Incompressible Encryption

Presentation by:

Harihar S	2020CS10878
Dhairya Gupta	2019CS50428
Abhinav Kumar	2019CS50415

Supervisor:
Prof. Venkata Koppula

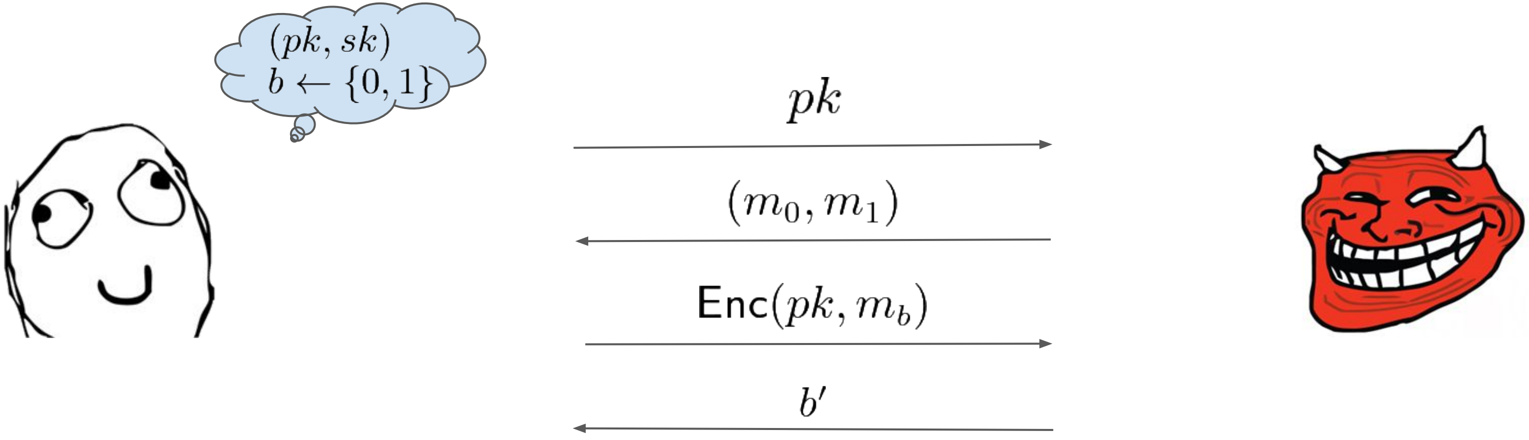
Public Key Encryption Scheme

- $\mathsf{KeyGen}(1^\lambda) \to (pk, sk)$

- $\mathsf{Enc}(pk, m) \to \mathsf{ct}$

- $\mathsf{Dec}(sk, \mathsf{ct}) \to m$

PKE Security



(pk, sk)

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

pk

(m_0, m_1)

$\mathsf{Enc}(pk, m_b)$

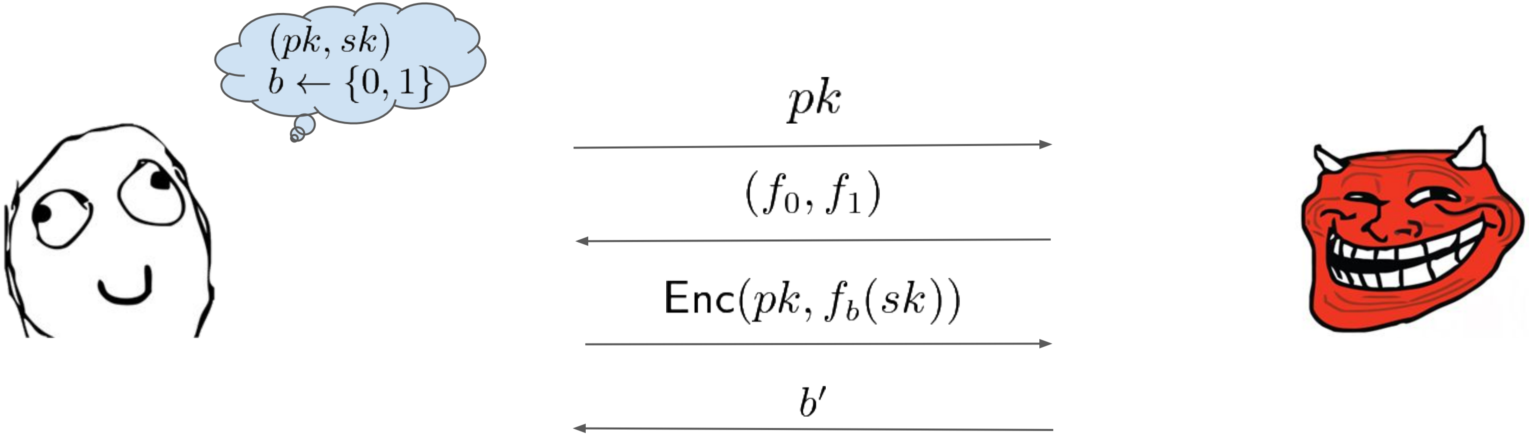
b'

- Put forward by Micali and Goldwasser in 1982

The need for stronger notions of security

- What if m depends on sk ?
- Disk encryption of Windows Vista stored self encryption on the disk
- Careless key management in backup systems
    - Backup key encrypted in the disk
    - Disk key encrypted on the backup system

Key Dependent Message PKE Security



(pk, sk)

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

pk

(f_0, f_1)

$\mathsf{Enc}(pk, f_b(sk))$

b'

- Has been widely studied in literature
- Efficient constructions from assumptions such as
    - Using DDH by Boneh et al
    - Using LWE by Applebaum et al

Security Guarantees Under Key Breaches

MICROSOFT — DATA BREACH — AI — CYBERSECURITY — NEWS

Microsoft exposed 38TB of private AI data, including passwords and secret keys

Microsoft itself warns that it is "not possible to audit the generation of SAS tokens"

ED TARGETT

September 18, 2023 . 4:10 PM — 3 min read

FORBES > INNOVATION > CYBERSECURITY

EDITORS' PICK

Zoom Gets Stuffed: Here's How Hackers Got Hold Of 500,000 Passwords

Davey Winder Senior Contributor

Co-founder, Straight Talking Cyber

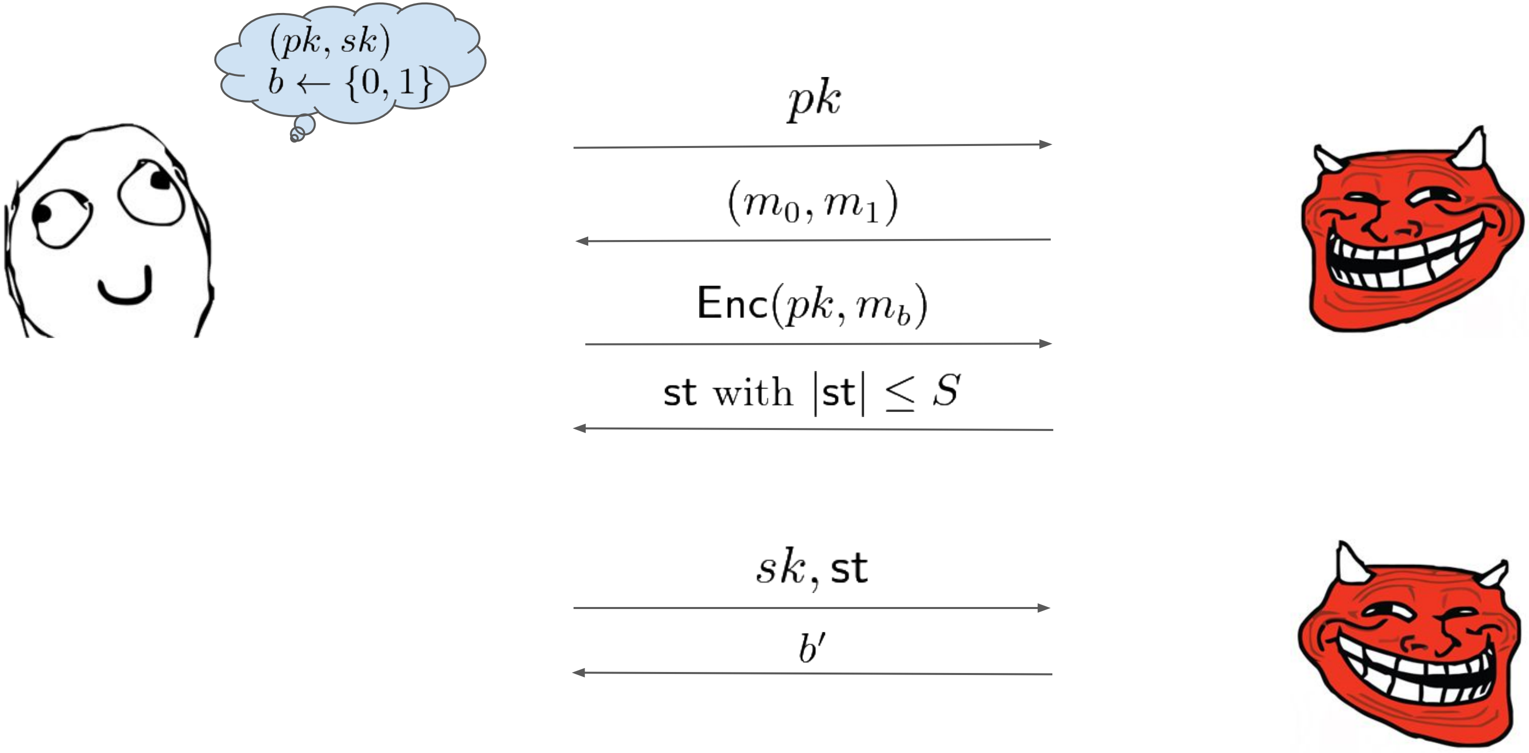
Apr 28, 2020, 06:46am EDT

- No guarantees if the hacker has the entire encrypted message.
- Can we force the adversary to store a large chunk of the ciphertext?

History of Incompressible Encryption

- Rivest [‘97] - All Or Nothing
  - It is hard to get any information on M, given K and most (but not all) of Enc(K, M)
- Dziembowski [‘06] - Symmetric Key Setting
  - The adversary gets no information on M, given K and a small state out of Enc(K, M)
- Guan, Wichs, Zhandry [‘22] - Public Key Setting
  - Public key version of Dziembowski work

Incompressible PKE Security



(pk, sk)

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

pk

(m_0, m_1)

$\mathsf{Enc}(pk, m_b)$

st with $|\mathsf{st}| \leq S$

sk, st

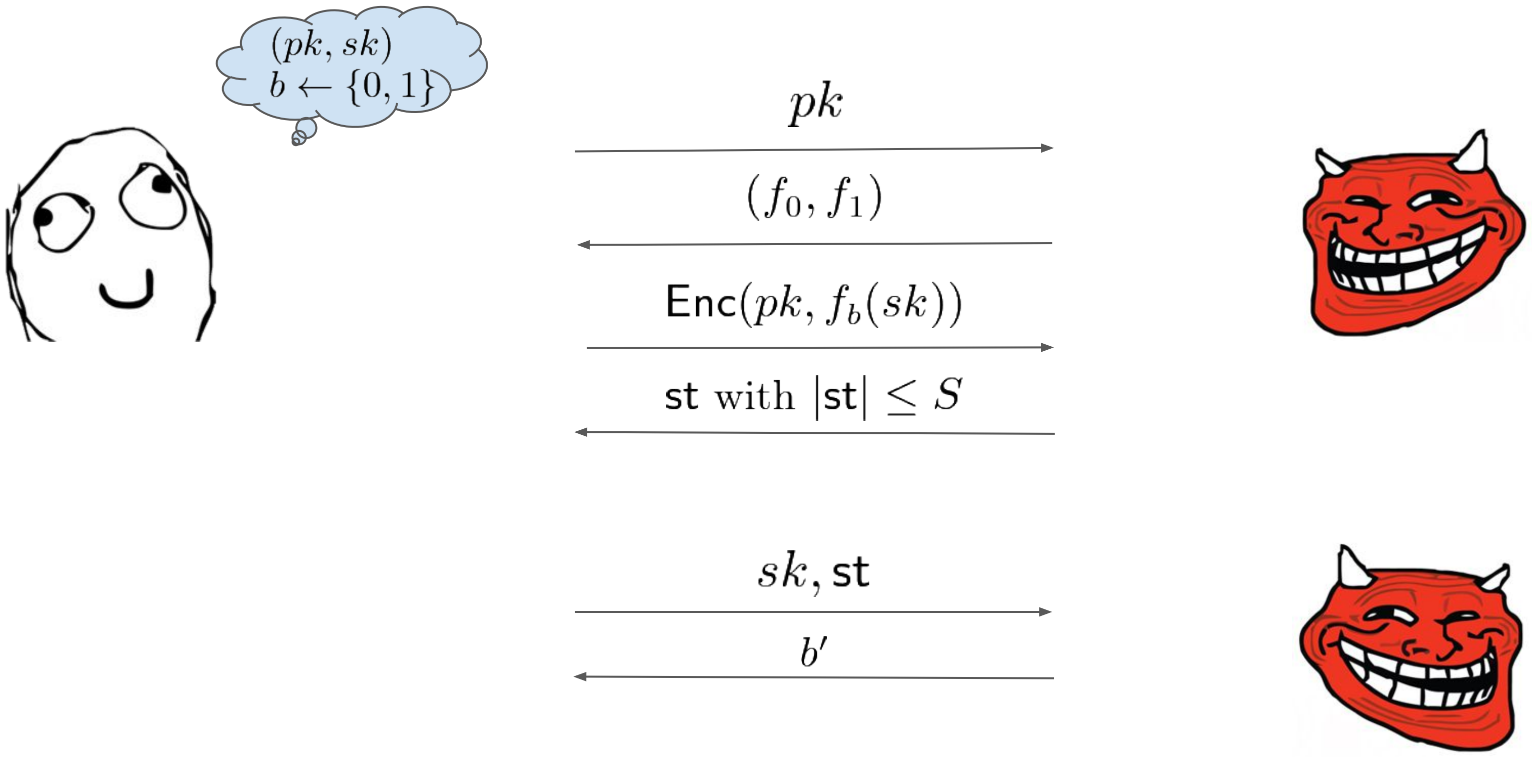
b'

One Time Pad is compressible

- Query on messages $(m_0, m_1) = (0^n, 1^n)$
- Receive $b^n \oplus k$
- Store a single bit state $\mathsf{st} = b \oplus k_0$
- Guess $b' = \mathsf{st} \oplus k_0$

This adversary can win with probability 1.

Incompressible KDM PKE Security



(pk, sk)

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

pk

(f_0, f_1)

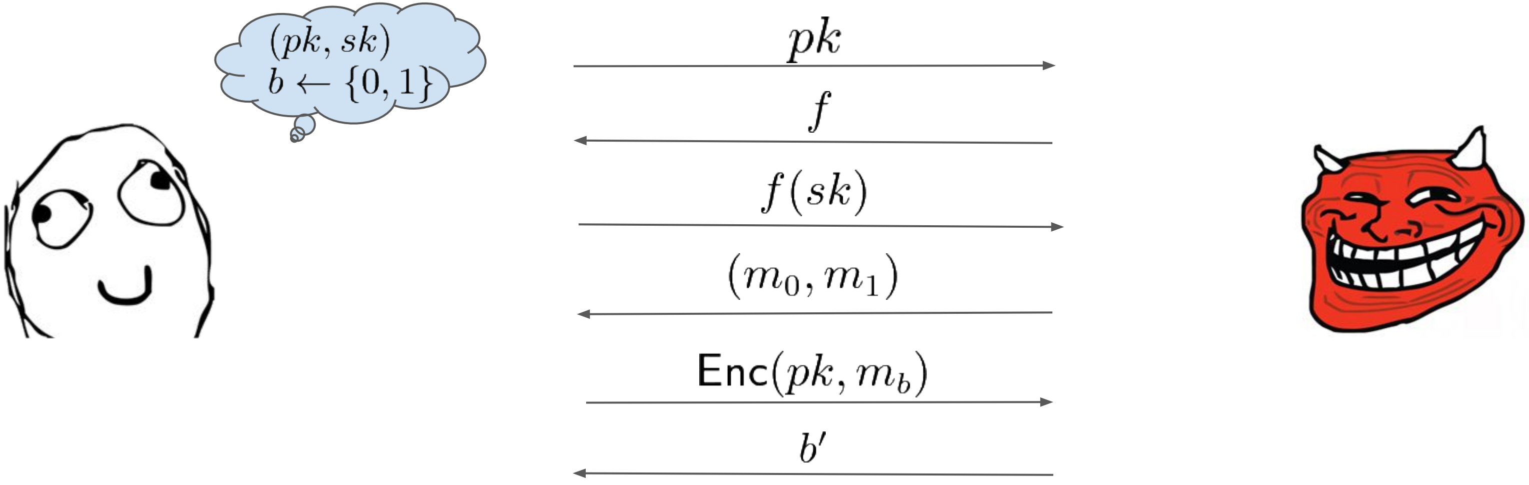
$\mathsf{Enc}(pk, f_b(sk))$

st with $|\mathsf{st}| \leq S$

sk, st

b'

Leakage Resilient PKE Security



(pk, sk)

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

pk

f

$f(sk)$

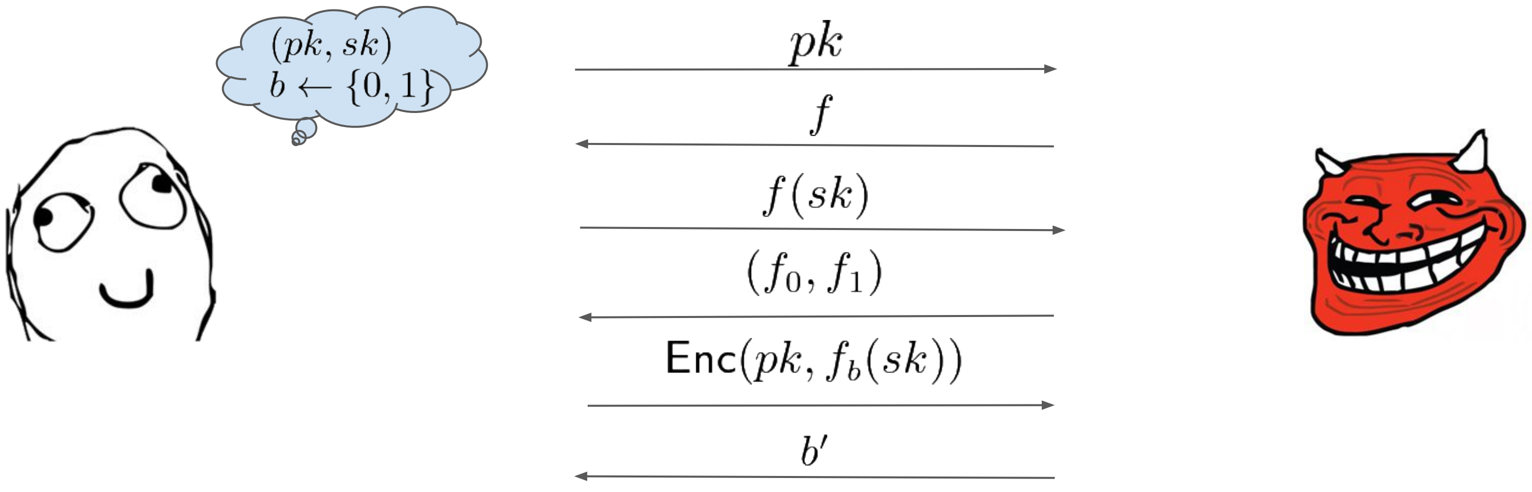
(m_0, m_1)

$\mathsf{Enc}(pk, m_b)$

b'

- Implementations often leak partial information

LR-KDM PKE



(pk, sk)

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

pk

f

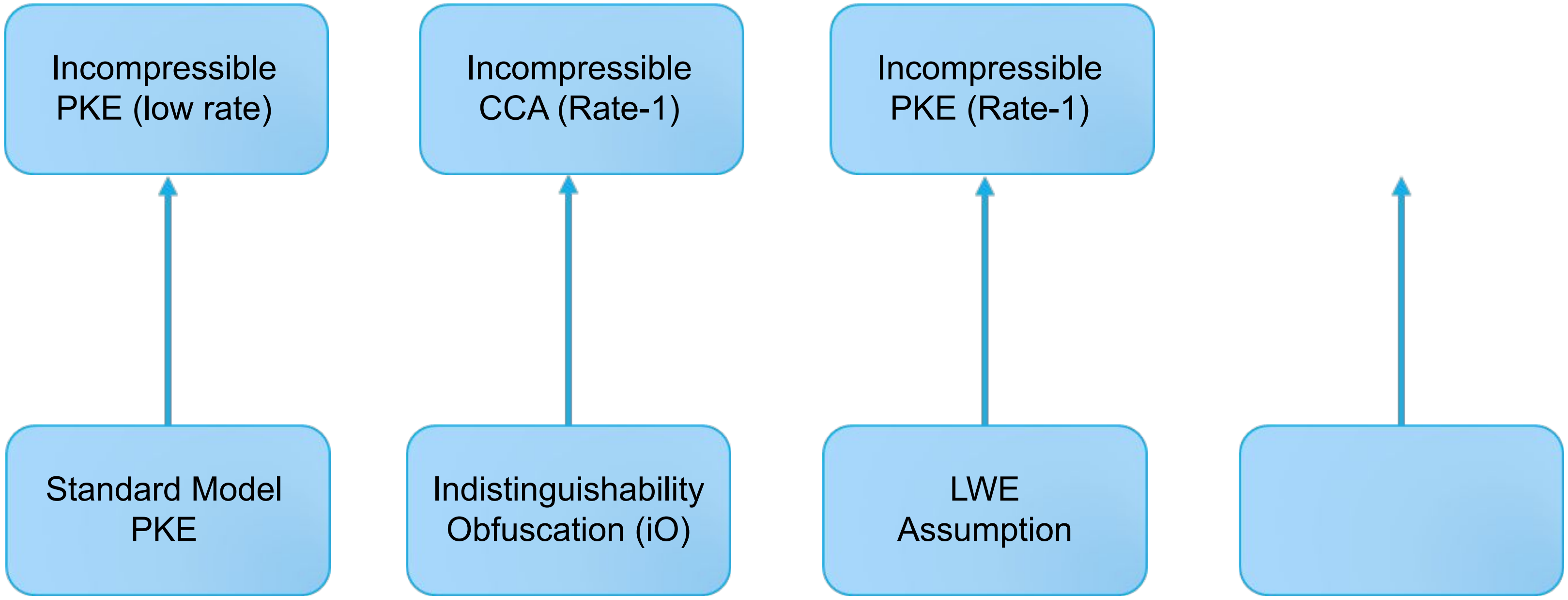
$f(sk)$

(f_0, f_1)

$\mathsf{Enc}(pk, f_b(sk))$

b'

Goal: Achieving incompressible KDM PKE

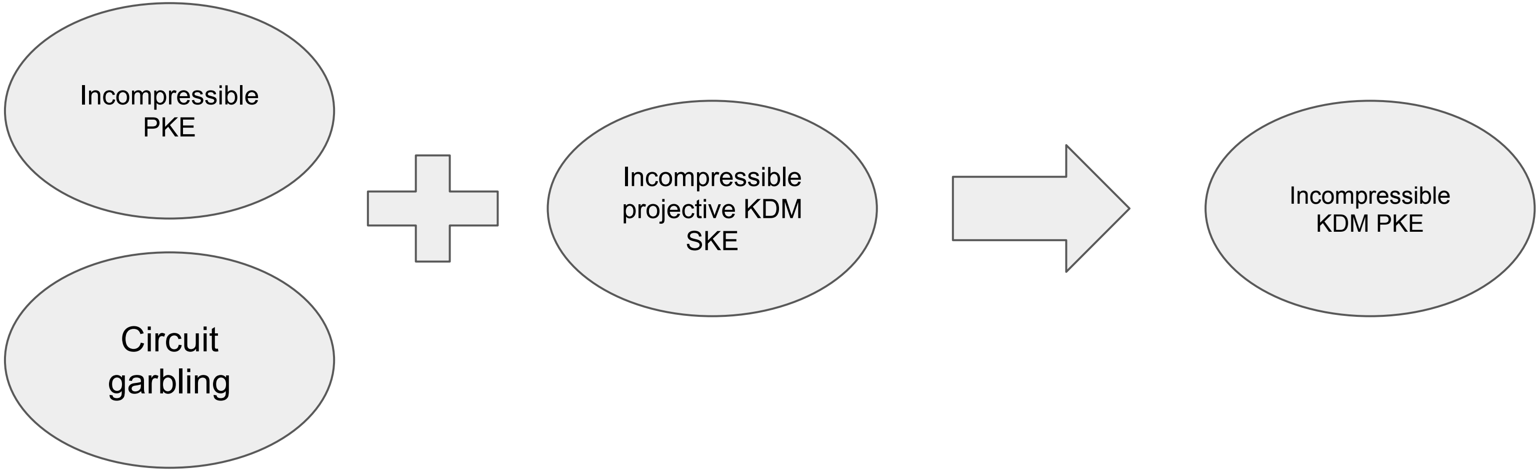

Incompressible PKE (low rate)
Incompressible CCA (Rate-1)
Incompressible PKE (Rate-1)
Standard Model PKE
Indistinguishability Obfuscation (iO)
LWE Assumption

[Guan, Wichs, Zhandry; May 22]

[Branco, Dottling, Dujmovic; Dec 22]

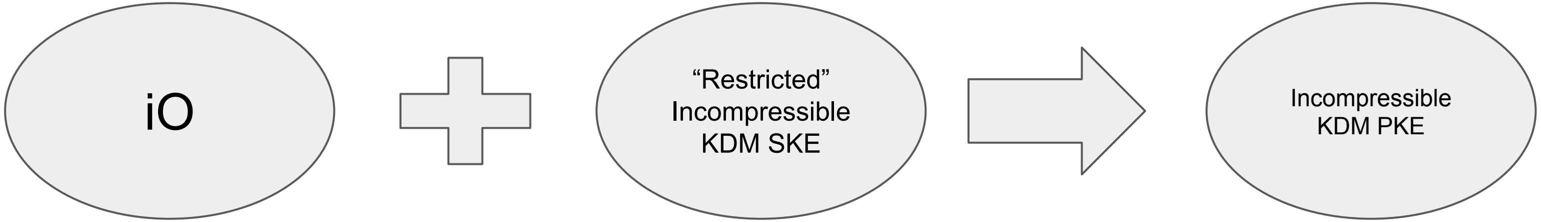
Our Results…

- Incompressible KDM PKE
  - Assuming an existing Incompressible PKE scheme
  - Assuming a secure garbling scheme
  - Assuming an incompressible projective KDM SKE



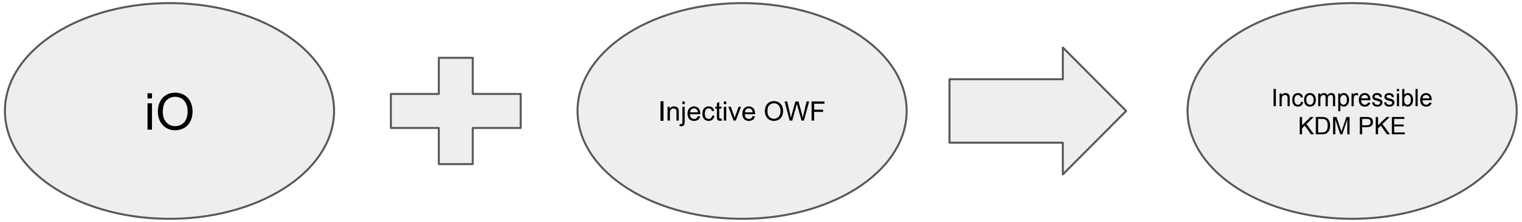
Our Results…

- Incompressible KDM PKE (from iO assumptions)
  - Assume existence of iO(indistinguishability obfuscation)
  - Assume existence of incompressible KDM SKE with injectivity



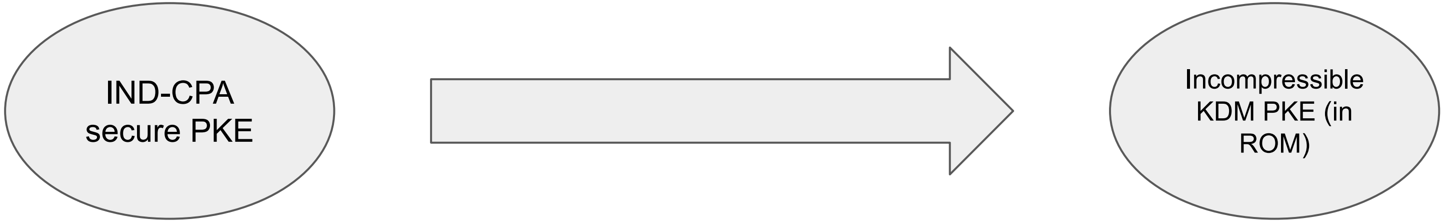
Our Results…

- Incompressible KDM PKE (from iO and heuristic obfuscation)
  - Assume existence of iO and heuristic obfuscation schemes
  - Assume existence of “injective” OWF(One Way functions)



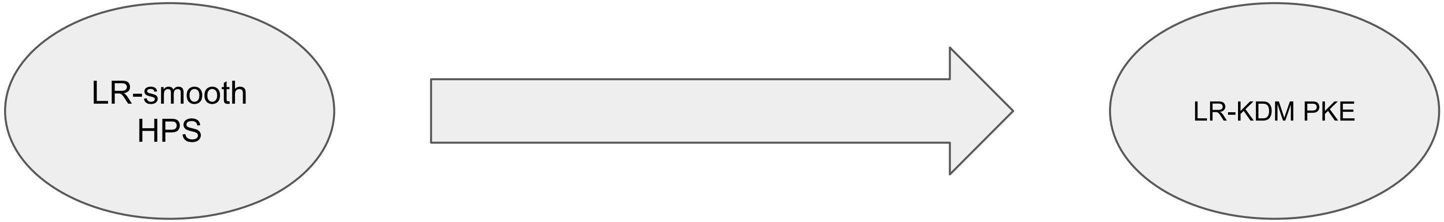
Our Results…

- Constant rate Incompressible KDM PKE(in the Random Oracle Model)
  - Assuming existence of a CPA secure PKE



Our Results…

- ➢ Leakage Resilient KDM PKE
  - ○ From standard number theoretic assumptions, we have LR-KDM PKE over affine functions
  - ○ From LR-smooth HPS(Hash Proof System), we have LR-KDM PKE over general functions



Unsuccessful attempts towards Incompressible KDM

- Amplification to Incompressible multi-key KDM secure from incompressible Circular 1-key secure SKE

- Using iO to build incompressible KDM PKE from incompressible KDM SKE. This approach needed the SKE scheme to be 'injective over the key space', which seems hard to obtain.

Next Steps…

➔ Achieving Incompressible KDM using only iO, without depending on heuristic obfuscation schemes or “restricted” incompressible KDM SKE

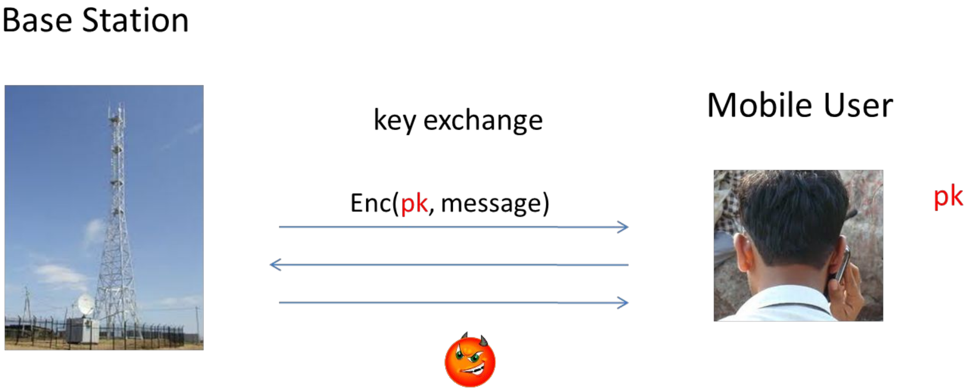
➔ Looking into Leakage resilience from other assumptions, such as CDH, LPN

➔ Looking into other variants of Incompressible encryption, possibly a stepping stone towards simpler constructions of Incomp. KDM.

Anonymous Encryption

Why do we need Anonymity?

- What if a receiver wants to be anonymous? Ex: Mobile Communication



Base Station

key exchange

Mobile User

Enc(pk, message)

pk

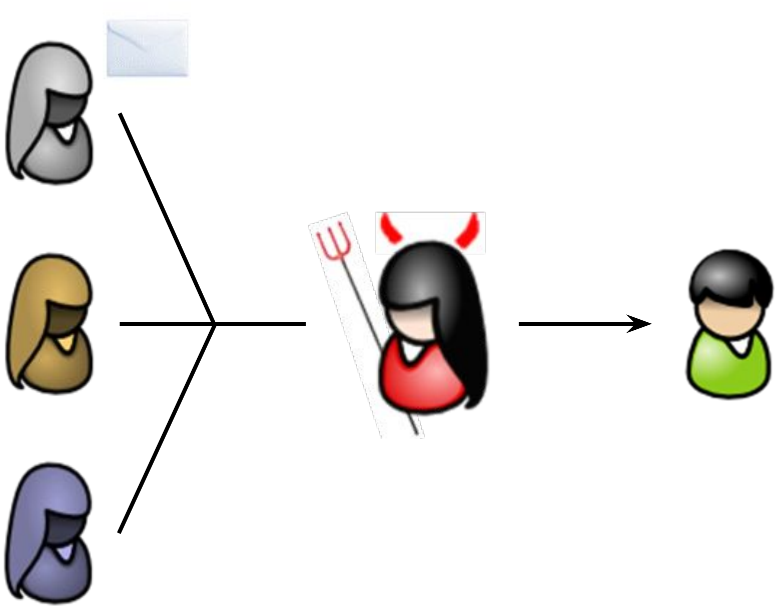
eavesdropper wants to learn identity of mobile user

- Consider a normal PKE scheme
  - Ciphertext hides the message
  - Does it also hide the recipient (the pk used for encryption)?

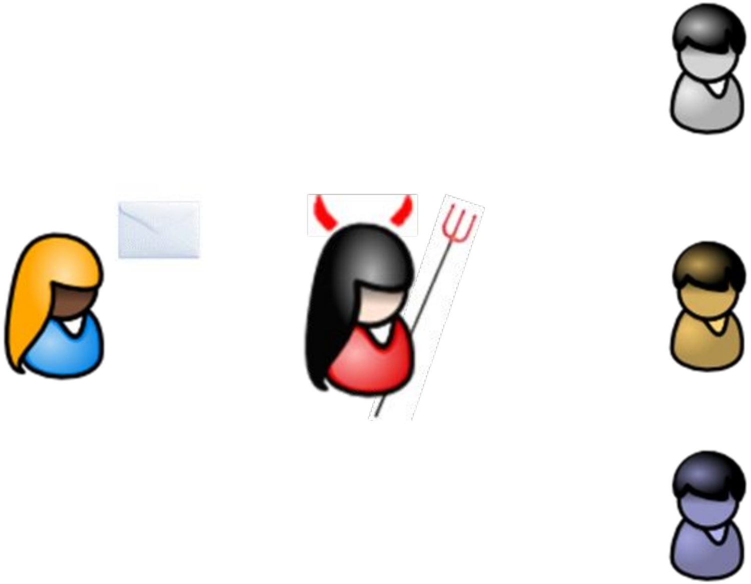
    Not necessarily! Counterexample: take $c = (pk, Enc(pk, m))$

What is Anonymous Encryption?

Sender Anonymity

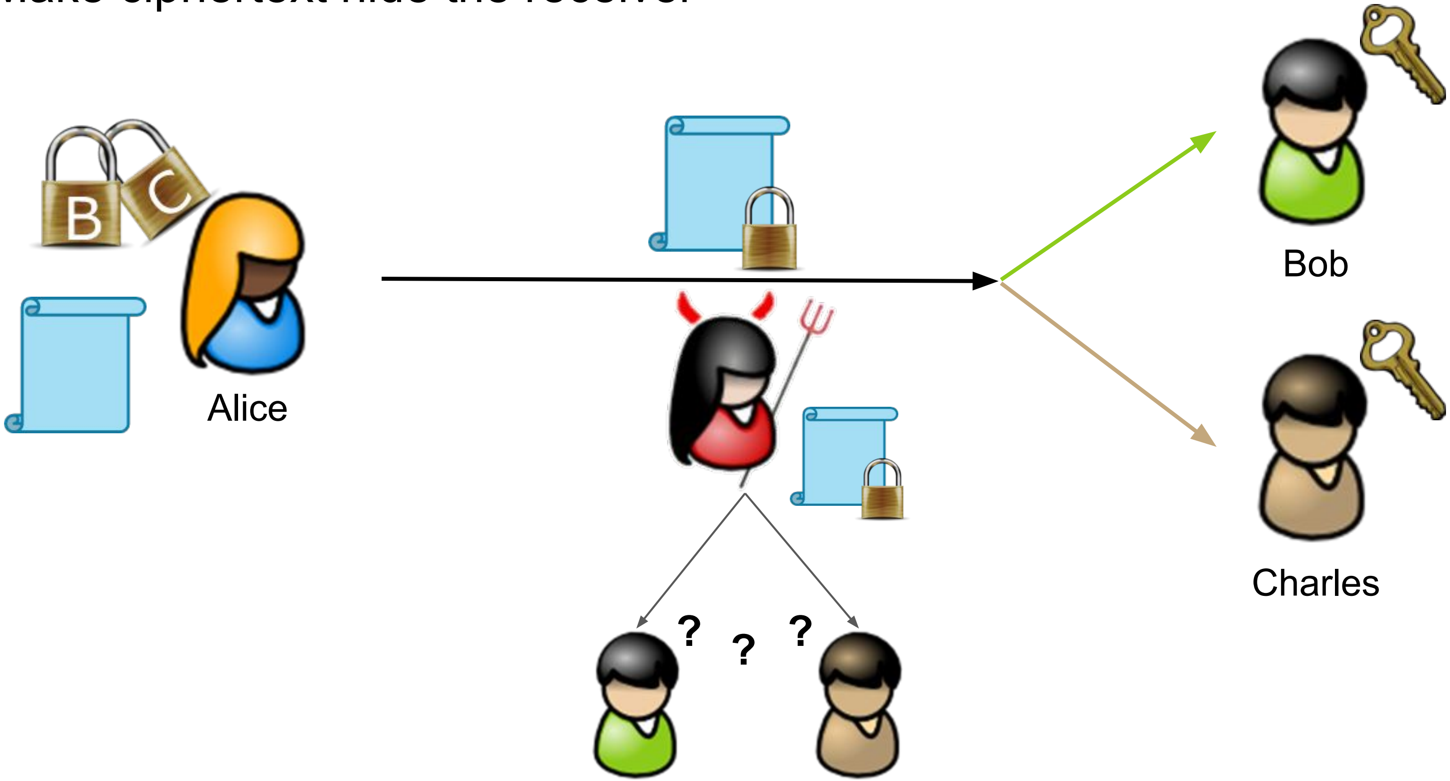


Receiver Anonymity

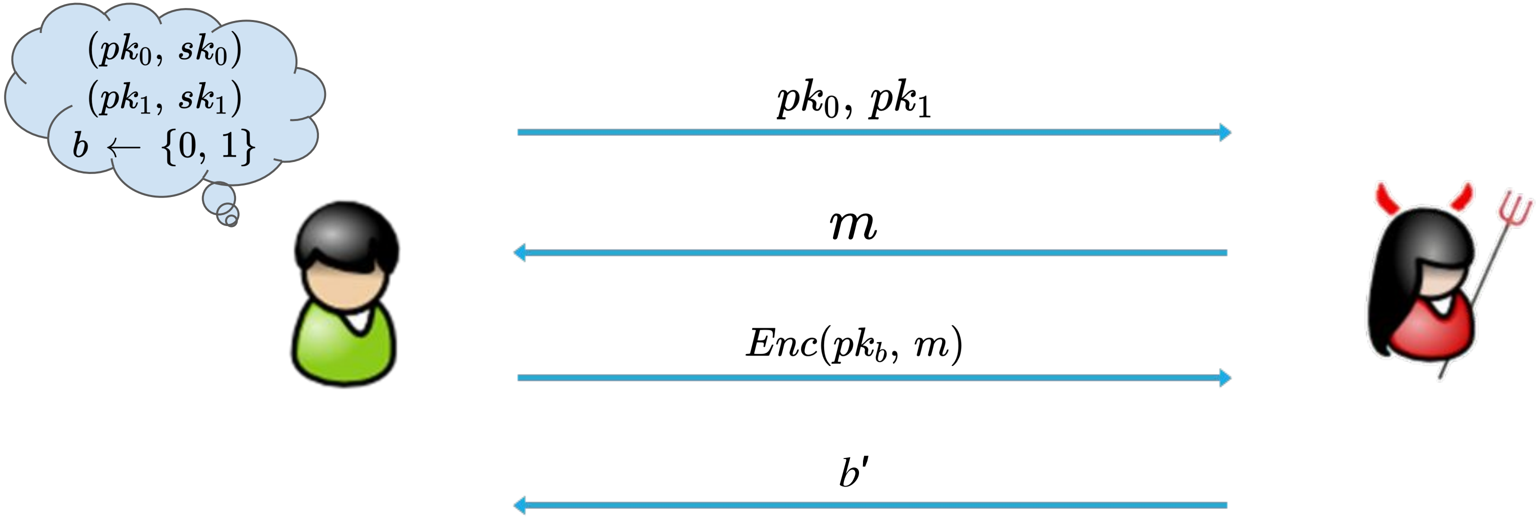


Goal: Receiver Anonymity

- Make ciphertext hide the receiver



Definition of Anonymous PKE



$(pk_0,\, sk_0)$

$(pk_1,\, sk_1)$

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

$pk_0,\, pk_1$

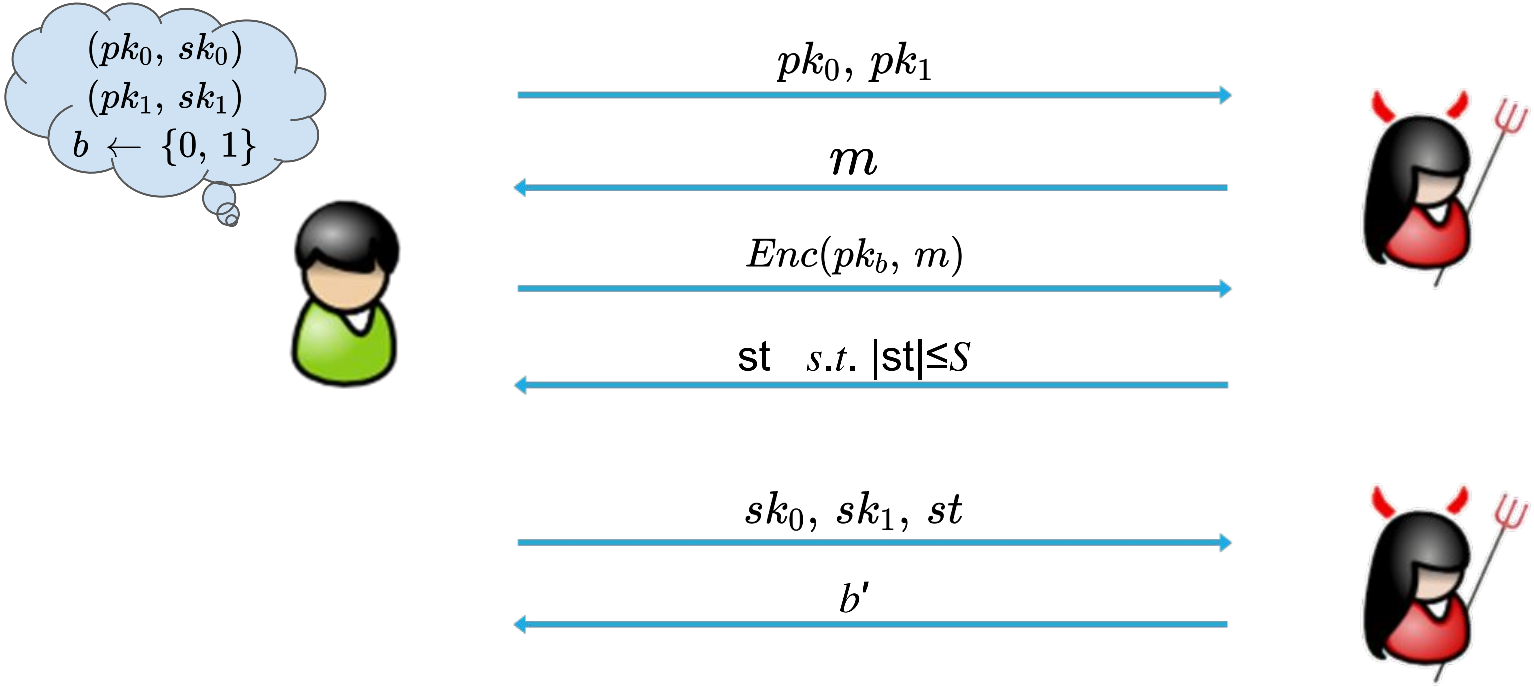
m

$Enc(pk_b,\, m)$

b'

Anonymity in Incompressible Setting

- No existing notion



(pk_0, sk_0)
(pk_1, sk_1)
$b \leftarrow \{0, 1\}$

pk_0, pk_1

m

$Enc(pk_b, m)$

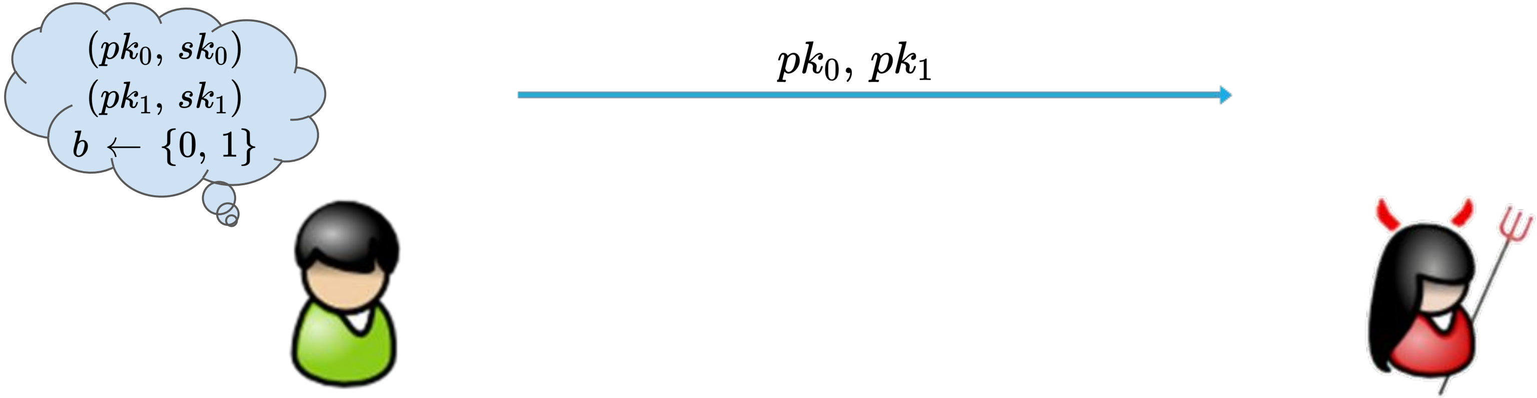
st $s.t.$ |st|≤S

sk_0, sk_1, st

b'

Anonymous Non-Committing Encryption

- No existing notion



$(pk_0,\, sk_0)$

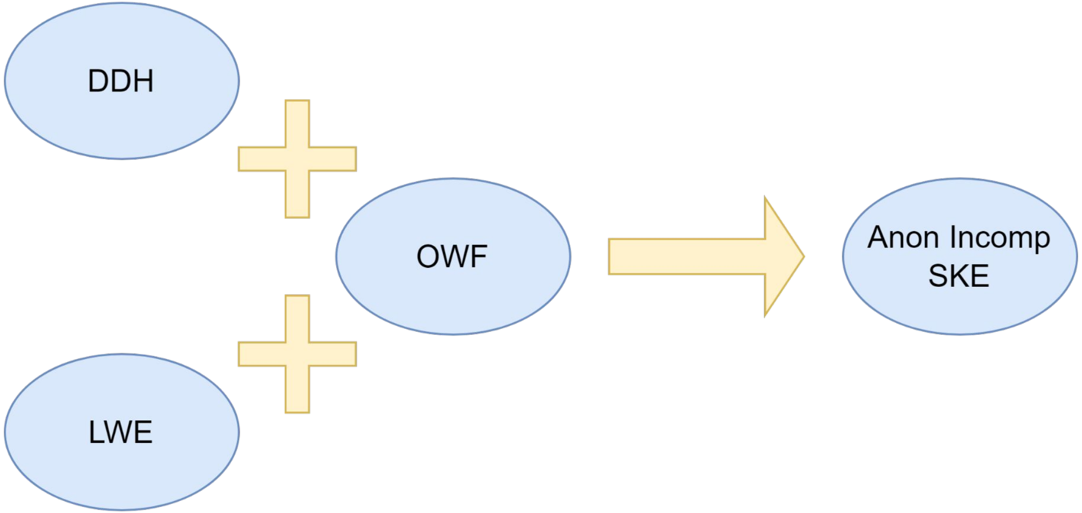
$(pk_1,\, sk_1)$

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

$pk_0,\, pk_1$

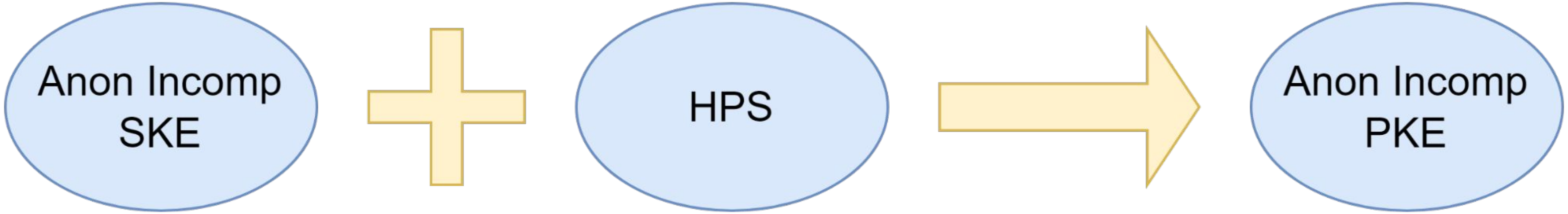
Our Results…

- Incompressible Anonymous SKE (Low Rate)
  - Construction using Strong Extractor
- Incompressible Anonymous SKE (Rate-1)



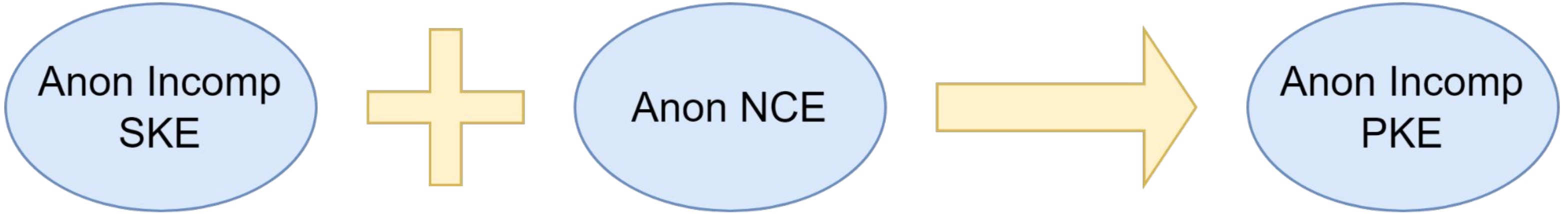
Our Results…

- Incompressible Anonymous PKE (Rate-1)
  - Construction from DDH/LWE



Our Results…

- Incompressible Anonymous PKE
  - Construction from DDH



Next Steps…

- Rate-1 Anon INC SKE in Random Oracle Model?
- Looking at constructing IB-NCE
- INC IBE via NCE
  - IB-NCE + INC SKE => INC IBE?
- Rate-½ strong INC IBE from DDH/LWE?

Thank you!