Public Key Encryption with Equality Test from Tag-Based Encryption

○Masayuki Tezuka　　Keisuke Tanaka

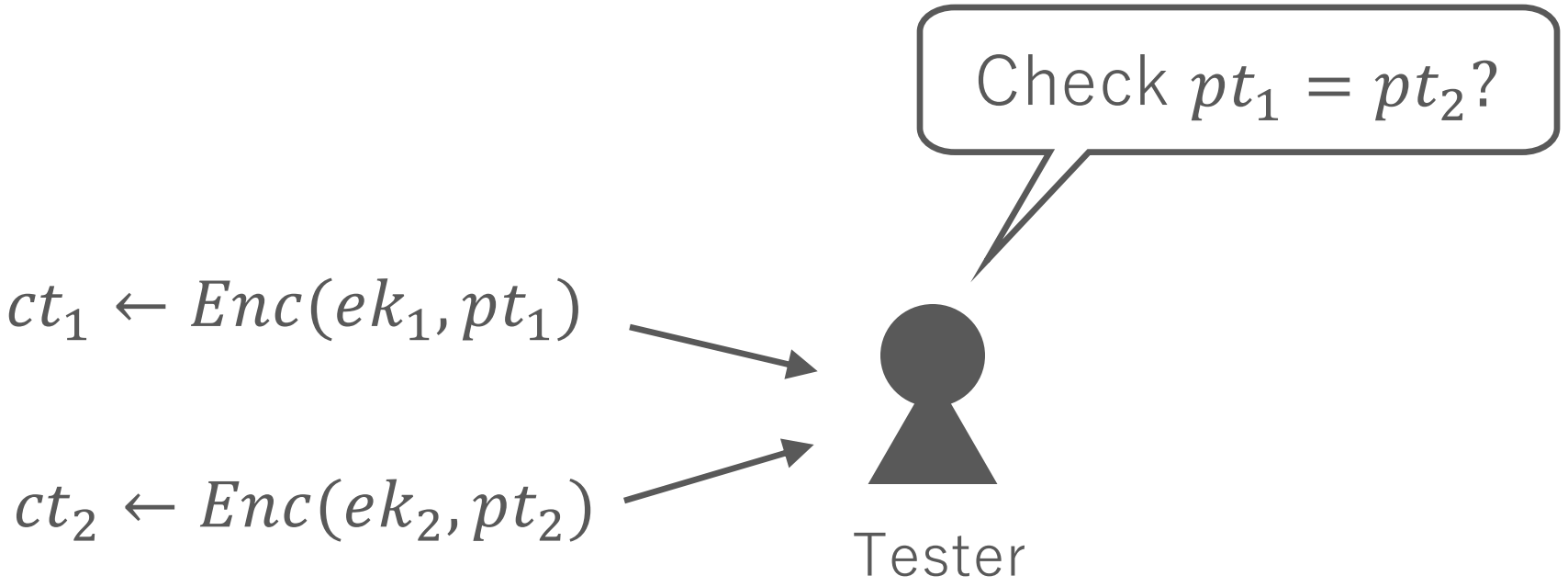
Institute of Science Tokyo

Version 2025/11/26

@ IWSEC 2025, Fukuoka, Japan

Background of PKEET

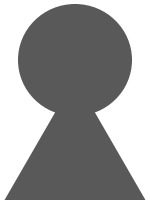
Public Key Encryption with Equality Test (PKEET) [YTHW10]



This test can be applied ciphertexts generated by different encryption keys.

[YTHW10] Yang, Tan, Huang, and Wong. Probabilistic public key encryption with equality test. CT-RSA 2010

Problem in Original PKEET by Yang et al. [YTHW10]



Tester (Anyone)

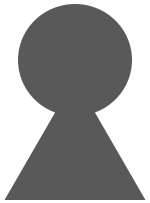
Anyone can be a tester and perform equality tests.

Problem

Anyone can obtain information related to plaintext.

[YTHW10] Yang, Tan, Huang, and Wong. Probabilistic public key encryption with equality test. CT-RSA 2010

Problem in Original PKEET by Yang et al. [YTHW10]



Tester (Anyone)

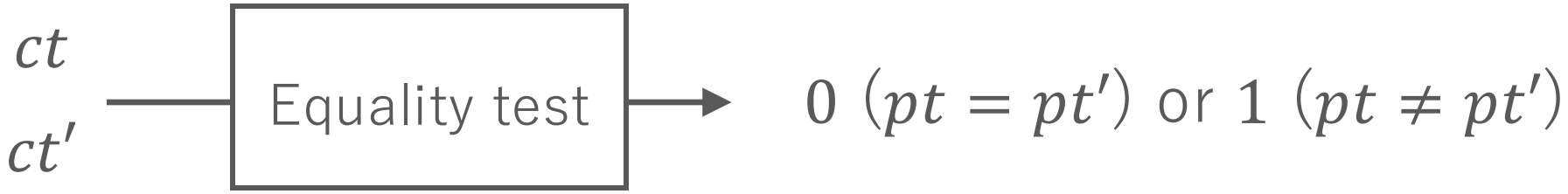
Anyone can be a tester and perform equality tests.

Problem

Anyone can obtain information related to plaintext.

For example, there is ct on an unknown plaintext pt.

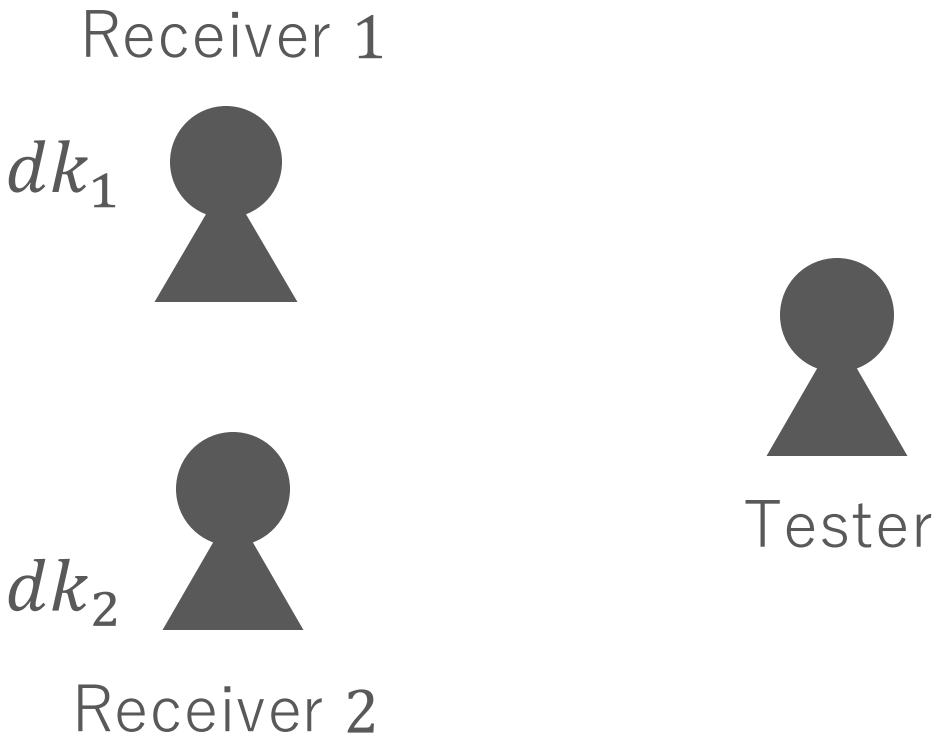
Anyone can freely choose pt' and generates its ciphertext ct'.



[YTHW10] Yang, Tan, Huang, and Wong. Probabilistic public key encryption with equality test. CT-RSA 2010

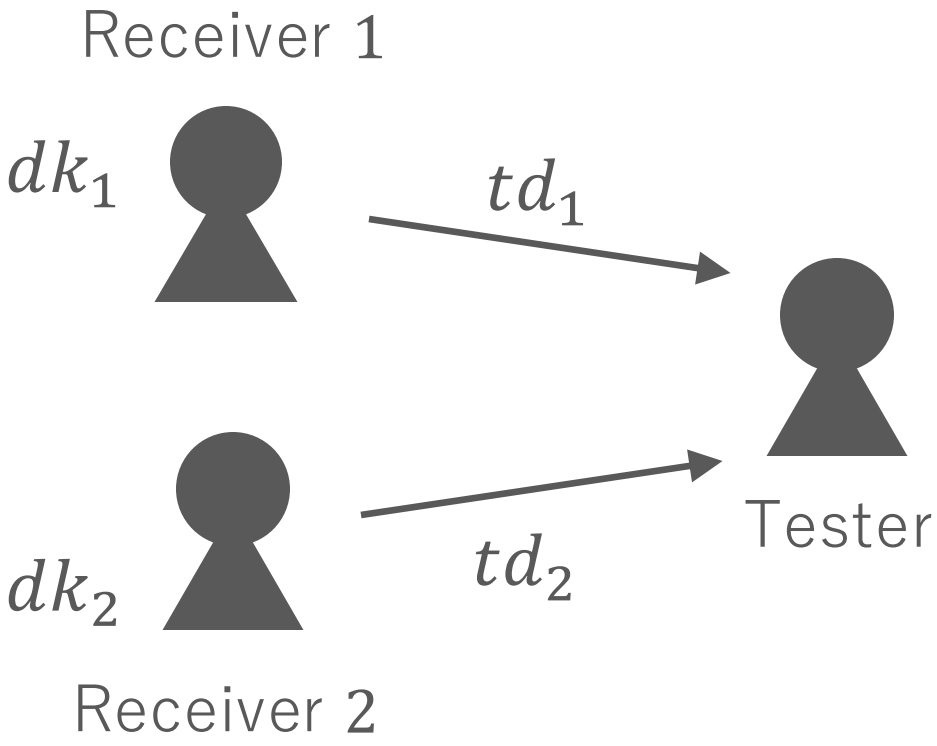
Restricting Equality Tests with Trapdoor Delegation

Trapdoor was introduced to limit who can perform equality tests.



Restricting Equality Tests with Trapdoor Delegation

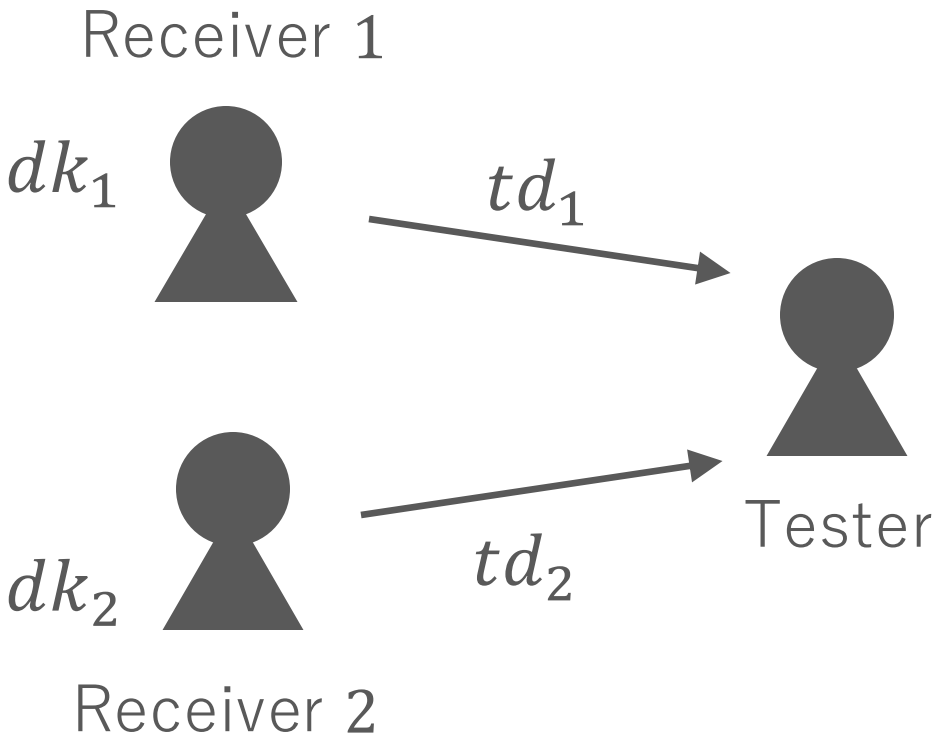
Trapdoor was introduced to limit who can perform equality tests.



The tester who is delegated trapdoors from receivers performs equality tests.

Restricting Equality Tests with Trapdoor Delegation

Trapdoor was introduced to limit who can perform equality tests.



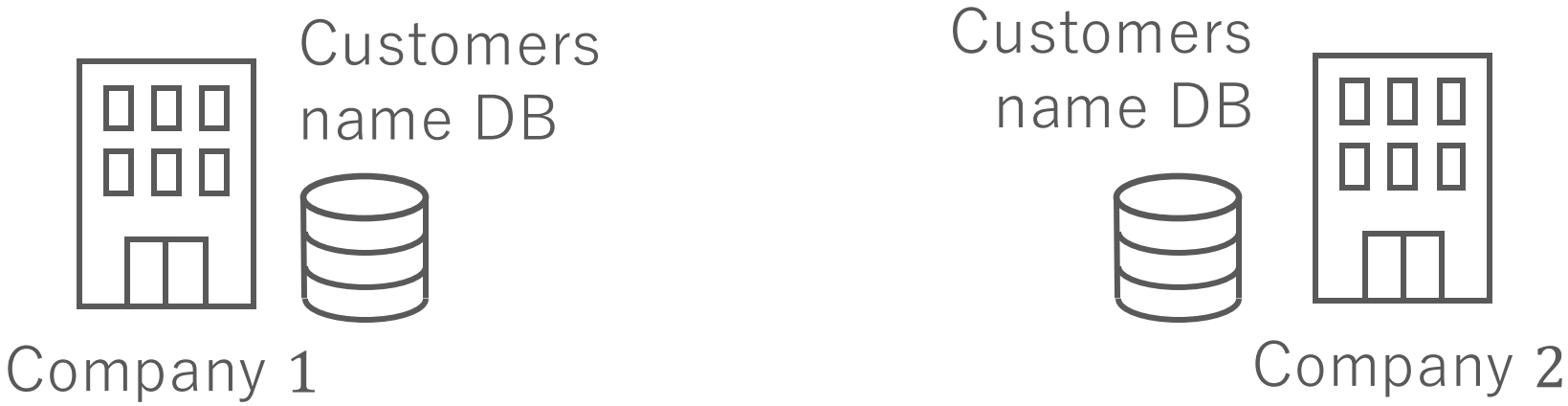
The tester who is delegated trapdoors from receivers performs equality tests.

From now on, we consider this type of PKEET.

Application of PKEET

- Keyword search on encrypted data
- Encrypted data partitioning
- Personal health record system
- Encrypted Database

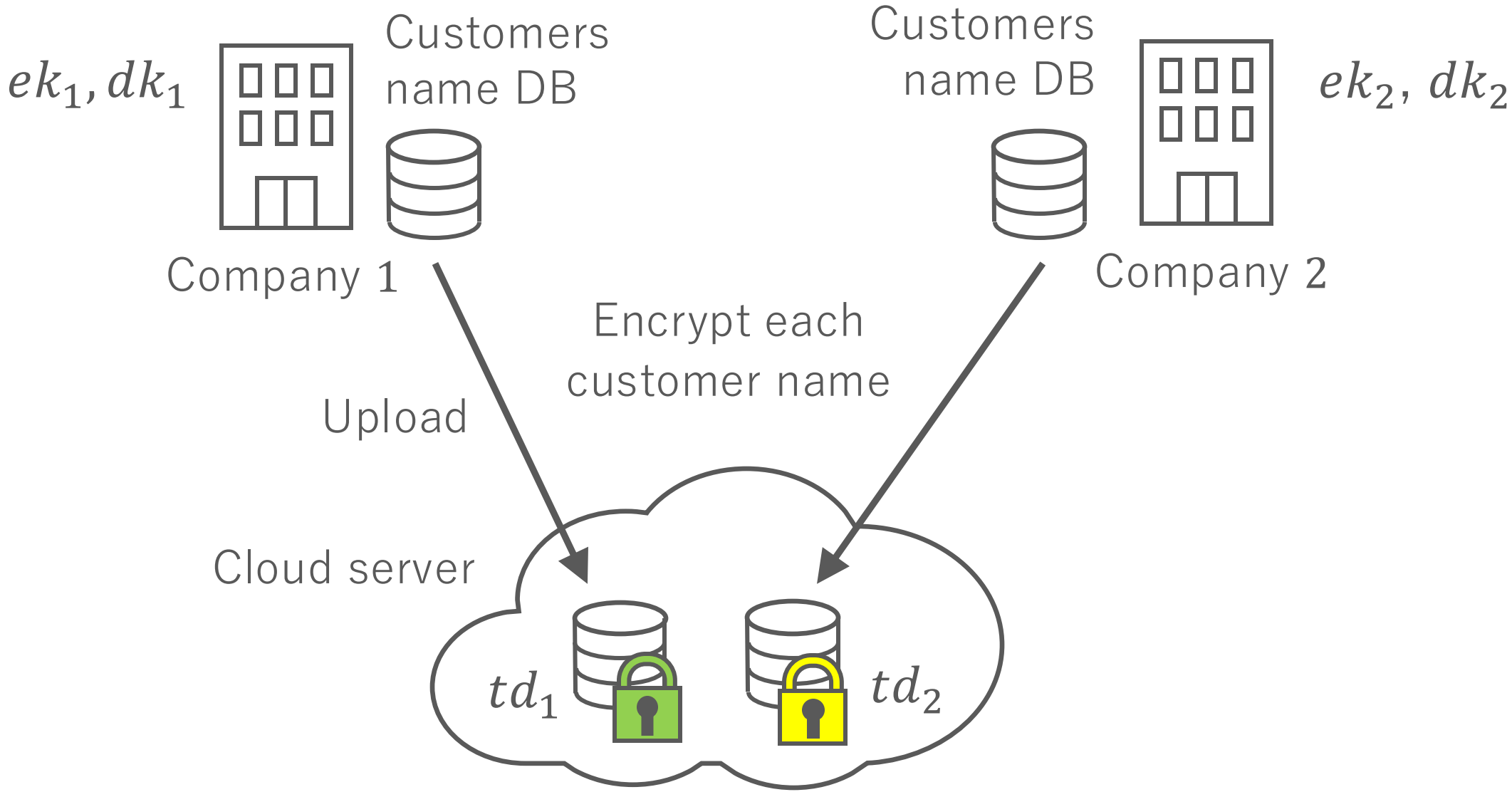
Application of PKEET: Encrypted Database



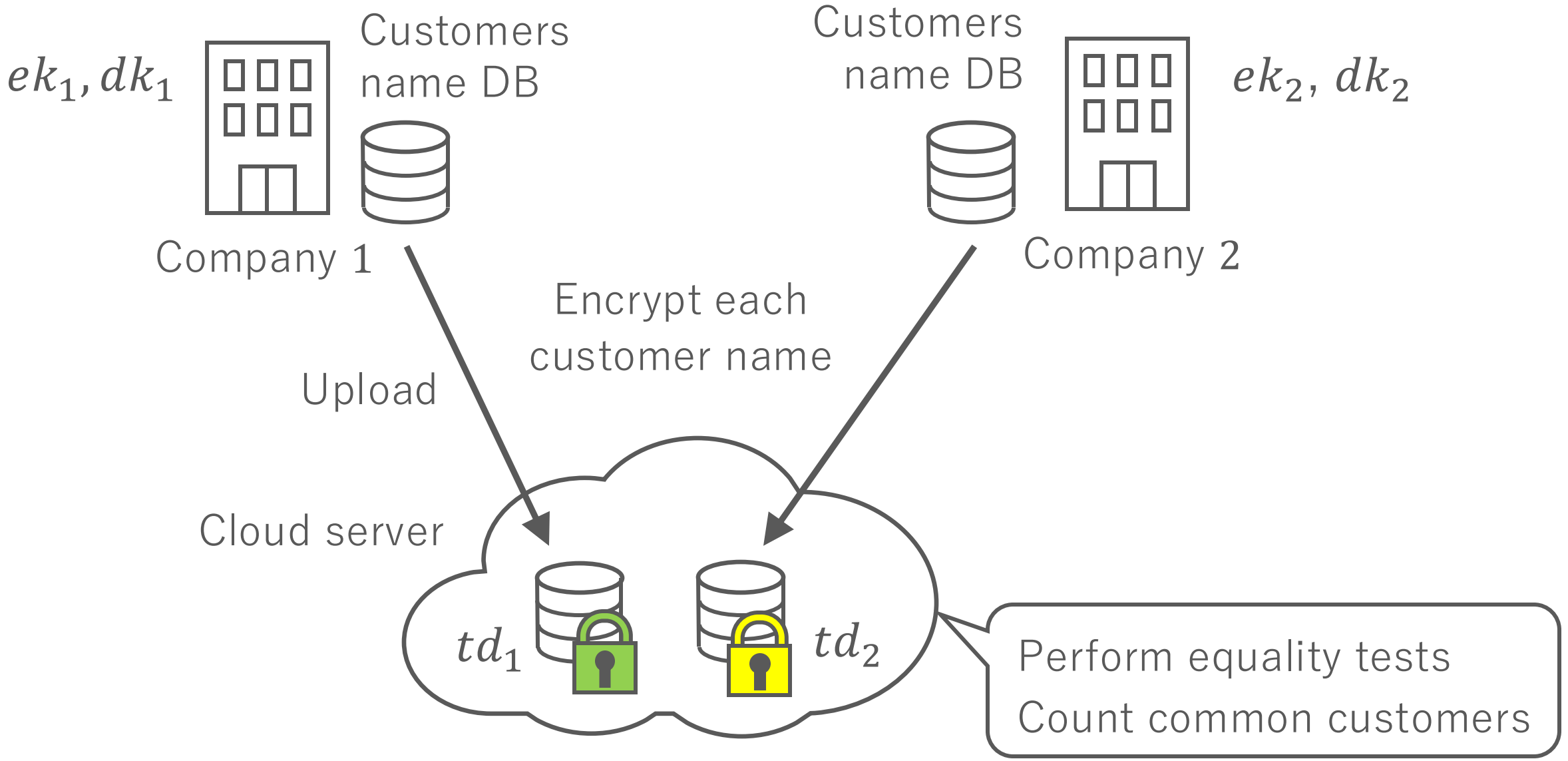
A data analysist want to know the number of common customers between two companies while maintaining privacy.

PKEET is useful for this scenario.

Application of PKEET: Encrypted Database



Application of PKEET: Encrypted Database



PKEET Constructions from Various Assumption

- Pairing Based Constructions

  e.g. [YTHW10], [Tan11]

- Lattice-Based Constructions

  e.g. [DFKRS19], [DRSFKS22]

We focus on generic constructions.

- Generic Constructions

  e.g. [LSQ18]. [LLSW19]

[YTHW10] Yang, Tan, Huang, and Wong. Probabilistic public key encryption with equality test. CT-RSA 2010
[Tan11] Tang. Towards public key encryption scheme supporting equality test with fine-grained authorization. ACISP 2011
[DFKRS19] Duong, Fukushima, Kiyomoto, Roy, and Susilo. A lattice-based public key encryption with equality test in standard model. ACISP 2019
[DRSFKS22] Duong, Roy, Susilo, Fukushima, Kiyomoto, and Sipasseuth. Chosen-ciphertext lattice-based public key encryption with equality test in standard model. Theor. Comput. Sci. 2022
[LSQ18] Lin, Sun, and Qu. Generic construction of public key encryption, identity-based encryption and signcryption with equality test. Inf. Sci. 2018
[LLSW19] Public key encryption with equality test from generic assumptions in the random oracle model. Inf. Sci. 2019.

Generic Construction of PKEET

Scheme	Primitives	Without ROM
[LSQ18]	IND-CCA PKE	**NO**
[LLSW19]	IND-CCA PKE	**NO**
[LLSWY20]	**sID-CPA HIBE** + OTS + OW&CR Hash	**Yes**
[CPL25]	OW-CPA PKE	**NO**

[LSQ18] Lin, Sun, and Qu. Generic construction of public key encryption, identity-based encryption and signcryption with equality test. Inf. Sci. 2018
[LLSW19] Public key encryption with equality test from generic assumptions in the random oracle model. Inf. Sci. 2019.
[LLSWY19] Lee, Ling, Seo, Wang, and Youn. Public key encryption with equality test in the standard model. Inf. Sci. 2020
[CPL25] Choi, Park, and Lee. An efficient and generic construction of public key encryption with equality test under the random oracle model. IEEE Access 2025

Generic Construction of PKEET

Scheme	Primitives	Without ROM
[LSQ18]	IND-CCA PKE	**NO**
[LLSW19]	IND-CCA PKE	**NO**
[LLSWY20]	**sID-CPA HIBE** + OTS + OW&CR Hash	**Yes**
[CPL25]	OW-CPA PKE	**NO**

[LSQ18] Lin, Sun, and Qu. Generic construction of public key encryption, identity-based encryption and signcryption with equality test. Inf. Sci. 2018
[LLSW19] Public key encryption with equality test from generic assumptions in the random oracle model. Inf. Sci. 2019.
[LLSWY19] Lee, Ling, Seo, Wang, and Youn. Public key encryption with equality test in the standard model. Inf. Sci. 2020
[CPL25] Choi, Park, and Lee. An efficient and generic construction of public key encryption with equality test under the random oracle model. IEEE Access 2025

Generic Construction of PKEET

Scheme	Primitives	Without ROM
[LSQ18]	IND-CCA PKE	**NO**
[LLSW19]	IND-CCA PKE	**NO**
[LLSWY20]	**sID-CPA HIBE** + OTS + OW&CR Hash	**Yes**
[CPL25]	OW-CPA PKE	**NO**

[LSQ18] Lin, Sun, and Qu. Generic construction of public key encryption, identity-based encryption and signcryption with equality test. Inf. Sci. 2018
[LLSW19] Public key encryption with equality test from generic assumptions in the random oracle model. Inf. Sci. 2019.
[LLSWY19] Lee, Ling, Seo, Wang, and Youn. Public key encryption with equality test in the standard model. Inf. Sci. 2020
[CPL25] Choi, Park, and Lee. An efficient and generic construction of public key encryption with equality test under the random oracle model. IEEE Access 2025

Generic Construction of PKEET

Scheme	Primitives	Without ROM
[LSQ18]	IND-CCA PKE	**NO**
[LLSW19]	IND-CCA PKE	**NO**
[LLSWY20]	**sID-CPA HIBE** + OTS + OW&CR Hash	**Yes**
[CPL25]	OW-CPA PKE	**NO**

Question Can we give a PKEET scheme from
weaker primitive then HIBE without the ROM?

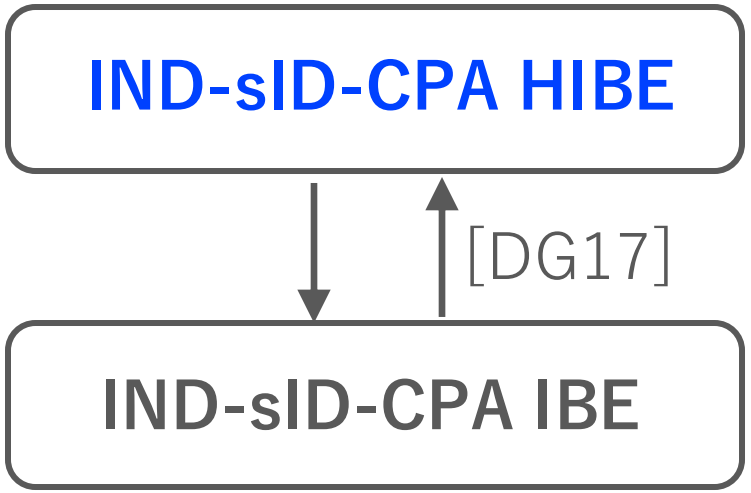
Generic Construction of PKEET

Scheme	Primitives	Without ROM
[LSQ18]	IND-CCA PKE	**NO**
[LLSW19]	IND-CCA PKE	**NO**
[LLSWY20]	**sID-CPA HIBE** + OTS + OW&CR Hash	**Yes**
[CPL25]	OW-CPA PKE	**NO**
Our Scheme	**IND-sTag-CCA TBE + OTS + OW&CR Hash**	**Yes**

Relationships among HIBE, IBE, PKE, TBE

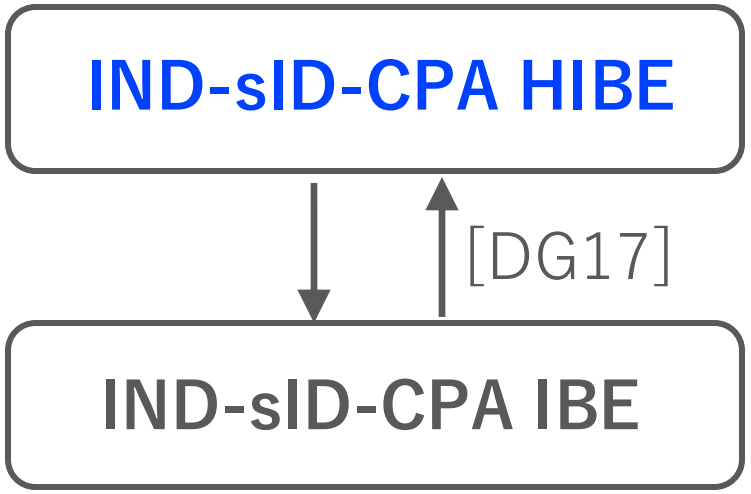
[DG17] Dottling and Garg. From selective IBE to full IBE and selective HIBE. TCC 2017
[Kil06] Kiltz. Chosen-ciphertext security from tag-based encryption. TCC 2006

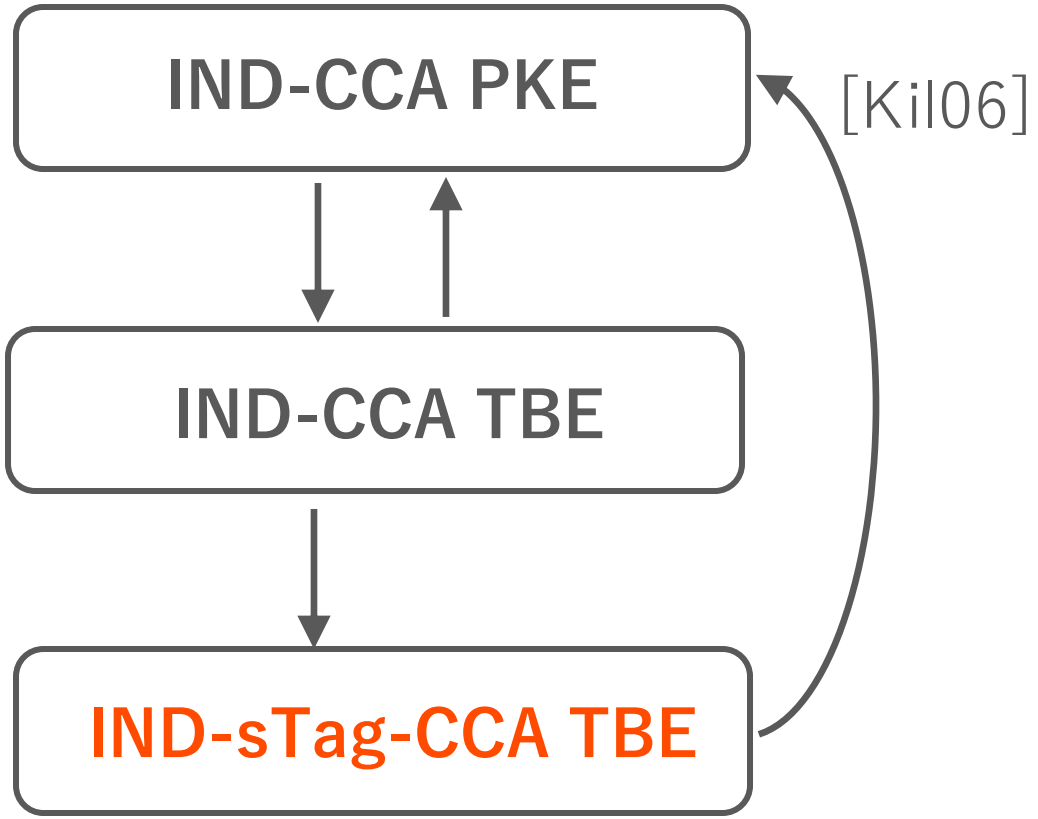
Relationships among HIBE, IBE, PKE, TBE



[DG17] Dottling and Garg. From selective IBE to full IBE and selective HIBE. TCC 2017
[Kil06] Kiltz. Chosen-ciphertext security from tag-based encryption. TCC 2006

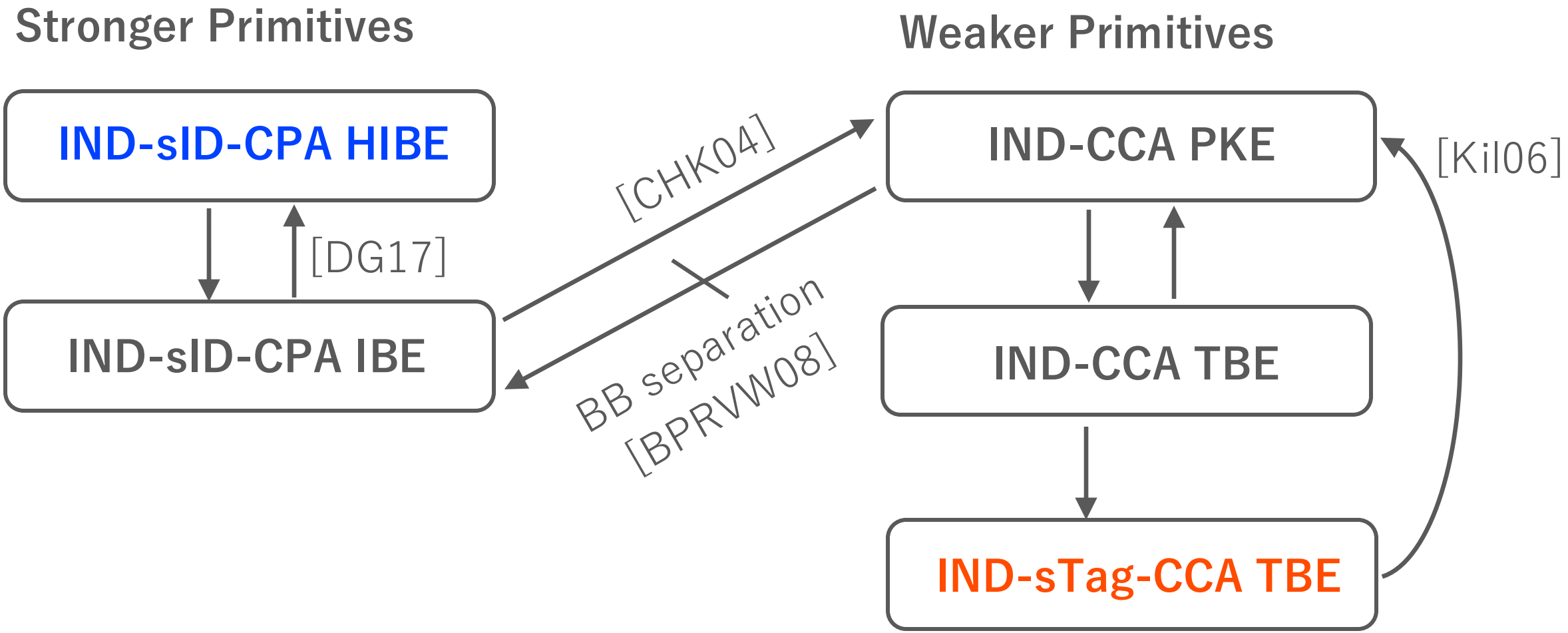
Relationships among HIBE, IBE, PKE, TBE





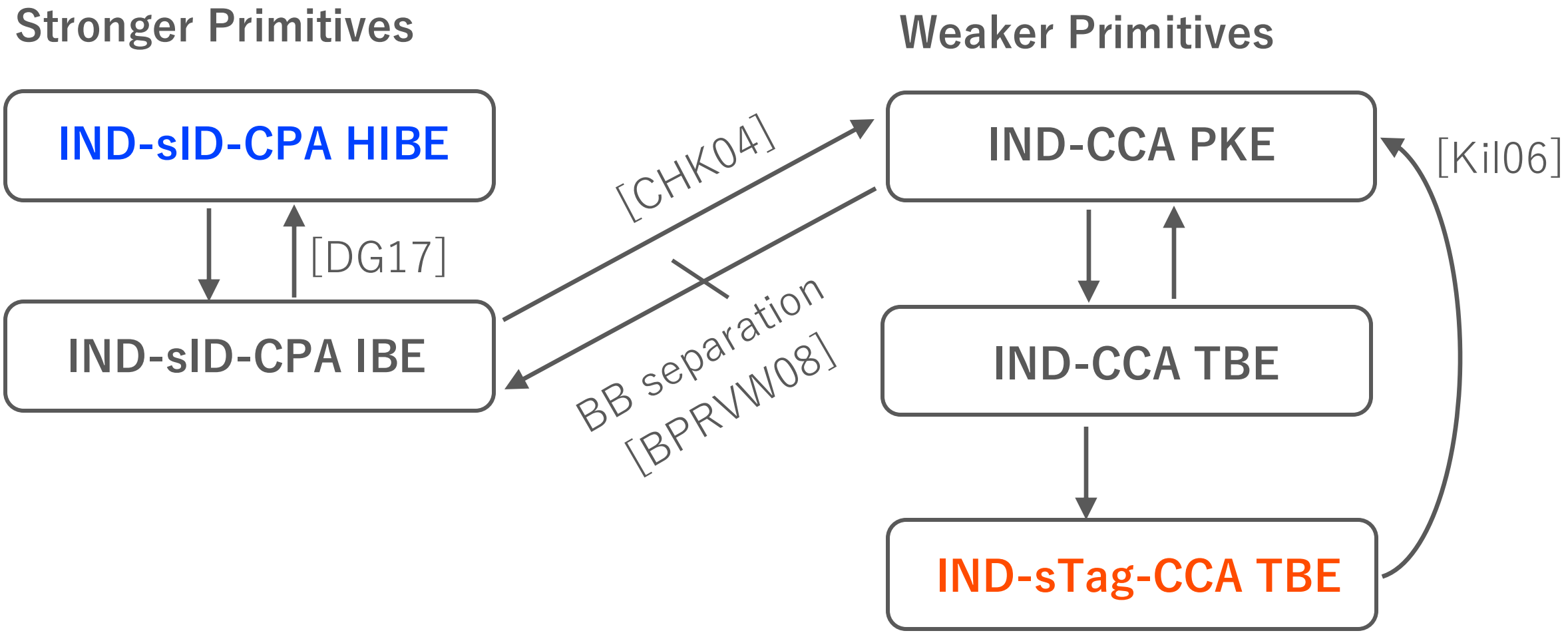
[DG17] Dottling and Garg. From selective IBE to full IBE and selective HIBE. TCC 2017
[Kil06] Kiltz. Chosen-ciphertext security from tag-based encryption. TCC 2006

Relationships among HIBE, IBE, PKE, TBE



[CHK04] Canetti, Halevi, and Katz. Chosen-ciphertext security from identity-based encryption. EUROCRYPT 2004
[BPRVW08] Boneh, Papakonstantinou, Rackoff, and Vahlis, and Waters. On the Impossibility of Basing Identity Based Encryption on Trapdoor Permutations. FOCS 2008

Relationships among HIBE, IBE, PKE, TBE



TBE is weaker than **HIBE**.

[CHK04] Canetti, Halevi, and Katz. Chosen-ciphertext security from identity-based encryption. EUROCRYPT 2004
[BPRVW08] Boneh, Papakonstantinou, Rackoff, and Vahlis, and Waters. On the Impossibility of Basing Identity Based Encryption on Trapdoor Permutations. FOCS 2008

Definition of PKEET and its Security

Syntax of PKEET

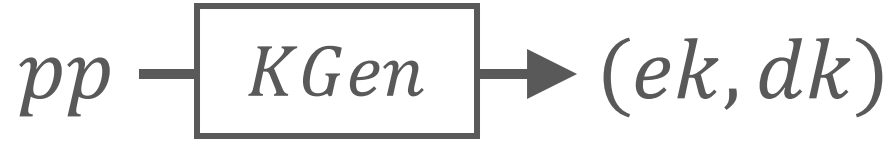
$$PKEET = (Setup, KGen, Enc, Dec, TDGen, Test)$$

Syntax of PKEET

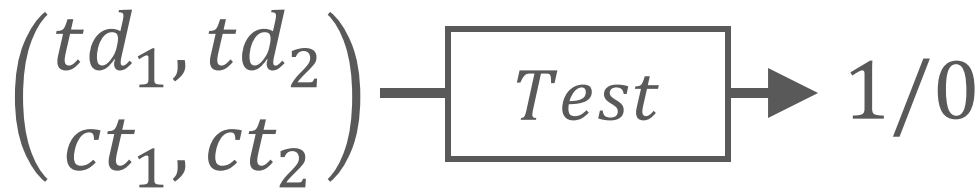
$$PKEET = (Setup, KGen, Enc, Dec, TDGen, Test)$$

Syntax of PKEET

$$PKEET = (Setup, KGen, Enc, Dec, TDGen, Test)$$

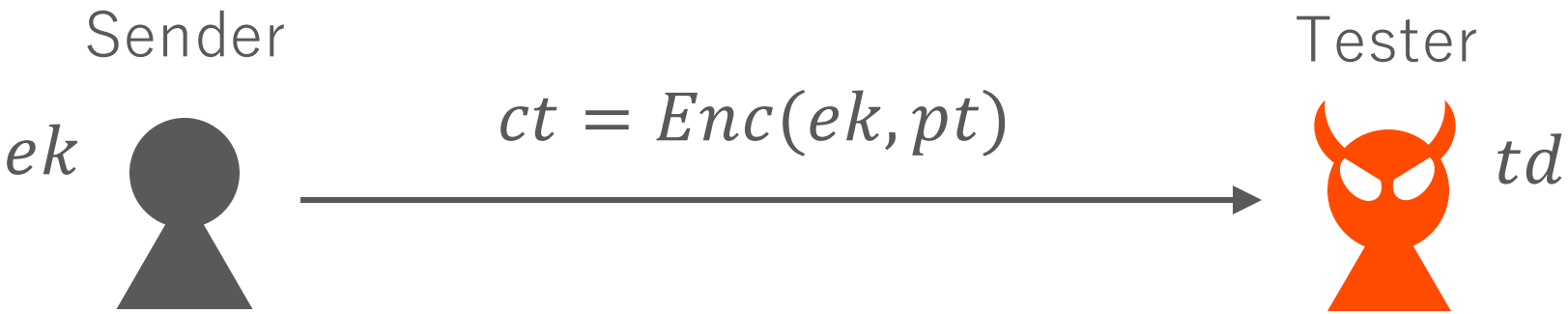


Algorithms for equality test



Security of PKEET (OW-CCA against Type-I Adversary)

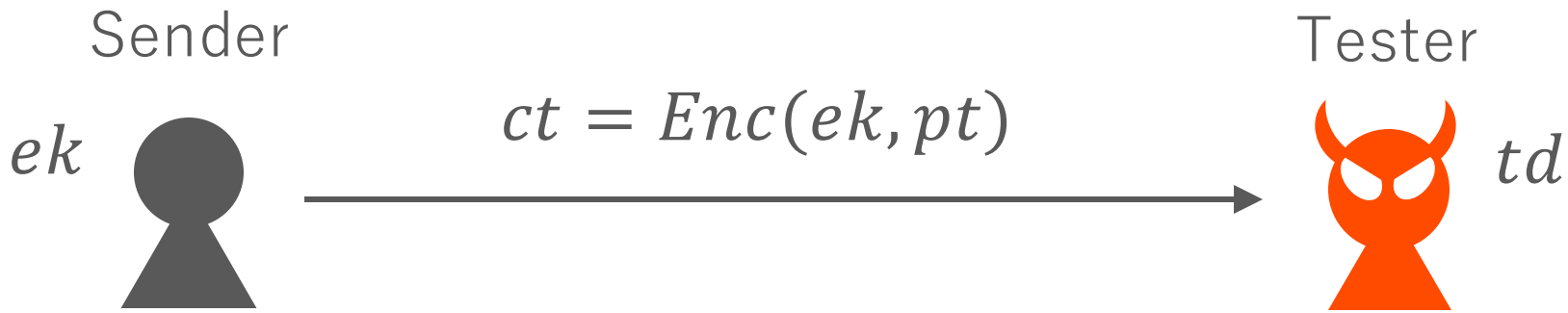
Type-I adversary (tester) has a trapdoor td.



Try to obtain information pt from ct by using td.

Security of PKEET (OW-CCA against Type-I Adversary)

Type-I adversary (tester) has a trapdoor td.

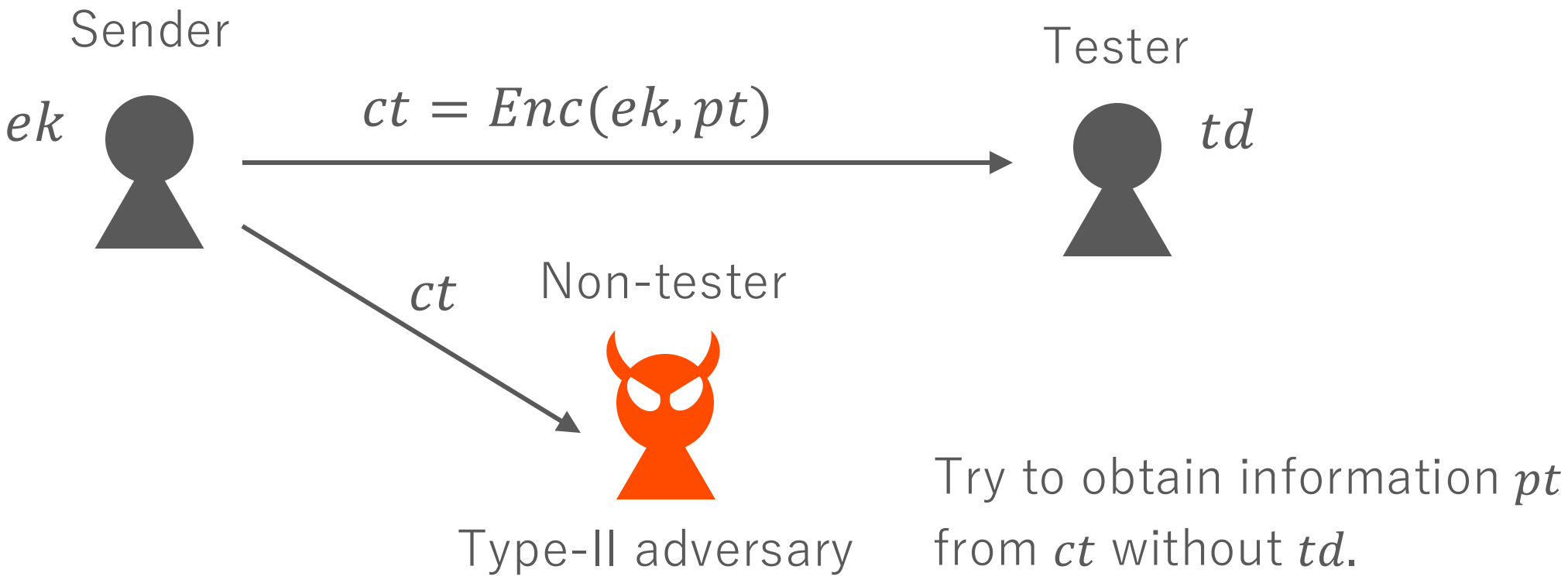


Try to obtain information pt from ct by using td.

The IND security is impossible for type-I adversaries.
Instead, we consider the **OW-CCA** security for **type-I adversaries**.

Security of PKEET (IND-CCA against Type-II Adversary)

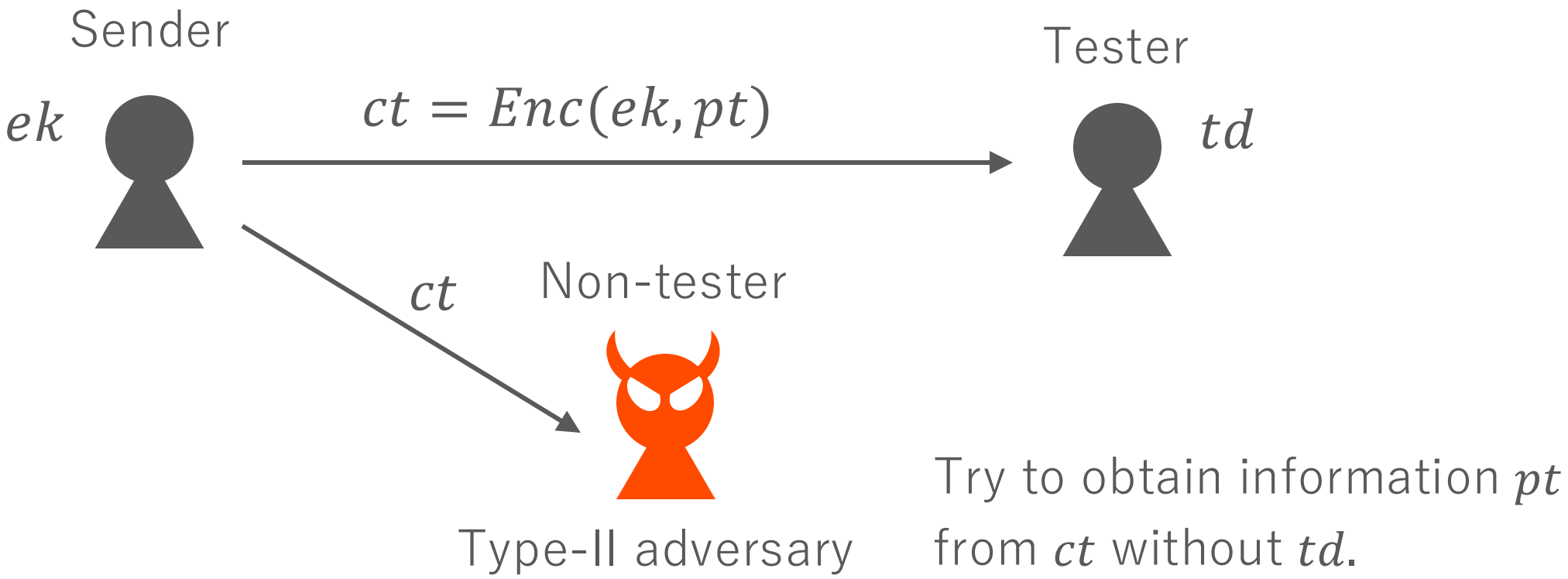
Type-II adversary (non-tester) does have a trapdoor td.



Try to obtain information pt from ct without td.

Security of PKEET (IND-CCA against Type-II Adversary)

Type-II adversary (non-tester) does have a trapdoor td.

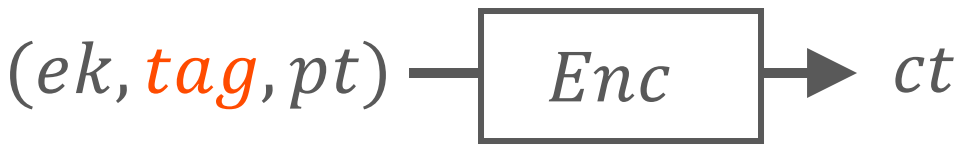


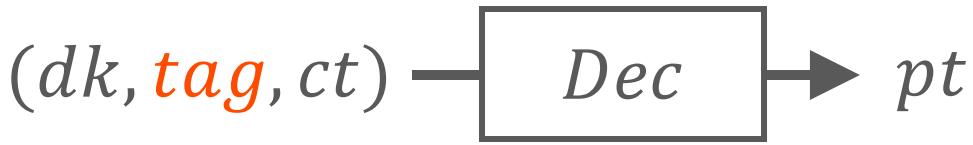
For **type-II adversaries**, we consider the **IND-CCA** security.

How to Obtain Our Construction

Key Primitive: Tag-Based Encryption (TBE) [Kil06]

A tag tag is an arbitrary bit strings.

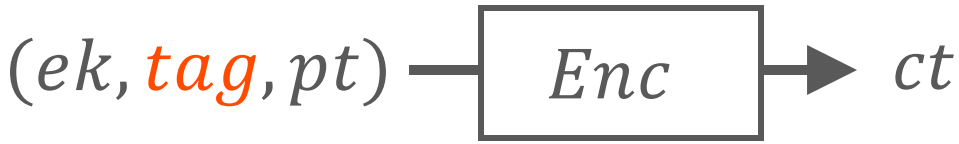


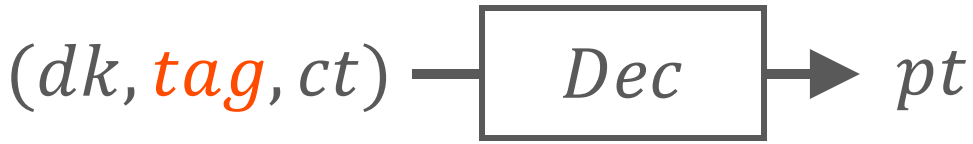


[Kil06] Kiltz. Chosen-ciphertext security from tag-based encryption. TCC 2006

Key Primitive: Tag-Based Encryption (TBE) [Kil06]

A tag tag is an arbitrary bit strings.





Correctness:

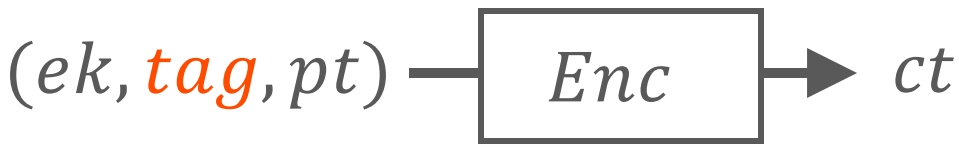
A ciphertext ct generated by $(ek,\ tag)$ can be decrypted by using $(dk,\ tag)$.

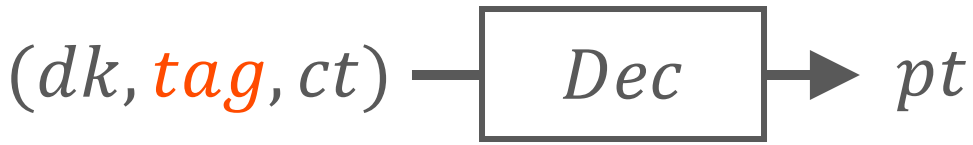
$$Dec\big(dk, tag\ Enc(ek, tag, pt)\big) = pt$$

[Kil06] Kiltz. Chosen-ciphertext security from tag-based encryption. TCC 2006

Key Primitive: Tag-Based Encryption (TBE) [Kil06]

A tag tag is an arbitrary bit strings.





Correctness:

A ciphertext ct generated by $(ek,\ tag)$ can be decrypted by using $(dk,\ tag)$.

$$Dec\big(dk, tag\ Enc(ek, tag, pt)\big) = pt$$

Observation [Kil06] :

TBE is sufficient for applying CHK transformation.

[Kil06] Kiltz. Chosen-ciphertext security from tag-based encryption. TCC 2006

Our Construction Approach

Based PKEET (Not CCA secure)

A receiver prepare two tuple of TBE keys $(ek_1, dk_1), (ek_2, dk_2)$.

A sender generates ciphertexts

$$ct_1 \leftarrow TBE.Enc(ek_1, tag, pt),\ ct_2 \leftarrow TBE.Enc(ek_2, tag, H(pt))$$

A trapdoor for equality tests is dk_2

Our Construction Approach

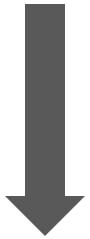
Based PKEET (Not CCA secure)

A receiver prepare two tuple of TBE keys $(ek_1, dk_1), (ek_2, dk_2)$.

A sender generates ciphertexts

$$ct_1 \leftarrow TBE.Enc(ek_1, tag, pt),\ ct_2 \leftarrow TBE.Enc(ek_2, tag, H(pt))$$

A trapdoor for equality tests is dk_2



CHK transformation [CHK04]

PKEET Scheme with IND-CCA security

[CHK04] Canetti, Halevi, and Katz. Chosen-ciphertext security from identity-based encryption. EUROCRYPT 2004

Our Construction: TBE + CHK Transformation

$pp = H$: OW & CR hash　　OTS : One-time signature

$PKEET.KGen(1^\lambda)$:

$(ek_1, dk_1) \leftarrow TBE_1.KGen(1^\lambda)$

$(ek_2, dk_2) \leftarrow TBE_2.KGen(1^\lambda)$

$(ek, dk) \leftarrow \big((ek_1, ek_2), (dk_1, dk_2)\big)$

Our Construction: TBE + CHK Transformation

$pp = H$: OW & CR hash　　OTS : One-time signature

$PKEET.KGen(1^\lambda)$:

$(ek_1, dk_1) \leftarrow TBE_1.KGen(1^\lambda)$

$(ek_2, dk_2) \leftarrow TBE_2.KGen(1^\lambda)$

$(ek, dk) \leftarrow ((ek_1, ek_2), (dk_1, dk_2))$

$PKEET.Enc(ek = (ek_1, ek_2), pt)$:

$(vk, sk) \leftarrow OTS.KGen(1^\lambda)$

$ct_1 \leftarrow TBE_1.Enc(ek_1, tag = vk, pt)$

$ct_2 \leftarrow TBE_2.Enc(ek_2, tag = vk, H(pt))$

$\sigma \leftarrow OTS.Sign(sk, (ct_1, ct_2))$

$ct \leftarrow (vk, ct_1, ct_2, \sigma)$

Our Construction: TBE + CHK Transformation

$pp = H$: OW & CR hash　　OTS : One-time signature

$PKEET.Dec\big(dk = (dk_1, dk_2), ct = (vk, ct_1, ct_2, \sigma)\big)$:

$OTS.Ver(vk, (ct_1, ct_2), \sigma) = 1$?

$pt \leftarrow TBE_1.Dec(dk_1, tag = vk, ct_1)$,

$h \leftarrow TBE_2.Dec(dk_2, tag = vk, ct_2)$,

If $H(pt) = h$ return pt.

Otherwise return $\perp$.

Validity check

Our Construction: TBE + CHK Transformation

$PKEET.TGen(dk = (dk_1, dk_2))$:

$td = dk_2$

Our Construction: TBE + CHK Transformation

$PKEET.TGen(dk = (dk_1, dk_2))$:

$td = dk_2$

$PKEET.Test\begin{pmatrix} td = dk_2, td' = dk'_2, \\ ct = (vk, ct_1, ct_2, \sigma), ct' = (vk', ct'_1, ct'_2, \sigma') \end{pmatrix}$:

$h \leftarrow TBE_2.Dec(dk_2, tag = vk, ct_2)$, $h' \leftarrow TBE_2.Dec(dk'_2, tag' = vk', ct'_2)$,

If $h = h'$ return 1.

Otherwise return 0.

Conclusion

We give a generic construction of PKEET.

Our construction is based on TBE.

The security is proven without the ROM.

Future work

Generic construction of PKEET with shorter ciphertext size in the standard model.

Thank you for listening !

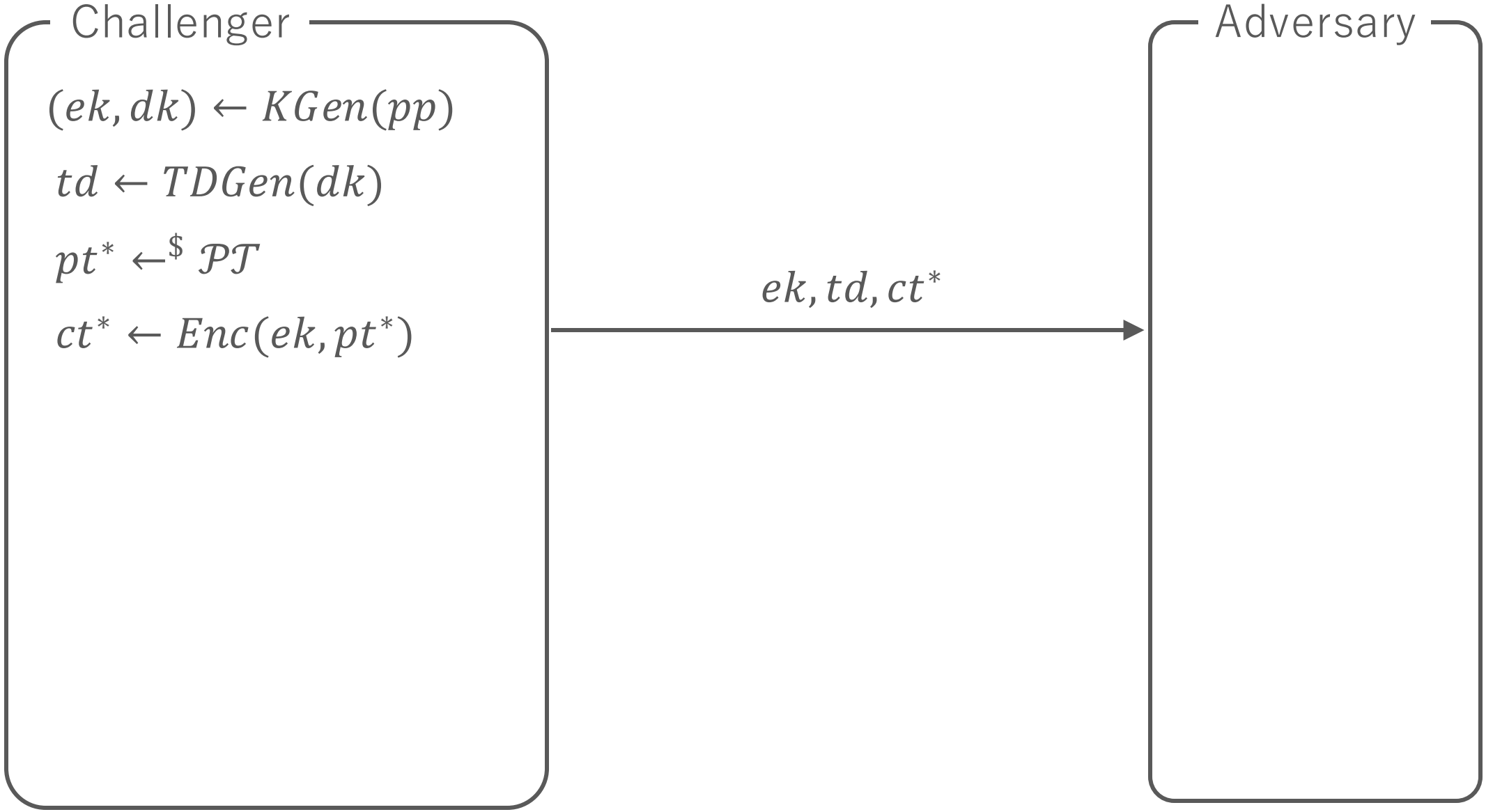
Appendix

OW-CCA Security Game against Type-I Adversary

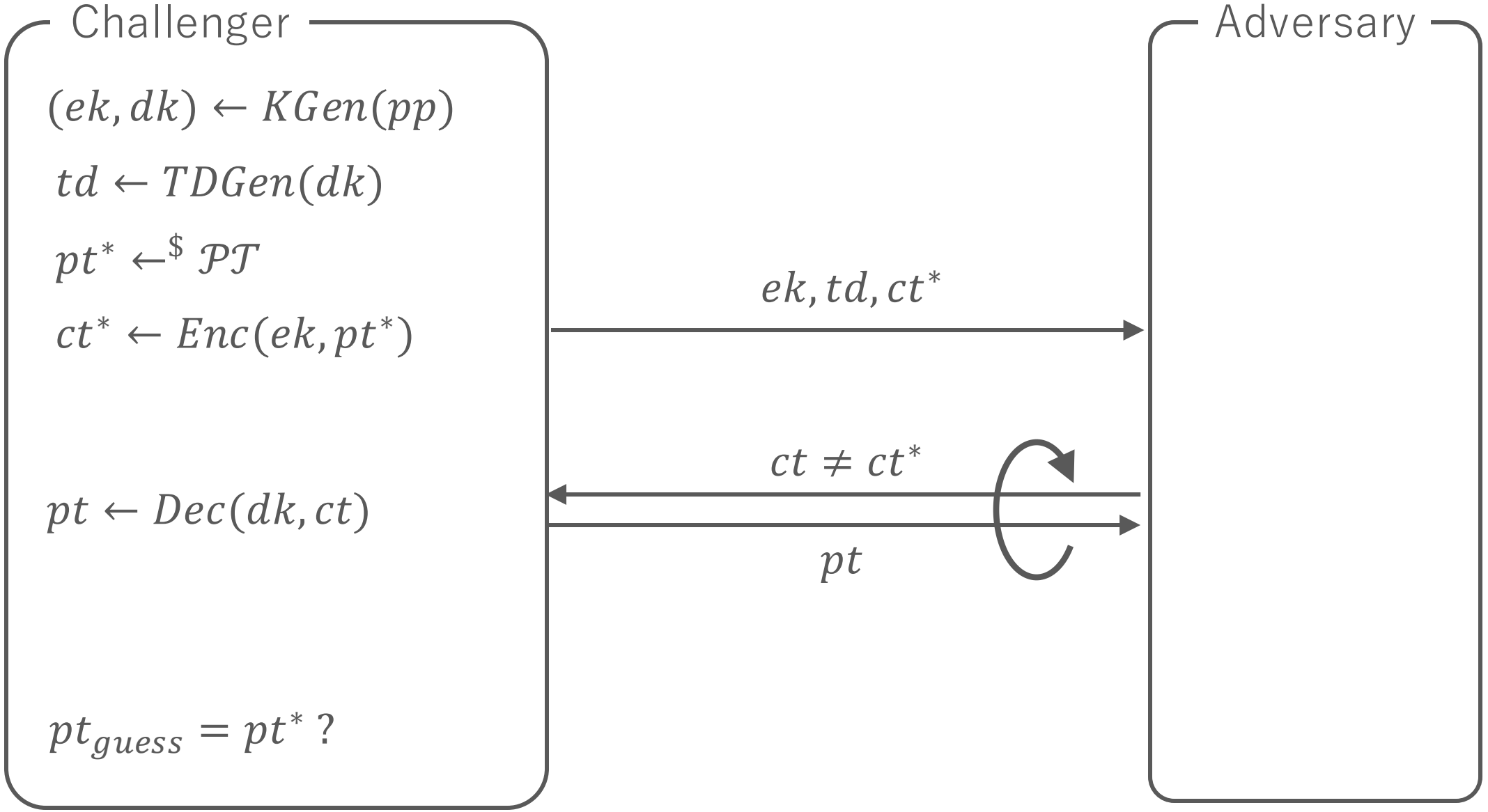
Challenger

Adversary

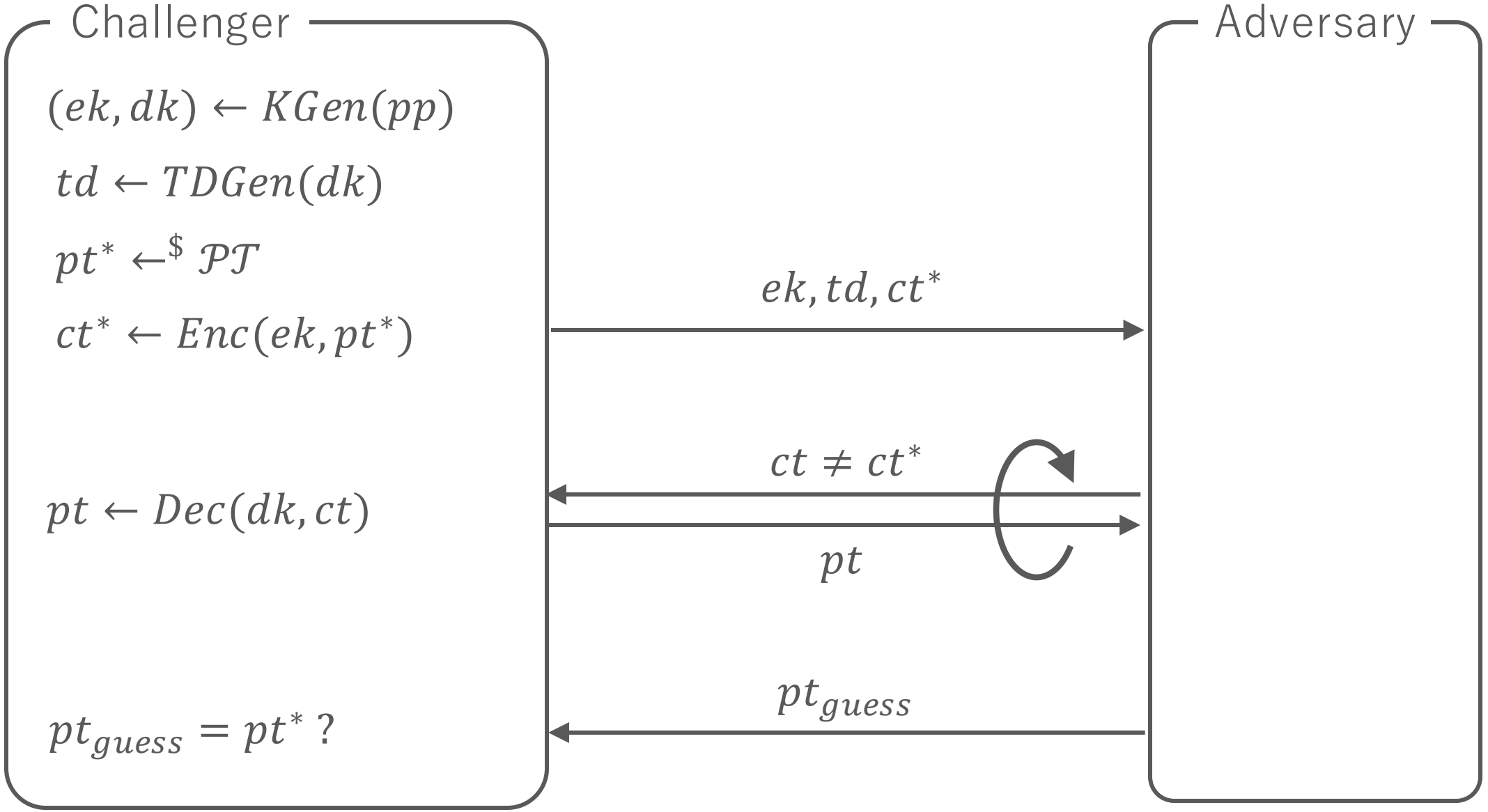
OW-CCA Security Game against Type-I Adversary



OW-CCA Security Game against Type-I Adversary



OW-CCA Security Game against Type-I Adversary

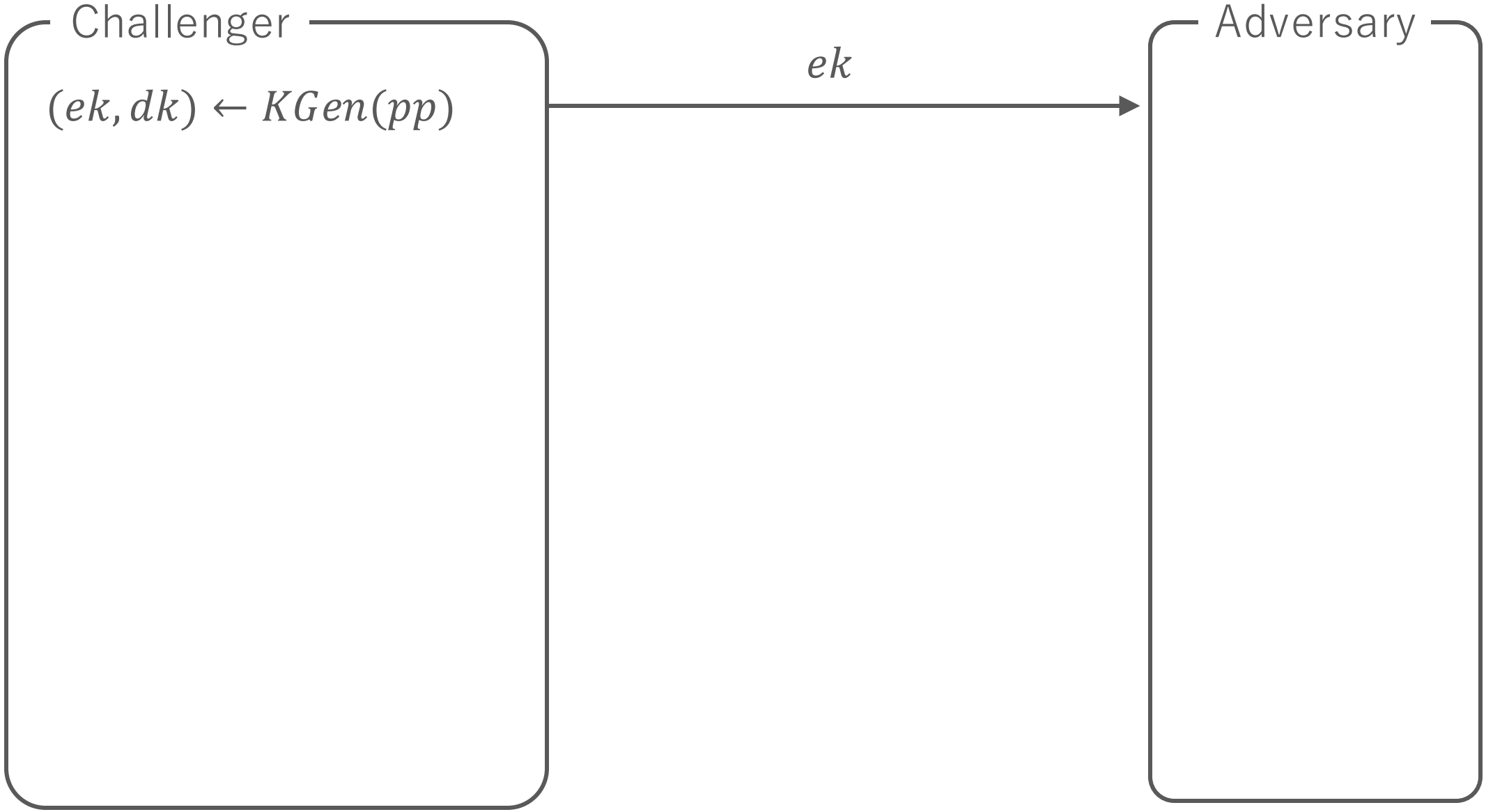


IND-CCA Security Game against Type-II Adversary

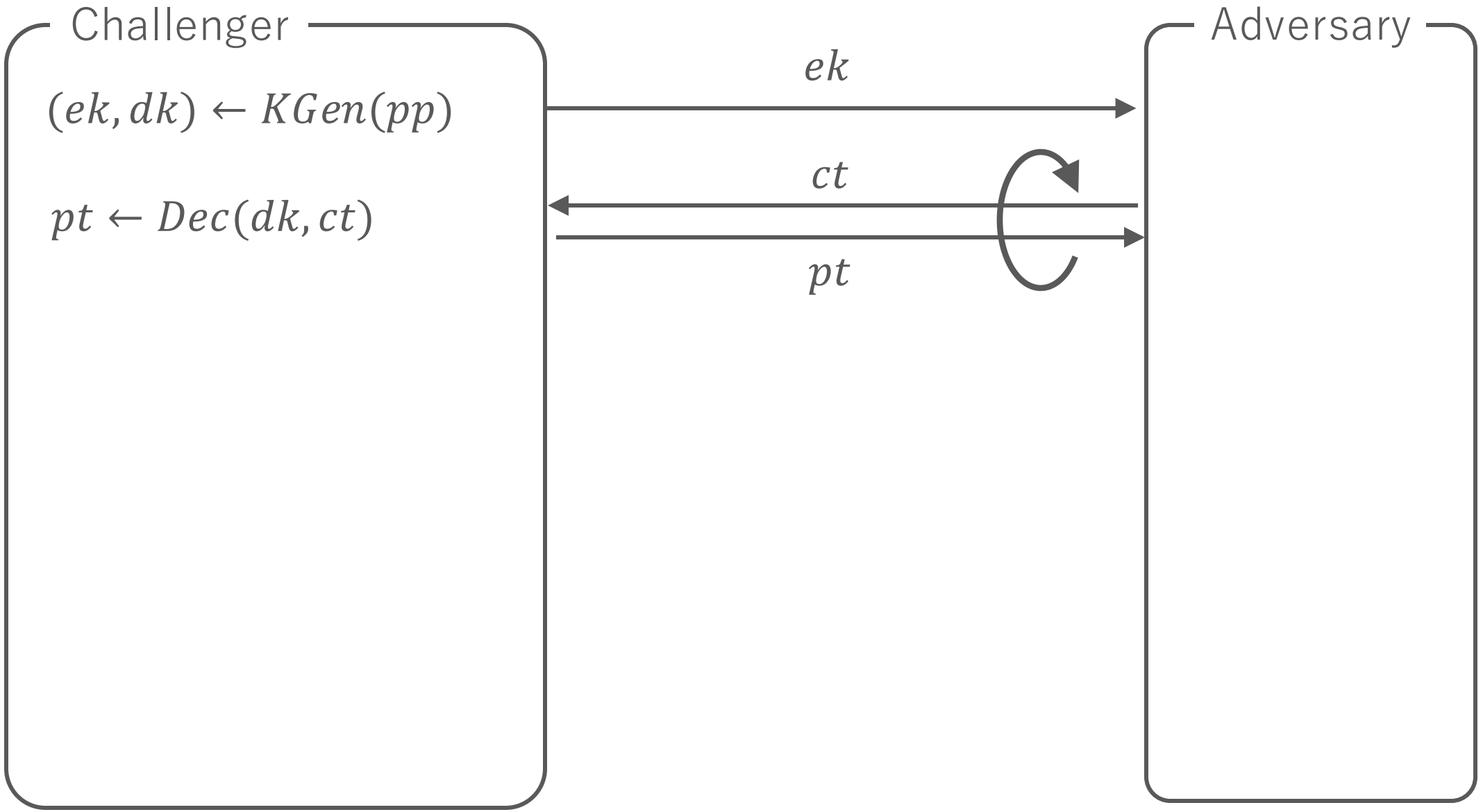
Challenger

Adversary

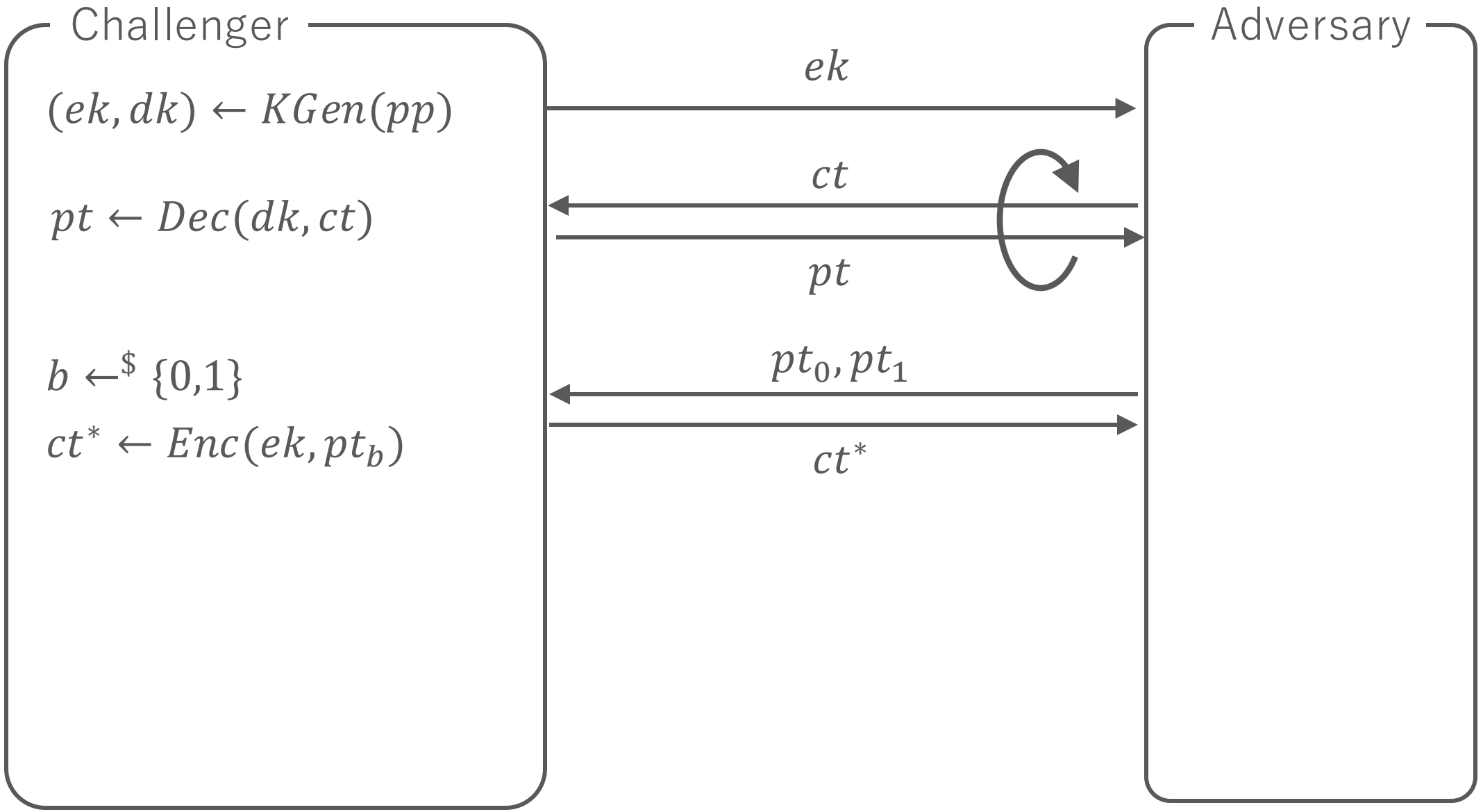
IND-CCA Security Game against Type-II Adversary



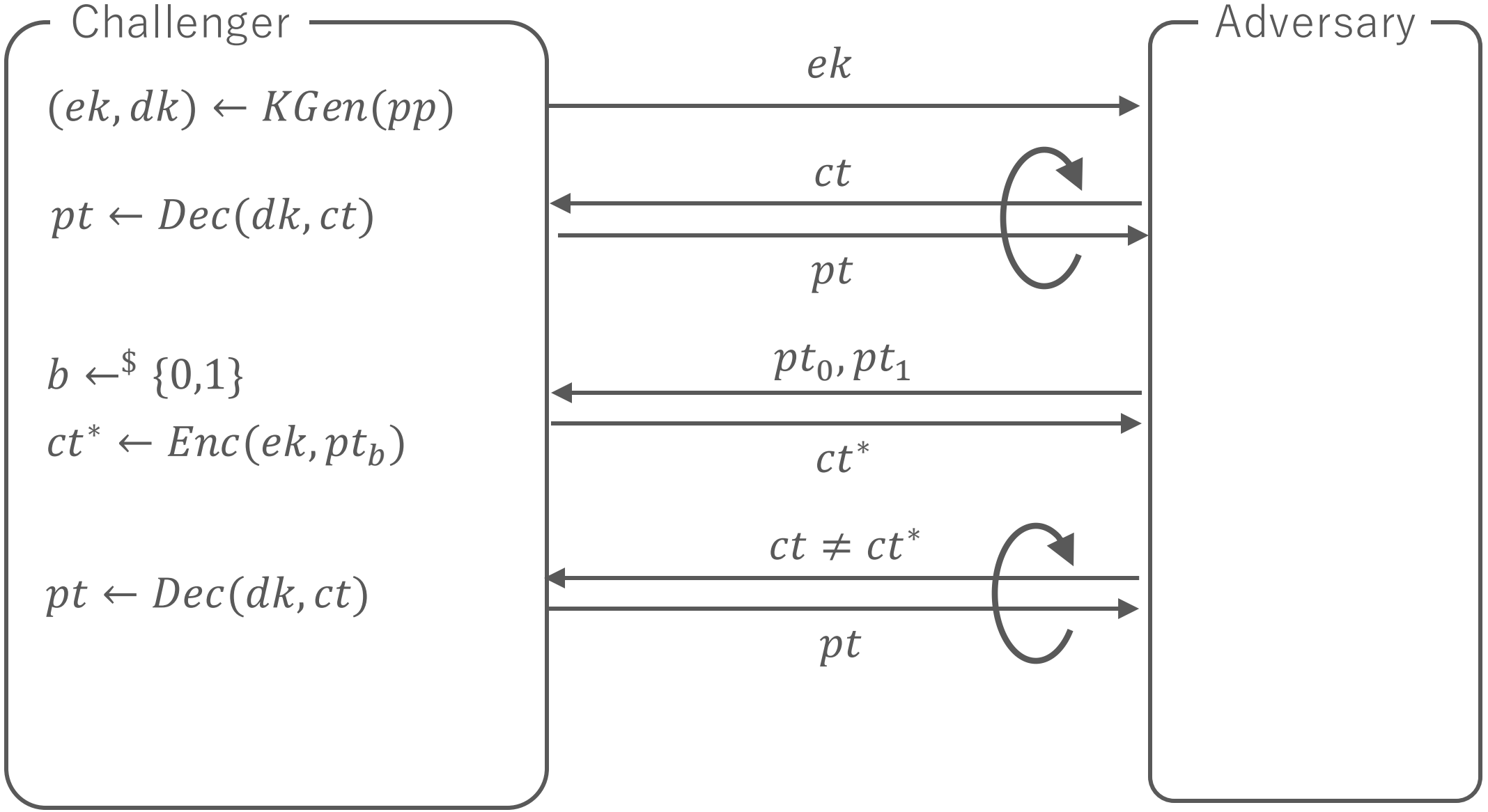
IND-CCA Security Game against Type-II Adversary



IND-CCA Security Game against Type-II Adversary



IND-CCA Security Game against Type-II Adversary



IND-CCA Security Game against Type-II Adversary