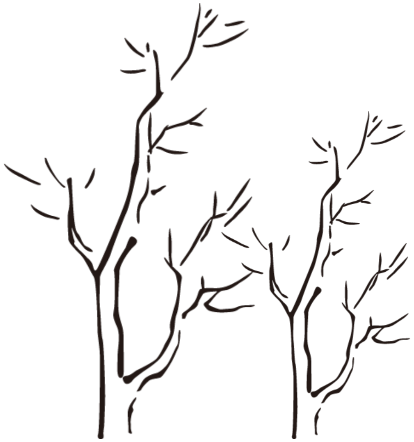
A Framework for Achieving KDM-CCA Secure Public-Key Encryption

Fuyuki Kitagawa (Tokyo Institute of Technology)
Keisuke Tanaka (Tokyo Institute of Technology)



Security notions for PKE

- It has been considered "IND-CCA security = standard"

  - ◆ takes active adversaries into consideration
  - ◆ implies non-malleability

Security notions for PKE

- It has been considered "IND-CCA security = standard"
  - ◆ takes active adversaries into consideration
  - ◆ implies non-malleability

- IND security falls short if an adversary can obtain side information of secret states

  → One typical example is encrypting secret keys

  e.g.) Anonymous credential, hard-disk encryption, FHE…

Security notions for PKE

- It has been considered “IND-CCA security = standard”

  ◆ takes active adversaries into consideration
  ◆ implies non-malleability

- IND security falls short if an adversary can obtain side information of secret states

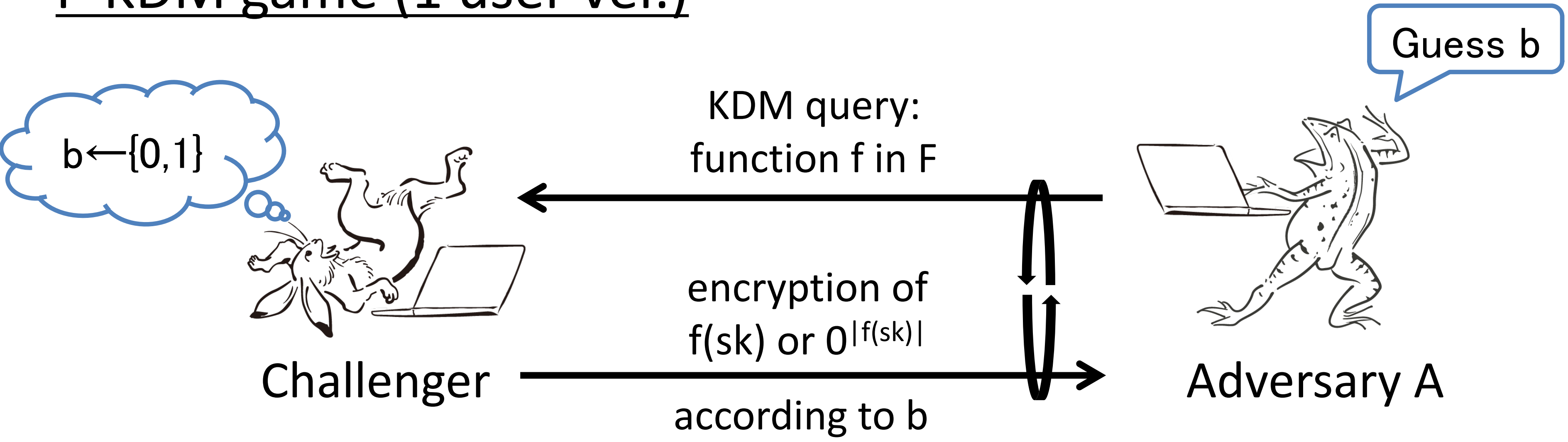
→ One typical example is encrypting secret keys

e.g.) Anonymous credential, hard-disk encryption, FHE…

⇨ Key dependent message (KDM) security [BRS02]

KDM security

F-KDM game (1 user ver.)



$b \leftarrow \{0,1\}$

Guess b

KDM query:
function f in F

encryption of
f(sk) or $0^{|f(sk)|}$
according to b

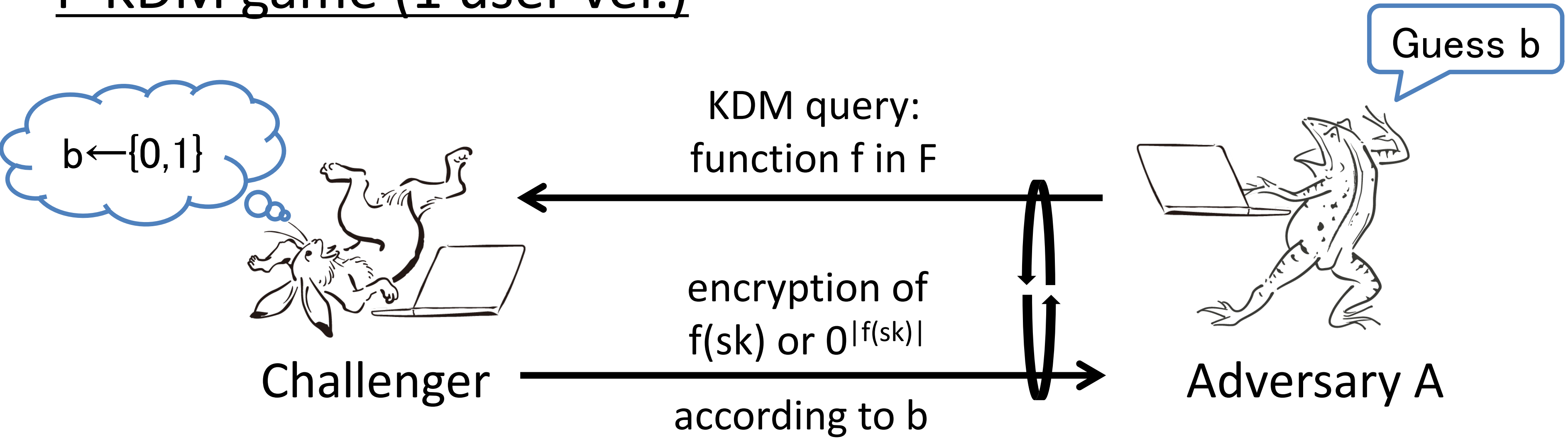
Challenger

Adversary A

A cannot guess b correctly with prob. greater than 1/2 $\Rightarrow$ F-KDM-CPA secure

KDM security

F-KDM game (1 user ver.)



Guess b

b←{0,1}

KDM query:
function f in F

encryption of
f(sk) or $0^{|f(sk)|}$
according to b

Challenger

Adversary A

A cannot guess b correctly
with prob. greater than 1/2 ⇒ F-KDM-CPA secure

- The adversary can also make a decryption query

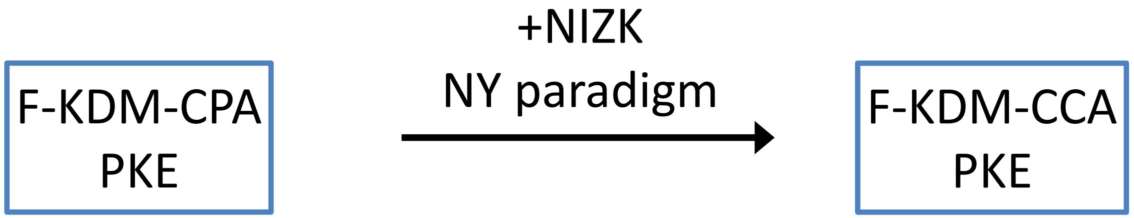
⇒ F-KDM-CCA secure

Our focus

Previous works on KDM-CCA

1. [CCS09]

F: any function class



F-KDM-CPA PKE

+NIZK NY paradigm

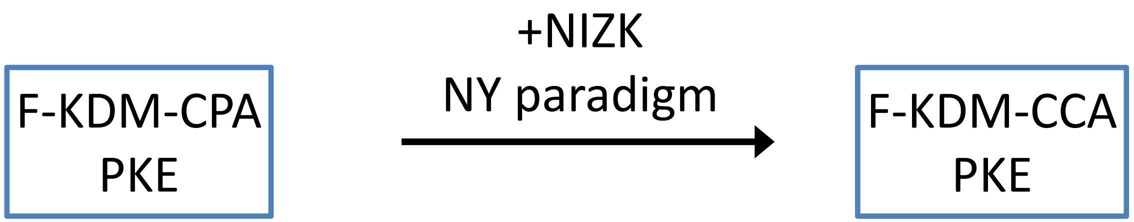
F-KDM-CCA PKE

Concrete instantiation: Aff-KDM-CCA PKE from DDH on pairing

Previous works on KDM-CCA

1. [CCS09]

F: any function class



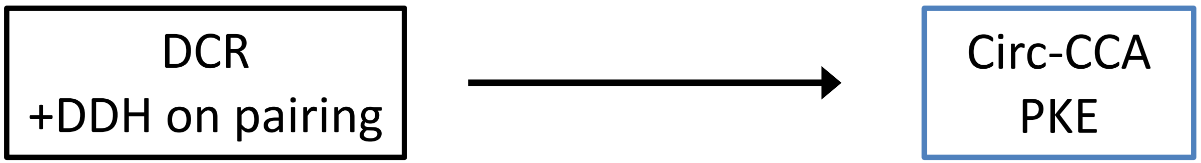
F-KDM-CPA PKE

+NIZK
NY paradigm

F-KDM-CCA PKE

Concrete instantiation: Aff-KDM-CCA PKE from DDH on pairing

2. [Hof13]



DCR
+DDH on pairing

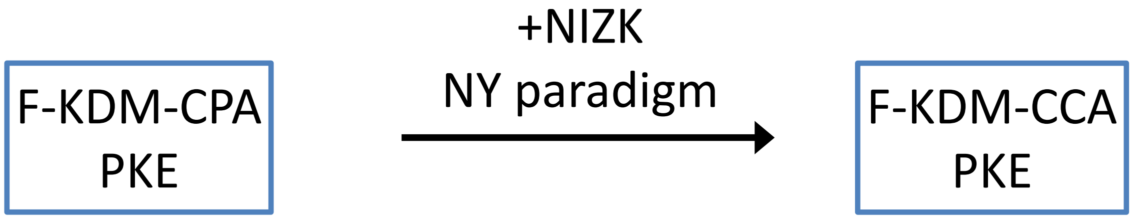
Circ-CCA PKE

More efficient than [CCS09]

Previous works on KDM-CCA

1. [CCS09]

F: any function class



F-KDM-CPA PKE

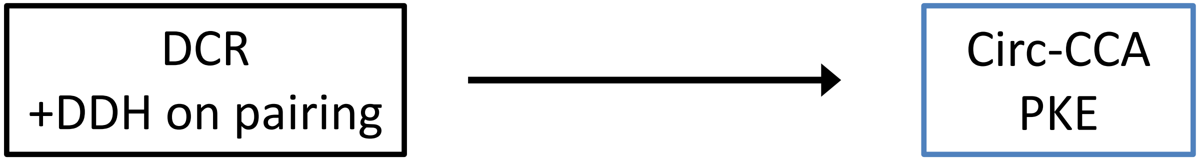
+NIZK

NY paradigm

F-KDM-CCA PKE

Concrete instantiation: Aff-KDM-CCA PKE from DDH on pairing

2. [Hof13]



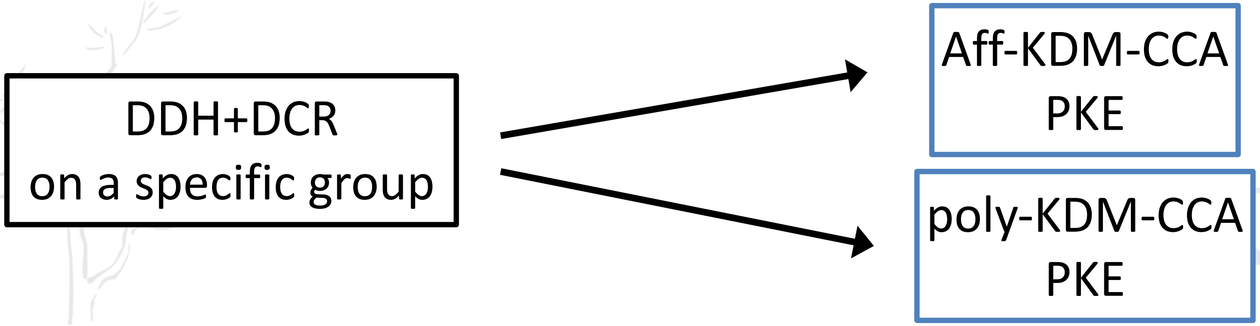
DCR +DDH on pairing

Circ-CCA PKE

More efficient than [CCS09]

3. [HLL16] (based on [LLJ15])

w/o pairing and efficient



DDH+DCR on a specific group

Aff-KDM-CCA PKE

poly-KDM-CCA PKE

Open problem

ALL existing KDM-CCA secure PKE rely on

NIZK
or
pairing
or
multiple assumptions
(DDH+DCR)

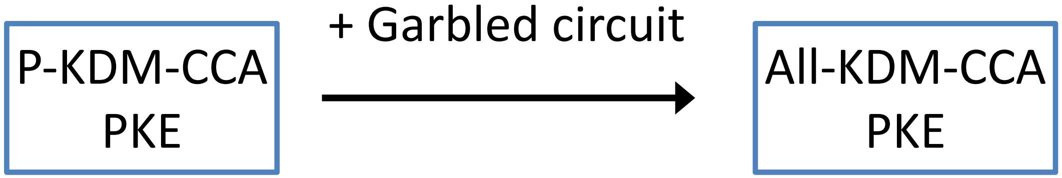
Especially,
all schemes w/o NIZK are proposed under DDH+DCR on a specific group
→ Removing one of them seems to be difficult

⇒ Construction based on a single assumption using neither NIZK nor pairing??

KDM-CCA for all functions

[App11] showed

P-KDM: KDM security w.r.t. projection functions



P-KDM-CCA PKE

+ Garbled circuit

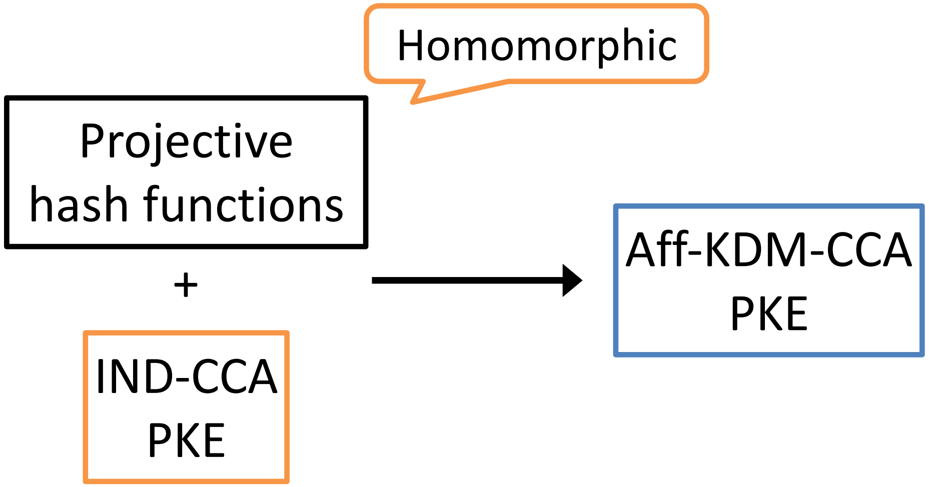
All-KDM-CCA PKE

Only [CCS09]'s scheme is compatible with this transformation
→ Need NIZK or pairing

All-KDM-CCA secure PKE
using neither NIZK nor pairing??

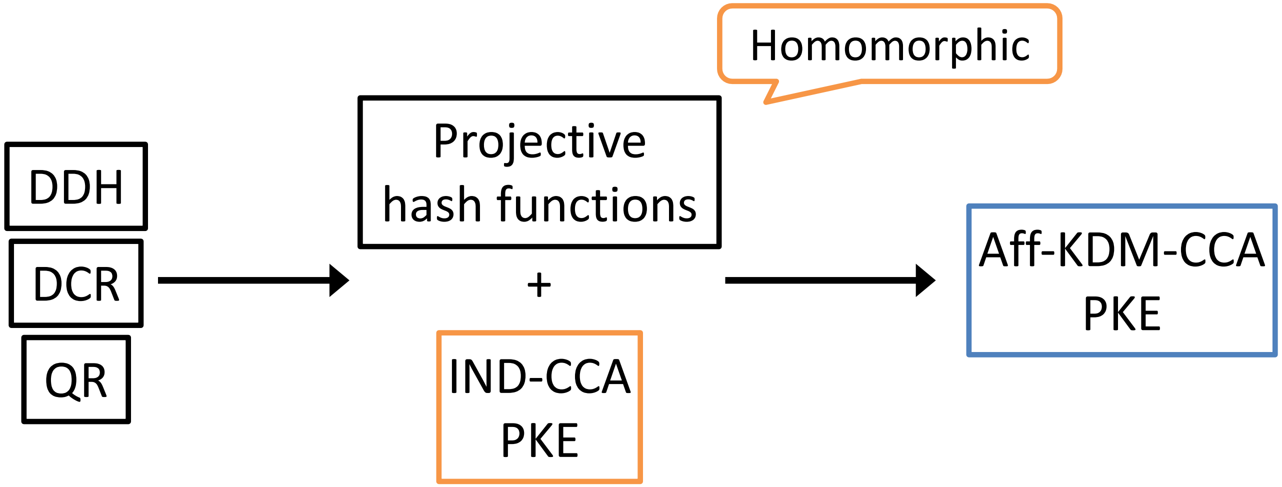
This work

1. A framework achieving KDM-CCA security in 1 user setting



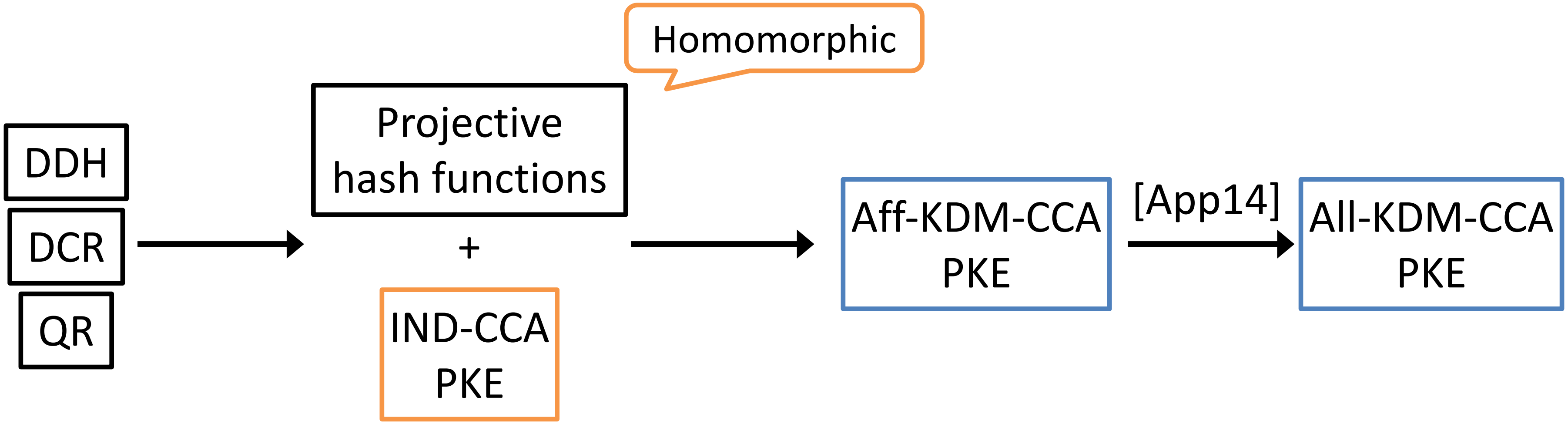
This work

1. A framework achieving KDM-CCA security in 1 user setting



This work

1. A framework achieving KDM-CCA security in 1 user setting



DDH

DCR

QR

Projective hash functions

+

IND-CCA PKE

Homomorphic

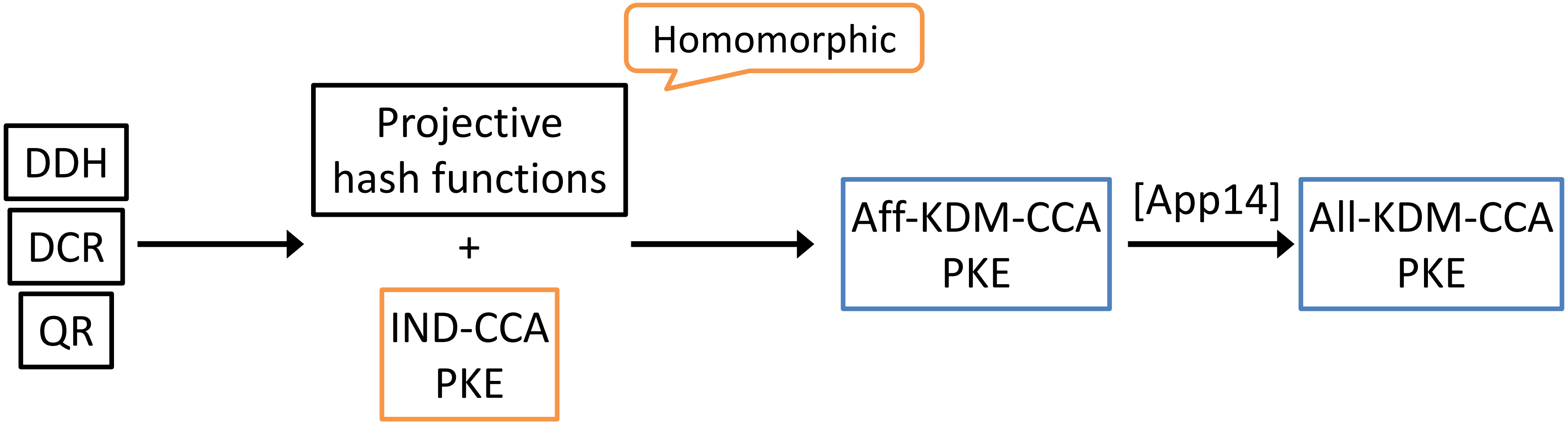
Aff-KDM-CCA PKE

[App14]

All-KDM-CCA PKE

This work

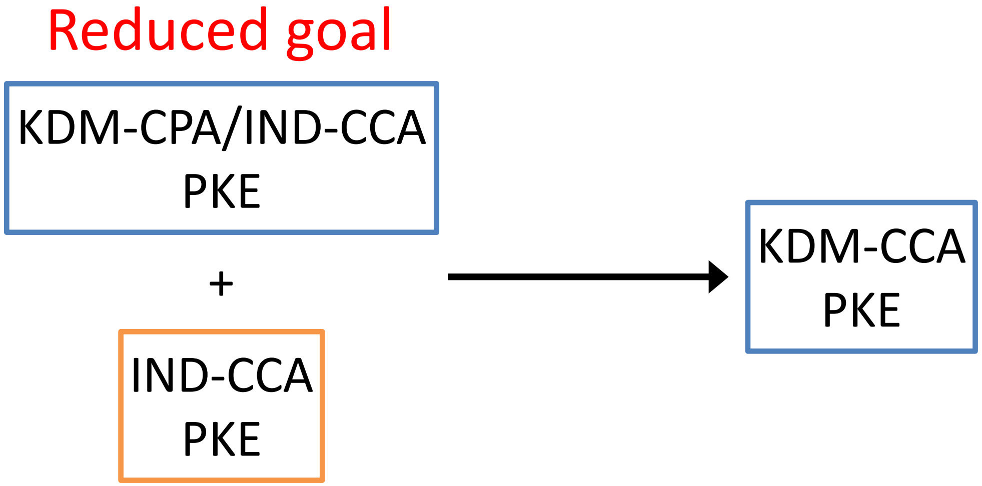
1. A framework achieving KDM-CCA security in 1 user setting



2. KDM-CCA security in multi user setting of concrete instantiations

Reduced goal

We essentially show



Reduced goal

KDM-CPA/IND-CCA PKE

+

IND-CCA PKE

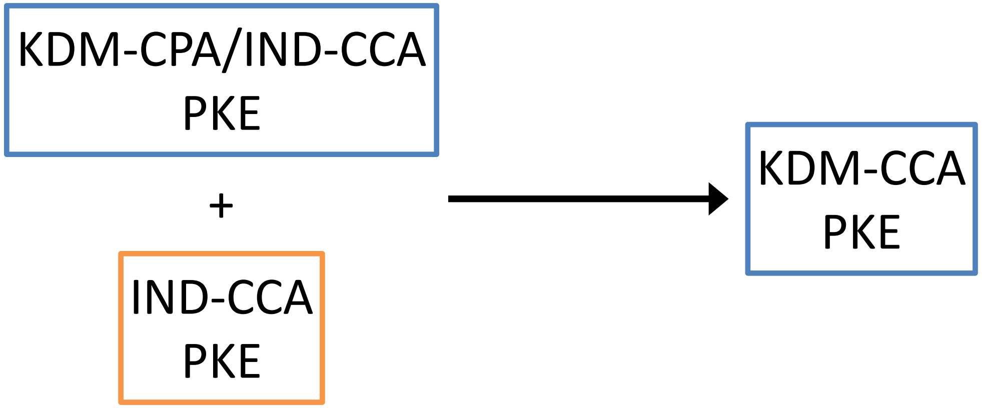
KDM-CCA PKE

Reduced goal

We essentially show

Much easier to construct than KDM-CCA PKE ☺

Reduced goal



KDM-CPA/IND-CCA PKE

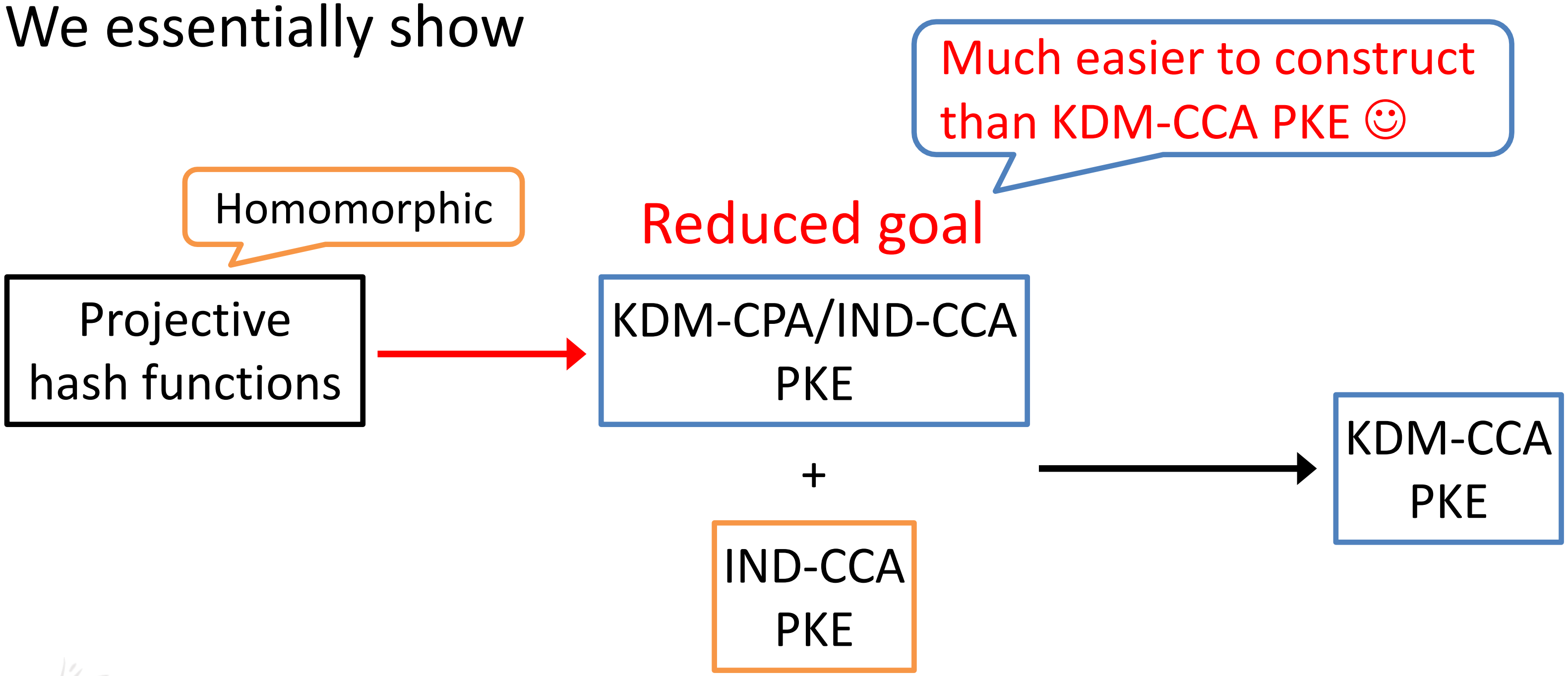
+

IND-CCA PKE

KDM-CCA PKE

Reduced goal

We essentially show



Much easier to construct than KDM-CCA PKE ☺

Homomorphic

Projective hash functions

Reduced goal

KDM-CPA/IND-CCA PKE

+

IND-CCA PKE

KDM-CCA PKE

Triple mode proof [MTY11]

- Framework for proving KDM-CPA

Standard mode

$$\boxed{\mathsf{E}(\mathsf{pk}, f(\mathsf{sk}))}$$

It is difficult to directly prove it based on the secrecy of secret-key… ☹

Hide mode

$$\boxed{\mathsf{E}(\mathsf{pk}, 0)}$$

Triple mode proof [MTY11]

- Framework for proving KDM-CPA

Standard mode

$$\boxed{\mathsf{E}(\mathsf{pk}, f(\mathsf{sk}))}$$

It is difficult to directly prove it based on the secrecy of secret-key… ☹

Fake mode

$$\boxed{\mathsf{Sim}(\mathsf{pk}, f)}$$

Hide mode

$$\boxed{\mathsf{E}(\mathsf{pk}, 0)}$$

Triple mode proof [MTY11]

- Framework for proving KDM-CPA

Standard mode

$$\boxed{\mathsf{E}(\mathsf{pk}, f(\mathsf{sk}))}$$

It is difficult to directly prove it based on the secrecy of secret-key… ☹

$\wr$ (1)

Using secrecy of encryption randomness

Fake mode

$$\boxed{\mathsf{Sim}(\mathsf{pk}, f)}$$

Hide mode

$$\boxed{\mathsf{E}(\mathsf{pk}, 0)}$$

Triple mode proof [MTY11]

- Framework for proving KDM-CPA

Standard mode

$\boxed{\mathsf{E}(\mathsf{pk}, f(\mathsf{sk}))}$

It is difficult to directly prove it based on the secrecy of secret-key… ☹

$\wr\wr$ (1)

Using secrecy of encryption randomness

Fake mode

$\boxed{\mathsf{Sim}(\mathsf{pk}, f)}$

$\wr\wr$ (2)

Using secrecy of secret-key

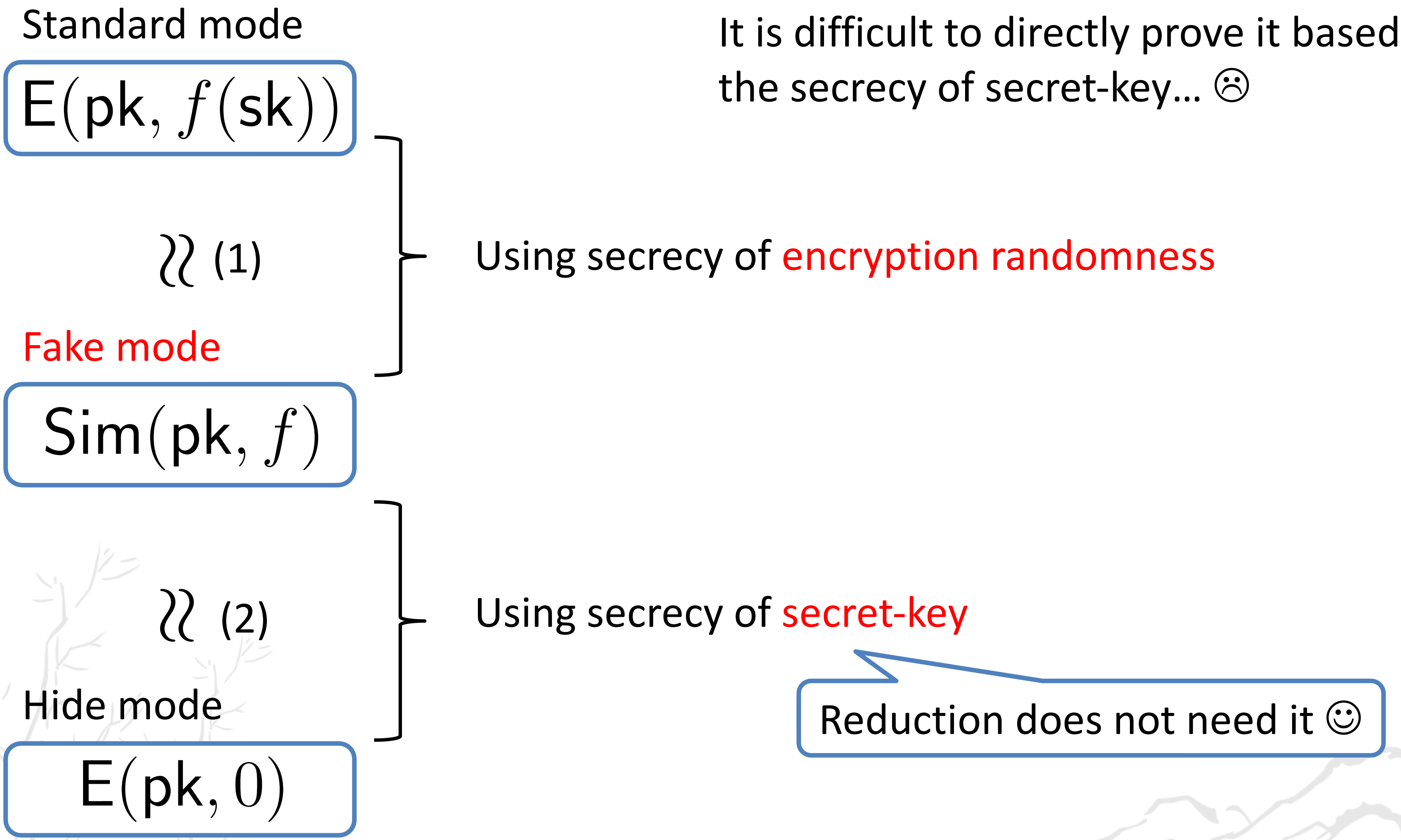
Hide mode

$\boxed{\mathsf{E}(\mathsf{pk}, 0)}$

Complete entire proof ☺

Triple mode proof [MTY11]

- Framework for proving KDM-CPA



Standard mode

$\mathsf{E}(\mathsf{pk}, f(\mathsf{sk}))$

It is difficult to directly prove it based on the secrecy of secret-key… ☹

$\approx$ (1)

Using secrecy of encryption randomness

Fake mode

$\mathsf{Sim}(\mathsf{pk}, f)$

$\approx$ (2)

Using secrecy of secret-key

Reduction does not need it ☺

Hide mode

$\mathsf{E}(\mathsf{pk}, 0)$

Complete entire proof ☺

Extension to CCA setting

Standard mode

$\boxed{\mathsf{E}(\mathsf{pk}, f(\mathsf{sk}))}$

$\approx$ (1)

Fake mode

$\boxed{\mathsf{Sim}(\mathsf{pk}, f)}$

Using secrecy of encryption randomness

→ Reduction knows secret-key

→ This step goes through when proving KDM-CCA ☺

$\approx$ (2)

Hide mode

$\boxed{\mathsf{E}(\mathsf{pk}, 0)}$

Using secrecy of secret-key

→Reduction does not know secret-key

→ This step fails when proving KDM-CCA ☹

Extension to CCA setting

Standard mode

$\mathrm{E}(\mathrm{pk}, f(\mathrm{sk}))$

$\wr$ (1)

Using secrecy of encryption randomness

→ Reduction knows secret-key
→ This step goes through when proving KDM-CCA ☺

Fake mode

$\mathrm{Sim}(\mathrm{pk}, f)$

$\wr$ (2)

Using secrecy of secret-key

→Reduction does not know secret-key
→ This step fails when proving KDM-CCA ☹

Hide mode

$\mathrm{E}(\mathrm{pk}, 0)$

We need new technique

First try

$$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$$

Outer scheme:
IND-CCA

Inner scheme:
KDM-CPA

Shown using
triple mode proof

Standard mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, f(\mathsf{sk})))}$$

$\wr$ (1)

Fake mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{Sim}(\mathsf{pk}, f))}$$

$\wr$ (2)

Hide mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, 0))}$$

Using IND-CCA of outer scheme

First try

$$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$$

Outer scheme:
IND-CCA

Inner scheme:
KDM-CPA

Shown using
triple mode proof

Standard mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, f(\mathsf{sk})))}$$

$\approx$ (1)

Fake mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{Sim}(\mathsf{pk}, f))}$$

$\approx$ (2)

Hide mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, 0))}$$

Using IND-CCA of outer scheme

Reduction can use

- sk of inner scheme
- decryption oracle for outer scheme

→ Simulate decryption oracle ☺

First try

$$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$$

Outer scheme:
IND-CCA

Inner scheme:
KDM-CPA

Shown using
triple mode proof

Standard mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, f(\mathsf{sk})))}$$

But, this idea has a problem…

$\wr\wr$ (1)

Fake mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{Sim}(\mathsf{pk}, f))}$$

$\wr\wr$ (2)

Hide mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, 0))}$$

Using IND-CCA of outer scheme

Reduction can use

- sk of inner scheme
- decryption oracle for outer scheme

→ Simulate decryption oracle ☺

How to maintain sk_{cca}??

$$\mathsf{E}_{\text{cca}}(\text{pk}_{\text{cca}}, \mathsf{E}(\text{pk}, m))$$

Outer scheme:
IND-CCA

Inner scheme:
KDM-CPA

Shown using
triple mode proof

Standard mode

$$\boxed{\mathsf{E}_{\text{cca}}(\text{pk}_{\text{cca}}, \mathsf{E}(\text{pk}, f(\text{sk})))}$$

$\approx$ (1)

Fake mode

$$\boxed{\mathsf{E}_{\text{cca}}(\text{pk}_{\text{cca}}, \text{Sim}(\text{pk}, f))}$$

$\approx$ (2)

Hide mode

$$\boxed{\mathsf{E}_{\text{cca}}(\text{pk}_{\text{cca}}, \mathsf{E}(\text{pk}, 0))}$$

How to maintain $\mathsf{sk}_{\mathsf{cca}}$??

$$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$$

Outer scheme:
IND-CCA

Inner scheme:
KDM-CPA

Shown using triple mode proof

Standard mode

$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, \cancel{f(\mathsf{sk})}))$

If we maintain $\mathsf{sk}_{\mathsf{cca}}$ as a part of secret-key

$f(\mathsf{sk}||\mathsf{sk}_{\mathsf{cca}})$ is encrypted

$\approx$ (1)

Fake mode

$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{Sim}(\mathsf{pk}, f))$

$\approx$ (2)

Hide mode

$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, 0))$

How to maintain $\mathsf{sk}_{\mathsf{cca}}$??

$$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$$

Outer scheme:
IND-CCA

Inner scheme:
KDM-CPA

Shown using triple mode proof

Standard mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, \cancel{f(\mathsf{sk})}))}$$

$\wr\wr$ (1)

Fake mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{Sim}(\mathsf{pk}, f))}$$

$\wr\wr$ (2)

Hide mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, 0))}$$

If we maintain $\mathsf{sk}_{\mathsf{cca}}$ as a part of secret-key

$f(\mathsf{sk} \| \mathsf{sk}_{\mathsf{cca}})$ is encrypted

→ There is a circularity involving outer scheme

Not KDM secure

How to maintain $\mathsf{sk}_{\mathsf{cca}}$??

$$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$$

Outer scheme:
IND-CCA

Inner scheme:
KDM-CPA

Shown using
triple mode proof

Standard mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, \cancel{f(\mathsf{sk})}))}$$

$\wr$ (1)

Fake mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{Sim}(\mathsf{pk}, \cancel{f}))}$$

$\wr$ (2)

Hide mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, 0))}$$

If we maintain $\mathsf{sk}_{\mathsf{cca}}$ as a part of secret-key

$f(\mathsf{sk}\|\mathsf{sk}_{\mathsf{cca}})$ is encrypted

→ There is a circularity involving outer scheme

Not KDM secure

$f(\cdot\|\mathsf{sk}_{\mathsf{cca}})$ The circularity remains after completing step (1)

→ We need to remove it to use IND-CCA of outer scheme

How to maintain $\mathsf{sk}_{\mathsf{cca}}$??

$$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$$

Outer scheme:
IND-CCA

Inner scheme:
KDM-CPA

Shown using
triple mode proof

Standard mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, \cancel{f(\mathsf{sk})}))}$$

$\wr$ (1)

Fake mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{Sim}(\mathsf{pk}, \cancel{f}))}$$

$\wr$ (2)

Hide mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, 0))}$$

If we maintain $\mathsf{sk}_{\mathsf{cca}}$ as a part of secret-key

$f(\mathsf{sk}\|\mathsf{sk}_{\mathsf{cca}})$ is encrypted

→ There is a circularity involving outer scheme

Not KDM secure

$f(\cdot\|\mathsf{sk}_{\mathsf{cca}})$

The circularity remains
after completing step (1)
→ We need to remove it
to use IND-CCA of outer scheme

⇨ We can do it if inner scheme is also IND-CCA

Remove circularity

- We maintain $\mathsf{sk}_{\mathsf{cca}}$ as a part of public-key after encrypted by inner scheme

Public-key: $(\mathsf{pk}, \mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, \mathsf{sk}_{\mathsf{cca}}))$

Secret-key: sk

Encryption: $\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$

Decryption: Reject $\mathsf{E}(\mathsf{pk}, \mathsf{sk}_{\mathsf{cca}})$

Remove circularity

- We maintain $\mathsf{sk}_{\mathsf{cca}}$ as a part of public-key after encrypted by inner scheme

Public-key: $(\mathsf{pk}, \mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, \mathsf{sk}_{\mathsf{cca}}))$

Encryption: $\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$

Secret-key: sk

Decryption: Reject $\mathsf{E}(\mathsf{pk}, \mathsf{sk}_{\mathsf{cca}})$

Standard mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, f(\mathsf{sk})))}$$

Remove circularity

- We maintain $\mathsf{sk}_{\mathsf{cca}}$ as a part of public-key after encrypted by inner scheme

Public-key: $(\mathsf{pk}, \mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, \mathsf{sk}_{\mathsf{cca}}))$

Secret-key: sk

Encryption: $\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$

Decryption: Reject $\mathsf{E}(\mathsf{pk}, \mathsf{sk}_{\mathsf{cca}})$

Standard mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, f(\mathsf{sk})))}$$

$\wr\wr$ (1)

Fake mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{Sim}(\mathsf{pk}, f))}$$

Using the property of inner scheme

Remove circularity

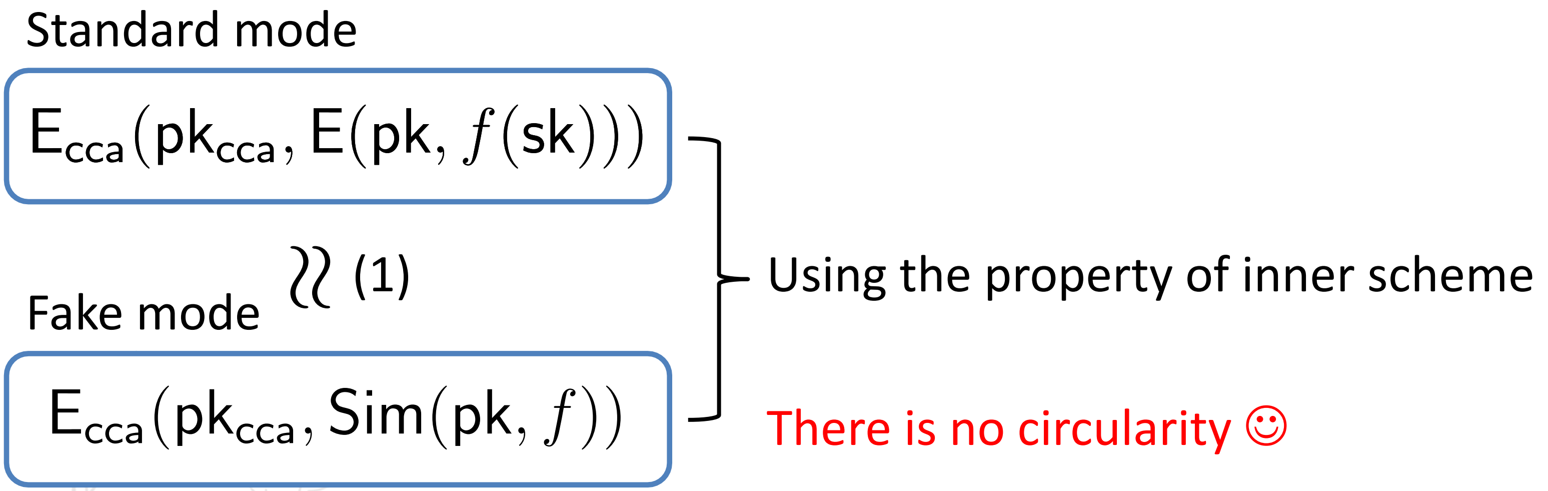
- We maintain $\mathsf{sk}_{\mathsf{cca}}$ as a part of public-key after encrypted by inner scheme

Public-key: $(\mathsf{pk}, \mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, \mathsf{sk}_{\mathsf{cca}}))$

Secret-key: sk

Encryption: $\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$

Decryption: Reject $\mathsf{E}(\mathsf{pk}, \mathsf{sk}_{\mathsf{cca}})$



Standard mode

$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, f(\mathsf{sk})))$

$\wr\wr$ (1)

Fake mode

$\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{Sim}(\mathsf{pk}, f))$

Using the property of inner scheme

There is no circularity ☺

Remaining proof strategy

- We maintain $\mathsf{sk}_{\mathsf{cca}}$ as a part of public-key after encrypted by inner scheme

Public-key: $(\mathsf{pk}, \mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, \mathsf{sk}_{\mathsf{cca}}))$ Encryption: $\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, m))$

Secret-key: sk

Eliminate it before step (2)
using IND-CCA of inner scheme

Reduction needs
to simulate decryption oracle

Standard mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{E}(\mathsf{pk}, f(\mathsf{sk})))}$$

$\wr\wr$ (1)

Fake mode

$$\boxed{\mathsf{E}_{\mathsf{cca}}(\mathsf{pk}_{\mathsf{cca}}, \mathsf{Sim}(\mathsf{pk}, f))}$$

Using the property of inner scheme

There is no circularity ☺

Remaining proof strategy

- We maintain $\mathrm{sk}_{\mathrm{cca}}$ as a part of public-key after encrypted by inner scheme

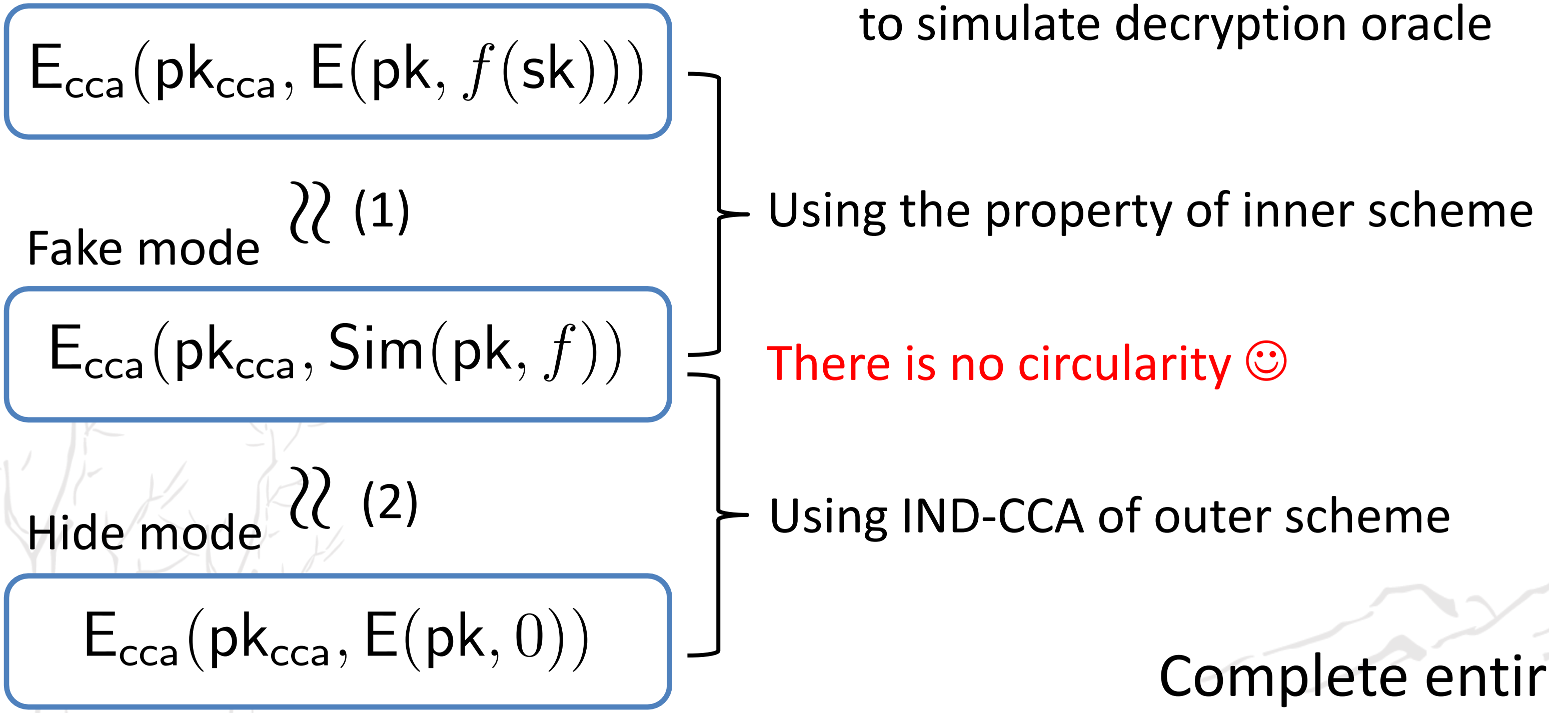
Public-key: $(\mathrm{pk}, \mathrm{pk}_{\mathrm{cca}}, \mathrm{E}(\mathrm{pk}, \mathrm{sk}_{\mathrm{cca}}))$ Encryption: $\mathrm{E}_{\mathrm{cca}}(\mathrm{pk}_{\mathrm{cca}}, \mathrm{E}(\mathrm{pk}, m))$

Secret-key: sk

Eliminate it before step (2)
using IND-CCA of inner scheme

Reduction needs
to simulate decryption oracle

Standard mode



$\mathrm{E}_{\mathrm{cca}}(\mathrm{pk}_{\mathrm{cca}}, \mathrm{E}(\mathrm{pk}, f(\mathrm{sk})))$

$\approx$ (1)

Fake mode

$\mathrm{E}_{\mathrm{cca}}(\mathrm{pk}_{\mathrm{cca}}, \mathrm{Sim}(\mathrm{pk}, f))$

$\approx$ (2)

Hide mode

$\mathrm{E}_{\mathrm{cca}}(\mathrm{pk}_{\mathrm{cca}}, \mathrm{E}(\mathrm{pk}, 0))$

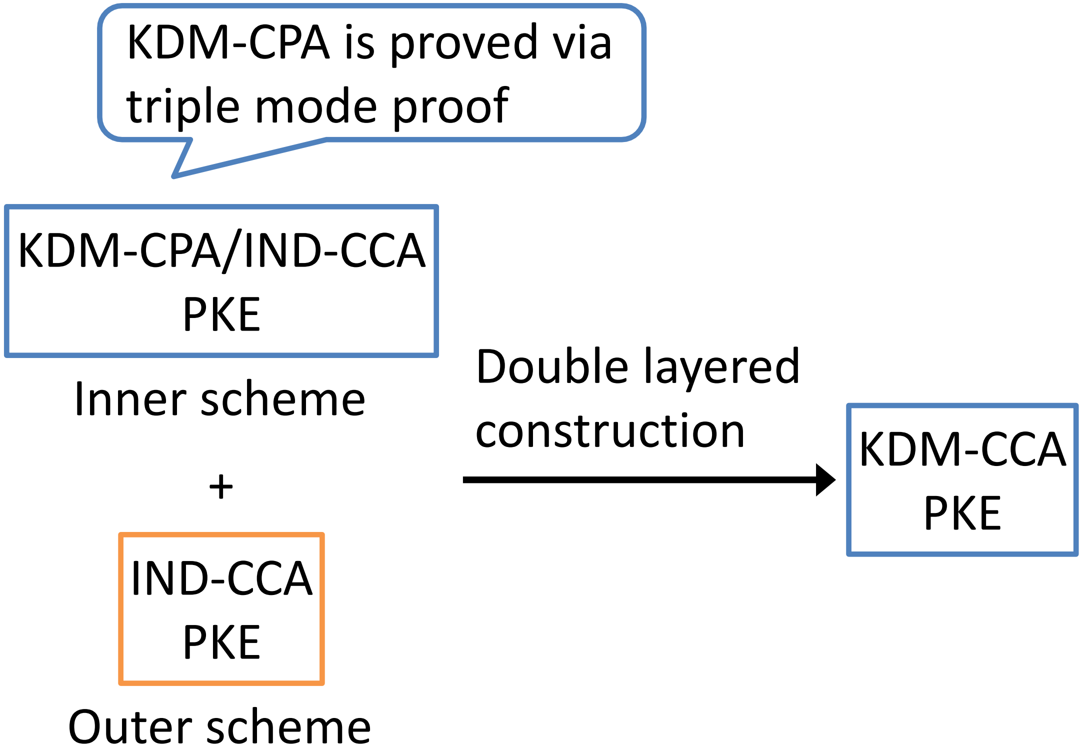
Using the property of inner scheme

There is no circularity ☺

Using IND-CCA of outer scheme

Complete entire proof ☺

Reduced goal



KDM-CPA is proved via triple mode proof

KDM-CPA/IND-CCA PKE

Inner scheme

+

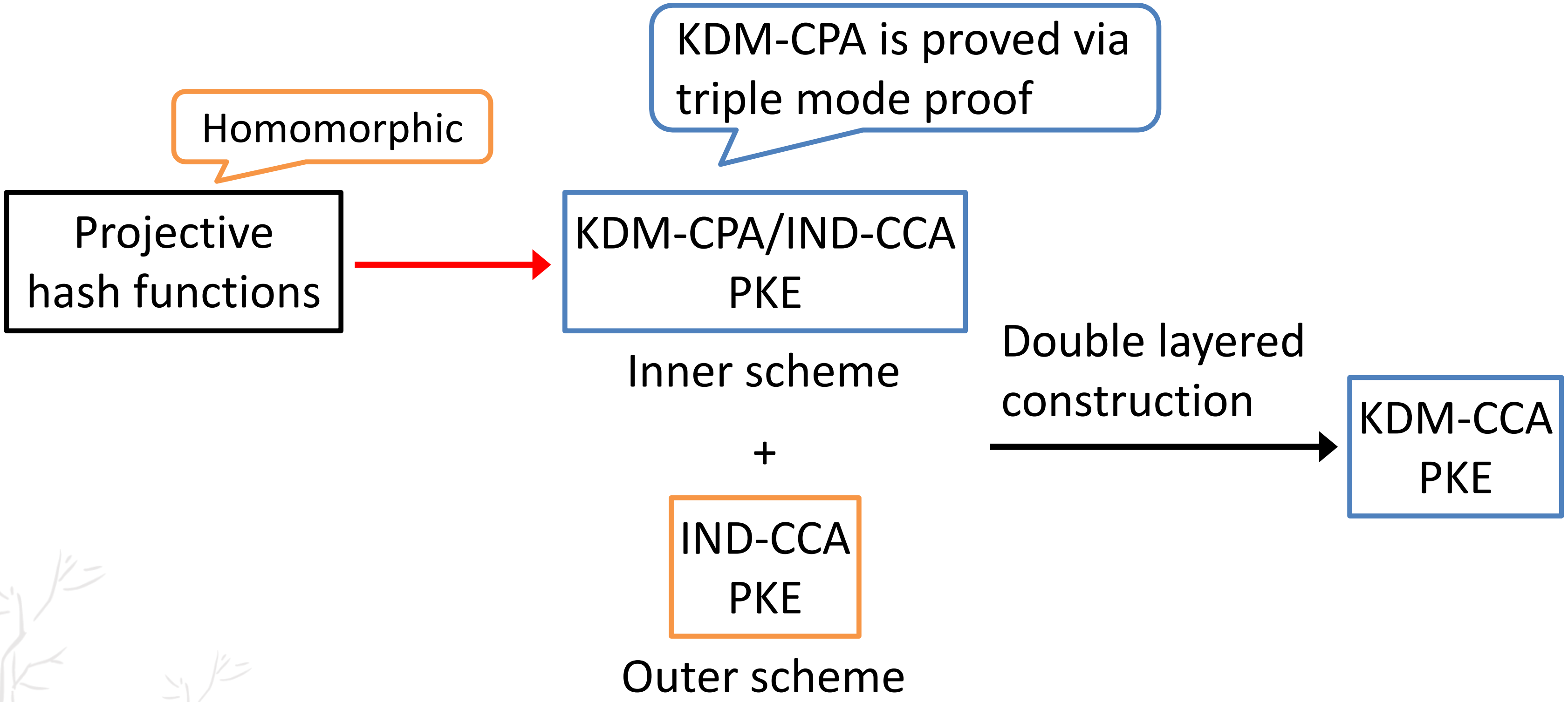
IND-CCA PKE

Outer scheme

Double layered construction

KDM-CCA PKE

Reduced goal



Homomorphic

Projective hash functions

KDM-CPA is proved via triple mode proof

KDM-CPA/IND-CCA PKE

Inner scheme

+

IND-CCA PKE

Outer scheme

Double layered construction

KDM-CCA PKE

Inner scheme from PHF

- Extend [Wee16] that is a generalization of [BHHO08,BG10]

Homomorphic and Smooth PHF

KDM-CPA PKE

Inner scheme from PHF

- Extend [Wee16] that is a generalization of [BHHO08,BG10]

Homomorphic and Smooth PHF ⇒ KDM-CPA PKE

- KDM-CPA of Wee’s scheme can be shown using triple mode proof

$$\begin{array}{c}\boxed{\mathsf{E}(\mathsf{pk}, f(\mathsf{sk}))} \\ \wr\wr \\ \boxed{\mathsf{Sim}(\mathsf{pk}, f)}\end{array}\Bigg\}$$

Homomorphism
projective property
Subset membership problem

Using the secrecy of encryption randomness

Inner scheme from PHF

- Extend [Wee16] that is a generalization of [BHHO08,BG10]

Homomorphic and Smooth PHF ⇒ KDM-CPA PKE

- KDM-CPA of Wee's scheme can be shown using triple mode proof

$$\left.\begin{array}{c}\boxed{\mathsf{E}(\mathsf{pk}, f(\mathsf{sk}))} \\ \wr\wr \\ \boxed{\mathsf{Sim}(\mathsf{pk}, f)}\end{array}\right\}$$

Homomorphism

projective property

Subset membership problem

Using the secrecy of encryption randomness

- It is also IND-CPA

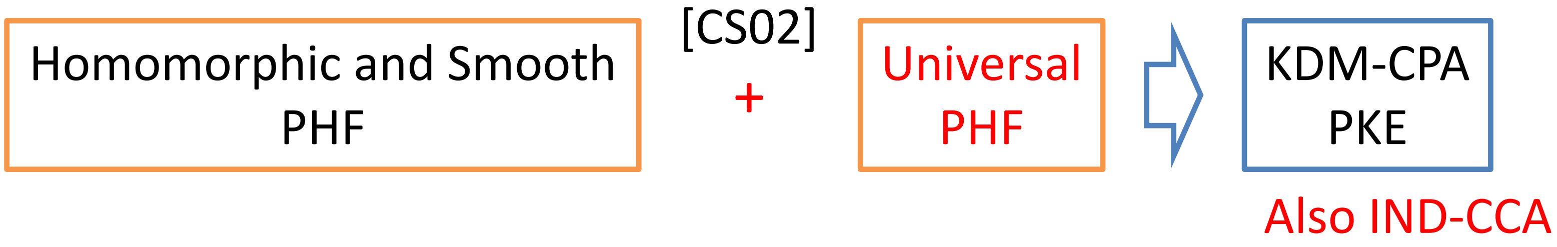
$$\left.\begin{array}{c}\boxed{\mathsf{E}(\mathsf{pk}, \mathsf{sk}_{\mathsf{cca}})} \\ \wr\wr \\ \boxed{\mathsf{E}(\mathsf{pk}, 0)}\end{array}\right\}$$

Smoothness

Subset membership problem

Inner scheme from PHF

- Extend [Wee16] that is a generalization of [BHHO08,BG10]


Homomorphic and Smooth PHF

[CS02]

+

Universal PHF

KDM-CPA PKE

Also IND-CCA

- KDM-CPA of Wee's scheme can be shown using triple mode proof

$\mathrm{E}(\mathrm{pk}, f(\mathrm{sk}))$

$\approx$

$\mathrm{Sim}(\mathrm{pk}, f)$

Homomorphism
projective property
Subset membership problem

Using the secrecy of encryption randomness

- It is also IND-CPA

$\mathrm{E}(\mathrm{pk}, \mathrm{sk}_{\mathrm{cca}})$

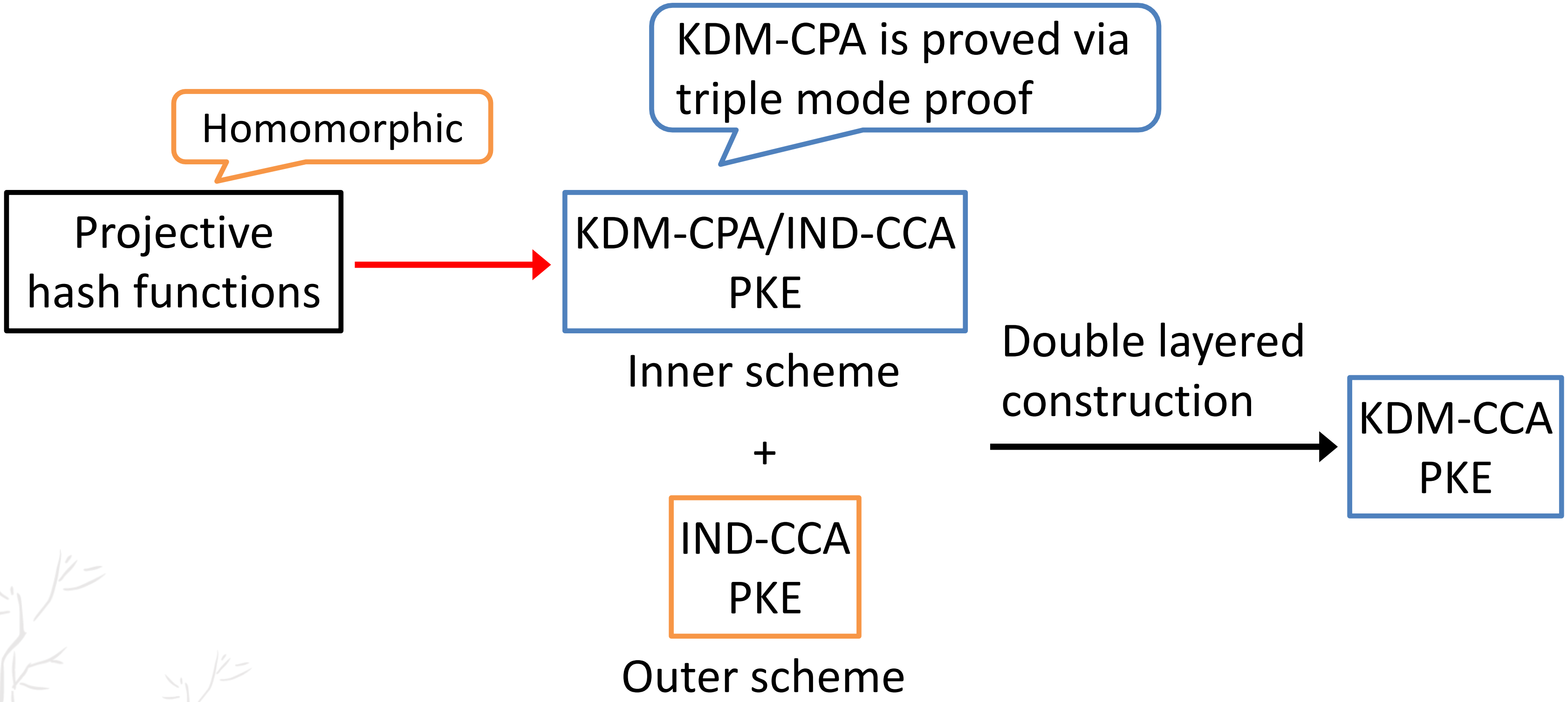
$\approx$

$\mathrm{E}(\mathrm{pk}, 0)$

Smoothness
Subset membership problem

By using universal PHF,
we can upgrade it into IND-CCA ☺

Our generic construction



Homomorphic

Projective hash functions

KDM-CPA is proved via triple mode proof

KDM-CPA/IND-CCA PKE

Inner scheme

+

IND-CCA PKE

Outer scheme

Double layered construction

KDM-CCA PKE

Instantiations and multi-user security

Instantiations

We can instantiate inner scheme based on instantiations of [Wee16]

DDH [BHHO08]

DCR/QR [BG10]

Instantiations and multi-user security

Instantiations

We can instantiate inner scheme based on instantiations of [Wee16]

DDH [BHHO08]
DCR/QR [BG10]

⇒ We obtain KDM-CCA PKE based on [DDH / DCR / QR

Instantiations and multi-user security

Instantiations

We can instantiate inner scheme based on instantiations of [Wee16]

DDH [BHHO08]
DCR/QR [BG10]

⇒ We obtain KDM-CCA PKE based on [DDH
DCR
QR

1-user/multi-user

Our generic construction achieves only KDM-CCA in 1-user setting
However, we can prove KDM-CCA in multi-user setting of instantiations

Instantiations and multi-user security

Instantiations

We can instantiate inner scheme based on instantiations of [Wee16]

DDH [BHHO08]
DCR/QR [BG10]

⇨ We obtain KDM-CCA PKE based on

- DDH
- DCR
- QR

1-user/multi-user

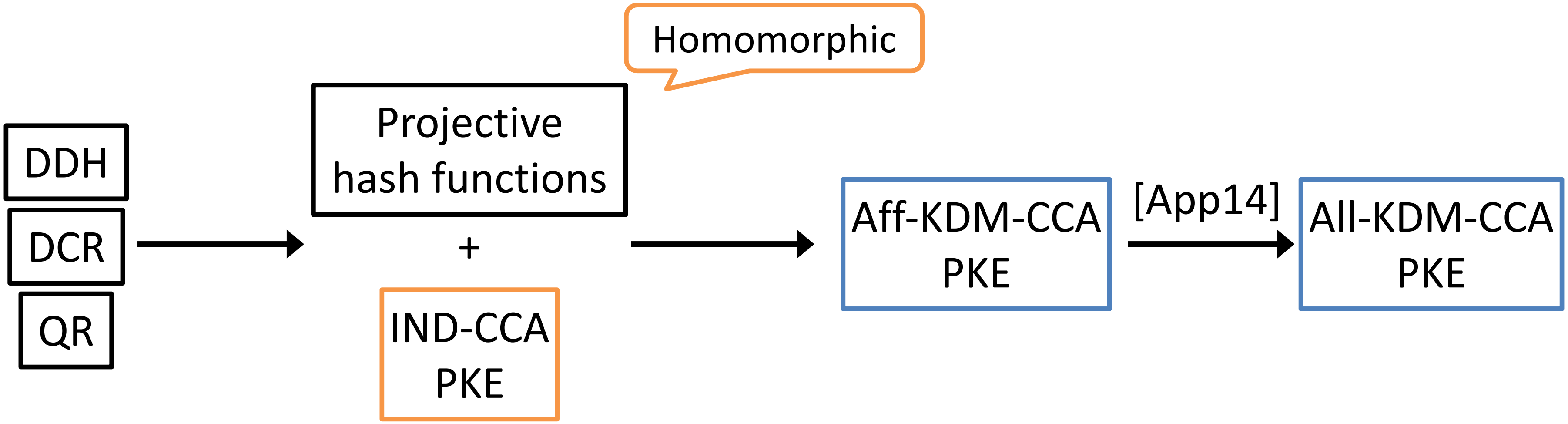
Our generic construction achieves only KDM-CCA in 1-user setting
However, we can prove KDM-CCA in multi-user setting of instantiations

Our instantiations are extensions of [BHHO08, BG10]

Using similar technique as them,
we can prove multi-user security ☺

Summary

1. A framework achieving KDM-CCA security in 1 user setting



Homomorphic

DDH

DCR

QR

Projective hash functions

+

IND-CCA PKE

Aff-KDM-CCA PKE

[App14]

All-KDM-CCA PKE

2. KDM-CCA security in multi user setting of concrete instantiations