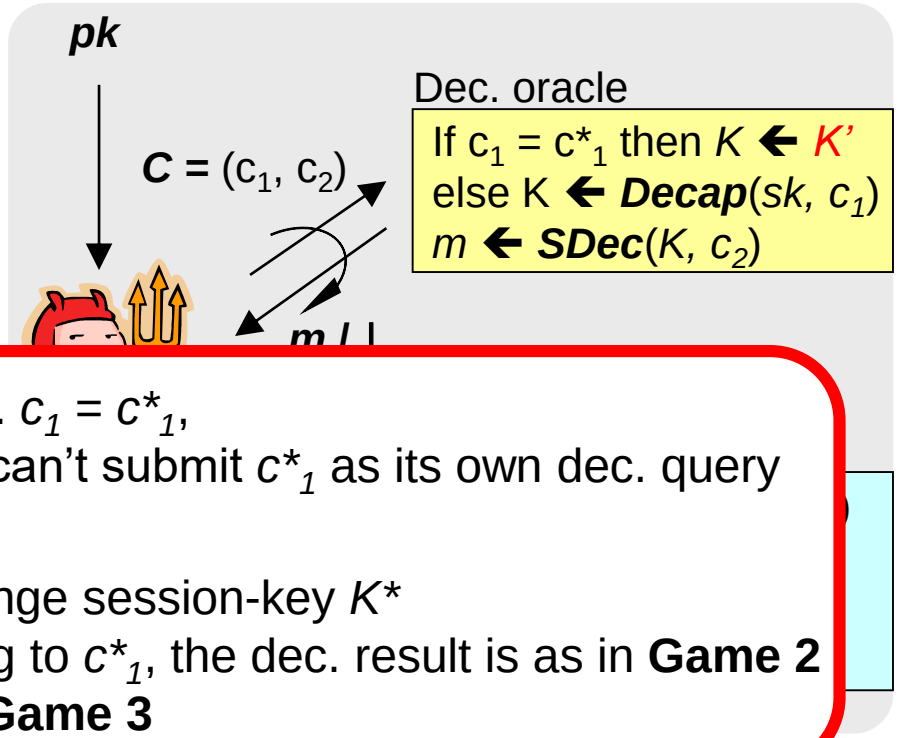
Proof of CCAKEM + CCASKE = CCAPKE (2/4)



- **Game 2**



- **Game 3**



Note: If ***A*** submits a dec. query (c_1, c_2) s.t. $c_1 = c^*_1$,
the reduction (CCA adversary of KEM **Γ**) can't submit c^*_1 as its own dec. query

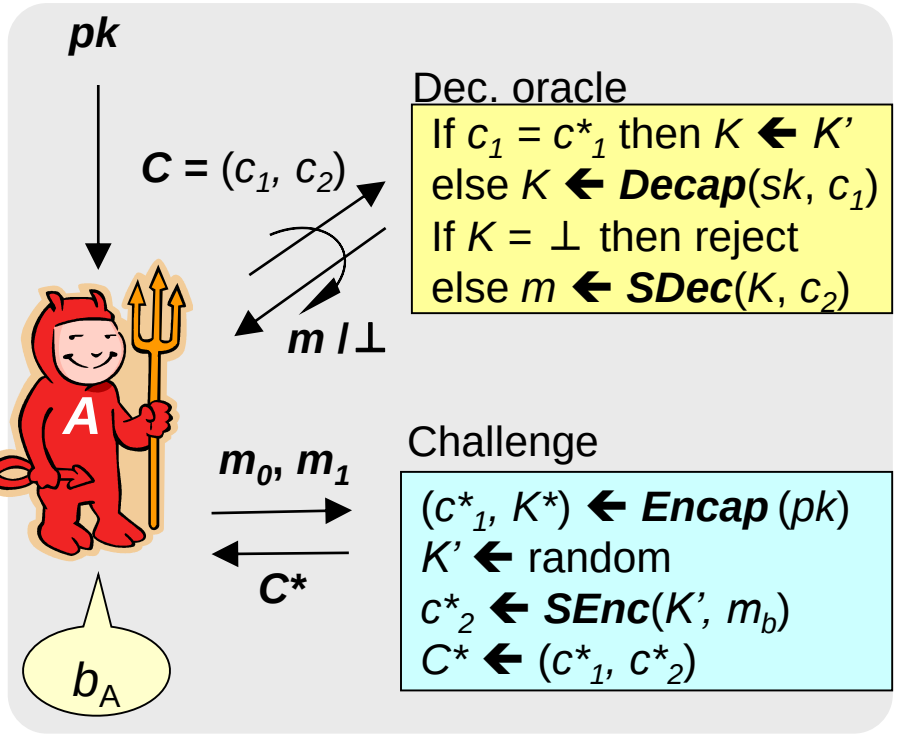
But, it can just decrypt c_2 using the challenge session-key K^*
-- If K^* is a real session-key corresponding to c^*_1, the dec. result is as in **Game 2**
-- If K^* is random, the dec. result is as in **Game 3**

Due to IND-CCA security of KEM *Γ*

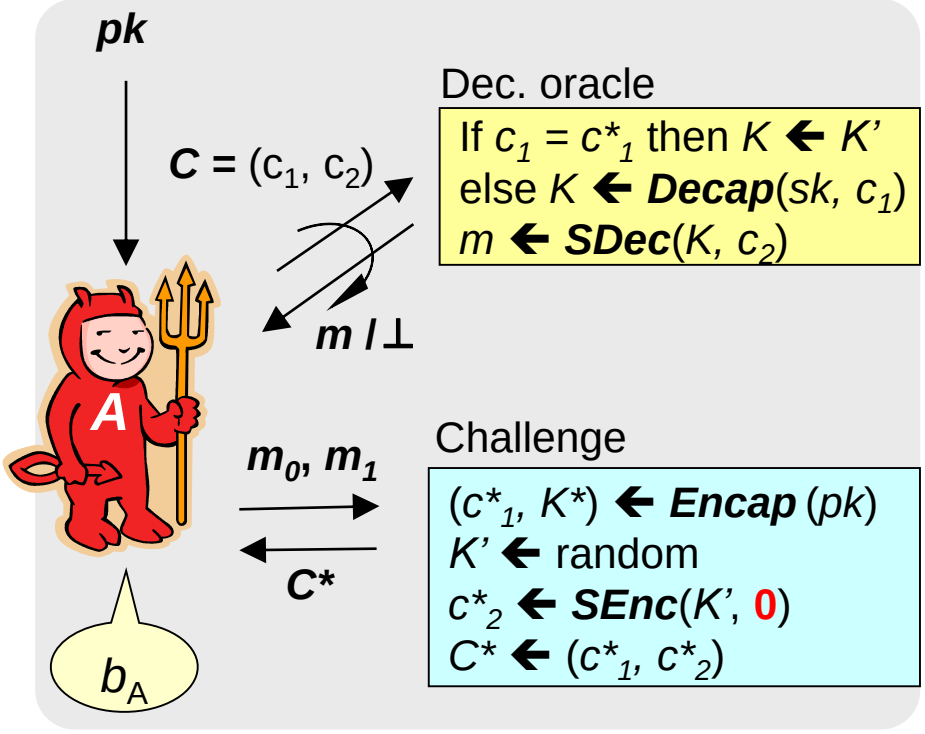
$\Pr[b_A = b$ in **Game 2**$]$ $\approx$ $\Pr[b_A = b$ in **Game 3**$]$

Proof of CCAKEM + CCASKE = CCAPKE (3/4)

- **Game 3**



- **Game 4**

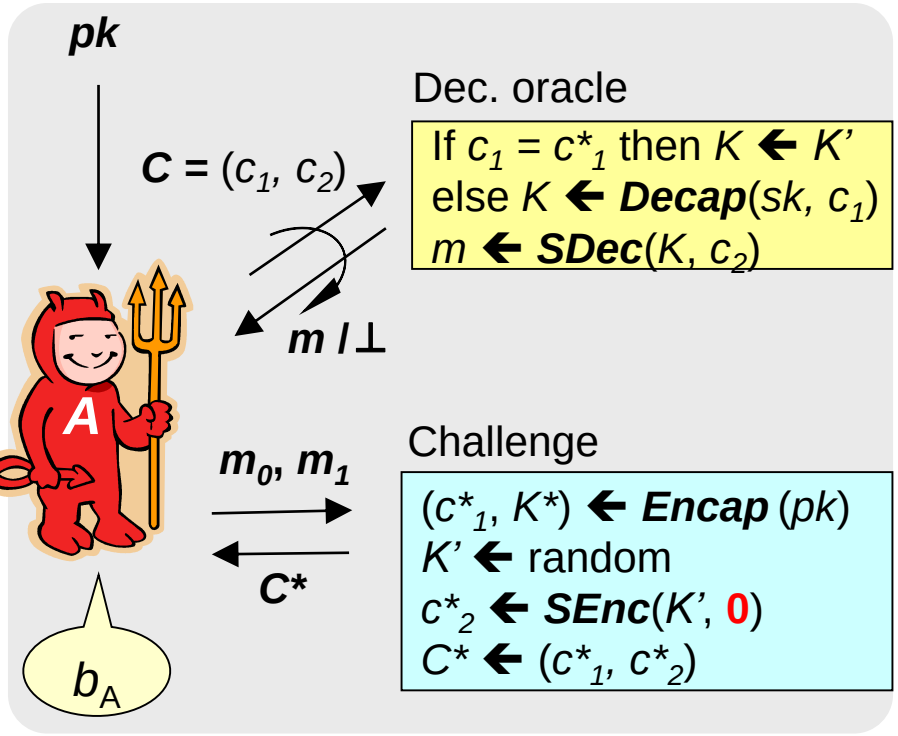


Due to IND-OTCCA security of SKE *E*

Pr[$b_A = b$ in **Game 3**] ≈ Pr[$b_A = b$ in **Game 4**]

Proof of CCAKEM + CCASKE = CCAPKE (4/4)

- **Game 4**



- In **Game 4**, m_b appears nowhere

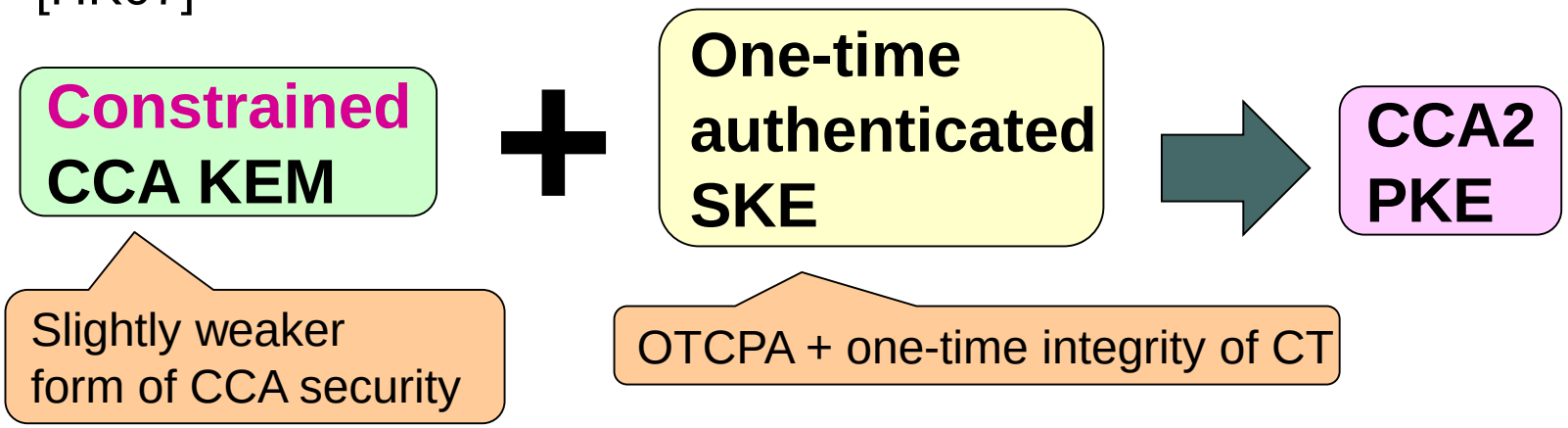
➔ Pr[$b_A = b$ in **Game 4**] = 1/2

- In summary,
  Pr[$b_A = b$ in **Game 1**]
  ≈ … ≈
  Pr[$b_A = b$ in **Game 4**] = 1/2

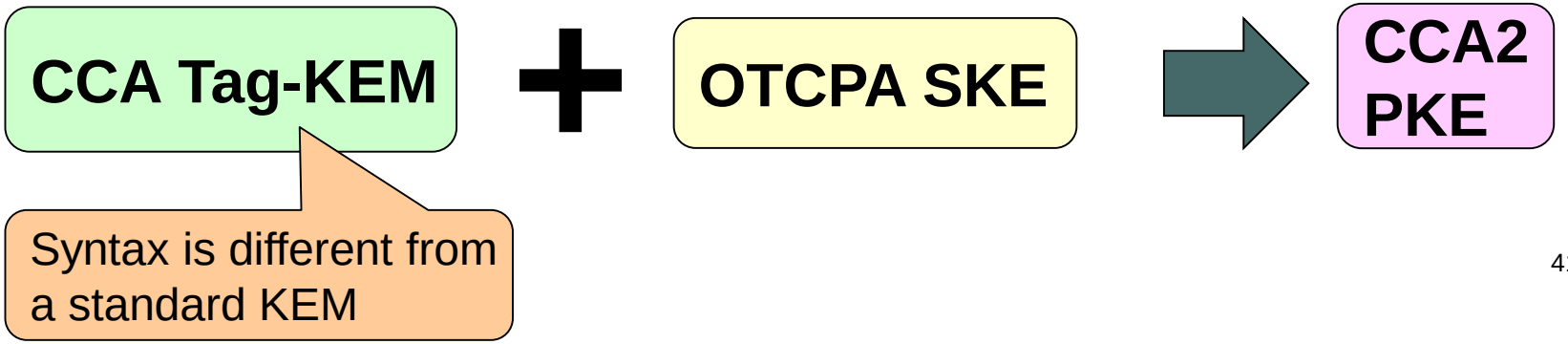
➔ ***A***'s CCA advantage =
|Pr[$b_A = b$ in **Game 1**] − 1/2| = neg.

Other Hybrid Composition Paradigms

- [HK07]



- [AGKS05]



Part 1 Outline

- Naor-Yung construction
- KEM & Hybrid Encryption
- Hash proof systems
- Fujisaki-Okamoto construction

CCA PKE/KEM Based on Hash Proof System

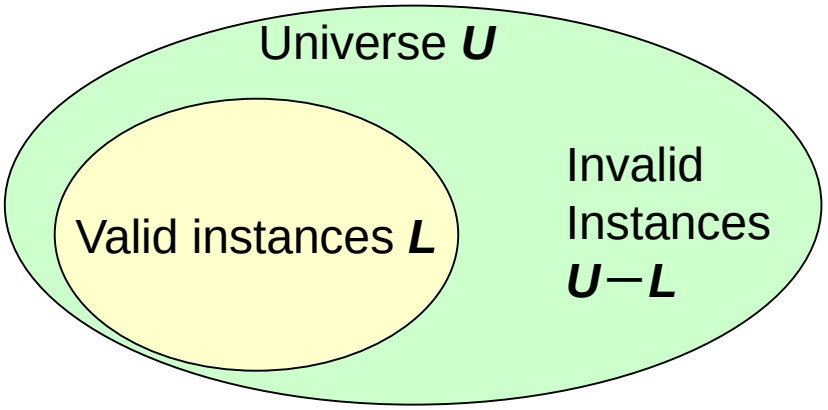
- Cramer and Shoup showed the first practical CCA secure PKE based on the DDH assumption [CS98]

- They generalized and abstract their construction using the notion of **Hash Proof System (HPS)** [CS02]

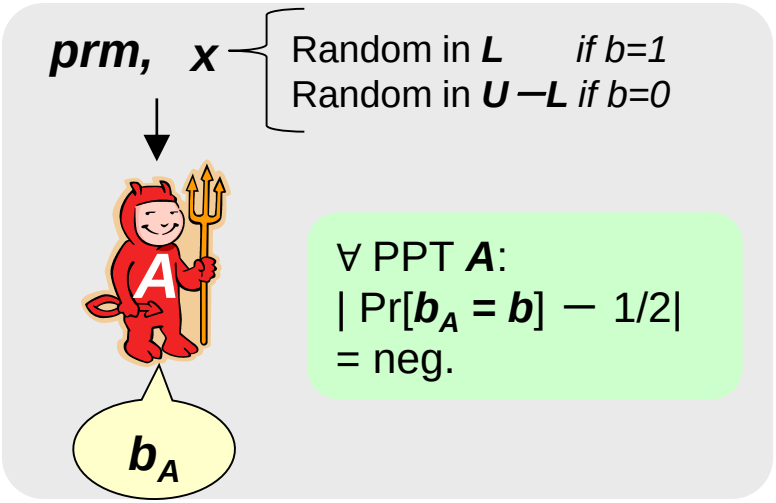
- HPS is a special kind of NIZK proof system for a specific NP language called a **subset membership problem**

Subset Membership Problem (SMP)

- Given $c \in \boldsymbol{U}$ (universe), tell whether $c \in \boldsymbol{L}$ *or* $c \in \boldsymbol{U}-\boldsymbol{L}$



- Hardness of SMP:

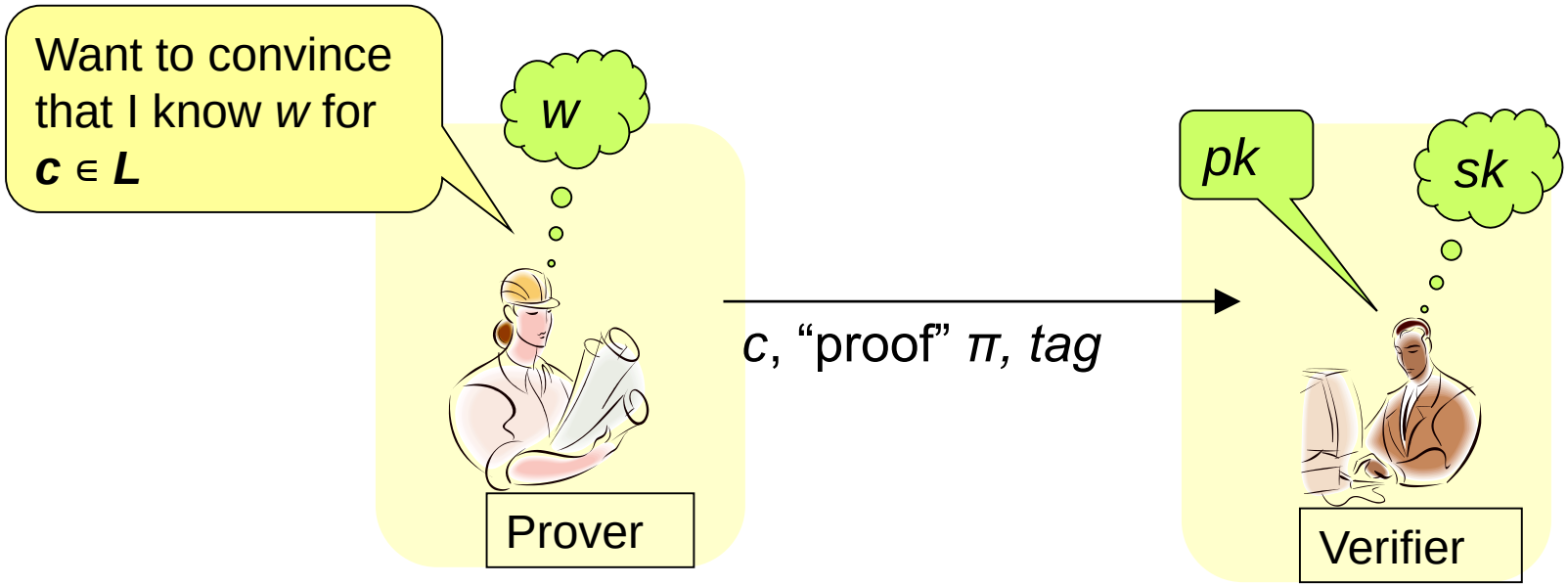


- Example: Diffie-Hellman tuple

  Let **G** be a cyclic group of prime order p *and* $prm = (g, h = g^{\alpha}) \in \mathbf{G^2}$
  - $U = \mathbf{G^2}$
  - $L = \{ (c_1, c_2) \in \mathbf{G}^2 \mid c_2 = c_1^{\alpha} \}$ witness is $w = \mathrm{Dlog}_g c_1$ s.t. $(c_1, c_2) = (g^w, h^w) = (g^w, g^{\alpha w})$
  - Assuming the hardness of this SMP is nothing but the DDH assumption

Hash Proof System [CS02]

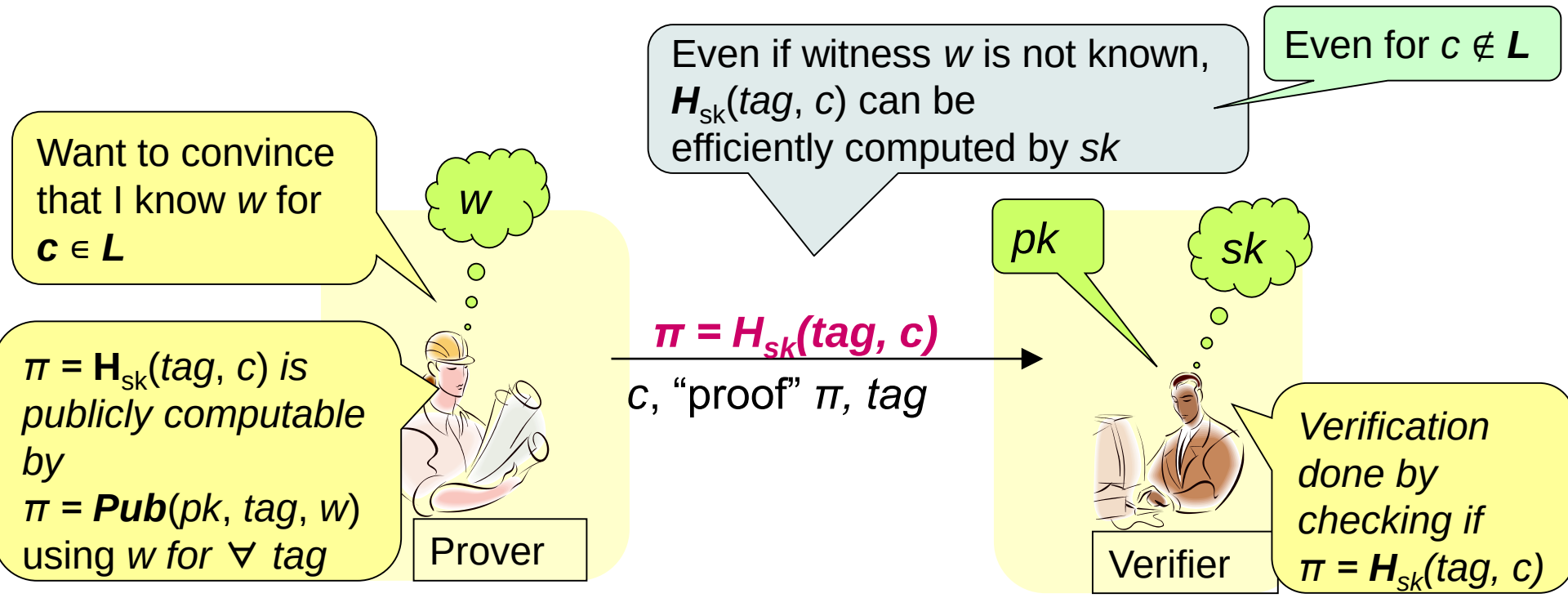
- A special kind of designated-verifier non-interactive ZK proof system for a SMP (***U, L,*** *prm*)



- A proof *π* is associated with a string "*tag*" (it could be empty string)

Hash Proof System [CS02]

- A special kind of designated-verifier non-interactive ZK proof system for a SMP (***U, L,*** *prm*)



- A proof *π* is associated with a string "*tag*" (it could be empty string)

Hash Proof System

- 3 *deterministic* algorithms

Given sk, π for any $c \in \boldsymbol{U}$ (not only $\boldsymbol{L}$) can be computed

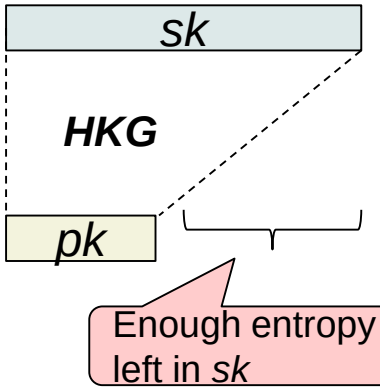
Key Generation	pk	$\leftarrow \boldsymbol{HKG}(sk)$	
Private Evaluation	π	$\leftarrow \boldsymbol{H}_{sk}(tag, c)$ $c \in \boldsymbol{U}$	
Public Evaluation	π	$\leftarrow \boldsymbol{Pub}(pk, tag, w)$	w: witness for $c \in \boldsymbol{L}$

Correctness: $\forall x \in \boldsymbol{L}$ *and witness w, tag, sk,* $pk = \mathbf{HKG}(sk)$: it holds that $\boldsymbol{H}_{sk}(tag, x) = \boldsymbol{Pub}(pk, tag, w)$

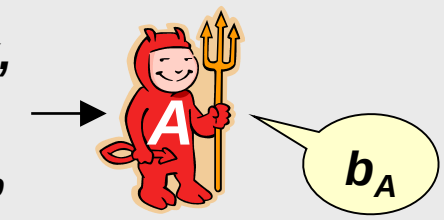
- In this tutorial, we will recall 2 statistical properties
  - **Smoothness** (next page) and **Universal-2** (later)

Smoothness

- Intuition:
  If *sk* and "invalid" $c \in \boldsymbol{U}-\boldsymbol{L}$ are chosen randomly,
  $\pi = \boldsymbol{H}_{sk}(tag, c)$ is statistically indistinguishable from random,
  even given *pk* (and *tag)*

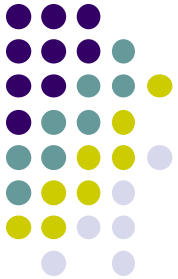


$b \leftarrow \{0,1\}$
$sk \leftarrow$ random
$pk \leftarrow \boldsymbol{HKG}(sk)$
$c \leftarrow \boldsymbol{U}-\boldsymbol{L}$
$\pi_1 \leftarrow \boldsymbol{H}_{sk}(tag,c)$
$\pi_0 \leftarrow$ random



∀ *tag* and comp. unbounded ***A***:
$|\Pr[\mathbf{b}_A = \mathbf{b}] - 1/2| = neg.$

Concrete HPS for DH-SMP with Smoothness



- **HKG:**
  Given $sk = (x_1, x_2) \in (\mathbf{Z}_p)^2$, compute $pk = g^{x1}h^{x2} \in \mathbf{G}$

- **Private evaluation:**
  $\boldsymbol{H}_{sk}(c_1, c_2) = c_1^{x1}c_2^{x2} \in \mathbf{G}$

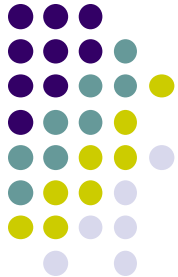
- **Public evaluation:**
  Given w s.t. $(c_1, c_2) = (g^w, h^w)$, compute $\pi = (pk)^w \in \mathbf{G}$
  - Note: $(pk)^w = (g^{x1}h^{x2})^w = (g^w)^{x1}(h^w)^{x2} = (c_1^{x1}c_2^{x2}) = \boldsymbol{H}_{sk}(c_1, c_2)$

- Intuition for smoothness
  - pk reveals one linear equation $\text{Dlog}_g(pk) = x_1 + \alpha \cdot x_2$, but otherwise hides $sk = (x_1, x_2)$

➔ If $(c^*_1, c^*_2) \in \boldsymbol{U} - \boldsymbol{L}$, $\boldsymbol{H}_{sk}(c_1,c_2) = c_1^{x1}c_2^{x2}$ constitutes another linearly independent equation and the remaining entropy of sk makes it look random

Diffie-Hellman SMP
- $prm = (g, h) = (g, g^\alpha) \in \boldsymbol{G}^2$
- $\boldsymbol{U} = \mathbf{G}^2$,
- $\boldsymbol{L} = \{ (c_1, c_2) \in \mathbf{G}^2 \mid c_2 = c_1^\alpha \}$,
  with witness $w = \text{Dlog}_g c_1$

Warm up: CPA KEM from HPS



- Building Blocks:
  - **SMP** (*prm*, ***L***, ***U***)
  - **HPS** = (***HKG***, ***H***, ***Pub***) for **SMP** (with empty-tag)

- KEM construction: ***Γ*** = (***KKG***, ***Encap***, ***Decap***)

- ***KKG***:
  1. *sk* ← random
  2. *pk* ← ***HKG***(*sk*)
  3. Return (*pk*, *sk*)

- ***Encap***(*pk*)
  1. Pick random c ∈ ***L*** and witness *w*
  2. *K* ← ***Pub***(*pk*, *w*)
  3. Return (*c*, *K*)

- ***Decap***(*sk*, *c*)
  1. Return $K \leftarrow H_{sk}(c)$

- Thm:
  If **SMP** is hard and **HPS** satisfies smoothness, then KEM ***Γ*** is CPA secure

Intuition for CPA Security

- ***KKG***:
  1. *sk* ← random
  2. *pk* ← ***HKG***(*sk*)
  3. Return (*pk*, *sk*)

- ***Encap***(*pk*)
  1. Pick random c ∈ ***L*** and witness *w*
  2. *K* ← ***Pub***(*pk*, *w*)
  3. Return (*c*, *K*)

- ***Decap***(*sk*, *c*)
  1. Return *K* ← $\boldsymbol{H}_{sk}(c)$

- We want to use **smoothness** of **HPS** to say that *K* looks random
- Before using **smoothness**, we need to make *c* invalid (i.e. *c* ∈ ***U*** − ***L***)

➔ Hardness of **SMP** can help to switch *c* ∈ ***L*** into *c* ∈ ***U*** − ***L***,
but before doing so, we have to ensure that the witness *w* is not used for computing *K*

➔ Before using so, we switch the computation of *K* from **public evaluation** (using *pk* and *w*) to **private evaluation** (using *sk* and *c*)

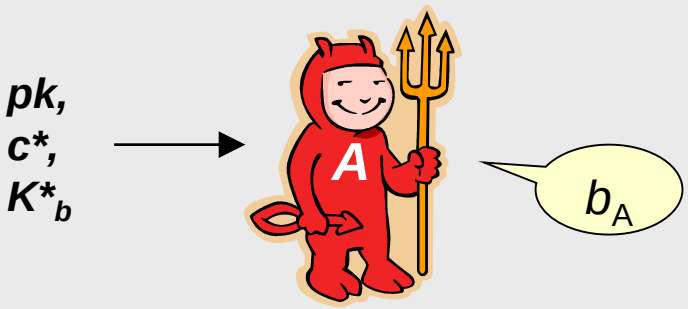
Proof of CPA KEM Based on HPS (1/3)

- **Game 1** (CPA game)

$b \leftarrow \{0,1\}$
$sk \leftarrow$ random
$pk \leftarrow \boldsymbol{HKG}(sk)$

Challenge

Pick random $c^* \in \boldsymbol{L}$
with witness w^*
$K^*_1 \leftarrow \boldsymbol{Pub}(pk, w^*)$
$K^*_0 \leftarrow$ random

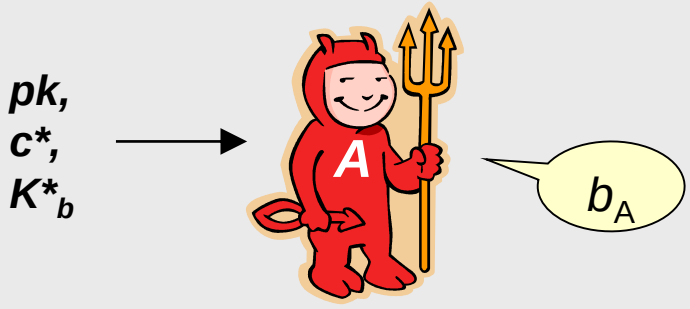


- **Game 2**

$b \leftarrow \{0,1\}$
$sk \leftarrow$ random
$pk \leftarrow \boldsymbol{HKG}(sk)$

Challenge

Pick random $c^* \in \boldsymbol{L}$
with witness w^*
$K^*_1 \leftarrow \boldsymbol{H_{sk}}(c^*)$
$K^*_0 \leftarrow$ random



Due to Correctness of HPS
(If $c^* \in \boldsymbol{L}$, then $\boldsymbol{Pub}(pk, w) = \boldsymbol{H_{sk}}(c^*)$)

$\Pr[b_A = b \text{ in } \mathbf{Game\ 1}] = \Pr[b_A = b \text{ in } \mathbf{Game\ 2}]$

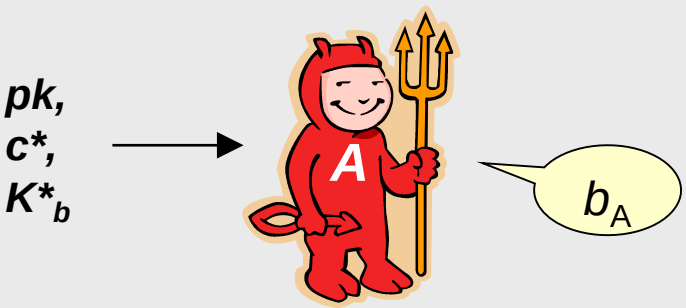
Proof of CPA KEM Based on HPS (2/3)

- **Game 2**

$b \leftarrow \{0,1\}$
$sk \leftarrow$ random
$pk \leftarrow \boldsymbol{HKG}(sk)$

Challenge

Pick random $c^* \in \boldsymbol{L}$
with witness w^*
$K^*_1 \leftarrow \boldsymbol{H_{sk}}(c^*)$
$K^*_0 \leftarrow$ random

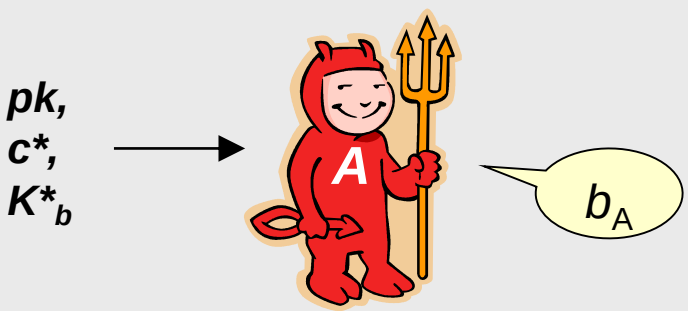


- **Game 3**

$b \leftarrow \{0,1\}$
$sk \leftarrow$ random
$pk \leftarrow \boldsymbol{HKG}(sk)$

Challenge

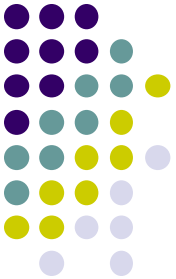
Pick random $c^* \in \boldsymbol{U-L}$
$K^*_1 \leftarrow \boldsymbol{H_{sk}}(c^*)$
$K^*_0 \leftarrow$ random



Due to Hardness of SMP

$\Pr[b_A = b \text{ in } \mathbf{Game\ 2}] \approx \Pr[b_A = b \text{ in } \mathbf{Game\ 3}]$

Proof of CPA KEM Based on HPS (2/3)



- **Game 2**

$b \leftarrow \{0,1\}$
$sk \leftarrow$ random
$pk \leftarrow \boldsymbol{HKG}(sk)$

Challenge

Pick random $c^* \in \boldsymbol{L}$
with witness w^*
$K^*_1 \leftarrow \boldsymbol{H_{sk}}(c^*)$
$K^*_0 \leftarrow$ random

- **Game 3**

$b \leftarrow \{0,1\}$
$sk \leftarrow$ random
$pk \leftarrow \boldsymbol{HKG}(sk)$

Challenge

Pick random $c^* \in \boldsymbol{U-L}$
$K^*_1 \leftarrow \boldsymbol{H_{sk}}(c^*)$
$K^*_0 \leftarrow$ random

Interestingly, this game hop can be shown even if ***A*** make dec. queries, because the reduction (the SMP solver) can know *sk*.

(Indistinguishability comes only from the instance c^* used as the challenge CT)

Due to Hardness of SMP

$\Pr[b_A = b \text{ in } \mathbf{Game\ 2}] \approx \Pr[b_A = b \text{ in } \mathbf{Game\ 3}]$

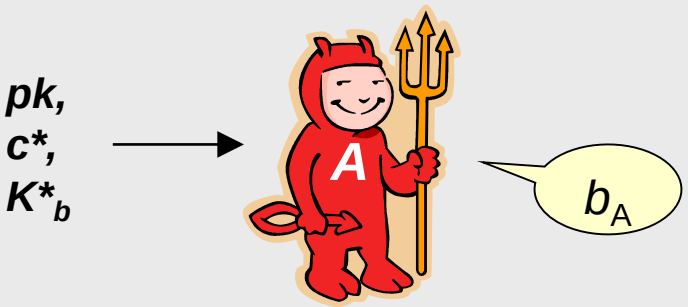
Proof of CPA KEM Based on HPS (3/3)

- **Game 3**

$b \leftarrow \{0,1\}$
$sk \leftarrow$ random
$pk \leftarrow \boldsymbol{HKG}(sk)$

Challenge

Pick random $c^* \in \boldsymbol{U} - \boldsymbol{L}$
$K^*_1 \leftarrow \boldsymbol{H_{sk}}(c^*)$
$K^*_0 \leftarrow$ random



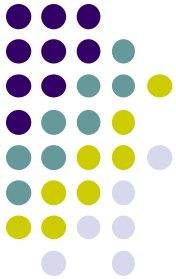
- In **Game 3**, the view of ***A*** is exactly as in the smoothness game

➔ **Smoothness** of **HPS** implies
$|\Pr[b_A = b \text{ in } \textbf{Game 3}] - 1/2| = \text{neg}$.

- In summary,
$\Pr[b_A = b \text{ in } \textbf{Game 1}]$
$\approx \ldots \approx$
$\Pr[b_A = b \text{ in } \textbf{Game 3}]$, and
$|\Pr[b_A = b \text{ in } \textbf{Game 3}] - 1/2| = \text{neg}$.

➔ ***A***'s CPA advantage =
$|\Pr[b_A = b \text{ in } \textbf{Game 1}] - 1/2| = \text{neg}$.

Towards CCA Security



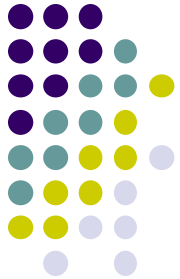
- ***KKG***:
  1. *sk* ← random
  2. *pk* ← ***HKG***(*sk*)
  3. Return (*pk*, *sk*)

- ***Encap***(*pk*)
  1. Pick random *c*∈***L*** and witness w
  2. *K* ← ***Pub***(*pk*, *w*)
  3. Return (*c*, *K*)

- ***Decap***(*sk*, *c*)
  1. Return *K* ← $\boldsymbol{H}_{sk}(c)$

- This KEM can't be proved CCA secure
- Where do we get stuck?
  - **Game1** and **Game 2** are identical ☺
  - The hop from **Game 2** to **Game 3** can be performed even if ***A*** makes dec. queries ☺
  - We can't argue | Pr[b_A = *b* in **Game 3**] – 1/2| = neg. if ***A*** makes dec. queries ☹

Observations in Game 3



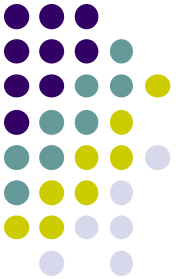
- If $\boldsymbol{A}$ submits a valid $c \in \boldsymbol{L}$ as a dec. query,
  its answer doesn't leak the info. of sk beyond pk, because
  - Correctness of HPS ensures $K = \boldsymbol{H}_{sk}(c)$ can be computed only from pk and witness w for $c \in \boldsymbol{L}$
  - Smoothness is a statistical property (against a comp. unbounded adversary)
  - Comp. unbounded adv. can find w for $c \in \boldsymbol{L}$ and $K = \boldsymbol{H}_{sk}(c)$ by itself

- However, if $\boldsymbol{A}$ submits an invalid $c \in \boldsymbol{U} - \boldsymbol{L}$ (other than c^*) as a dec. query,
  evaluation of invalid $\mathbf{H}_{sk}(c)$ may leak the info. of sk

  ➔ Smoothness of HPS can't ensure that $K^*_1 = \boldsymbol{H}_{sk}(c^*)$ looks random

- To achieve CCA security, we use another HPS for validity checking of c,
  so that dec. queries don't leak the information of sk
  - [CS02]

CCA KEM Based on HPS



- Building Blocks:
  - **SMP** (*prm*, ***L***, ***U***)
  - **HPS** = (***HKG***, ***H***, ***Pub***) for **SMP** (with empty-tag)
  - **HPS'** = (***HKG'***, ***H'***, ***Pub'***) for **SMP**

- CCA KEM construction: ***Γ*** = (***KKG***, ***Encap***, ***Decap***)

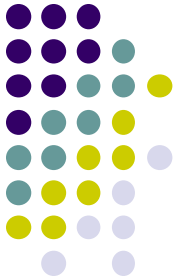
- ***KKG***:
  1. *sk* ← random
  2. *sk'* ← random
  3. *SK* ← (*sk*, *sk'*)
  4. *pk* ← ***HKG***(*sk*)
  5. *pk'* ← ***HKG'***(*sk'*)
  6. *PK* ← (*pk*, *pk'*)
  7. Return (*PK*, *SK*)

- ***Encap***(*PK*)
  1. Pick random *c*∈***L*** and witness *w*
  2. *K* ← ***Pub***(*pk, w*)
  3. *tag* ← c (*)
  4. *π* ← ***Pub'***(*pk'*, *tag*, w)
  5. *C* ← (*c*, *π*)
  6. Return (*C, K*)

- ***Decap***(*sk*, *C*)
  1. (*c*,*π*) ← *C*
  2. *tag* ← *c*
  3. If $H'_{sk'}(tag,c) \neq \pi$ reject
  4. Return *K* ← $H_{sk}(c)$

(*) If *c* doesn't fit the tag-space, we use a collision-resistant hash

CCA Security of HPS-Based KEM



- Thm [CS02]:
  - Assume
    - **SMP** is hard
    - **HPS** satisfies smoothness
    - **HPS'** satisfies **universal-2**

  ➔ HPS-based KEM is **IND-CCA**

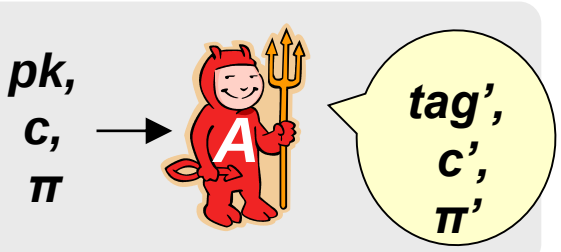
(*) slightly simplified and weakened from the definition of [CS02]

Universal-2

- Similar to simulation-soundness of NIZK proofs

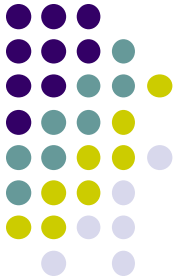
- If *sk* is chosen randomly,
  even after observing $pk = \boldsymbol{HKG}(sk)$
  and $\pi = \boldsymbol{H}_{sk}(tag,c)$ for invalid $c \in \boldsymbol{U}-\boldsymbol{L}$,
  the "second" evaluation $\boldsymbol{H}_{sk}(tag',c')$ is statistically hard to guess
  for any $(c', tag') \neq (c, tag)$ and $c' \in \boldsymbol{U}-\boldsymbol{L}$

$sk \leftarrow$ random
$pk \leftarrow \boldsymbol{HKG}(sk)$
$c \leftarrow \boldsymbol{U}-\boldsymbol{L}$
$\pi \leftarrow \boldsymbol{H}_{sk}(tag,c)$



∀ *tag* and comp. unbounded $\boldsymbol{A}$:
$\Pr[\boldsymbol{H}_{sk}(tag',c') = \pi'$
$\wedge\ c' \in \boldsymbol{U}-\boldsymbol{L}$
$\wedge\ (\underline{tag'},c') \neq (tag,c)\ ] = \text{neg}.$

Concrete HPS for DH-SMP with Universal-2



- ***HKG***:
  Given $sk = (y_1, y_2, z_1, z_2) \in (\mathbf{Z}_p)^4$,
  compute $pk = (g^{y1}h^{z1}, g^{y2}h^{z2}) \in \mathbf{G}^2$

Diffie-Hellman SMP
- $prm = (g, h) = (g, g^{\alpha}) \in \boldsymbol{G}^2$
- $\mathbf{U} = \mathbf{G}^2$,
- $\boldsymbol{L} = \{ (c_1, c_2) \in \mathbf{G}^2 \mid c_2 = c_1^{\alpha} \}$,
  with witness $w = \mathrm{Dlog}_g c_1$

- **Private evaluation:**
  $\boldsymbol{H}_{sk}(tag, c_1, c_2) = c_1^{y1 + tag \cdot y2} c_2^{z1 + tag \cdot z2} \in \mathbf{G}$

- **Public evaluation**:
  Given w s.t. $(c_1, c_2) = (g^w, h^w)$ and
  $pk = (pk_1, pk_2) \in \mathbf{G}^2$,
  compute $(pk_1)^w \cdot (pk_2)^{tag \cdot w} \in \mathbf{G}$

High-level structure:
Parallelly use
the previous HPS
with smoothness

- **Universal-2** can be shown with a similar reasoning about linear equations of *sk* leaked from *pk* and $\boldsymbol{H}_{sk}(tag, c_1, c_2)$

How Universal-2 Help?

- ***KKG***:
1. *sk* ← random
2. *sk'* ← random
3. *SK* ← (*sk*, *sk'*)
4. *pk* ← ***HKG***(*sk*)
5. *pk'* ← ***HKG'***(*sk'*)
6. *PK* ← (*pk*, *pk'*)
7. Return (*PK*, *SK*)

- ***Encap***(*PK*)
1. Pick random $c \in \boldsymbol{L}$ and witness w
2. *K* ← ***Pub***(*pk, w*)
3. *tag* ← c
4. π ← ***Pub'***(*pk'*, *tag*, w)
5. *C* ← (*c*, π)
6. Return (*C, K*)

- ***Decap***(*sk*, *C*)
1. (c,π) ← *C*
2. tag ← *c*
3. If $\boldsymbol{H'}_{sk'}(tag,c) \neq \pi$ reject
4. Return $K \leftarrow \boldsymbol{H}_{sk}(c)$

- Consider Game 3-like situation
  - Adversary ***A*** is given the challenge $C^* = (c^*, \pi^*)$ and K^*_b, where $c^* \in \boldsymbol{U}-\boldsymbol{L}$, $\pi^* = \boldsymbol{H'}_{sk'}(tag^*=c^*, c^*)$, and $K^*_1 = \boldsymbol{H}(c^*)$
- Even if ***A*** submits a dec. query (c, π) s.t. $c \in \boldsymbol{U}-\boldsymbol{L}$, *it is almost always rejected* because
  - If $c = c^*$, then $\pi \neq \pi^*$ must hold, and the validity check cannot hold
  - If $c \neq c^*$, then **universal-2** of **HPS'** guarantees that it is unlikely that $\boldsymbol{H'}_{sk'}(tag=c,c) = \pi$ holds
    - Finding such π is statistically hard

How Universal-2 Help?

- ***KKG***:

1. *sk* ← random
2. *sk'* ← random
3. *SK* ← (*sk*, *sk'*)
4. *pk* ← ***HKG***(*sk*)
5. *pk'* ← ***HKG'***(*sk'*)
6. *PK* ← (*pk*, *pk'*)
7. Return (*PK*, *SK*)

- ***Encap***(*PK*)

1. Pick random $c \in \boldsymbol{L}$ and witness w
2. *K* ← ***Pub***(*pk, w*)
3. *tag* ← c
4. π ← ***Pub'***(*pk'*, *tag*, w)
5. *C* ← (*c*, π)
6. Return (*C, K*)

- ***Decap***(*sk*, *C*)

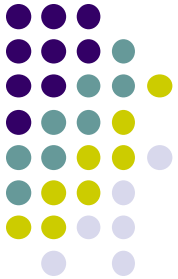
1. (*c*,π) ← *C*
2. tag ← *c*
3. If $\boldsymbol{H'}_{sk'}(tag,c) \neq \pi$ reject
4. Return *K* ← $\boldsymbol{H}_{sk}(c)$

- Hence, we can add a game in which all dec. queries *C* = (*c*, π) s.t. $c \in \boldsymbol{U}-\boldsymbol{L}$ is rejected, before we use **smoothness** of **HPS**

➔ **Universal-2** of **HPS'** ensures that the difference of Pr[$b_A = b$] before/after this game is neg.

➔ Since dec. queries *C* = (*c*, π) with invalid $c \in \boldsymbol{U}-\boldsymbol{L}$ no longer leaks the info. of *sk*, we can use **smoothness** of **HPS** to say that K^*_1 is random

Analogy between Naor-Yung and HPS-Based KEM



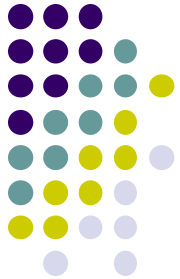
- NY-construction
  - Two key pairs of CPA PKE
  - Message finally hidden by the CPA security of PKE $\boldsymbol{\pi_1}$
  - Invalid dec. queries handled by sk_2 of PKE $\boldsymbol{\pi_2}$ and simulation-soundness of NIZK proof **P**
  - $C = (c_1, c_2, \pi)$

analogy

- HPS-based KEM
  - Two HPSes
  - Session-key random by smoothness of **HPS**
  - Invalid dec. queries handled by Universal-2 of **HPS'**
  - $C = (c, \pi)$

We can in fact view this part as a "proof"-part of a NY-ciphertext

Other Remarks on HPS



- [CS02] showed several other instantiations of SMPs and HPSes
  - QR assumption [GM84], DCR assumption [Pai99]

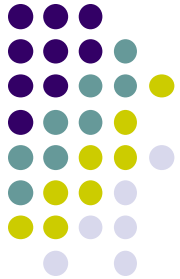
- There are non-Cramer-Shoup-type formalizations of HPS whose purpose is not CCA secure KEM, but other crypto. primitives & protocols
  - Key exchange [KV09]
  - Oblivious transfer [HK12]
  - …

  - For more information, see recent papers on HPS
    - E.g. Benhamouda et al.@PKC'18 [BBDQ18]

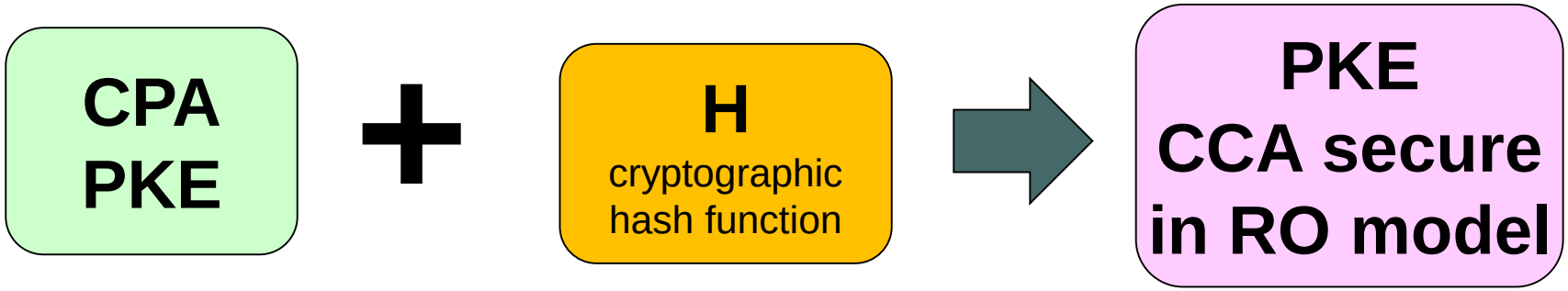
Part 1 Outline

- Naor-Yung construction
- KEM & Hybrid Encryption
- Hash proof systems
- Fujisaki-Okamoto construction

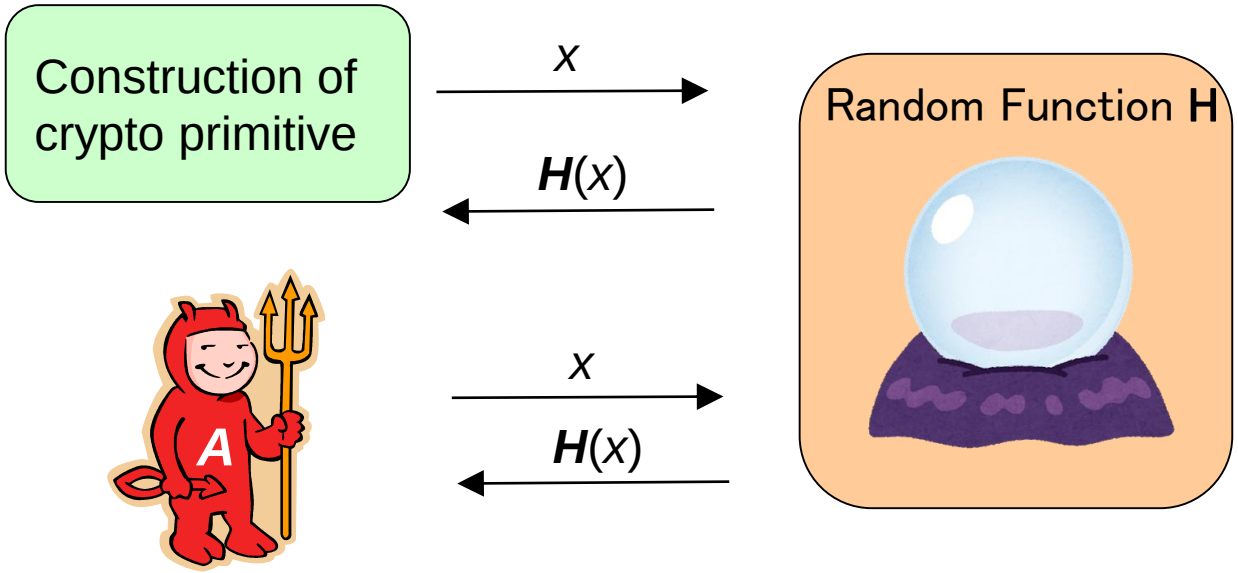
Fujisaki-Okamoto Constructions [FO99a,FO99b,FO13]



- Transformations for converting any CPA secure PKE into one achieving IND-CCA security in the random oracle (RO) model [BR93]
  - Simple, powerful, and useful

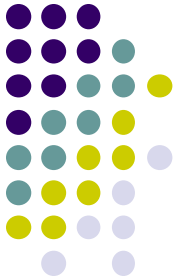


Random Oracle Model [BR93]



- Idealization of a cryptographic hash function
- Very useful for constructing systems with practical efficiency
- Some inconsistencies to the standard model are known (e.g. [CGH98]), but still widely-used

Fujisaki-Okamoto Constructions

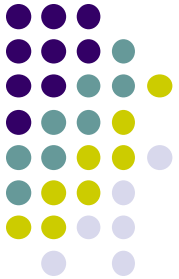


- There are 3 versions by the authors themselves
  - PKC'99 [FO99a]
    - Simplest, applicable to IND-CPA PKE
  - CRYPTO'99 [FO99b]
    - Slightly more complicated, adopting hybrid encryption, applicable to any one-way CPA PKE
  - JoC'13 [FO13]
    - Slight extension of CRYPTO'99

- The transformations are applicable to not only PKE but other advanced form of encryption schemes (such as IBE, ABE, etc.)

- On the other hand, its proof is not so simple (in my opinion)

Fujisaki-Okamoto Construction (PKC'99 ver.)



- Building blocks:
  - PKE $\boldsymbol{\pi}$ = (***KG***, ***Enc***, ***Dec***) s.t. Randomness space of ***Enc*** is $\{0,1\}^k$
  - Cryptographic hash function ***H*** : $\{0,1\}^*$ ➔ $\{0,1\}^k$ (modeled as RO)

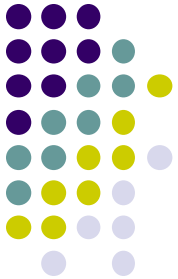
- FO construction: $\boldsymbol{\Pi_{FO}}$ = ($\boldsymbol{KG_{FO}}$, $\boldsymbol{Enc_{FO}}$, $\boldsymbol{Dec_{FO}}$)

- $\boldsymbol{KG_{FO}}$:
  1. (*pk*, *sk*) ← ***KG***
  2. *PK* ← (*pk*, ***H***)
  3. Return (*PK*, *sk*)

- $\boldsymbol{Enc_{FO}}(PK, m)$
  1. $r \leftarrow \{0,1\}^k$
  2. $R \leftarrow \boldsymbol{H}(r||m)$
  3. $c \leftarrow \boldsymbol{Enc}(pk, r||m; R)$
  4. Return *c*

- $\boldsymbol{Dec_{FO}}(sk, c)$
  1. $x \leftarrow \boldsymbol{Dec}(sk, c)$
  2. If $x = \perp$ then return $\perp$
  3. Parse *x* as *r*||*m*
  4. If $\boldsymbol{Enc}(pk, r||m; R) = c$ then return *m* else $\perp$

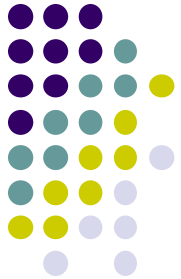
Several Variants Transforming "weak encryption" into CCA



- Bellare-Rogaway [BR93]
- OAEP [BR94]
  - There are also several variants of OAEP
- REACT [OP01]
- GEM [CHJ+02]
- Dent [Den03]
  - KEM-version of FO

- Recently, there are also several results on constructions in the Quantum RO model (where an adversary may have quantum-access to the RO)
  - [TU16], [HHK17], [SXY18], [JZC+18]

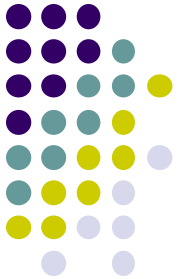
CCA Security of Fujisaki-Okamoto



- Thm:
  - Assume the underlying PKE π is **IND-CPA** and has "large ciphertext entropy".
    Then, the FO construction π_{FO} is **IND-CCA in the RO model**

- In the original paper, the above theorem is proved via the notion of "Plaintext-Awareness" [BDPR98]

Large Ciphertext Entropy



- Informally, it requires the output of ***Enc*** has large entropy (over the choice of randomness) s.t. no particular value is output too frequently
- Formally,

$$\mathbf{max}_{pk,\, m,\, c'} \Pr_{c \leftarrow \mathbf{Enc}(pk,\, m)}[c' = c] = \text{neg}.$$

  - The original paper [FO99a] calls this property (in a slightly different form) the γ-uniformity

  - Any natural construction of PKE satisfies this notion
    - e.g. ElGamal $C = (g^r, (pk)^r \cdot m)$
  - Even if a PKE does not satisfy the above, we can generically transform it to satisfy the above by attaching randomness to CT
    - $c' = c||r$

Intuition for Security (1/2)

$c = \boldsymbol{Enc}(pk, r||m\ ;\ \boldsymbol{H}(r||m))$

- Plaintext m is encrypted by a CPA secure PKE $\boldsymbol{\pi}$, but its randomness is derived from m (Circularity! ☹)
- Since $\boldsymbol{H}$ is an RO, unless adversary $\boldsymbol{A}$ queries $(r^*||m_b)$ to $\boldsymbol{H}$, $R^* = \boldsymbol{H}(r^*||m_b)$ should behave like a random string
- ➔If there is no dec. query, we can expect that $r^*||m_b$ encrypted in the challenge CT $c^* = \boldsymbol{Enc}(pk, r^*||m_b; R^*)$ is hidden due to the **CPA security** of $\boldsymbol{\pi}$

Intuition for Security (2/2)

$c = \boldsymbol{Enc}(pk, r||m\ ;\ \boldsymbol{H}(r||m))$

- Also, due to the validity check by re-encryption in $\boldsymbol{Dec_{FO}}$ and the large ciphertext entropy, ***A*** can't create a ciphertext $c = \boldsymbol{Enc}(pk, r||m; \boldsymbol{H}(r||m))$, without ever querying $(r||m)$ to the RO ***H*** (plaintext-awareness)
  - If $(r||m)$ is known to ***A***, the dec. result m doesn't give ***A*** any new info.
  - This property ensures that sk need not be used to answer dec. queries

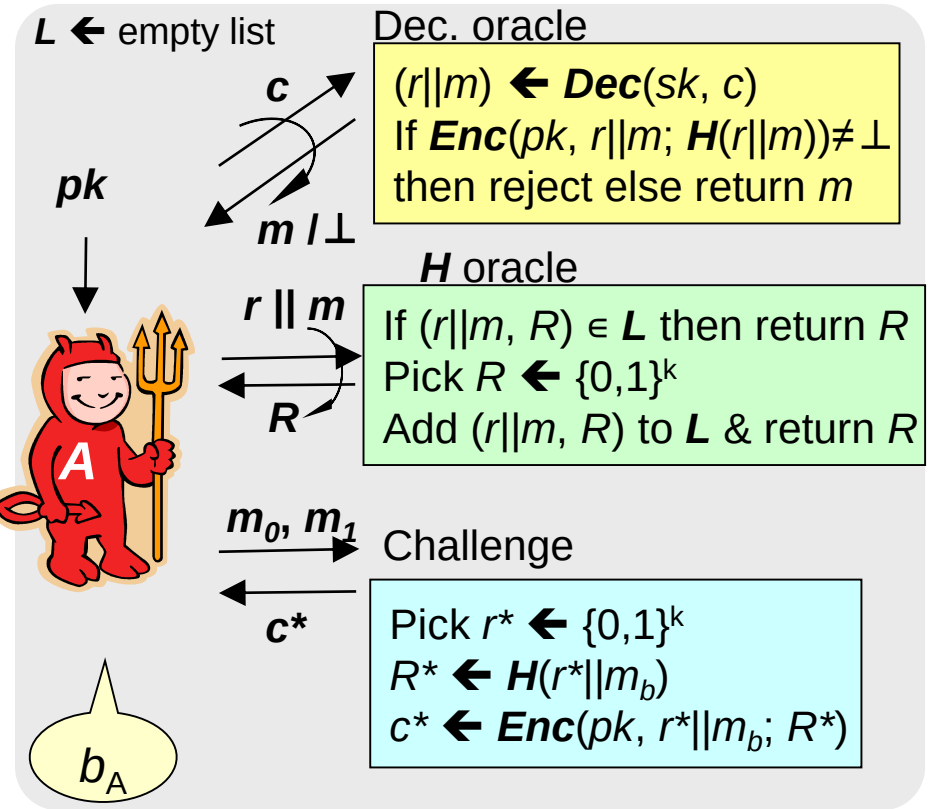
- The last concern: ***A*** may somehow submit $(r^*||m_b)$ to the RO ***H*** after given c^*
  - Since the answer to the query leaks R^* used to generate c^*, we can no longer say $(r^*||m_b)$ is hidden from ***A***'s view

- But, extracting r^* from c^* is possible only after breaking the **IND-CPA** security of $\boldsymbol{\pi}$

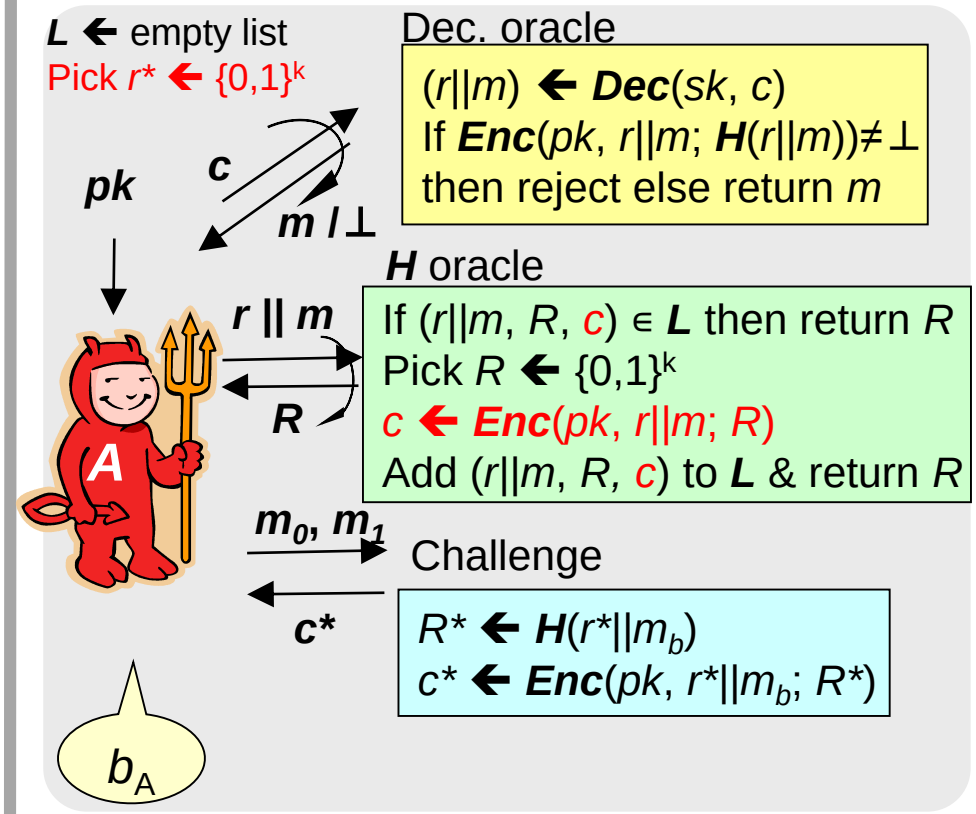
  ➔ We can bound the probability that ***A*** makes such bad ***H***-query by the **CPA security** of $\boldsymbol{\pi}$

Security Proof of FO (1/7)

- **Game 1** (CCA in RO game)



- **Game 2**

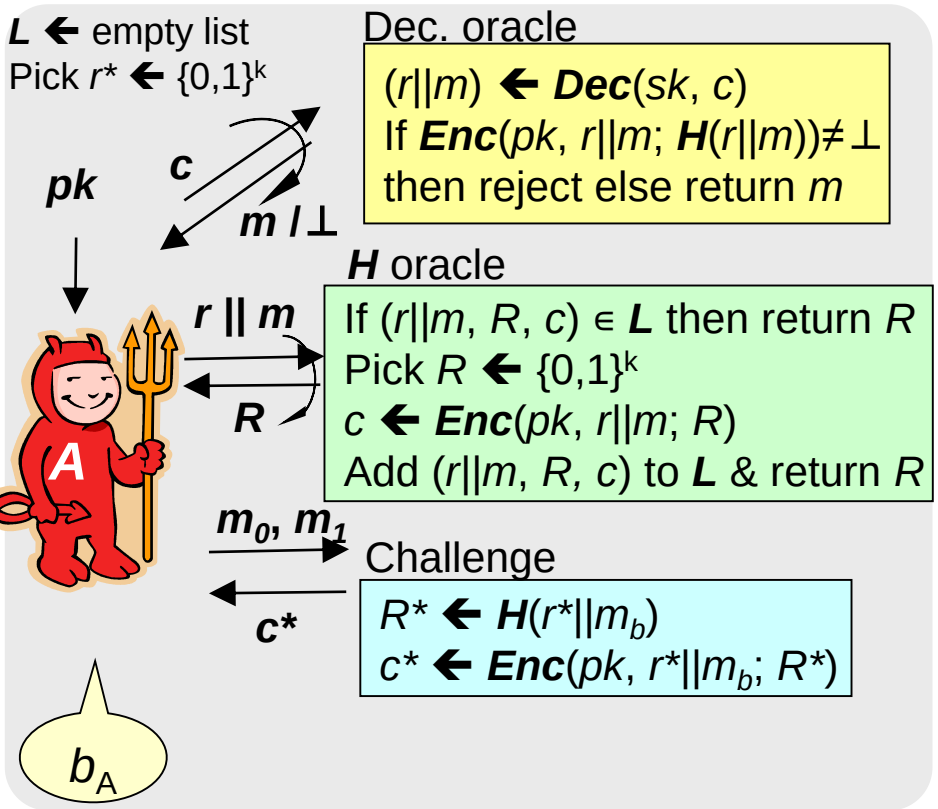


No difference in the behaviors of oracles in *A*'s viewpoint

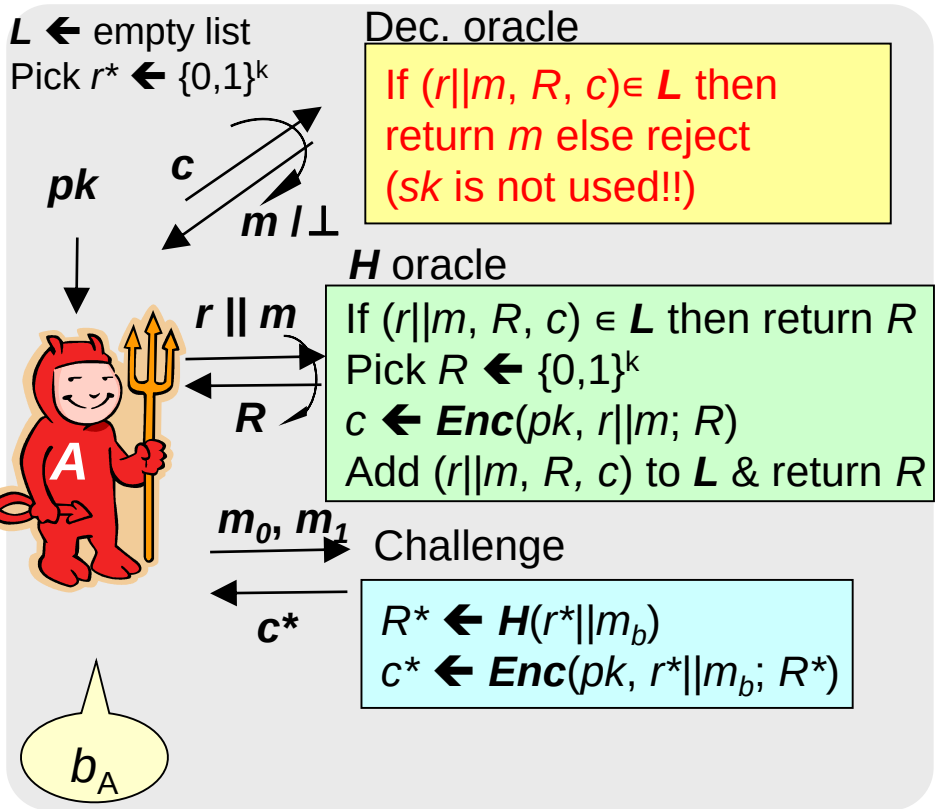
$\Pr[b_A = b \text{ in } \textbf{Game 1}] = \Pr[b_A = b \text{ in } \textbf{Game 2}]$

Security Proof of FO (2/7)

- **Game 2**



- **Game 3**



Pr[$b_A = b$ in **Game 2**]

Pr[$b_A = b$ in **Game 3**]

Security Proof of FO (2/7)

- **Game 2**

L ← empty list
Pick r^* ← $\{0,1\}^k$

Dec. oracle
$(r||m)$ ← ***Dec***(sk, c)
If ***Enc***$(pk, r||m; \boldsymbol{H}(r||m)) \neq \perp$

pk ***c***

b_A

- **Game 3**

L ← empty list
Pick r^* ← $\{0,1\}^k$

Dec. oracle
If $(r||m, R, c) \in$ ***L*** then return m else reject

c

urn *R*

urn *R*

)

Game 2 and **Game 3** are identical unless ***A*** submits a dec. query c s.t. $(r||m, R, c)$ is not in ***L*** but it is not rejected **in Game 2**

➔ Since corresponding $(r||m)$ has not been queried to ***H*** oracle,
$R = \boldsymbol{H}(r||m)$ is completely hidden from ***A***'s view at the point of the query

➔ Since R is uniform random, it is statistically hard to find c s.t.
$c = \boldsymbol{Enc}(pk, (r||m); R)$ due to the large ciphertext entropy of PKE ***π***

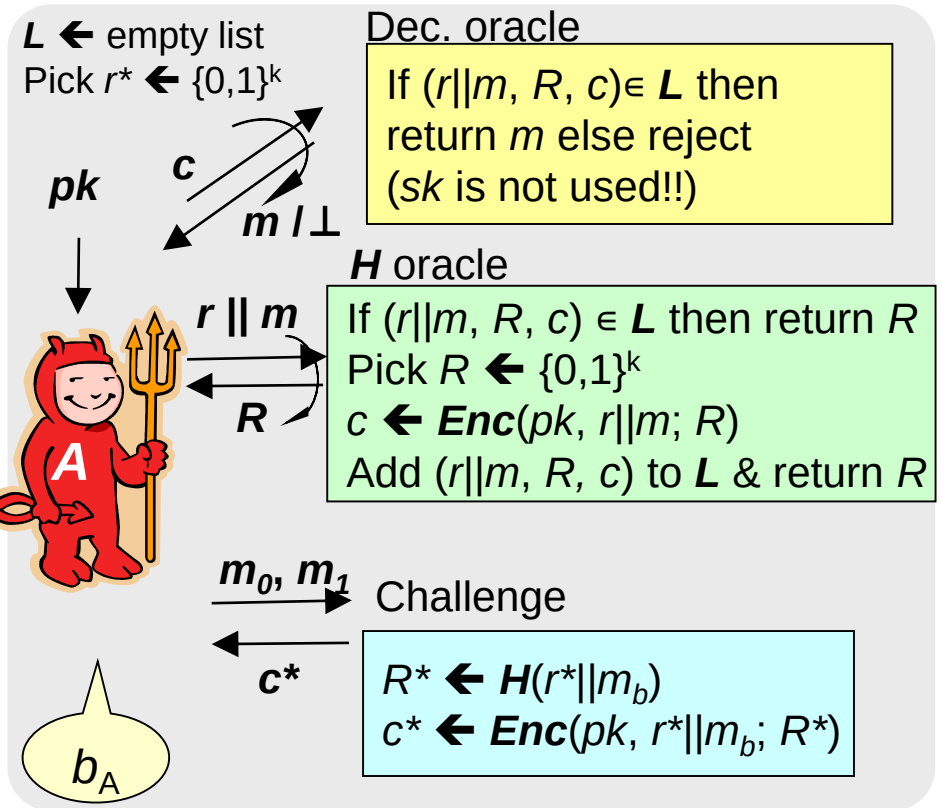
➔ Probability that ***A*** submits such a bad ***H***-query c is negligible

Due to large ciphertext entropy of PKE *π*

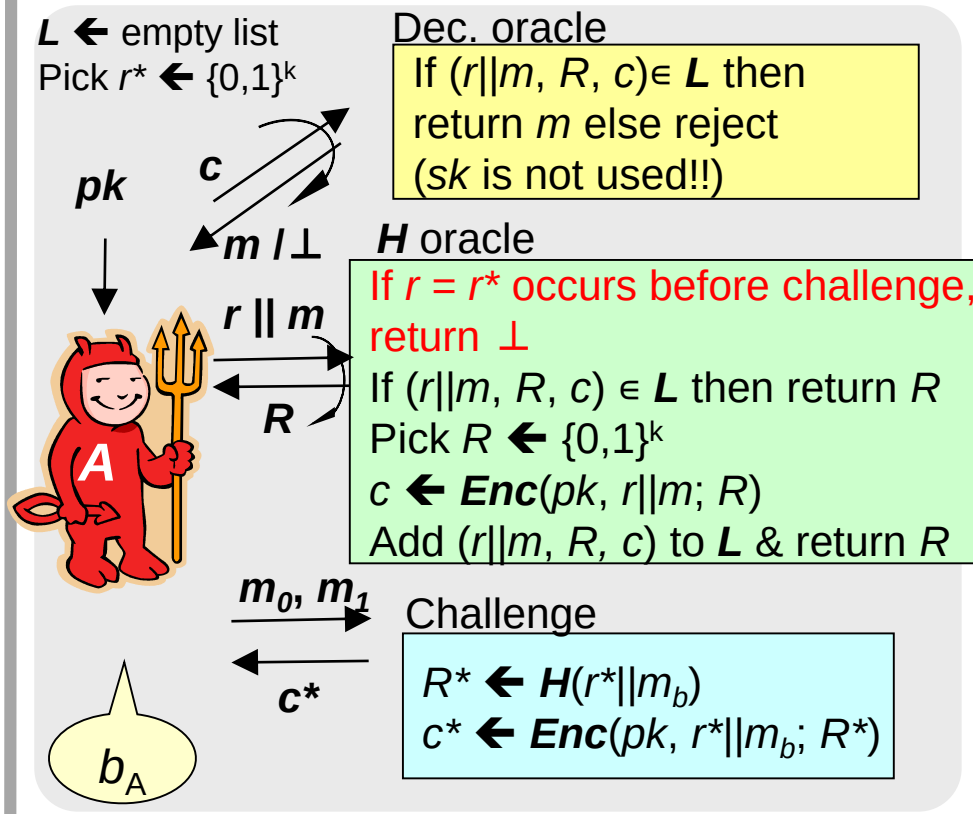
$\Pr[b_A = b$ in **Game 2**$]$ ≈ $\Pr[b_A = b$ in **Game 3**$]$

Security Proof of FO (3/7)

- **Game 3**



- **Game 4**

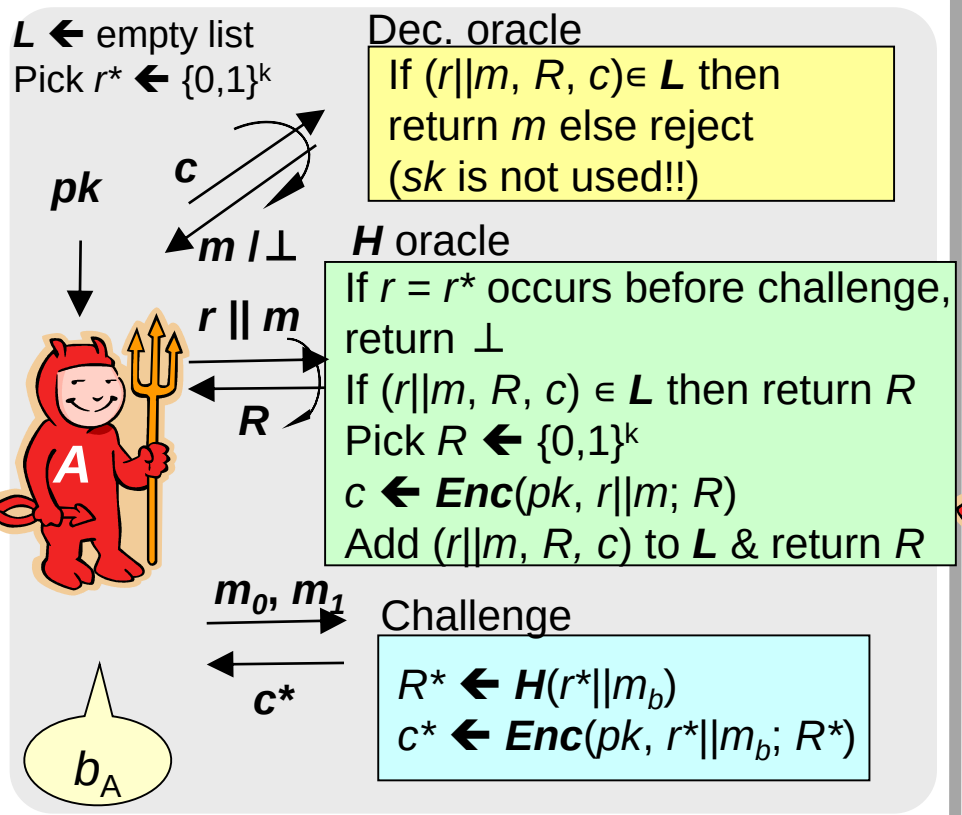


r* is info.-theoretically hard to guess before challenge

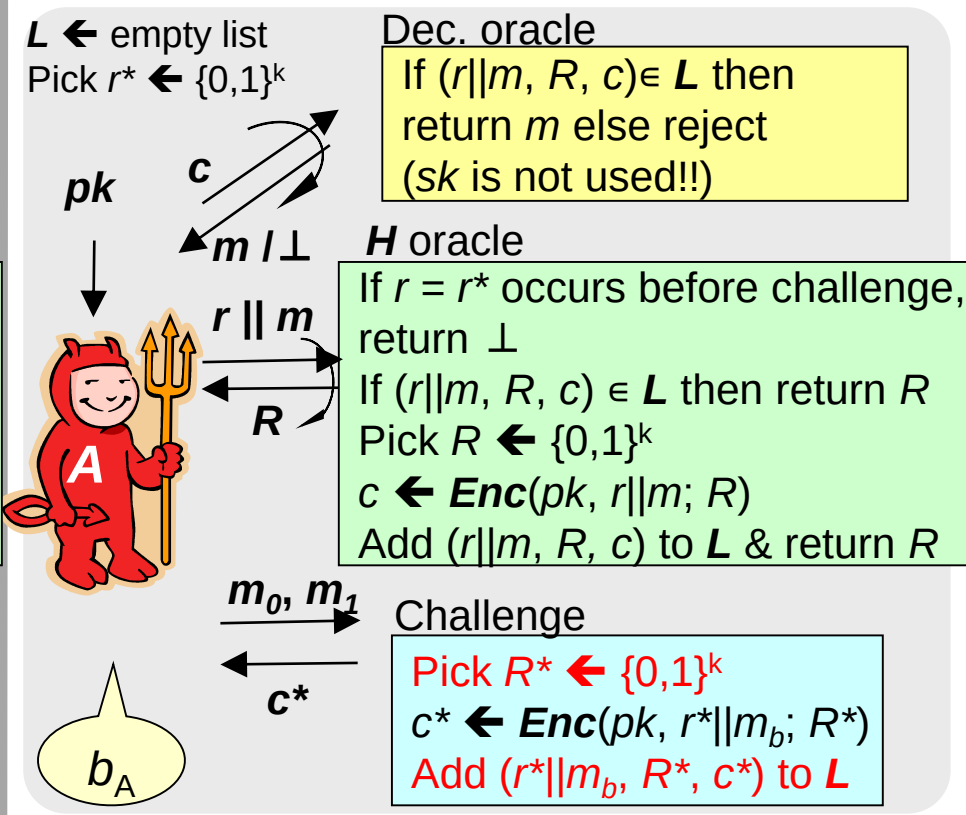
$\Pr[b_A = b \text{ in } \textbf{Game 3}] \approx \Pr[b_A = b \text{ in } \textbf{Game 4}]$

Security Proof of FO (4/7)

- **Game 4**



- **Game 5**

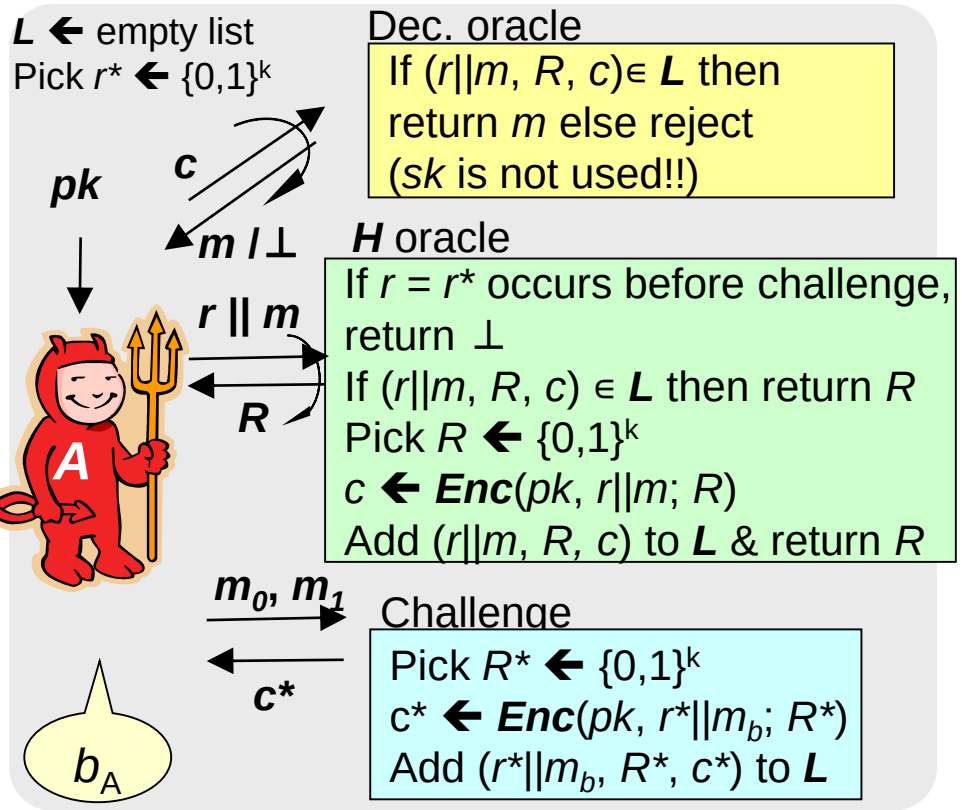


The distribution of *A*'s view is identical

$\Pr[b_A = b \text{ in } \textbf{Game 4}] = \Pr[b_A = b \text{ in } \textbf{Game 5}]$

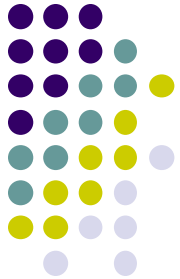
Security Proof of FO (5/7)

- **Game 5**



- We want to show
  $|\Pr[b_A = b \text{ in } \textbf{Game 5}] - 1/2| = \text{neg}$.
- It seems **CPA security** of PKE $\boldsymbol{\pi}$ straightforwardly implies this
- However, the reduction (CPA attacker of $\boldsymbol{\pi}$) can' t properly handle ***H*** queries containing r^* made after challenge, since it doesn't know R^*
- *Intuition again:*
  1. *If **A** doesn't make **H**-query with r*, then **CPA security** of **π** helps*
  2. *To make **H**-query with r*, **A** has to break **CPA** of **π** in the first place*

How to Bound $|\Pr[b_A = b \text{ in Game 5}] - 1/2|$? (6/7)

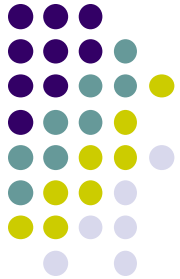


- For **Game 5,** we define the event:
  - $\boldsymbol{S_5}$: $b_A = b$ occurs
  - $\boldsymbol{Q_5}$: ***A*** submits an ***H***-query containing ***r**** after challenge
- We can decompose $|\Pr[b_A = b \text{ in } \mathbf{Game\ 5}] - 1/2|$ as

$$= |\Pr[\mathbf{S_5}] - 1/2|$$

$$\leqq |\Pr[\mathbf{S_5} \wedge \neg\mathbf{Q_5}] + (1/2)\cdot\Pr[\mathbf{Q_5}] - 1/2| + (1/2)\cdot\Pr[\mathbf{Q_5}]$$

How to Bound $|\Pr[b_A = b \text{ in Game 5}] - 1/2|$? (6/7)



- For **Game 5,** we define the event:
  - $\boldsymbol{S_5}$: $b_A = b$ occurs
  - $\boldsymbol{Q_5}$: ***A*** submits an ***H***-query containing ***r**** after challenge
- We can decompose $|\Pr[b_A = b \text{ in } \mathbf{Game\ 5}] - 1/2|$ as

$$= |\Pr[\mathbf{S_5}] - 1/2|$$

$$\leqq |\Pr[\mathbf{S_5} \wedge \neg\mathbf{Q_5}] + (1/2)\cdot\Pr[\mathbf{Q_5}] - 1/2| + (1/2)\cdot\Pr[\mathbf{Q_5}]$$

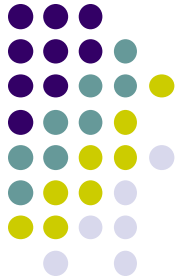
Fact: For any events **S** and **E**, we have

$$|\Pr[\mathbf{S}] - 1/2| \leqq |\Pr[\mathbf{S} \wedge \neg\mathbf{E}] + (1/2)\cdot\Pr[\mathbf{E}] - 1/2| + (1/2)\cdot\Pr[\mathbf{E}]$$

Proof: If $\Pr[\mathbf{E}] = 0$ then trivially true. Otherwise,

$$\begin{aligned}|\Pr[\mathbf{S}] - 1/2| &= |\Pr[\mathbf{S} \wedge \neg\mathbf{E}] + \Pr[\mathbf{S} \mid \mathbf{E}]\cdot\Pr[\mathbf{E}] - 1/2| \\ &= |\Pr[\mathbf{S} \wedge \neg\mathbf{E}] + (1/2)\cdot\Pr[\mathbf{E}] - (1/2)\cdot\Pr[\mathbf{E}] + \Pr[\mathbf{S} \mid \mathbf{E}]\cdot P[\mathbf{E}] - 1/2| \\ &\leqq |\Pr[\mathbf{S} \wedge \neg\mathbf{E}] + (1/2)\cdot\Pr[\mathbf{E}] - 1/2| + |\Pr[\mathbf{S}|\mathbf{E}] - 1/2|\cdot\Pr[\mathbf{E}] \\ &\leqq |\Pr[\mathbf{S} \wedge \neg\mathbf{E}] + (1/2)\cdot\Pr[\mathbf{E}] - 1/2| + (1/2)\cdot\Pr[\mathbf{E}]\end{aligned}$$

How to Bound $|\Pr[b_A = b \text{ in Game 5}] - 1/2|$? (6/7)



- For **Game 5,** we define the event:
  - $\boldsymbol{S_5}$: $b_A = b$ occurs
  - $\boldsymbol{Q_5}$: ***A*** submits an ***H***-query containing ***r**** after challenge
- We can decompose $|\Pr[b_A = b \text{ in } \mathbf{Game\ 5}] - 1/2|$ as

$$= |\Pr[\mathbf{S_5}] - 1/2|$$

$$\leqq |\Pr[\mathbf{S_5} \wedge \neg \mathbf{Q_5}] + (1/2) \cdot \Pr[\mathbf{Q_5}] - 1/2| + (1/2) \cdot \Pr[\mathbf{Q_5}]$$

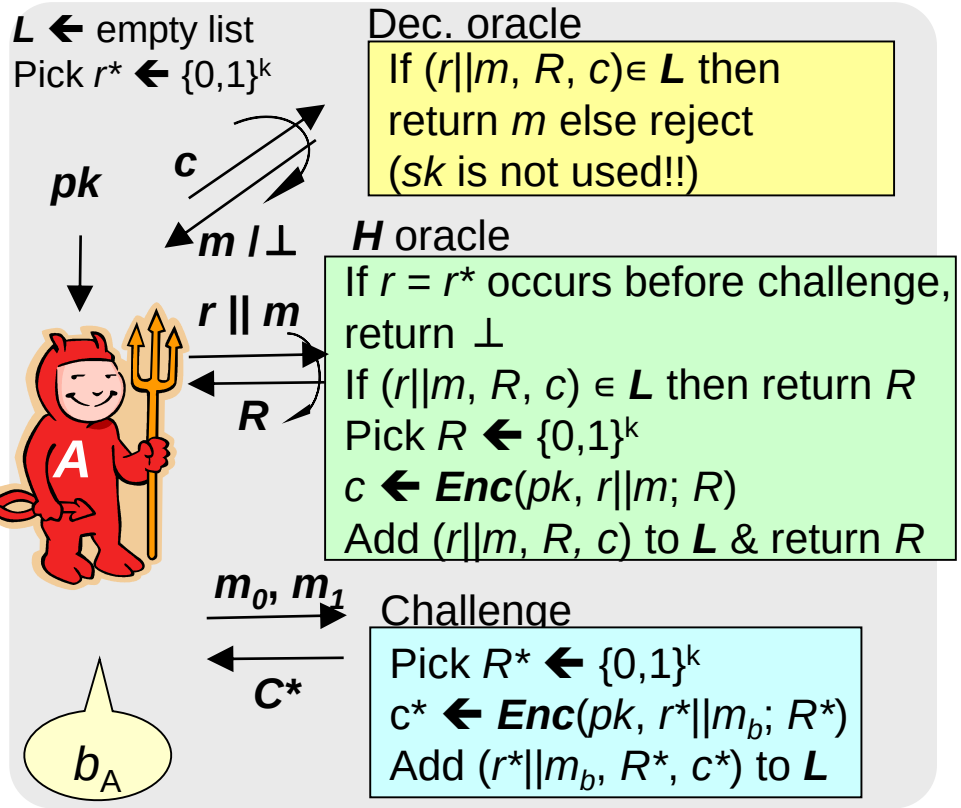
≒ ***A***'s advantage in **Game 5** w/o making ***H***-query containing r* after challenge

Probability that ***A*** submits an ***H***-query containing *r** after challenge

- We can bound both terms to be neg. due to **CPA security** of PKE $\boldsymbol{\pi}$, because the reduction (CPA attacker) need not know *R** if $\mathbf{Q_5}$ does not occur
  - If $\mathbf{Q_5}$ occurs, the reduction for the left just outputs a random bit

Security Proof of FO (7/7)

- **Game 5**



- In summary,
  $\Pr[b_A = b \text{ in } \textbf{Game 1}]$
  $\approx \ldots \approx$
  $\Pr[b_A = b \text{ in } \textbf{Game 5}]$,
  and
  $|\Pr[b_A = b \text{ in } \textbf{Game 5}] - 1/2| = \text{neg}$.

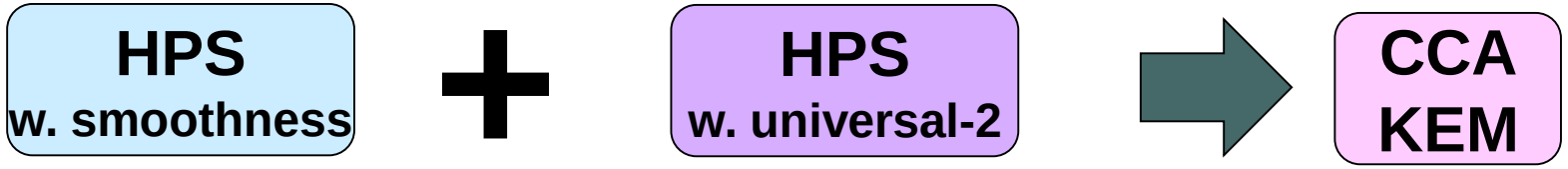
➔ ***A***'s CCA advantage in RO model =
$|\Pr[b_A = b \text{ in } \textbf{Game 1}] - 1/2| = \text{neg}$.

Summary of Part 1

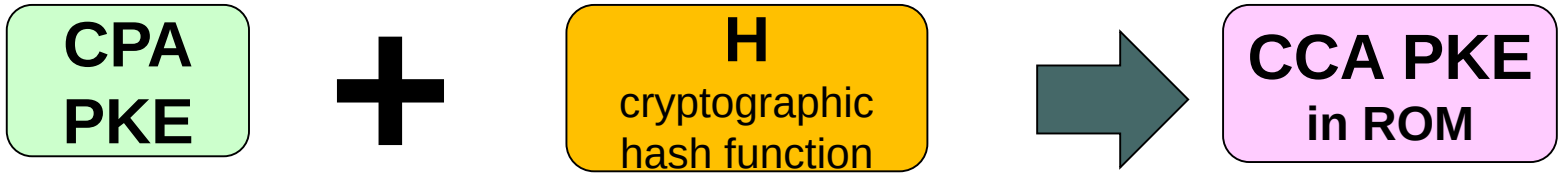
- Naor-Yung

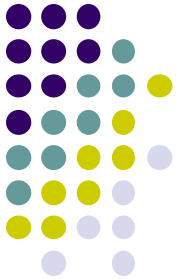
- Hybrid Encryption

- HPS-Based KEM



- Fujisaki-Okamoto





Part 2:
Brief Survey of Recent Topics on CCA Secure PKE

Outline

- Generic assumptions
- Tight security
- Post-quantum security

Fundamental Question on CCA PKE

Can we construct CCA PKE only from CPA PKE?

CPA PKE → **??** → **CCA PKE**

- To ultimately solve this open problem, we want to collect insights by considering what "**generic assumptions**" are sufficient to achieve CCA PKE

≒ cryptographic primitives

??? → **CCA PKE**

Generic Assumptions that Imply CCA PKE

Q. Which primitive(s) implies CCA secure PKE/KEM ??

??? → **CCA PKE/KEM**

[NY90,DDN91]
CPA PKE + NIZK

[CHK04,Kiltz06]
IBE (or TBE)

[PW08,RS09,KMO10,Wee10]
TDF w/ additional properties

[HLW12]
Detectable CCA PKE

[DG17,DGHM18]
Hash Enc.

[HO12]
Hom. PKE w/ additional properties

[HO13]
Lossy PKE w. large PT space

[MH14a]
CPA PKE + Point Obf.

[MH14b]
CPA PKE + UCE

[KW18]
CPA PKE + Hinting PRG

[Dac14,MH16]
PKE satisfying sPA1 (for many keys) & weak simulatability

[SW14]
iO + OWF

[MH15]
Sender NCE + KDM SKE

[HK15]
1-bit PKE w/ circular security & reproducibility

One Possibility…

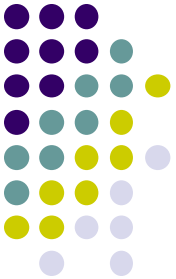
What if constructing CCA PKE from CPA PKE is ***impossible***?

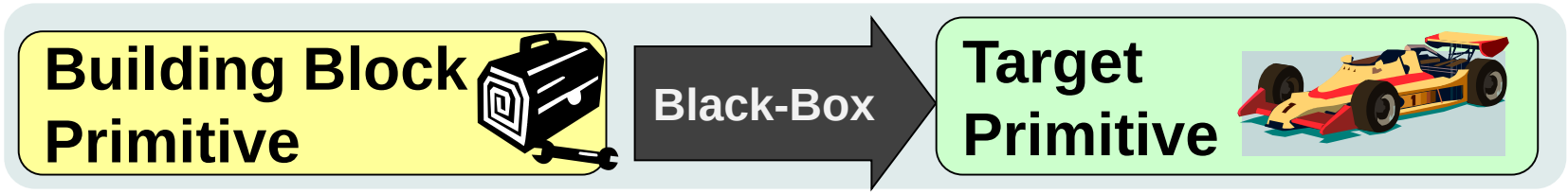
- Currently, we do not have techniques to completely rule out the possibility of constructing a **primitive A** from another **primitive B**

- However, if we focus only on **black-box** constructions, we can rule out the possibilities [IR89]

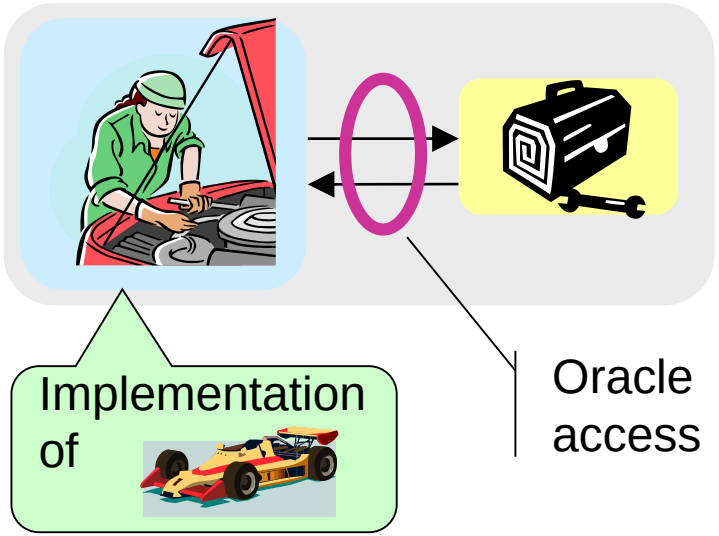
(Fully)

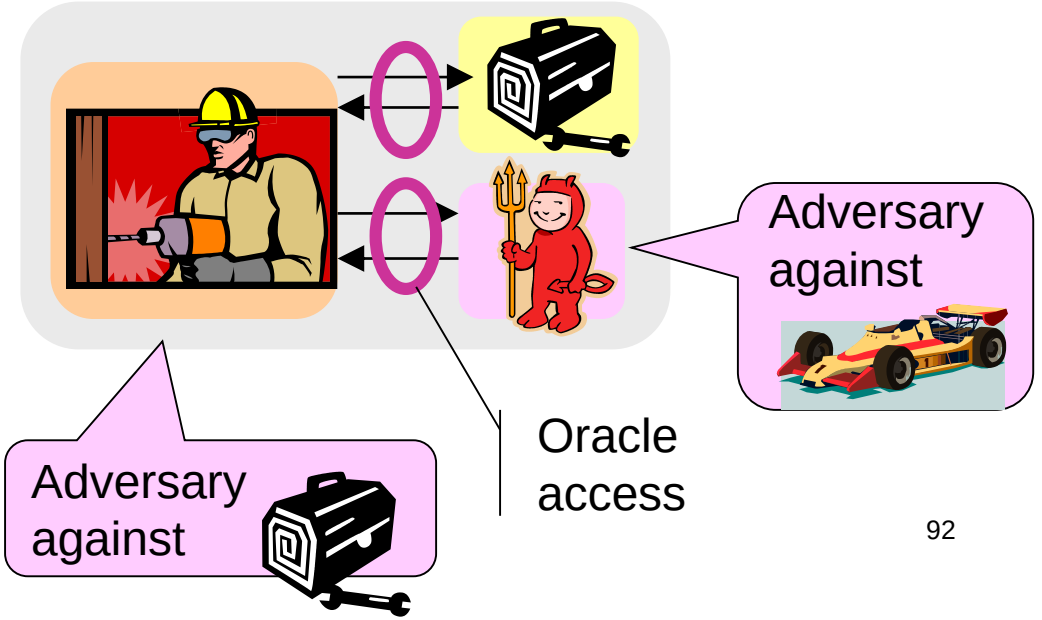
Black-Box Construction





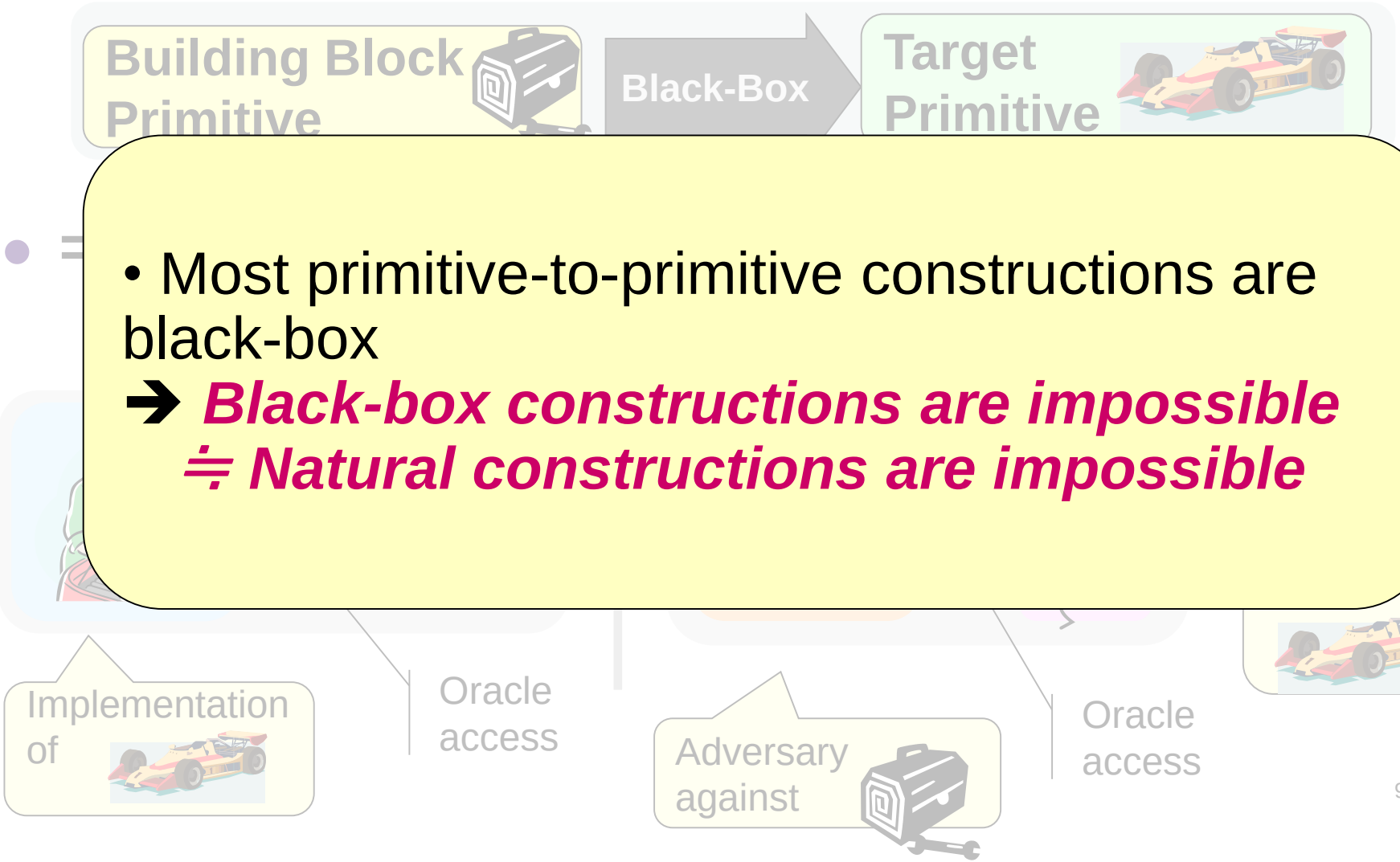
= **Construction** and **Reduction**
(security proof)





(Fully)

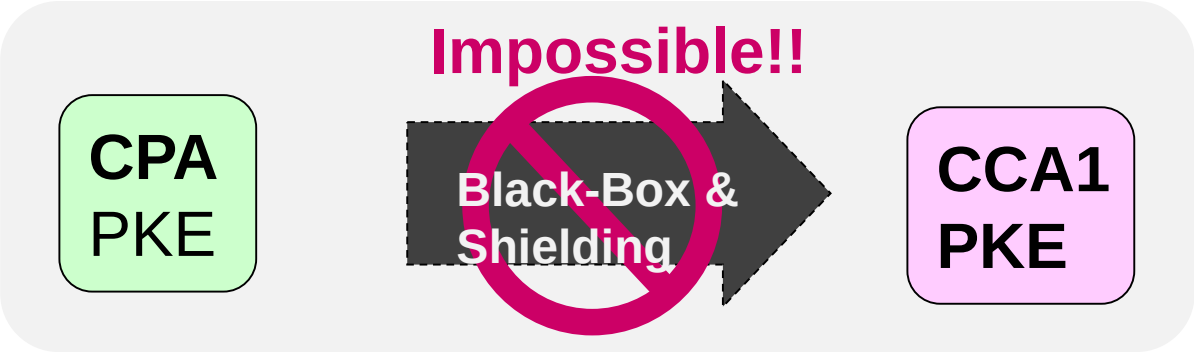
Black-Box Construction



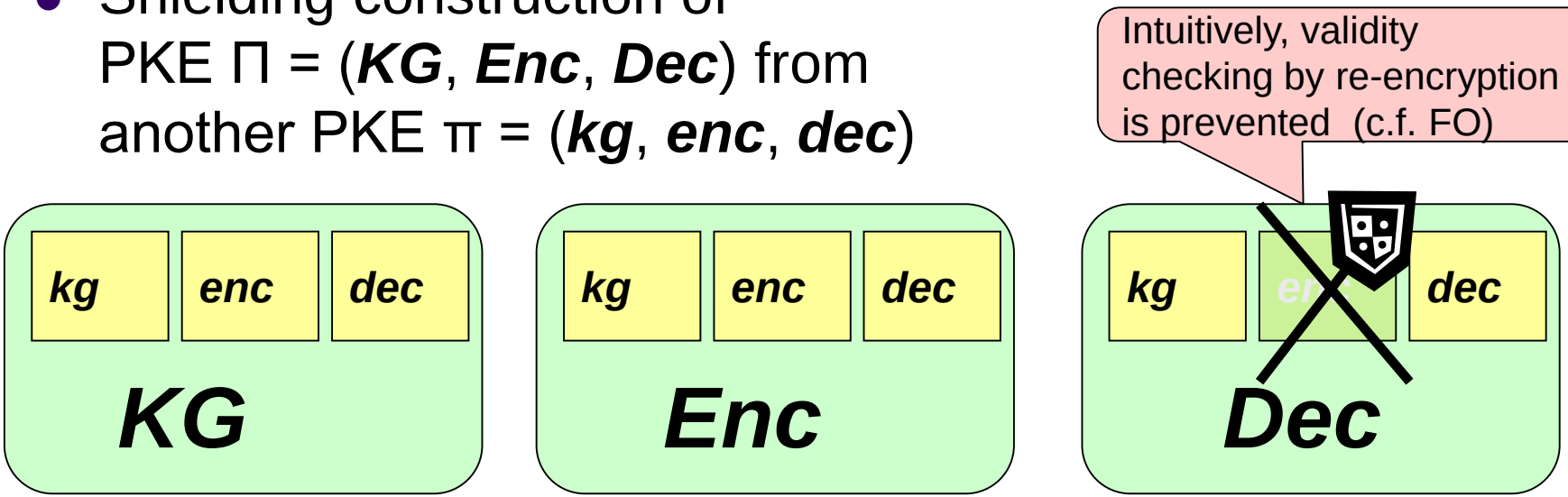
- Most primitive-to-primitive constructions are black-box

➔ ***Black-box constructions are impossible***
≒ Natural constructions are impossible

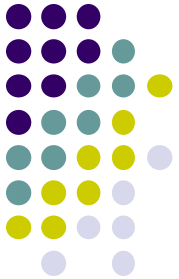
Negative Result [GMM07]



- Shielding construction of PKE Π = (***KG***, ***Enc***, ***Dec***) from another PKE π = (***kg***, ***enc***, ***dec***)



Known Methods that Bypass Black-Box Impossibilities



- (NIZK) Proof systems
- Garbled Circuits
- Obfuscation

These primitives treat the **descriptions** of underlying primitive

- Knowledge assumptions

E.g. Plaintext-awareness (Adversary is treated as non-black-box)

- Security properties that involve "functions" in security definitions
  (e.g. KDM security, leakage-resilience)

[MH15]: The construction is black-box, but the reduction has to treat the description of the underlying primitive as a KDM-query in KDM security game

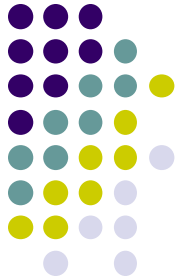
Open Problems

- Can we construct CCA PKE from CPA PKE?
- Can we weaken the assumptions of the known constructions?
- Can we strengthen the negative result of [GMM07] ?

Outline

- Generic assumptions
- Tight security
- Post-quantum security

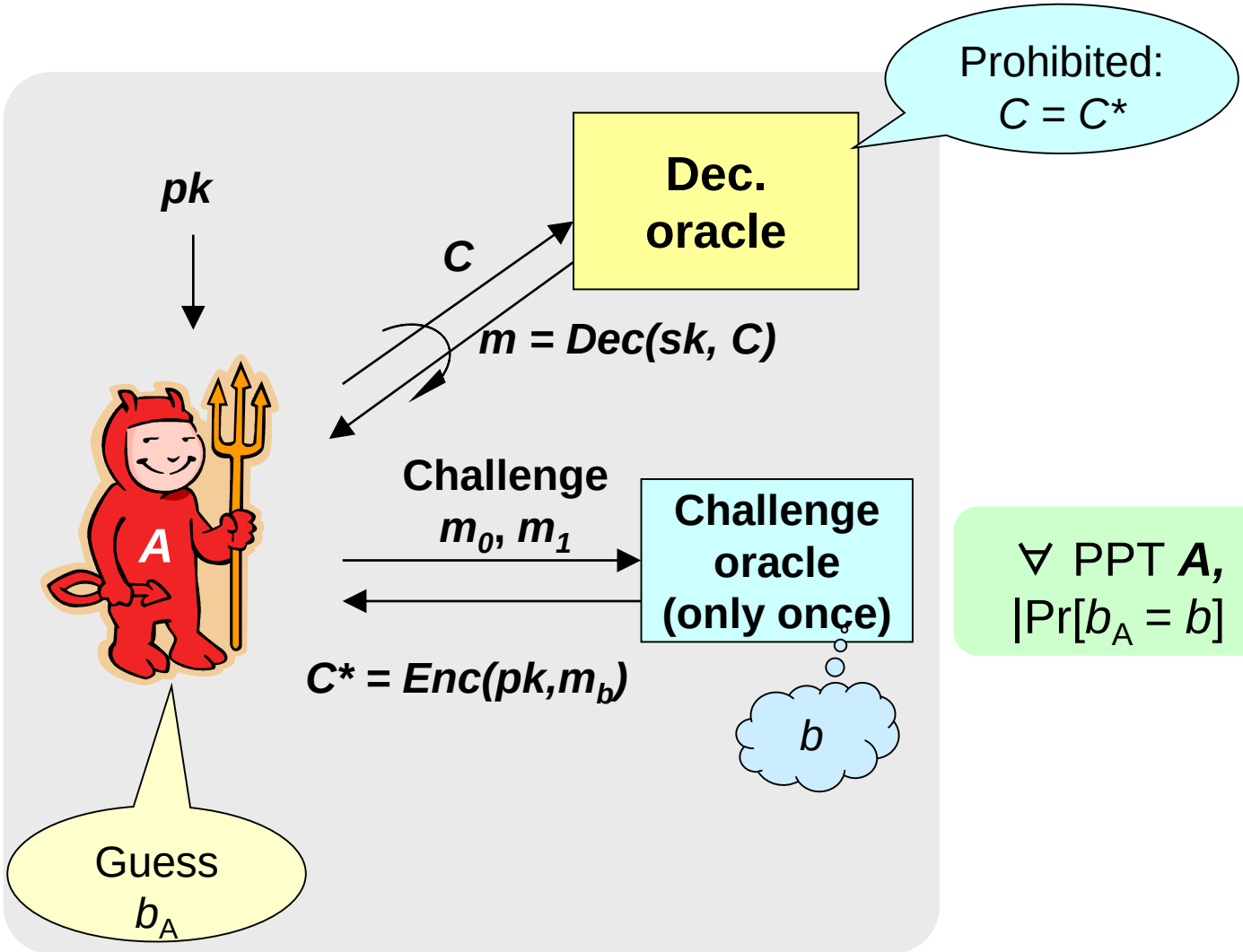
Multi-User/Challenge Security [BBM00]



- The standard security model of IND-CCA considers single-key and single challenge CT

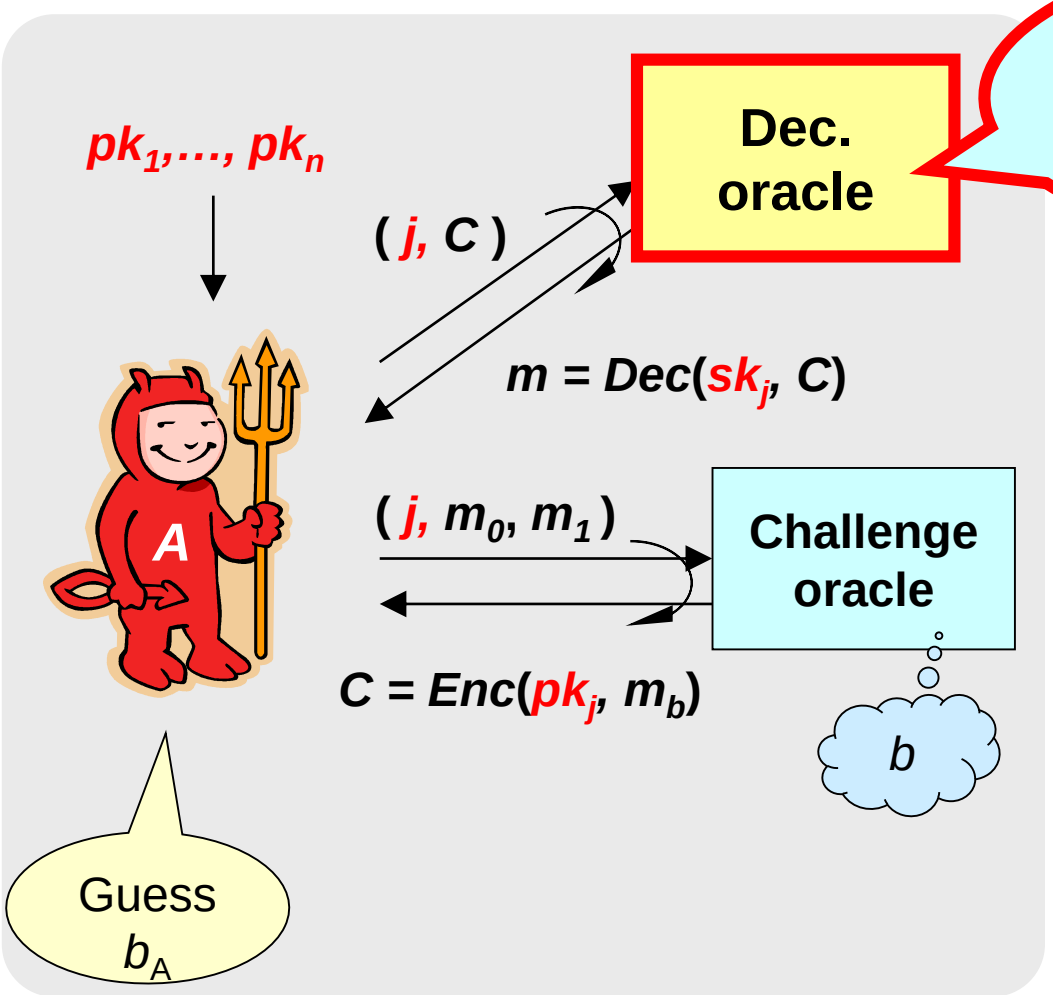
- Bellare et al. [BBM00] defined **multi-user/multi-challenge** version of CCA security

IND-CCA Security



∀ PPT ***A***,
$|\Pr[b_A = b] - 1/2| = \text{neg}.$

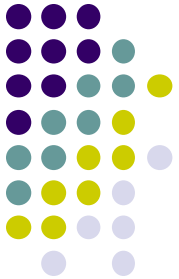
Multi-User/Multi-Challenge IND-CCA Security



Prohibited:
If C is returned as a response to challenge query with index j

$\forall$ PPT $\boldsymbol{A}$,
$|\Pr[b_A = b] - 1/2| = \text{neg}.$

Why We Care Multi-User/Challenge Setting?



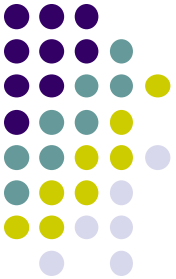
- Multi-user/challenge ver. is implied by single-user/challenge ver., but the "reduction loss" is proportional to **#users** and **#challenge**
  - $\forall$ $\boldsymbol{A_{multi}}$ attacking the $\boldsymbol{n}$-user/$\boldsymbol{q}$-challenge CCA,
    $\exists$ $\boldsymbol{B_{single}}$ attacking the single (i.e. standard) CCA, s.t.

$$\mathbf{Adv}(\boldsymbol{A_{multi}}) \leqq \boldsymbol{n} \cdot \boldsymbol{q} \cdot \mathbf{Adv}(\boldsymbol{B_{single}})$$

- The factors $\boldsymbol{n}$ and $\boldsymbol{q}$ could have huge effects in **concrete security**
  - Suppose $\boldsymbol{n} = 2^{32}$ and $\boldsymbol{q} = 2^{40}$ — Each of half the world population sends almost 1 ciphertext/day
  - In order to achieve 128-bit security in $\boldsymbol{n}$-user/$\boldsymbol{q}$-challenge setting, we need to use 200-bit secure scheme in the single setting

➔ Affects the parameter size

“Tight” Security



- In general, to prove that a PKE scheme is secure against adversaries $\boldsymbol{A_{cca}}$ that make $\boldsymbol{q_e}$-enc. (challenge) queries, and $\boldsymbol{q_d}$ dec. queries in the $\boldsymbol{n}$-user setting
  based on an underlying hard problem $\boldsymbol{P}$,
  we show a reduction $\boldsymbol{B_P}$ (solving $\boldsymbol{P}$) s.t.

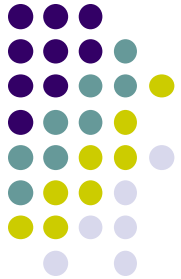
$$\boldsymbol{Adv}(\boldsymbol{A_{cca}}) \leqq F(\boldsymbol{q_e},\boldsymbol{q_d},\boldsymbol{n},\boldsymbol{\lambda}) \cdot \mathbf{Adv}(\boldsymbol{B_P}),$$

  where F is a polynomial and $\boldsymbol{\lambda}$ is a security parameter

Ideally, just a constant

- If $F(\boldsymbol{q_e},\boldsymbol{q_d},\boldsymbol{n},\boldsymbol{\lambda}) = F'(\boldsymbol{\lambda})$ (i.e. independent of $\boldsymbol{q_e},\boldsymbol{q_d},\boldsymbol{n}$), and $\boldsymbol{Time}(\boldsymbol{A_{cca}}) \approx \boldsymbol{Time}(\boldsymbol{B_P})$, we say that the reduction is **tight**, and PKE is often called **tightly CCA secure** based on $\boldsymbol{P}$

Tight Security



- The first tightly secure CCA secure PKE based on a simple “static” assumption (concretely, DLIN in bilinear groups) was achieved by Hofheinz and Jager [HJ12]
  - Based on NY construction, but the scheme is very inefficient

- After [HJ12], one of the recent trends is to construct a practical PKE scheme whose multi-user/challenge security is tightly reduced to a simple assumption

Tightly Secure Schemes

#group elements

Scheme	$\|pk\|$	$\|C\| - \|m\|$	Sec. loss	Assumption	Pairing?
[HJ12]	O(1)	O(λ)	O(1)	DLIN	yes
[ADKNO13]	O(1)	O(λ)	O(1)	DLIN	yes
[HKS15]	O(λ)	2	O(λ)	subgroup	yes
[LPJY15]	O(λ)	47	O(λ)	DLIN	yes
[AHY15]	O(λ)	12	O(λ)	DLIN	yes
[GCDCT16]	O(λ)	6k + 4	O(λ)	k-LIN (k≧1)	yes
[Hofheinz16]	2	60	O(λ)	SXDH	yes
[GHKW16]	2kλ	3k	O(λ)	k-LIN (k≧1)	no
[Hofheinz17]	2k(k + 5)	k + 4	O(λ)	k-LIN (k≧2)	yes
[Hofheinz17]	20	28	O(λ)	DCR	---
[GHK17]	6	3	O(λ)	DDH	no
[GHK17]	$k^2(k+1) + 4k$	k(k+2)	O(λ)	k-LIN (k≧1)	no

1-LIN = DDH, 2-LIN = DLIN

Information taken from [Hof17,GHK17]

Open Problem

- Is it possible to construct better schemes?
  - [GHK17] based on DDH seems as efficient as Cramer-Shoup [CS98]
  - There seems to be room for improvement for
    - constructions based on DCR
    - “truly” tight schemes (reduction loss O(1))
- Do random oracles help to improve efficiency?
  - Note:
    If RO is used, we have to also take into account #RO-queries of adversary
    ➔ ROs do not necessarily make the problem easier

Outline

- Generic assumptions
- Tight security
- Post-quantum security

Quantum Computer

- Quantum computer breaks many of widely-used cryptography [Shor94]
  - RSA, elliptic curve…
- Need **Post-quantum** cryptography
  - Lattice, Code, Isogeny, Hash-based…
  - NIST started standardization

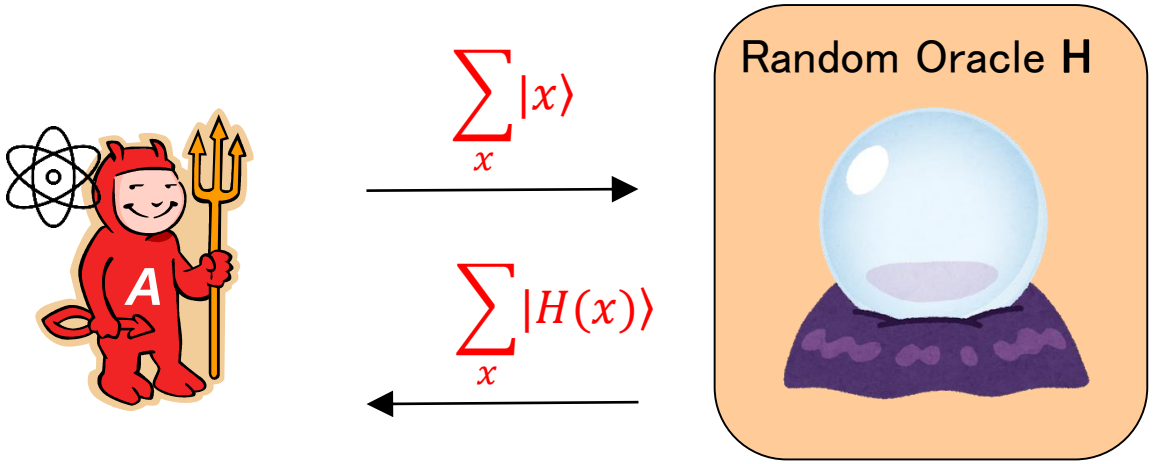
Q. Why is a quantum computer so awesome?

Because it can do a kind of parallel computation!

$$\sum_x |x\rangle \rightarrow \sum_x |F(x)\rangle$$

Based on EC'18 presentation slide by Takashi Yamakawa

Quantum Random Oracle (QRO) Model [BDF+11]



- In the real world, ROs are instantiated by a real hash functions

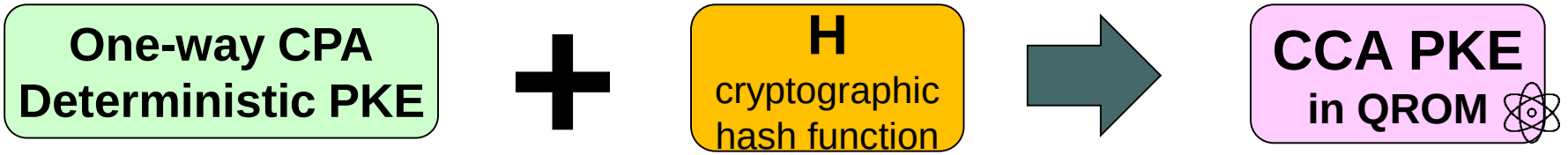
➔ An adversary may quantumly compute it!

- **QRO model** allows the adversary to evaluate the RO "in superposition"
  - Submit quantum states, and receives the evaluated quantum state

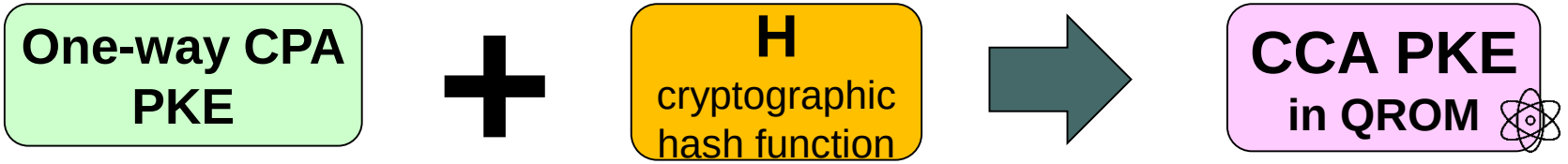
Based on EC'18 presentation slide by Takashi Yamakawa

CCA Secure PKE Schemes in QRO Model

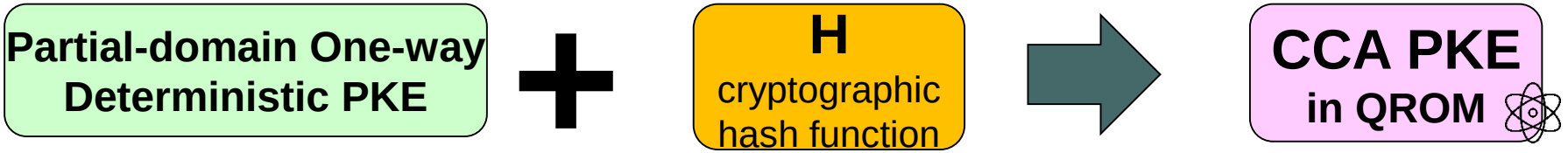
- Several works show security analyses of existing RO-based constructions in the QRO model
  - Bellare-Rogaway [BDH+11,SXY18]



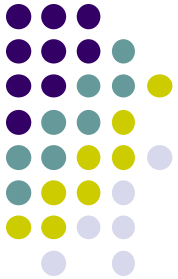
  - Fujisaki-Okamoto (and variants) [TU16,HHK17,JZC+18]



  - OAEP [TU16]



Difficulty in QRO Model



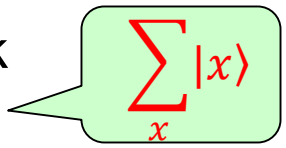
- Suppose that we want to prove
  $\boldsymbol{H}(x) \approx_c$ random when x is hard to compute

- Let $\boldsymbol{\varepsilon_{ind}}$ = best adversary's advantage to break indistinguishability
  $\boldsymbol{\varepsilon_{search}}$ = best adversary's advantage to find x

- In classical RO model, $\boldsymbol{H}(x)$ looks random as long as x is hard to find

➔ $\boldsymbol{\varepsilon_{ind}} \leqq \boldsymbol{\varepsilon_{search}}$

- In Quatum RO model, the above argument doesn't work because an adversary may query superposition of all x



- **OW2H Lemma [Unr14]:** $\boldsymbol{\varepsilon_{ind} \leq 2q \cdot \sqrt{\varepsilon_{search}}}$

q = #$\boldsymbol{H}$-query

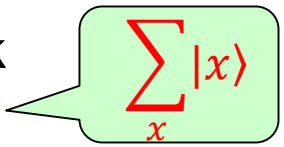
Based on EC'18 presentation slide by Takashi Yamakawa

Difficulty in QRO Model

- Suppose that we want to prove
- Suppose $q = 2^{60}$.
- If we want to achieve $\varepsilon_{ind} = 2^{-128}$ (128-bit security), we have to assume $\varepsilon_{search} = 2^{-378}$ (378-bit security)

Huge reduction loss can occur!!

- In Quatum RO model, the above argum[illegible]oesn't work because an adversary may query super[illegible]osition of all x

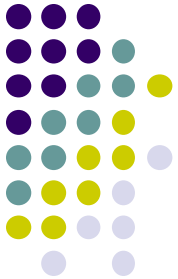


- **OW2H Lemma [Unr14]:** $\varepsilon_{ind} \leq 2q \cdot \sqrt{\varepsilon_{search}}$

q = #***H***-query

Based on EC'18 presentation slide by Takashi Yamakawa

CCA Secure PKE Constructions in QRO Model



Paper	Construction	Assumption	Reduction Loss
[BDH+11]	Bellare-Rogaway	One-way D-PKE	$\epsilon_{cca} \approx q\sqrt{\epsilon_{assump}}$
[SXY18]	Bellare-Rogaway	Disjoint- Simulatable D-PKE	$\epsilon_{cca} \approx \epsilon_{assump}$
[TU16,HHK17]	Fujisaki-Okamoto	One-way CPA	$\epsilon_{cca} \approx q^{\frac{3}{2}}(\epsilon_{assump})^{\frac{1}{4}}$
[JZC+18]	Fujisaki-Okamoto	One-way CPA	$\epsilon_{cca} \approx q\sqrt{\epsilon_{assump}}$
[TU16]	OAEP	Partial-domain One-way D-PKE	$\epsilon_{cca} \approx q^{\frac{7}{4}}(\epsilon_{assump})^{\frac{1}{8}}$

Note: CCA security is in single-user/challenge setting

Disjoint simulatability [SXY18] is an indistinguishability-type notion for D-PKE, satisfied by D-PKE whose ciphertext is pseudorandom and its CT-space is sparse (but it is not necessarily satisfied by most NIST PQ-competition candidates as it is)

Based on EC'18 presentation slide by Takashi Yamakawa

Open Problems

- Constructions with better reductions?
  - Can we avoid using Unruh's OW2H lemma?
- Can we prove that the reduction losses of the existing constructions are inherent?
- What about multi-user/challenge setting?
  - Current works focus only on single-user/challenge setting
- Are other existing RO-constructions not listed in the previous slide also secure in the QRO model?

Summary

- CCA secure PKE has been one of the central topics in public-key cryptography

- Part1 reviewed basic constructions of CCA PKE
  and how their security is proved
  - Naor-Yung, Hybrid Enc., HPS-based KEM, Fujisaki-Okamoto

- Part 2 briefly reviewed recent hot topics on CCA PKE
  - General assumptions
  - Tight security
  - Post-quantum security

Thank you!

- Still many interesting open problems are left

References

- [AGKS05]: Masayuki Abe, Rosario Gennaro, Kaoru Kurosawa, Victor Shoup: Tag-KEM/DEM: A New Framework for Hybrid Encryption and A New Analysis of Kurosawa-Desmedt KEM. EUROCRYPT 2005: 128-146
- [ADKNO13] Masayuki Abe, Bernardo David, Markulf Kohlweiss, Ryo Nishimaki, Miyako Ohkubo: Tagged One-Time Signatures: Tight Security and Optimal Tag Size. Public Key Cryptography 2013: 312-331
- [AHY15]: Nuttapong Attrapadung, Goichiro Hanaoka, Shota Yamada: A Framework for Identity-Based Encryption with Almost Tight Security. ASIACRYPT (1) 2015: 521-549
- [BFK+12]: Romain Bardou, Riccardo Focardi, Yusuke Kawamoto, Lorenzo Simionato, Graham Steel, Joe-Kai Tsay: Efficient Padding Oracle Attacks on Cryptographic Hardware. CRYPTO 2012: 608-625
- [BBDQ18]: Fabrice Benhamouda, Olivier Blazy, Léo Ducas, Willy Quach: Hash Proof Systems over Lattices Revisited. Public Key Cryptography (2) 2018: 644-674
- [BBM00]: Mihir Bellare, Alexandra Boldyreva, Silvio Micali: Public-Key Encryption in a Multi-user Setting: Security Proofs and Improvements. EUROCRYPT 2000: 259-274
- [BDPR98]: Mihir Bellare, Anand Desai, David Pointcheval, Phillip Rogaway: Relations Among Notions of Security for Public-Key Encryption Schemes. CRYPTO 1998: 26-45
- [BR93]: Mihir Bellare, Phillip Rogaway: Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. ACM Conference on Computer and Communications Security 1993: 62-73
- [BR94]: Mihir Bellare, Phillip Rogaway: Optimal Asymmetric Encryption. EUROCRYPT 1994: 92-111
- [BR06]: Mihir Bellare, Phillip Rogaway: The Security of Triple Encryption and a Framework for Code-Based Game-Playing Proofs. EUROCRYPT 2006: 409-426
- [BS99]: Mihir Bellare, Amit Sahai: Non-malleable Encryption: Equivalence between Two Notions, and an Indistinguishability-Based Characterization. CRYPTO 1999: 519-536
- [BS06]: Mihir Bellare, Amit Sahai: Non-Malleable Encryption: Equivalence between Two Notions, and an Indistinguishability-based Characterization. IACR Cryptology ePrint Archive 2006: 228 (2006)
- [Ble98]: Daniel Bleichenbacher: Chosen Ciphertext Attacks Against Protocols Based on the RSA Encryption Standard PKCS #1. CRYPTO 1998: 1-12

References

- [BDH+11]: Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, Mark Zhandry: Random Oracles in a Quantum World. ASIACRYPT 2011: 41-69
- [Can01]: Ran Canetti: Universally Composable Security: A New Paradigm for Cryptographic Protocols. FOCS 2001: 136-145
- [CGH98]: Ran Canetti, Oded Goldreich, Shai Halevi: The Random Oracle Methodology, Revisited (Preliminary Version). STOC 1998: 209-218
- [CHK04]: Ran Canetti, Shai Halevi, Jonathan Katz: Chosen-Ciphertext Security from Identity-Based Encryption. EUROCRYPT 2004: 207-222
- [CKN03]: Ran Canetti, Hugo Krawczyk, Jesper Buus Nielsen: Relaxing Chosen-Ciphertext Security. CRYPTO 2003: 565-582
- [CL18]: Ran Canetti, Amit Lichtenberg: Certifying Trapdoor Permutations, Revisited. TCC (1) 2018: 476-506
- [CHJ+02]: Jean-Sébastien Coron, Helena Handschuh, Marc Joye, Pascal Paillier, David Pointcheval, Christophe Tymen: GEM: A Generic Chosen-Ciphertext Secure Encryption Method. CT-RSA 2002: 263-276
- [CS98]: Ronald Cramer, Victor Shoup: A Practical Public Key Cryptosystem Provably Secure Against Adaptive Chosen Ciphertext Attack. CRYPTO 1998: 13-25
- [CS02]: Ronald Cramer, Victor Shoup: Universal Hash Proofs and a Paradigm for Adaptive Chosen Ciphertext Secure Public-Key Encryption. EUROCRYPT 2002: 45-64
- [CS03]: Ronald Cramer, Victor Shoup: Design and Analysis of Practical Public-Key Encryption Schemes Secure against Adaptive Chosen Ciphertext Attack. SIAM J. Comput. 33(1): 167-226 (2003)
- [Dac14]: Dana Dachman-Soled: A Black-Box Construction of a CCA2 Encryption Scheme from a Plaintext Aware (sPA1) Encryption Scheme. Public Key Cryptography 2014: 37-55
- [Den03]: Alexander W. Dent: A Designer's Guide to KEMs. IMA Int. Conf. 2003: 133-151
- [DDOPS01]: Alfredo De Santis, Giovanni Di Crescenzo, Rafail Ostrovsky, Giuseppe Persiano, Amit Sahai: Robust Non-interactive Zero Knowledge. CRYPTO 2001: 566-598
- [DH76]: Whitfield Diffie, Martin E. Hellman: New directions in cryptography. IEEE Trans. Information Theory 22(6): 644-654 (1976)

References

- [DDN91]: Danny Dolev, Cynthia Dwork, Moni Naor: Non-Malleable Cryptography (Extended Abstract). STOC 1991: 542-552
- [DG17]: Nico Döttling, Sanjam Garg: From Selective IBE to Full IBE and Selective HIBE. TCC (1) 2017: 372-408
- [DGHM18]: Nico Döttling, Sanjam Garg, Mohammad Hajiabadi, Daniel Masny: New Constructions of Identity-Based and Key-Dependent Message Secure Encryption Schemes. Public Key Cryptography (1) 2018: 3-31
- [FO99a]: Eiichiro Fujisaki, Tatsuaki Okamoto: How to Enhance the Security of Public-Key Encryption at Minimum Cost. Public Key Cryptography 1999: 53-68
- [FO99b]: Eiichiro Fujisaki, Tatsuaki Okamoto: Secure Integration of Asymmetric and Symmetric Encryption Schemes. CRYPTO 1999: 537-554
- [FO13]: Eiichiro Fujisaki, Tatsuaki Okamoto: Secure Integration of Asymmetric and Symmetric Encryption Schemes. J. Cryptology 26(1): 80-101 (2013)
- [GHKW16]: Romain Gay, Dennis Hofheinz, Eike Kiltz, Hoeteck Wee: Tightly CCA-Secure Encryption Without Pairings. EUROCRYPT (1) 2016: 1-27
- [GHK17]: Romain Gay, Dennis Hofheinz, Lisa Kohl: Kurosawa-Desmedt Meets Tight Security. CRYPTO (3) 2017: 133-160
- [GCDCT16]: Junqing Gong, Jie Chen, Xiaolei Dong, Zhenfu Cao, Shaohua Tang: Extended Nested Dual System Groups, Revisited. Public Key Cryptography (1) 2016: 133-163
- [GMM07]: Yael Gertner, Tal Malkin, Steven Myers: Towards a Separation of Semantic and CCA Security for Public Key Encryption. TCC 2007: 434-455
- [Gol01]: Oded Goldreich: The Foundations of Cryptography - Volume 1, Basic Techniques. Cambridge University Press 2001
- [Gol04]: Oded Goldreich: The Foundations of Cryptography - Volume 2, Basic Applications. Cambridge University Press 2004
- [Gol11]: Oded Goldreich: Basing Non-Interactive Zero-Knowledge on (Enhanced) Trapdoor Permutations: The State of the Art. Studies in Complexity and Cryptography 2011: 406-421

References

- [GM84]: Shafi Goldwasser, Silvio Micali: Probabilistic Encryption. J. Comput. Syst. Sci. 28(2): 270-299 (1984)
- [GR13]: Oded Goldreich, Ron D. Rothblum: Enhancements of Trapdoor Permutations. J. Cryptology 26(3): 484-512 (2013)
- [GOS06]: Jens Groth, Rafail Ostrovsky, Amit Sahai: Perfect Non-interactive Zero Knowledge for NP. EUROCRYPT 2006: 339-358
- [HK15]: Mohammad Hajiabadi, Bruce M. Kapron: Reproducible Circularly-Secure Bit Encryption: Applications and Realizations. CRYPTO (1) 2015: 224-243
- [HK12]: Shai Halevi, Yael Tauman Kalai: Smooth Projective Hashing and Two-Message Oblivious Transfer. J. Cryptology 25(1): 158-193 (2012)
- [HO12]: Brett Hemenway, Rafail Ostrovsky: On Homomorphic Encryption and Chosen-Ciphertext Security. Public Key Cryptography 2012: 52-65
- [HO13]: Brett Hemenway, Rafail Ostrovsky: Building Lossy Trapdoor Functions from Lossy Encryption. ASIACRYPT (2) 2013: 241-260
- [HHK10]: Javier Herranz, Dennis Hofheinz, Eike Kiltz: Some (in)sufficient conditions for secure hybrid encryption. Inf. Comput. 208(11): 1243-1257 (2010)
- [HKS15]: Dennis Hofheinz, Jessica Koch, Christoph Striecks: Identity-Based Encryption with (Almost) Tight Security in the Multi-instance, Multi-ciphertext Setting. Public Key Cryptography 2015: 799-822
- [HHK17]: Dennis Hofheinz, Kathrin Hövelmanns, Eike Kiltz: A Modular Analysis of the Fujisaki-Okamoto Transformation. TCC (1) 2017: 341-371
- [HK07]: Dennis Hofheinz, Eike Kiltz: Secure Hybrid Encryption from Weakened Key Encapsulation. CRYPTO 2007: 553-571
- [Hof16]: Dennis Hofheinz: Algebraic Partitioning: Fully Compact and (almost) Tightly Secure Cryptography. TCC (A1) 2016: 251-281
- [Hof17]: Dennis Hofheinz: Adaptive Partitioning. EUROCRYPT (3) 2017: 489-518
- [HJ12]: Dennis Hofheinz, Tibor Jager: Tightly Secure Signatures and Public-Key Encryption. CRYPTO 2012: 590-607

References

- [HLW12]: Susan Hohenberger, Allison B. Lewko, Brent Waters: Detecting Dangerous Queries: A New Approach for Chosen Ciphertext Security. EUROCRYPT 2012: 663-681
- [IR89]: Russell Impagliazzo, Steven Rudich: Limits on the Provable Consequences of One-Way Permutations. STOC 1989: 44-61
- [KV09]: Jonathan Katz, Vinod Vaikuntanathan: Smooth Projective Hashing and Password-Based Authenticated Key Exchange from Lattices. ASIACRYPT 2009: 636-652
- [Kil06]: Eike Kiltz: Chosen-Ciphertext Security from Tag-Based Encryption. TCC 2006: 581-600
- [KMO10]: Eike Kiltz, Payman Mohassel, Adam O'Neill: Adaptive Trapdoor Functions and Chosen-Ciphertext Security. EUROCRYPT 2010: 673-692
- [KW18]: Venkata Koppula, Brent Waters: Realizing Chosen Ciphertext Security Generically in Attribute-Based Encryption and Predicate Encryption. IACR Cryptology ePrint Archive 2018: 847 (2018)
- [JSC+18]: Haodong Jiang, Zhenfeng Zhang, Long Chen, Hong Wang, Zhi Ma: IND-CCA-Secure Key Encapsulation Mechanism in the Quantum Random Oracle Model, Revisited. CRYPTO (3) 2018: 96-125
- [Lin03]: Yehuda Lindell: A Simpler Construction of CCA2-Secure Public-Key Encryption under General Assumptions. EUROCRYPT 2003: 241-254
- [LPJY15]: Benoît Libert, Thomas Peters, Marc Joye, Moti Yung: Compactly Hiding Linear Spans - Tightly Secure Constant-Size Simulation-Sound QA-NIZK Proofs and Applications. ASIACRYPT (1) 2015: 681-707
- [MH14a]: Takahiro Matsuda, Goichiro Hanaoka: Chosen Ciphertext Security via Point Obfuscation. TCC 2014: 95-120
- [MH14b]: Takahiro Matsuda, Goichiro Hanaoka: Chosen Ciphertext Security via UCE. Public Key Cryptography 2014: 56-76
- [MH15]: Takahiro Matsuda, Goichiro Hanaoka: Constructing and Understanding Chosen Ciphertext Security via Puncturable Key Encapsulation Mechanisms. TCC (1) 2015: 561-590
- [MH16]: Takahiro Matsuda, Goichiro Hanaoka: Trading Plaintext-Awareness for Simulatability to Achieve Chosen Ciphertext Security. Public Key Cryptography (1) 2016: 3-34

References

- [OP01]: Tatsuaki Okamoto, David Pointcheval: REACT: Rapid Enhanced-Security Asymmetric Cryptosystem Transform. CT-RSA 2001: 159-175
- [Pai99]: Pascal Paillier: Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. EUROCRYPT 1999: 223-238
- [PSV07]: Rafael Pass, Abhi Shelat, Vinod Vaikuntanathan: Relations Among Notions of Non-malleability for Encryption. ASIACRYPT 2007: 519-535
- [PW08]: Chris Peikert, Brent Waters: Lossy trapdoor functions and their applications. STOC 2008: 187-196
- [RS91]: Charles Rackoff, Daniel R. Simon: Non-Interactive Zero-Knowledge Proof of Knowledge and Chosen Ciphertext Attack. CRYPTO 1991: 433-444
- [RS09]: Alon Rosen, Gil Segev: Chosen-Ciphertext Security via Correlated Products. TCC 2009: 419-436
- [Sah99]: Amit Sahai: Non-Malleable Non-Interactive Zero Knowledge and Adaptive Chosen-Ciphertext Security. FOCS 1999: 543-553
- [SW14]: Amit Sahai, Brent Waters: How to use indistinguishability obfuscation: deniable encryption, and more. STOC 2014: 475-484
- [Sho94]: Peter W. Shor: Algorithms for Quantum Computation: Discrete Logarithms and Factoring. FOCS 1994: 124-134
- [Sho00]: Victor Shoup: Using Hash Functions as a Hedge against Chosen Ciphertext Attack. EUROCRYPT 2000: 275-288
- [Sho04]: Victor Shoup: Sequences of games: a tool for taming complexity in security proofs. IACR Cryptology ePrint Archive 2004: 332 (2004)
- [SXY18]: Tsunekazu Saito, Keita Xagawa, Takashi Yamakawa: Tightly-Secure Key-Encapsulation Mechanism in the Quantum Random Oracle Model. EUROCRYPT (3) 2018: 520-551
- [TU16]: Ehsan Ebrahimi Targhi, Dominique Unruh: Post-Quantum Security of the Fujisaki-Okamoto and OAEP Transforms. TCC (B2) 2016: 192-216
- [Unr14]: Dominique Unruh: Revocable Quantum Timed-Release Encryption. EUROCRYPT 2014: 129-146
- [Wee10]: Hoeteck Wee: Efficient Chosen-Ciphertext Security via Extractable Hash Proofs. CRYPTO 2010: 314-332