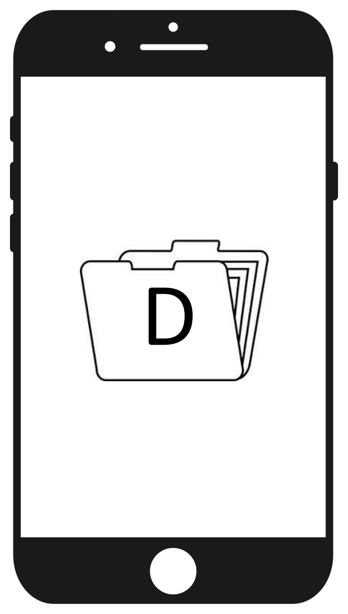
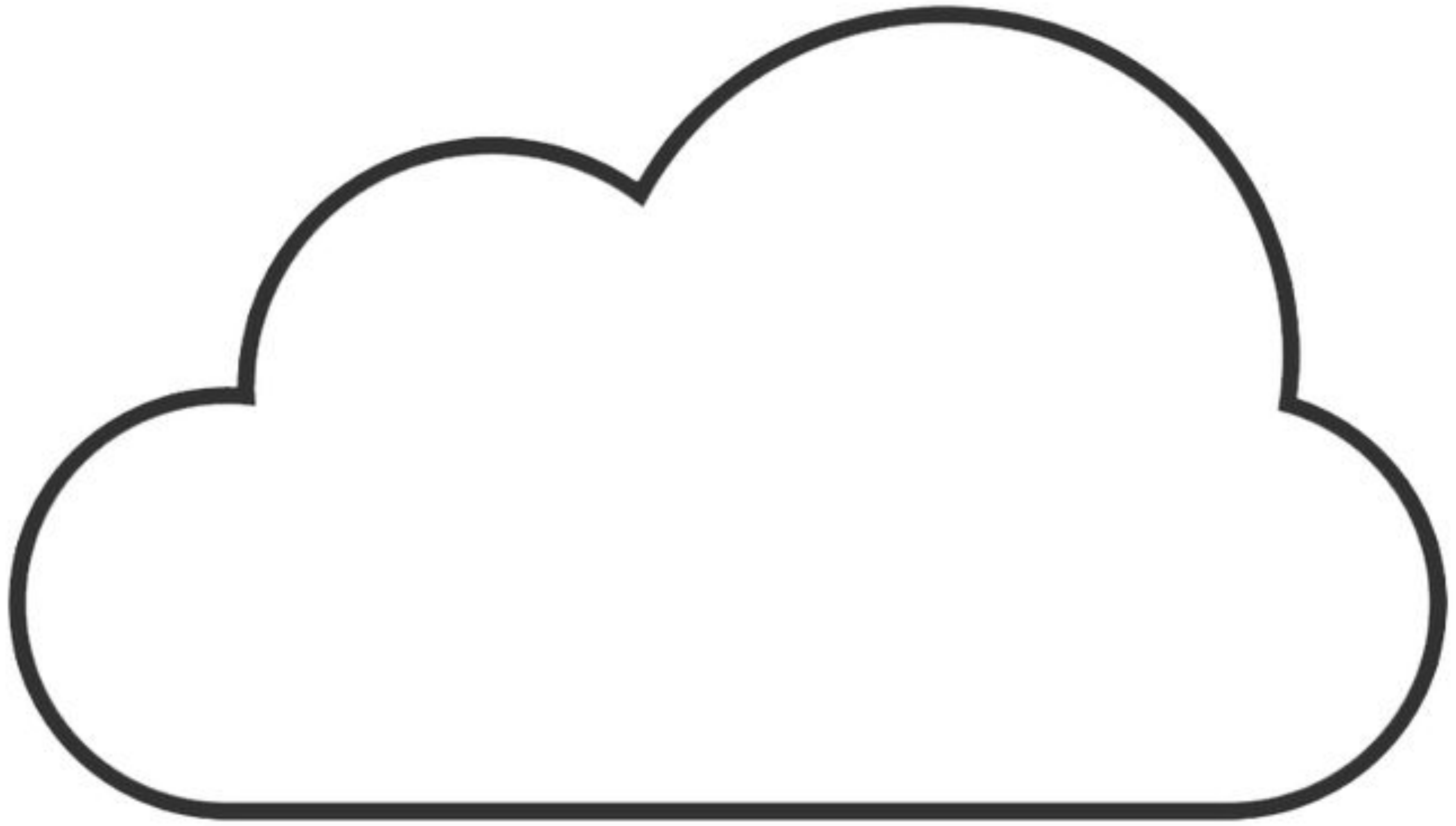
# Weakening Assumptions for Publicly-Verifiable Deletion

James Bartusek, Dakshita Khurana, Giulio Malavolta, Alexander Poremba, Michael Walter

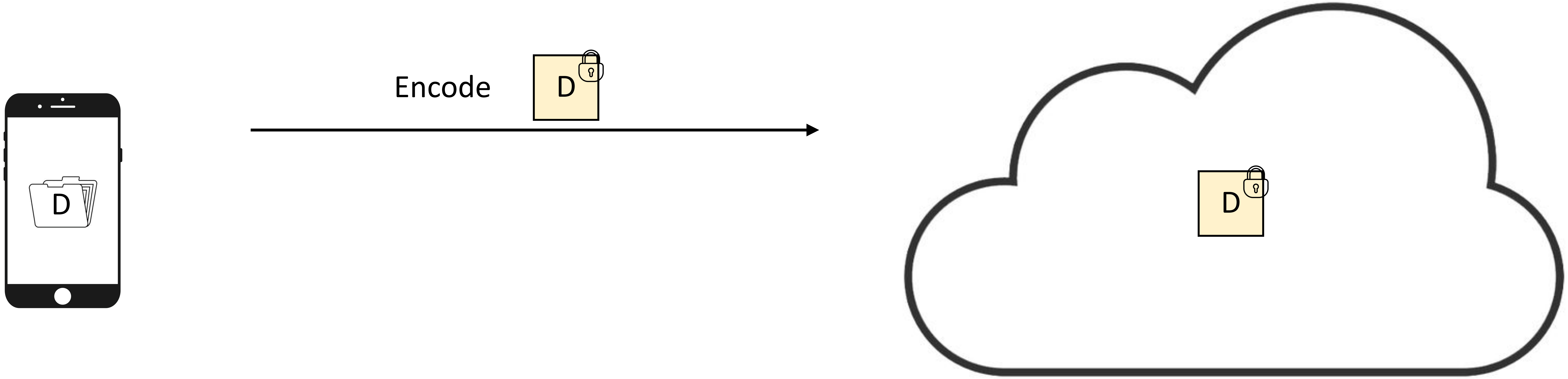
TCC 2023, Taipei

# How to Delete Data

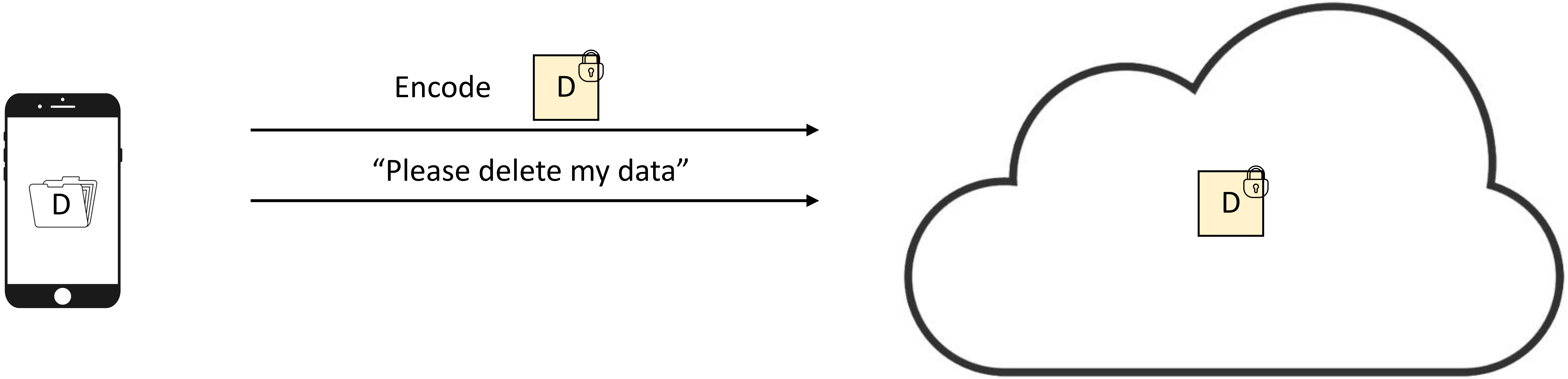


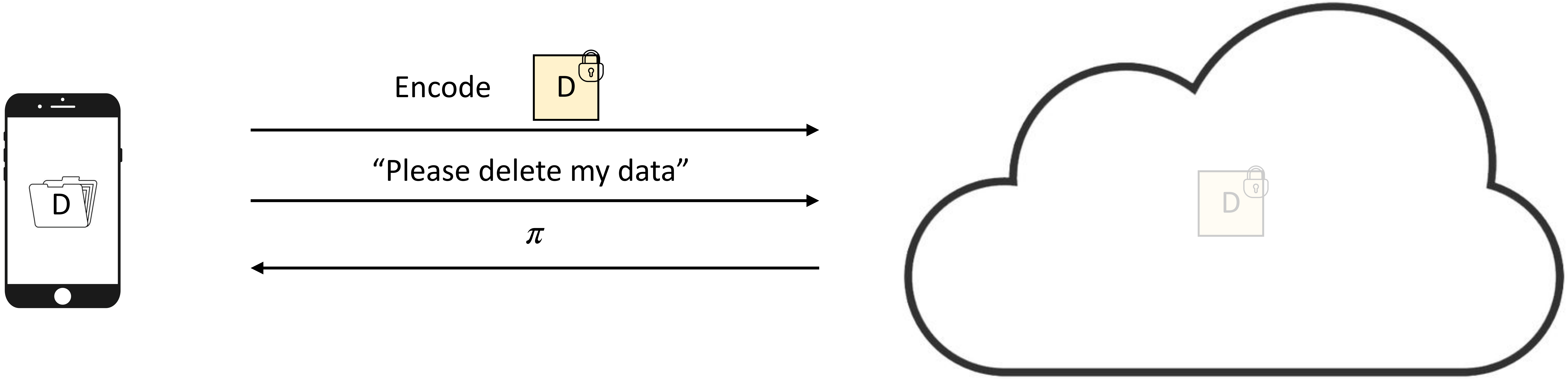
# How to Delete Data



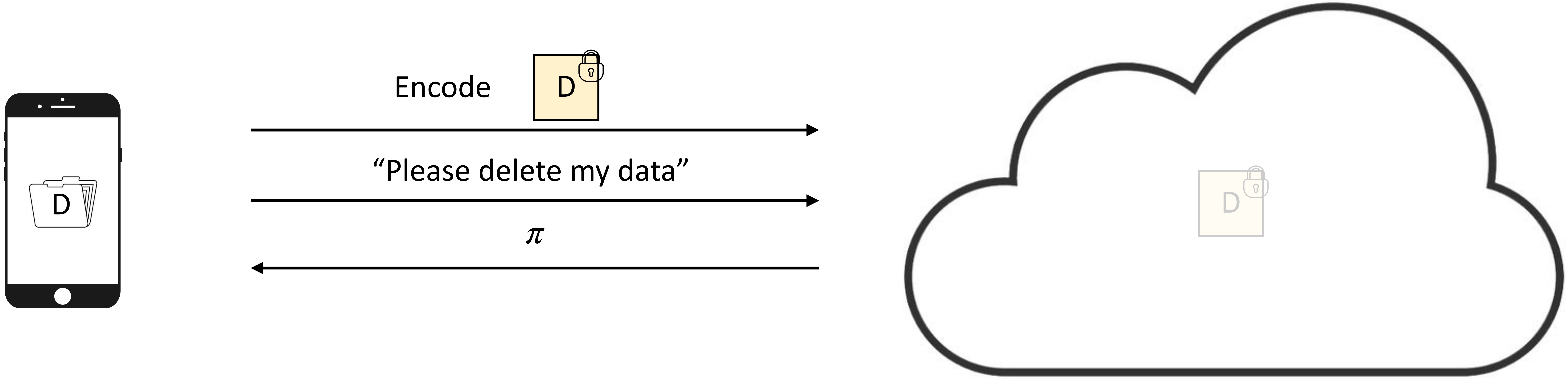
# How to Delete Data



# How to Delete Data

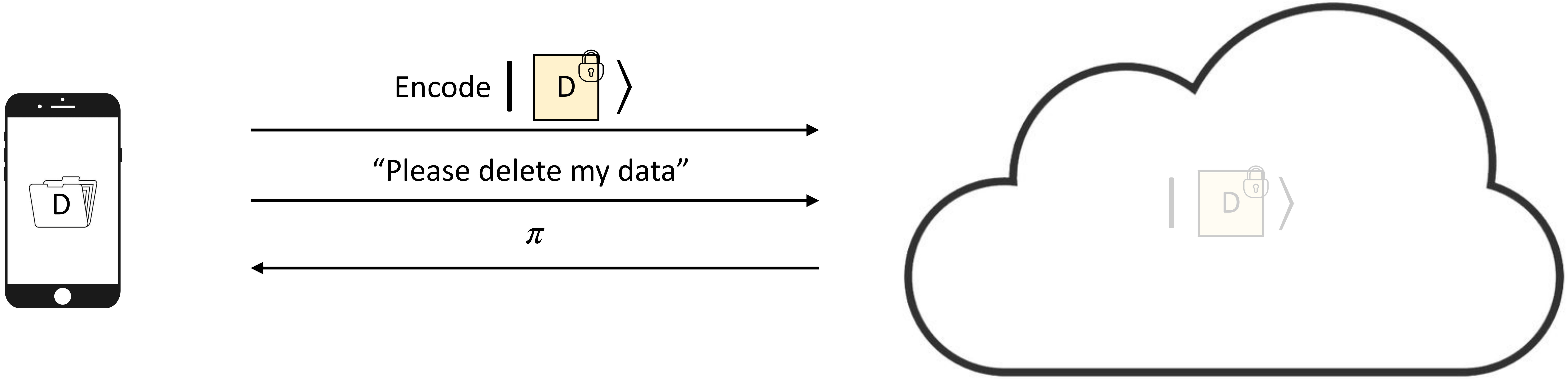


# How to Delete Data



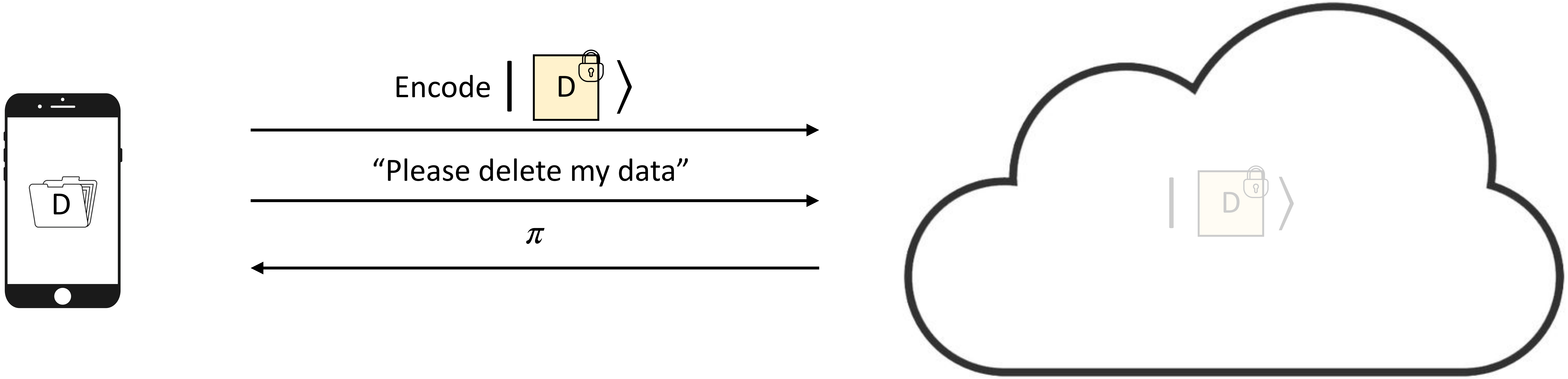

- Classically: Possible if the server is honest

# How to Delete Data




- Classically: Possible if the server is honest
- Quantumly: Possible if the server is malicious but computationally bounded

# How to Delete Data




- Classically: Possible if the server is honest
- Quantumly: Possible if the server is malicious but computationally bounded

$\{$ Commitments, SKE, PKE, ABE, WE, (Q)FHE, TimedE, … $\}$ with certified deletion

Privately-Verifiable Deletion

Privately-Verifiable Deletion

[Unruh13] [BI20] [HMNY21] [HMNY22]
[Poremba22] [BK23] [AKL23] …

Privately-Verifiable Deletion

[Unruh13] [BI20] [HMNY21] [HMNY22]
[Poremba22] [BK23] [AKL23] …

Publicly-Verifiable Deletion

Privately-Verifiable Deletion

[Unruh13] [BI20] [HMNY21] [HMNY22]
[Poremba22] [BK23] [AKL23] …

Publicly-Verifiable Deletion

- [BGGK**M**RR23] from obfuscation

Privately-Verifiable Deletion

[Unruh13] [BI20] [HMNY21] [HMNY22]
[Poremba22] [BK23] [AKL23] …

Publicly-Verifiable Deletion

- [BGGK**M**RR23] from obfuscation
- [BKP23] from almost-regular OWF

Privately-Verifiable Deletion

[Unruh13] [BI20] [HMNY21] [HMNY22] [Poremba22] [BK23] [AKL23] …

Publicly-Verifiable Deletion

- [BGGK**M**RR23] from obfuscation
- [BKP23] from almost-regular OWF

Our Results

Privately-Verifiable Deletion

[Unruh13] [BI20] [HMNY21] [HMNY22] [Poremba22] [BK23] [AKL23] …

Publicly-Verifiable Deletion

- [BGGK**M**RR23] from obfuscation
- [BKP23] from almost-regular OWF

Our Results

A general compiler for X with certified deletion assuming:

X = your favorite cryptographic primitive

Privately-Verifiable Deletion

[Unruh13] [BI20] [HMNY21] [HMNY22]
[Poremba22] [BK23] [AKL23] …

Publicly-Verifiable Deletion

- [BGGK**M**RR23] from obfuscation
- [BKP23] from almost-regular OWF

Our Results

A general compiler for X with certified deletion assuming:

- any OWF (with a classical public-key)

X = your favorite cryptographic primitive

Privately-Verifiable Deletion

[Unruh13] [BI20] [HMNY21] [HMNY22] [Poremba22] [BK23] [AKL23] …

Publicly-Verifiable Deletion

- [BGGK**M**RR23] from obfuscation
- [BKP23] from almost-regular OWF

Our Results

A general compiler for X with certified deletion assuming:

- any OWF (with a classical public-key)
- any OWSG (with a quantum public-key)

X = your favorite cryptographic primitive

Privately-Verifiable Deletion

[Unruh13] [BI20] [HMNY21] [HMNY22] [Poremba22] [BK23] [AKL23] …

Publicly-Verifiable Deletion

- [BGGK**M**RR23] from obfuscation
- [BKP23] from almost-regular OWF

Our Results

A general compiler for X with certified deletion assuming:

- any OWF (with a classical public-key)
- any OWSG (with a quantum public-key)

X = your favorite cryptographic primitive

Concurrent Work [KNY23]

Privately-Verifiable Deletion

[Unruh13] [BI20] [HMNY21] [HMNY22] [Poremba22] [BK23] [AKL23] …

Publicly-Verifiable Deletion

- [BGGK**M**RR23] from obfuscation
- [BKP23] from almost-regular OWF

Our Results

A general compiler for X with certified deletion assuming:

- any OWF (with a classical public-key)
- any OWSG (with a quantum public-key)

X = your favorite cryptographic primitive

Concurrent Work [KNY23]

- Similar (but not identical!) compiler from OWF

Privately-Verifiable Deletion

[Unruh13] [BI20] [HMNY21] [HMNY22] [Poremba22] [BK23] [AKL23] …

Publicly-Verifiable Deletion

- [BGGK**M**RR23] from obfuscation
- [BKP23] from almost-regular OWF

Our Results

A general compiler for X with certified deletion assuming:

- any OWF (with a classical public-key)
- any OWSG (with a quantum public-key)

X = your favorite cryptographic primitive

Concurrent Work [KNY23]

- Similar (but not identical!) compiler from OWF
- Minimality theorem from *hard quantum planted problems for NP*

# Our Compiler

# Our Compiler

$$\mathsf{vk} = y_0, y_1$$

# Our Compiler

$$\mathsf{vk} = y_0, y_1$$

$$|\,\mathsf{ct} > = \mathsf{Enc}(x_0 \oplus x_1) \quad \text{and} \quad \frac{1}{\sqrt{2}}\left(\,|\,x_0 > + \,(-1)^{\mathsf{msg}}\,|\,x_1 >\,\right)$$

# Our Compiler

$$\mathsf{vk} = y_0, y_1$$

$$|\,\mathsf{ct}> = \mathsf{Enc}(x_0 \oplus x_1) \quad \text{and} \quad \frac{1}{\sqrt{2}}\left(|x_0> + (-1)^{\mathsf{msg}}|x_1>\right)$$

- Decrypt: Measure in the Hadamard basis and decrypt the classical cipher

# Our Compiler

$$\mathsf{vk} = y_0, y_1$$

$$|\mathsf{ct}> = \mathsf{Enc}(x_0 \oplus x_1) \quad \text{and} \quad \frac{1}{\sqrt{2}}\left(|x_0> + (-1)^{\mathsf{msg}}|x_1>\right)$$

- Decrypt: Measure in the Hadamard basis and decrypt the classical cipher
- Delete: Measure in the comp. basis and check if $\mathsf{OWF}(x_b) = y_b$

# Our Compiler

$$\mathsf{vk} = y_0, y_1$$

$$|\,\mathsf{ct} > = \mathsf{Enc}(x_0 \oplus x_1) \quad \text{and} \quad \frac{1}{\sqrt{2}}\left(|\,x_0 > + (-1)^{\mathsf{msg}}|\,x_1 > \right)$$

- Decrypt: Measure in the Hadamard basis and decrypt the classical cipher
- Delete: Measure in the comp. basis and check if $\mathsf{OWF}(x_b) = y_b$

Technique inspired by recent works on Quantum PKE [BGH**M**SVW] [HKNY23] [**M**W23]

# Main Theorem

b = 0

$$\mathscr{A}\left\{ \begin{matrix} (y_0, y_1) \quad \mathsf{Enc}(x_0 \oplus x_1) \\ \frac{1}{\sqrt{2}}\left( |x_0> + |x_1> \right) \end{matrix} \right\}$$

$$\downarrow$$

$$x^* : \mathsf{OWF}(x^*) = y_0 \text{ OR } \mathsf{OWF}(x^*) = y_1$$

b = 1

$$\mathscr{A}\left\{ \begin{matrix} (y_0, y_1) \quad \mathsf{Enc}(x_0 \oplus x_1) \\ \frac{1}{\sqrt{2}}\left( |x_0> - |x_1> \right) \end{matrix} \right\}$$

$$\downarrow$$

$$x^* : \mathsf{OWF}(x^*) = y_0 \text{ OR } \mathsf{OWF}(x^*) = y_1$$

# Main Theorem

b = 0

$$\mathscr{A}\left\{ \begin{array}{ll} (y_0, y_1) & \mathsf{Enc}(x_0 \oplus x_1) \\ \frac{1}{\sqrt{2}}\left( |x_0 > + |x_1 > \right) & \end{array} \right\}$$

$$\downarrow$$

$x^*$ : OWF$(x^*) = y_0$ OR OWF$(x^*) = y_1$

b = 1

$$\mathscr{A}\left\{ \begin{array}{ll} (y_0, y_1) & \mathsf{Enc}(x_0 \oplus x_1) \\ \frac{1}{\sqrt{2}}\left( |x_0 > - |x_1 > \right) & \end{array} \right\}$$

$$\downarrow$$

$x^*$ : OWF$(x^*) = y_0$ OR OWF$(x^*) = y_1$

$$\boxed{\text{Claim: TD}\left(\mathrm{out}_0, \mathrm{out}_1\right) = \mathrm{negl}(\lambda)}$$

# Proof Sketch

- Step I: Delay the choice of the bit
- Step II: Measure the first register in the Hadamard basis, before measuring c

# Proof Sketch

- Step I: Delay the choice of the bit $$\frac{1}{2}\sum_{c} |c> \left( |x_0> + (-1)^c |x_1> \right)$$

- Step II: Measure the first register in the Hadamard basis, before measuring c

# Proof Sketch

- Step I: Delay the choice of the bit

$$\frac{1}{2}\sum_{c} |c> \left( |x_0> + (-1)^c |x_1> \right)$$

- Step II: Measure the first register in the Hadamard basis, before measuring c

$\boxed{\text{Success probability} = 1/2}$

# Proof Sketch

- Step I: Delay the choice of the bit $$\frac{1}{2}\sum_c |c> \Big( |x_0> + (-1)^c |x_1> \Big)$$

$$\approx_c \text{(by semantic security)}$$

- Step II: Measure the first register in the Hadamard basis, before measuring c

$$\boxed{\text{Success probability = 1/2}}$$

# Summary

# Summary

- We have X with certified deletion, assuming X and OWF (or OWSG)

# Summary

- We have X with certified deletion, assuming X and OWF (or OWSG)
  - X = {Commitments, PKE, ABE, FHE…}

# Summary

- We have X with certified deletion, assuming X and OWF (or OWSG)
  - X = {Commitments, PKE, ABE, FHE…}
- Open problems:

# Summary

- We have X with certified deletion, assuming X and OWF (or OWSG)
  - X = {Commitments, PKE, ABE, FHE…}
- Open problems:
  - Construction using product states?

# Summary

- We have X with certified deletion, assuming X and OWF (or OWSG)
  - X = {Commitments, PKE, ABE, FHE…}
- Open problems:
  - Construction using product states?
  - Even weaker assumptions?

# Summary

- We have X with certified deletion, assuming X and OWF (or OWSG)
  - X = {Commitments, PKE, ABE, FHE…}
- Open problems:
  - Construction using product states?
  - Even weaker assumptions?
  - More crypto with certified deletion?