

CCA Security and Trapdoor Functions

Payman Mohassel
University of Calgary

joint work with Eike Kiltz and Adam O'Neill

Public-key Encryption

pk



$$C = \text{Enc}_{pk}(m)$$

→



$(pk, sk) \leftarrow \text{PKG}(1^k)$



$m \leftarrow \text{Dec}_{sk}(C)$

$\text{PKE} = (\text{PKG}, \text{Enc}, \text{Dec})$

Public-key Encryption

- Key generation
 - $(pk, sk) \leftarrow \text{PKG}(1^n)$
- Encryption
 - $C \leftarrow \text{Enc}(pk, m; r)$
 - Randomized
- Decryption
 - $m \leftarrow \text{Dec}(sk, C)$
 - Deterministic

Semantic Security [GM '82]

- Given $Enc_{pk}(m)$ it is hard to learn any $f(m)$
 - No partial information is leaked
- Captures our intuition for confidentiality

IND-CPA PKE

$(pk, sk) \leftarrow \text{PKG}(1^k) ; b \leftarrow \{0,1\}$

Challenger

pk

m_0, m_1

$C = \text{Enc}_{pk}(m_b)$



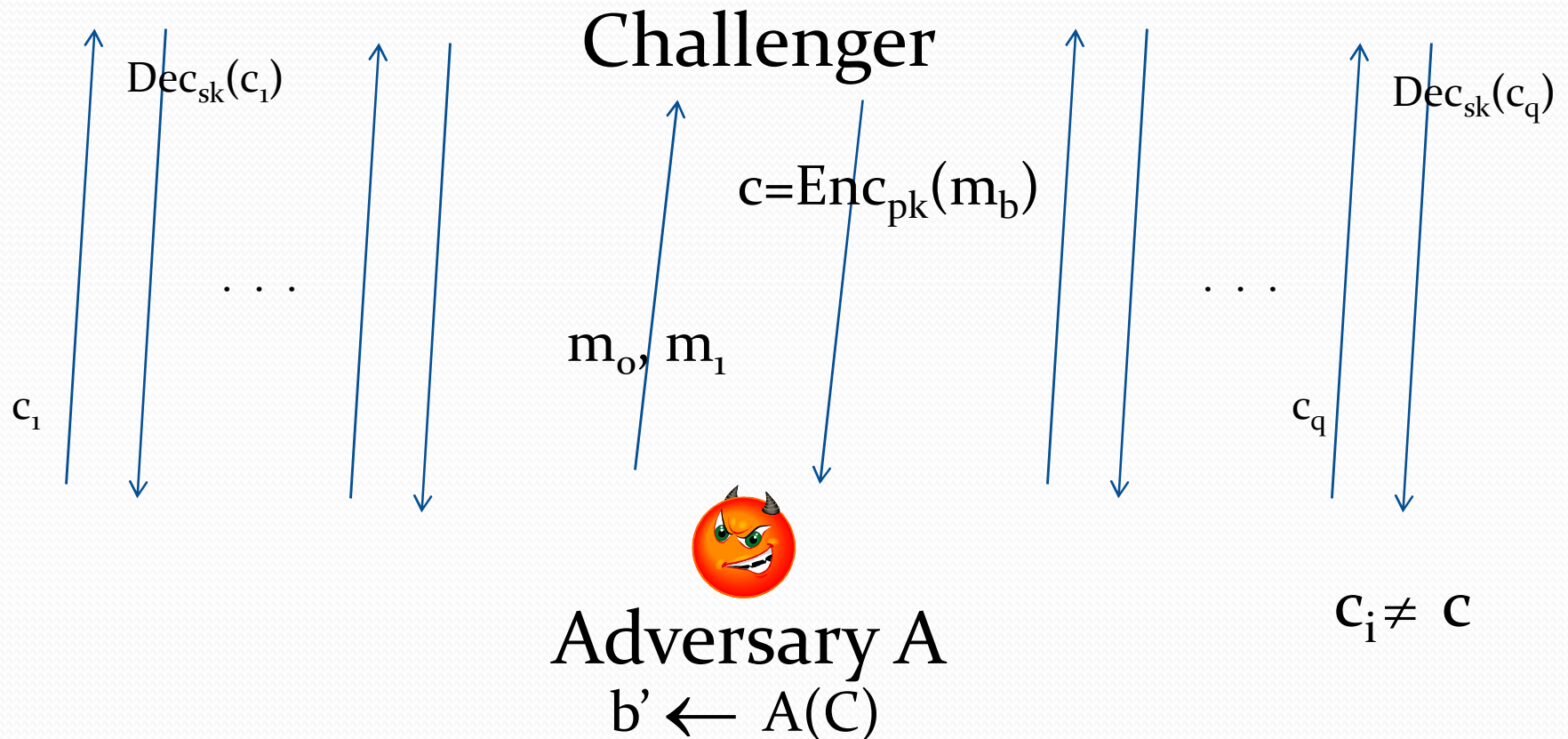
Adversary A

$b' \leftarrow A(C)$

$$\text{Adv}_{\text{ind-cpa, PKE}}(A) = |\Pr[b' = b] - 1/2|$$

IND-CCA PKE

$(pk, sk) \leftarrow \text{PKG}(1^k) ; b \leftarrow \{0,1\}$



$$\text{Adv}_{\text{ind-cca,PKE}}(A) = |\Pr[b' = b] - 1/2|$$

First General Constructions

- Naor-Yung paradigm [NY '90, DDN '91, ...]
 - IND-CPA PKE + NIZK
- $C_1 = \text{Enc}(pk_1, m) + C_2 = \text{Enc}(pk_2, m)$
 - Prove “ C_1 and C_2 encrypt the same message m ”
- Based on any one-way trapdoor permutation
- **Non-black-box and inefficient**

Number Theoretic Constructions

- Universal hash proofs
 - [CS '98, CS '02, ...]
 - Decisional assumptions
 - DDH, QR, DCR
- More recently
 - [CKS '08, HK '09, ...]
 - Computational assumptions
 - CDH, RSA, factoring

Black-Box Constructions

- Simplicity
- Efficiency preserving
- Variety of assumptions
- Point of comparison

IND-CCA PKE from IBE

- From any IND-CPA IBE scheme
 - [CHK '04, BK '05]
- IBE schemes from
 - Decisional and computational BDH
 - First, based on a computational assumption
- But IBE is not easy to achieve
 - Based on many assumptions

Trapdoor Functions [DH '76]

- TDF = (Gen, F, F⁻¹)
 - $F(ek,.) : \{0,1\}^k \rightarrow \{0,1\}^{k'}$
- Key generation
 - $(ek, td) \leftarrow \text{Gen}(1^k)$
- Evaluation on x
 - $y \leftarrow F(ek, x)$
- Inversion on y
 - $x' \leftarrow F^{-1}(td, y)$
 - s.t. $F(ek, x') = y$
- $F(ek,.)$ and $F^{-1}(td,.)$ are deterministic

One-Wayness

$(ek, td) \leftarrow \text{Gen}(1^k), x \leftarrow \{0,1\}^k$

Challenger

$ek, y = F(ek, x)$

x'



Adversary A

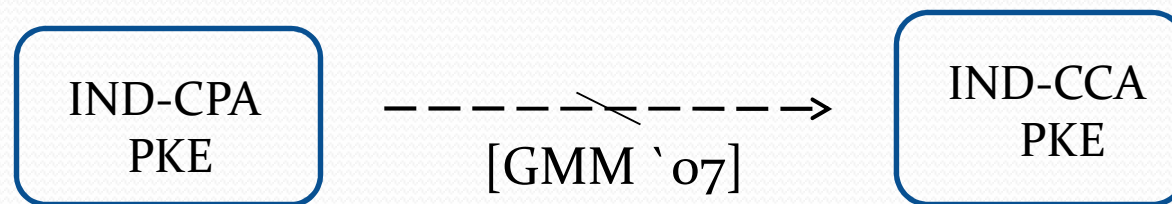
$x' \leftarrow A(ek, y)$

$$\text{Adv}_{\text{ow, TDF}}(A) = \Pr[x' = x]$$

OW-TDFs \Rightarrow IND-CPA PKE

- TDF = (Gen, F, F⁻¹)
- Hardcore bits hc(.)
 - Given F(ek,x), hc(x) looks random
- PKG(1^k)
 - (pk,sk) \leftarrow Gen(1^k)
- Enc(b;r)
 - C = (F(pk,r), b \oplus hc(r))
- Dec(c₁, c₂)
 - r = F⁻¹(sk,c₁) ; b = c₂ \oplus hc(r)
- **Black-box, efficient**

How about IND-CCA PKE?



- IND-CCA PKE from OW-TDFs has proven elusive!
 - In the standard model
- What assumption on TDFs leads to BB Construction of IND-CCA PKE?

Stronger Assumptions on TDFs

- Lossiness
- Correlation security
- Adaptiveness

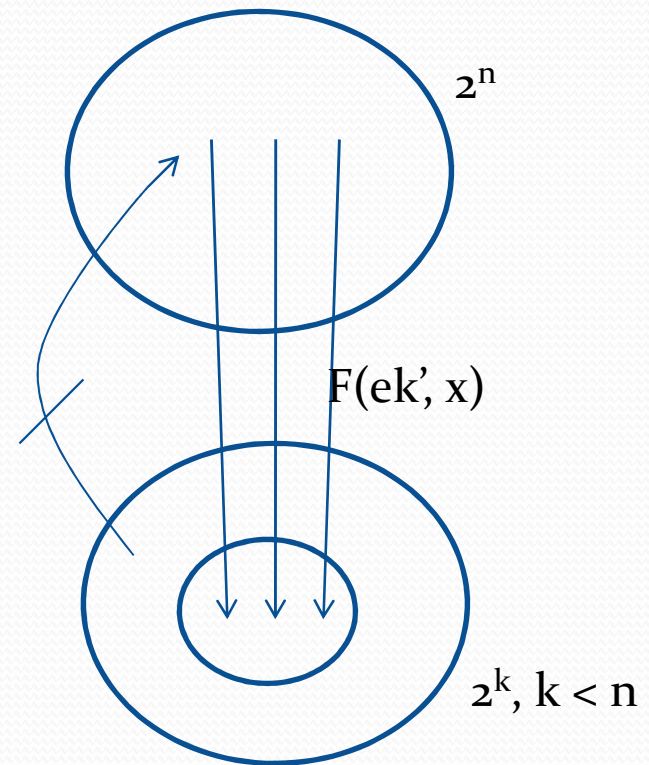
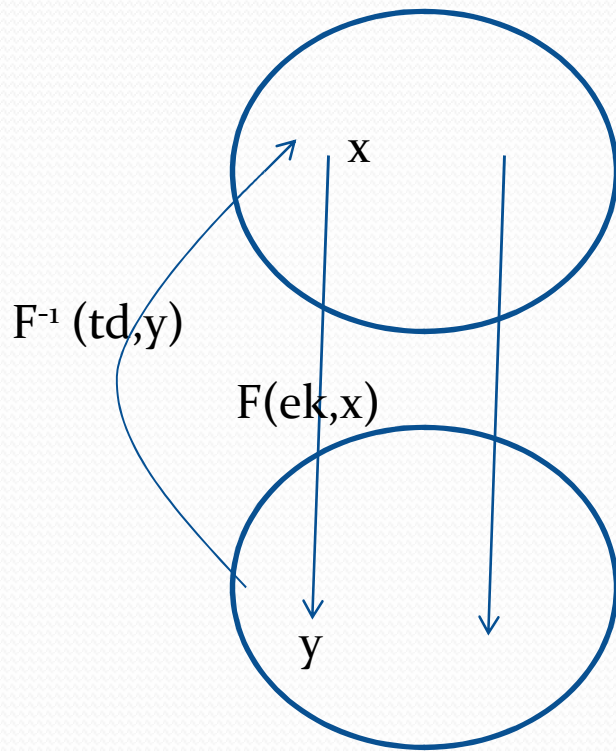
Lossy Trapdoor Functions [PW '08]

$$\text{TDF} = (G_{\text{inj}}, G_{\text{loss}}, F, F^{-1})$$

$$(ek, td) \leftarrow G_{\text{inj}}(1^k)$$

$$ek \approx ek'$$

$$ek' \leftarrow G_{\text{loss}}(1^k)$$



Lossy TDFs

- IND-CCA PKE from lossy TDFs [PW '08]
- Lossy TDFs
 - DDH assumption
 - d-linear generalization
 - Paillier's DCR
 - LWE
 - QR assumption [FGKRS '10]
 - RSA under phi-hiding [KO '10]
- The first IND-CCA PKE from lattice-based assumptions

Correlation Secure TDFs [RS '09]

For $1 \leq i \leq n$, $(ek_i, td_i) \leftarrow \text{Gen}(1^k)$, $(x_1, x_2, \dots, x_n) \leftarrow C(1^k)$

Challenger

$$x_1 = x_2 = \dots = x_n$$

$$ek_i, y = F(ek_i, x_i)$$

$$x_1, x_2, \dots, x_n$$



Adversary A

Why Interesting?

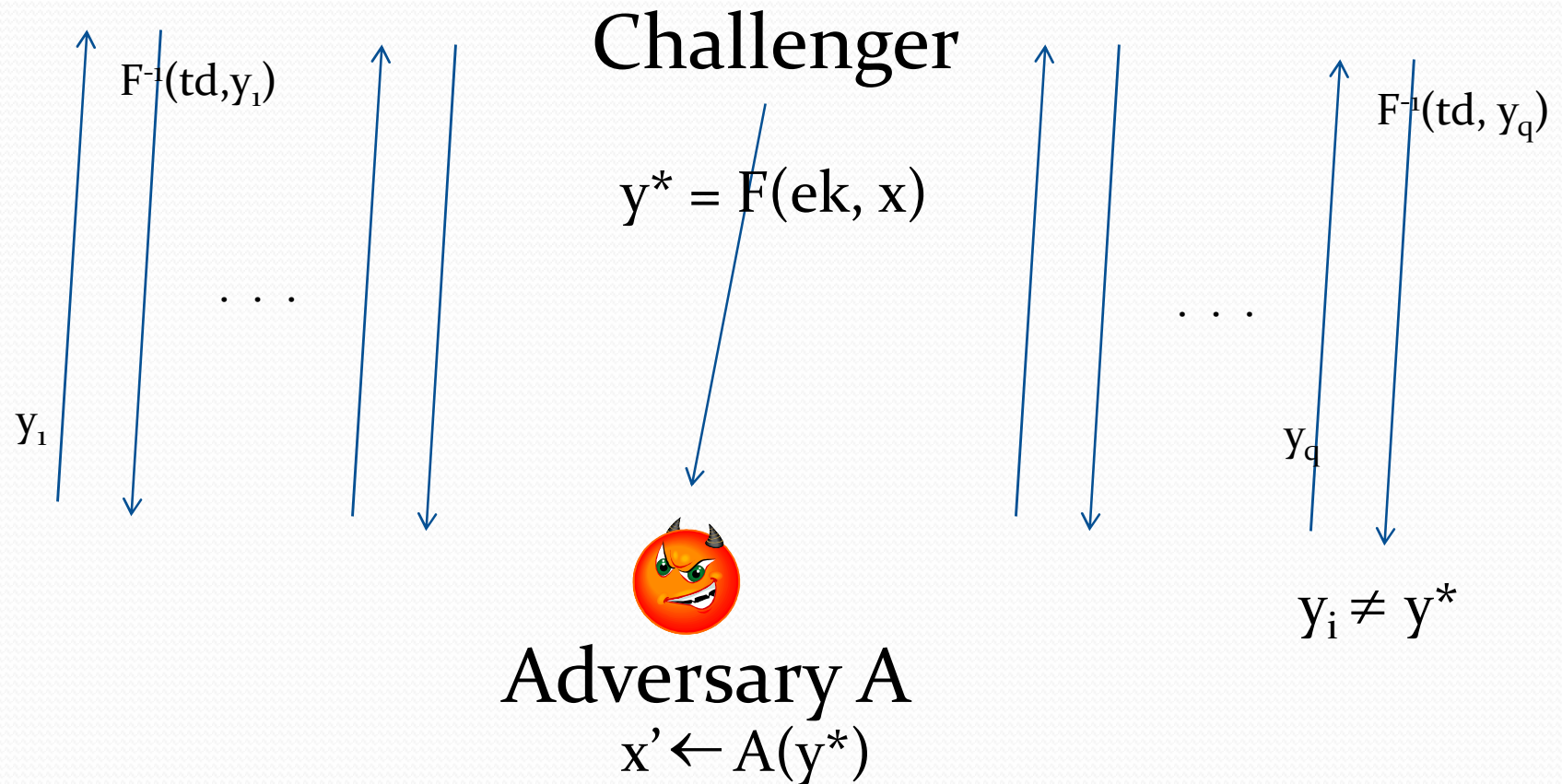
- LTDFs \Rightarrow CS-TDFs
- CS-TDFs \Rightarrow IND-CCA PKE
- LTDFs are too strong
 - Imply CRHFs, OT, ...
- BB separation [RS '09]
 - CS-TDFs $\not\Rightarrow$ LTDFs
- Other assumptions [FGKRS '10]
 - Syndrome decoding

Is CS-TDF too Strong?

- OW-TDFs $\not\Rightarrow$ CS-TDFs, [Vahlis '10]
- Is correlation security necessary for encryption?
- Why not consider CCA TDFs?
 - A natural choice for IND-CCA PKE
 - Does it imply IND-CCA PKE?
 - Where it stands relative to LTDFs and CS-TDFs?

Adaptive TDFs

$$(ek, td) \leftarrow \text{Gen}(1^n) ; x \leftarrow \{0,1\}^k$$



$$\text{Ad}_{\text{atdf}, \text{TDF}}(A) = \Pr[x' = x]$$

Deterministic Encryption [BBO '07]

- CPA and CCA variants
- CCA variant is a strong ATDF
 - Adaptively one-way
 - Hides partial information
 - Accepts high entropy inputs
- Different applications

ATDFs \Rightarrow IND-CCA PKE

- TDF = (Gen, F, F⁻¹)
 - Key Generation
 - (pk, sk) \leftarrow Gen(1^k)
 - Encrypt bit b
 - $r \leftarrow \{0,1\}^k$
 - $C = (F(\text{pk}, r), b \oplus \text{hc}(r))$
 - Decryption of (c₁, c₂)
 - $r = F^{-1}(\text{sk}, c_1)$; $m = c_2 \oplus \text{hc}(r)$
 - Malleable
- TDF = (Gen, F, F⁻¹)
 - Key Generation
 - (pk, sk) \leftarrow Gen(1^k)
 - Encrypt bit b
 - $r \leftarrow \{0,1\}^k$
 - If $\text{hc}(r) = b$, return $C = F(\text{pk}, r)$
 - Else repeat
 - Decryption of C
 - $r = F^{-1}(\text{sk}, c)$; return $\text{hc}(r)$
 - Multi-bit messages
 - [Myer-Shelat '09]

If linear hardcore bits available, ATDF is a CCA-KEM

Tag-based Variant

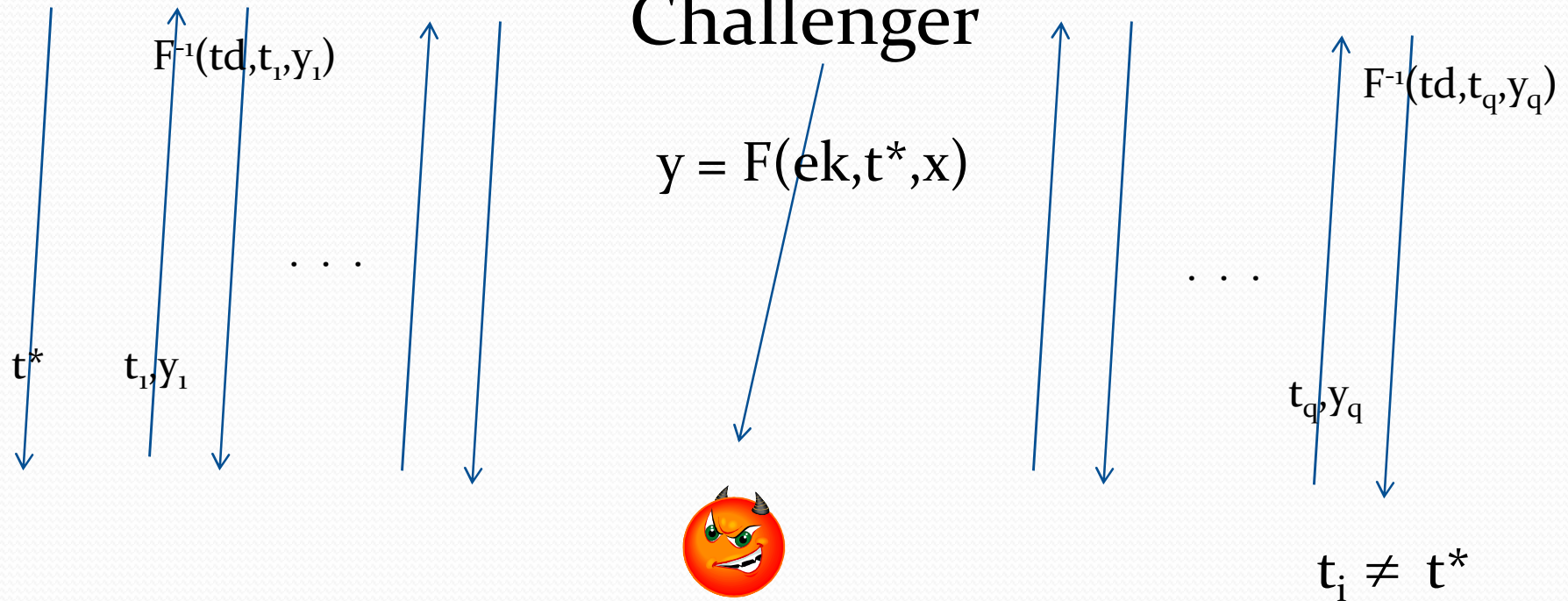
- Evaluation and inversion take a tag
 - $F(ek,t,x)$, $F^{-1}(td,t,y)$

tb-ATDFs

$$(ek, td) \leftarrow \text{Gen}(1^n) ; x \leftarrow \{0,1\}^k , t^* \leftarrow T(1^k)$$

Challenger

$$y = F(ek, t^*, x)$$



Adversary A
 $x' \leftarrow A(t^*, y)$

$$\text{Ad}_{\text{tb-atdf}, \text{TDF}}(A) = \Pr[x' = x]$$

tb-ATDFs \Rightarrow IND-CCA PKE

- TDF = (Gen, F, F⁻¹)
- OTS = (G, Sig, Ver)
- Key Generation
 - $(ek, td) \leftarrow \text{Gen}(1^k)$
- Encryption of bit b
 - $r \leftarrow \{0,1\}^k$
 - $(vk, sk) \leftarrow G(1^k)$
 - $C = (F(pk, vk, r), b \oplus hc(r))$
 - $sig \leftarrow \text{Sig}(sk, C)$
 - Return (vk, C, sig)
- Decryption of $(vk, (c_1, c_2), sig)$
 - If $\text{Ver}(vk, C, sig) = 1$
 - $r = F^{-1}(td, vk, c_1)$
 - Return $m = hc(r) \oplus c_2$

If linear hardcore bits are available, replace with BK transform

Now what?

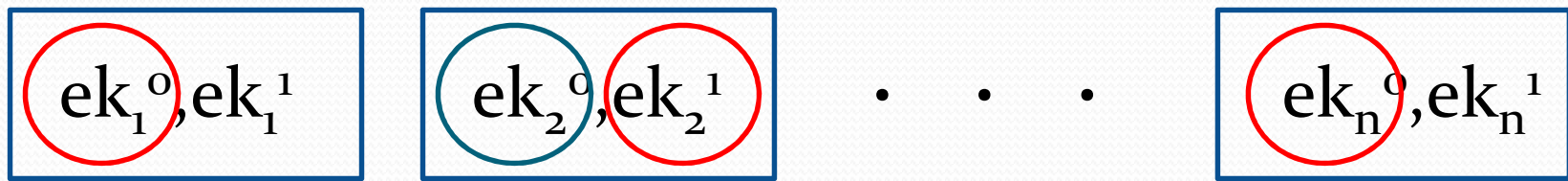
- $\text{LTDF} \Rightarrow \text{tb-ATDFs} \Rightarrow \text{IND-CCA PKE}$
 - [PW '08] encryption scheme
- $\text{CS-TDF} \Rightarrow \text{tb-ATDFs} \Rightarrow \text{IND-CCA PKE}$
 - [RS '09] encryption scheme
- Indirectly building a tb-ATDF

- A direct construction for ATDFs
 - More efficient

CS-TDFs \Rightarrow tb-ATDFs

$$\text{TDF} = (\text{Gen}, F, F^{-1})$$

$$\text{Gen}(1^k)$$



$$t \neq t^*$$

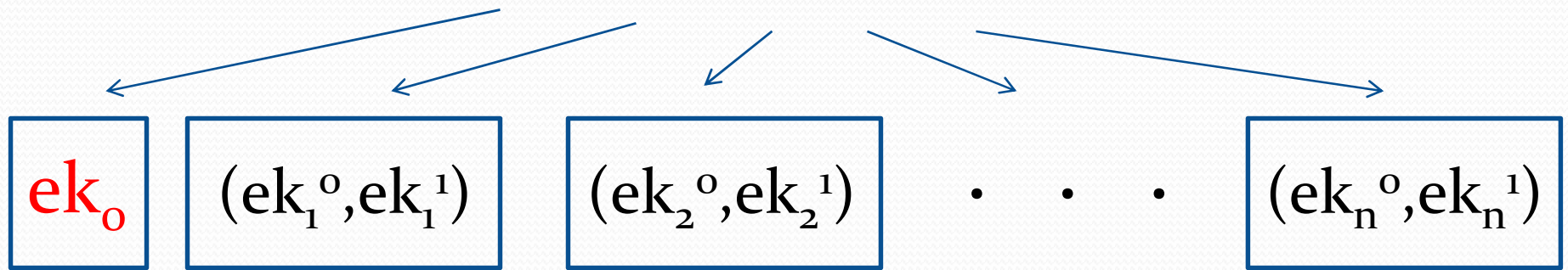
$$F'(t, x) = F(ek_1^{t_1}, x), \quad F(ek_2^{t_2}, x), \quad \dots, \quad F(ek_n^{t_n}, x)$$

$$t = t_1 t_2 \dots t_n$$

CS-TDF \Rightarrow ATDF

$$\text{TDF} = (\text{Gen}, F, F^{-1})$$

$$\text{Gen}(1^k)$$



$$F(t, x) = F(ek_0, x), F(ek_1^{t_1}, x), F(ek_2^{t_2}, x), \dots, F(ek_n^{t_n}, x)$$

$$F(ek_0, x) = t_1 t_2 \dots t_n$$

RSA-based Instantiation

- Instance Independent RSA
- Computing e^{th} root is hard
 - Even with access to an oracle for e'^{th} root ($e' \neq e$)
 - e, e' are primes
 - $\gcd(e, e') = 1$
- Considered before
 - [PV '06, MJ '09]
- Yields efficient tb-ATDF
 - Hash the tag to a prime exponent
 - Heuristic hashing [GHR '99]

Relations

- (ATDFs, tb-ATDFs) \Rightarrow CS-TDFs
 - Extension of Vahlis's result
- ATDFs and tb-ATDFs
 - tb-ATDFs \Rightarrow ATDFs
 - Other direction open

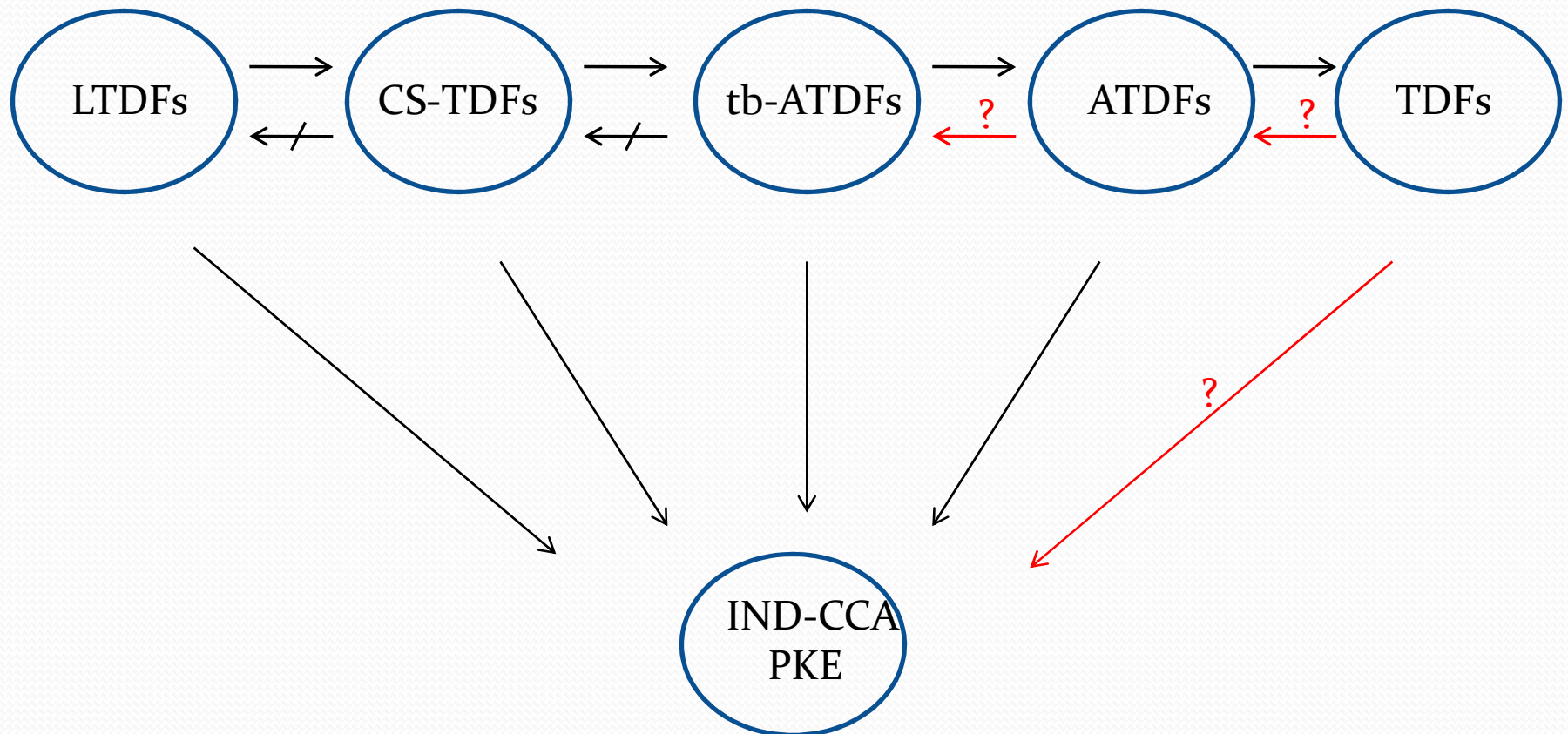
Optimizations

- If the tb -ATDF is a *permutation* or has *linear hardcore bits*
 - Use BK-transform instead of OTS
 - Shorter ciphertexts and better efficiency
- $LTDF \Rightarrow ATDF \Rightarrow IND\text{-}CCA$ PKE
 - Avoids OTS altogether
 - Use KEM/DEM if linear hardcore bits available

Interesting Questions

- Separating ATDFs and TDFs
- Designing ATDFs and tb-ATDFs
 - Weaker assumptions
 - Better efficiency
- Studying the II-RSA assumption
 - Better hashing strategies
- More applications of ATDFs
 - Signature schemes?
 - Non-malleable primitives?

Summary



Thank you!