Quantum encryption with certified deletion

Anne Broadbent, Rabib Islam

(University of Ottawa)

Motivation

Motivation

Motivation

Motivation

Motivation




Motivation

Motivation

Motivation

Motivation

"I deleted the ciphertext!"

Motivation

“I deleted the ciphertext!”

“How do I know?”

Motivation

Motivation

- With a classical ciphertext, Bob cannot prove deletion to Alice
  - Bob can always make a copy of the ciphertext that can be decrypted once the key is received

Motivation

- With a classical ciphertext, Bob cannot prove deletion to Alice
    - Bob can always make a copy of the ciphertext that can be decrypted once the key is received
- Therefore, we must consider a non-classical solution

A solution

A solution

- A quantum ciphertext?
  - No-cloning theorem: there is no map that will create an identical copy of an arbitrary quantum state

A solution

- A quantum ciphertext?
  - No-cloning theorem: there is no map that will create an identical copy of an arbitrary quantum state
- But what would a proof of deletion look like?

A solution

- A quantum ciphertext?
  - No-cloning theorem: there is no map that will create an identical copy of an arbitrary quantum state
- But what would a proof of deletion look like?
- Entropic uncertainty relations: measurement in one basis can cause loss of information about what the measurement outcome in another basis would have been

A solution

- A quantum ciphertext?
  - No-cloning theorem: there is no map that will create an identical copy of an arbitrary quantum state
- But what would a proof of deletion look like?
- Entropic uncertainty relations: measurement in one basis can cause loss of information about what the measurement outcome in another basis would have been
- Conjugate coding (Wiesner/BB84 states) and measurements will be integral to our scheme

Previous results

Previous results

Context for the idea

Previous results

Context for the idea

- [Unruh 2013] “Revocable quantum timed-release encryption”

Previous results

Context for the idea

- [Unruh 2013] “Revocable quantum timed-release encryption”
- [Fu and Miller 2018] “Local randomness: Examples and application”

Previous results

Context for the idea

- [Unruh 2013] “Revocable quantum timed-release encryption”
- [Fu and Miller 2018] “Local randomness: Examples and application”
- [Bennett and Brassard 1984] “Quantum cryptography: Public key distribution and coin-tossing”
  - [Tomamichel and Leverrier 2017] “A largely self-contained and complete security proof for quantum key distribution”

Previous results

Context for the idea

- [Unruh 2013] “Revocable quantum timed-release encryption”
- [Fu and Miller 2018] “Local randomness: Examples and application”
- [Bennett and Brassard 1984] “Quantum cryptography: Public key distribution and coin-tossing”
  - [Tomamichel and Leverrier 2017] “A largely self-contained and complete security proof for quantum key distribution”
- [Coiteux-Roy and Wolf 2019] “Proving Erasure”

Scheme: parameters

Scheme: parameters

- n: length of the message

Scheme: parameters

- n: length of the message
- m: number of qubits used in the quantum encoding

Scheme: key generation

Scheme: key generation

- $\theta \leftarrow \{\theta \in \{0,1\}^m \mid \omega(\theta) = k\}$, where k is less than m.
  - Basis for encoding qubits
  - Content of qubits: string of length m called r

Scheme: key generation

- $\theta \leftarrow \{\theta \ \in \ \{0, 1\}^m \mid \omega(\theta) \ = \ k\},$ where k is less than m.
  - Basis for encoding qubits
  - Content of qubits: string of length m called r
- $r_{diag} \leftarrow \{0, 1\}^k$
  - Also called “check bits”

Scheme: key generation

- $\theta \leftarrow \{\theta \in \{0, 1\}^m \mid \omega(\theta) = k\},$ where k is less than m.
  - Basis for encoding qubits
  - Content of qubits: string of length m called r
- $r_{diag} \leftarrow \{0, 1\}^k$
  - Also called "check bits"
- $u \leftarrow \{0, 1\}^n$

Scheme: key generation

- $\theta \leftarrow \{\theta \in \{0, 1\}^m \mid \omega(\theta) = k\},$ where k is less than m.
    - Basis for encoding qubits
    - Content of qubits: string of length m called r
- $r_{diag} \leftarrow \{0, 1\}^k$
    - Also called “check bits”
- $u \leftarrow \{0, 1\}^n$
- $H \leftarrow$ universal$_2$ family of hash functions
    - Domain: strings of length $m - k$; codomain: strings of length n

Scheme: encryption

Scheme: encryption

- $r_{comp} \leftarrow \{0, 1\}^{m-k}$

Scheme: encryption

- $r_{comp} \leftarrow \{0, 1\}^{m-k}$
- $x \leftarrow H(r_{comp})$

Scheme: encryption

- $r_{comp} \leftarrow \{0, 1\}^{m-k}$
- $x \leftarrow H(r_{comp})$
- Ciphertext: r encoded in basis θ, with $msg \bigoplus x \bigoplus u$.

Scheme: decryption

Scheme: decryption

- Measure qubits in basis θ to yield r, and hence r_{comp}

Scheme: decryption

- Measure qubits in basis θ to yield r, and hence r_{comp}
- Compute $H(r_{comp}) = x$.

Scheme: decryption

- Measure qubits in basis θ to yield r, and hence r_{comp}
- Compute $H(r_{comp}) = x$.
- Compute $msg \bigoplus x \bigoplus u \bigoplus x \bigoplus u = msg$.

Scheme: delete

Scheme: delete

- Measure qubits in the Hadamard basis and obtain a certificate of deletion $y \leftarrow \{0, 1\}^m$

Scheme: verification

Scheme: verification

- Using θ, take the substring of the received string that corresponds to the diagonal positions of the qubits (call the result y').

Scheme: verification

- Using θ, take the substring of the received string that corresponds to the diagonal positions of the qubits (call the result y').
- Accept if $\omega\left(r_{diag} \bigoplus y'\right) < \delta k$.

Error tolerance

Error tolerance

- We use linear error correcting codes and a hash function

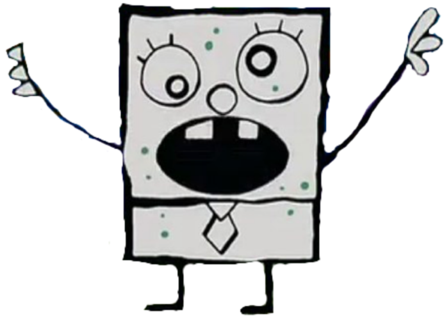
Error tolerance

- We use linear error correcting codes and a hash function
- More details in the paper

Certified deletion security: Game 1

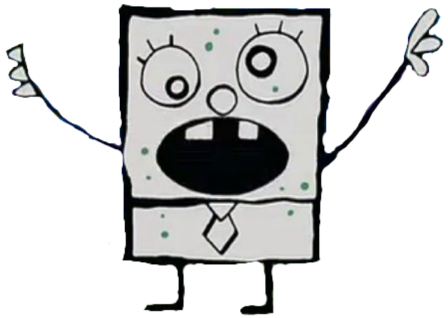
Certified deletion security: Game 1

Certified deletion security: Game 1



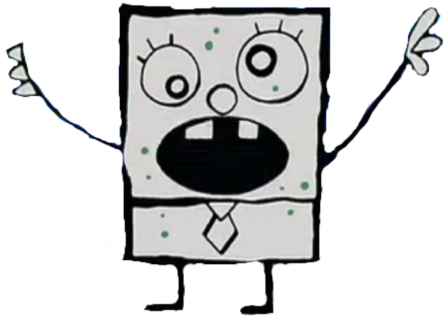
Certified deletion security: Game 1

msg_0



Certified deletion security: Game 1

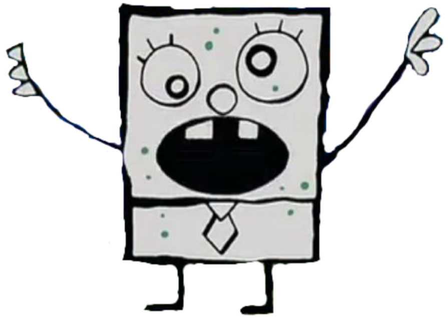
msg_0



Certified deletion security: Game 1

msg_0

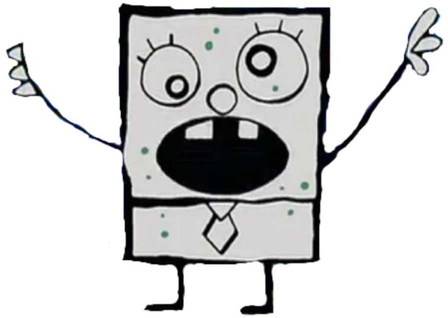
$\theta,\ u,\ H,$
r_{diag}



Certified deletion security: Game 1

msg_0

$\theta,\ u,\ H,$
r_{diag}



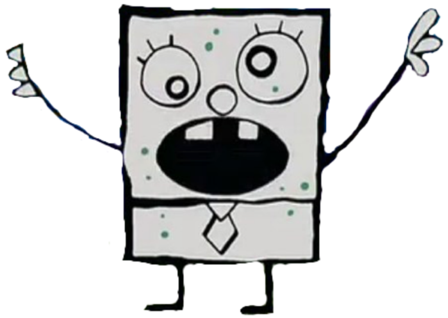
b

Certified deletion security: Game 1

msg_0

$\theta,\ u,\ H,$
r_{diag}
ciphertext

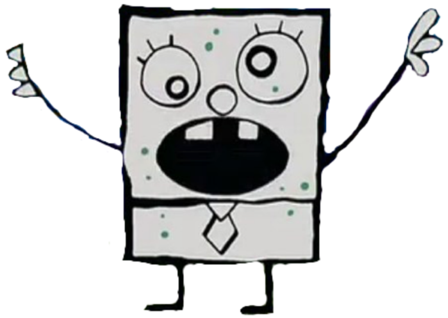
b



Certified deletion security: Game 1

msg_0

$\theta,\ u,\ H,$
r_{diag}



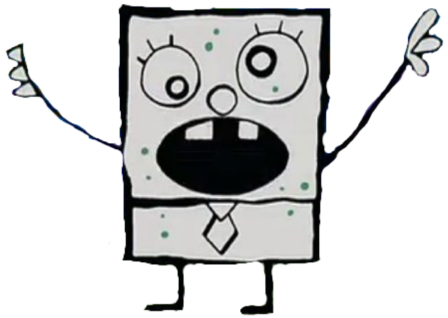
ciphertext

b

Certified deletion security: Game 1

msg_0

y



$\theta,\ u,\ H,$
r_{diag}

ciphertext

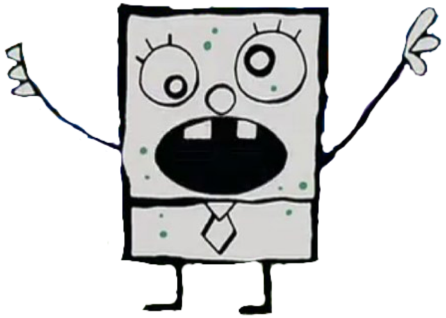
b

Certified deletion security: Game 1

msg_0

y

$\theta,\ u,\ H,$
r_{diag}



ciphertext

b

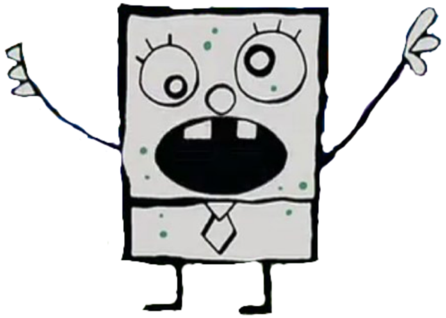
Certified deletion security: Game 1

msg_0

y

$\theta,\ u,\ H,$
r_{diag}

ciphertext

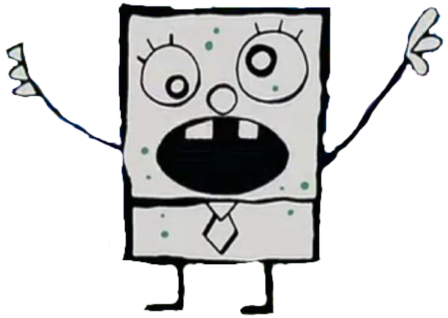


b ok

Certified deletion security: Game 1

msg_0

y



$\theta,\ u,\ H,$
r_{diag}

ciphertext

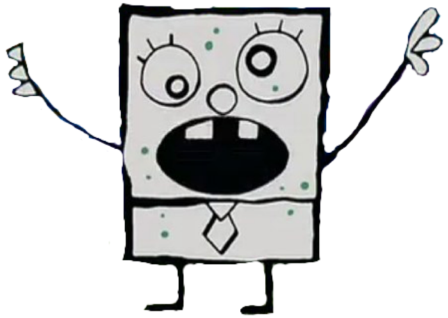
b ok

Certified deletion security: Game 1

msg_0

y

b'



$\theta,\ u,\ H,$
r_{diag}

ciphertext

b ok

Certified deletion security: Game 1

Certified deletion security: Game 1

- Bob can be seen as having two goals:
  1. Determine whether his message was encrypted in the ciphertext
  2. Convince Alice that he deleted the ciphertext prior to receiving the key

Certified deletion security: Game 1

- Bob can be seen as having two goals:
  1. Determine whether his message was encrypted in the ciphertext
  2. Convince Alice that he deleted the ciphertext prior to receiving the key
- Scheme is secure if the probabilities of the following two events are negligibly close:
  1. Verification passes and Bob's guess of b is 1, in the case that Alice encrypted the string of zeros
  2. Verification passes and Bob's guess of b is 1, in the case that Alice encrypted the candidate message.

Certified deletion security: Game 2

Certified deletion security: Game 2

- Game 1 is difficult to analyze

Certified deletion security: Game 2

- Game 1 is difficult to analyze
- We developed a Game 2 which is based on an entanglement-based series of interactions

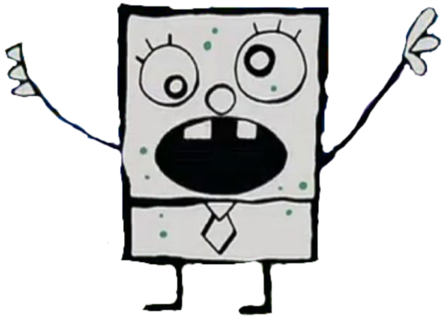
Certified deletion security: Game 2

- Game 1 is difficult to analyze
- We developed a Game 2 which is based on an entanglement-based series of interactions
- A reduction shows that statements about Game 2 can translate into statements about Game 1
  - We thereby achieve bounds relevant to our scheme

Certified deletion security: Game 2

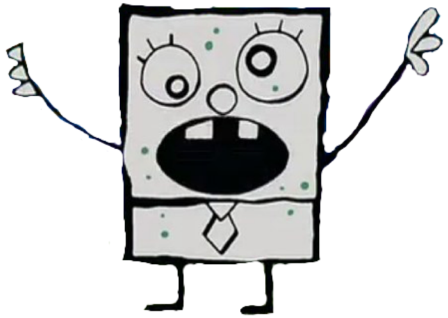
Certified deletion security: Game 2

Certified deletion security: Game 2



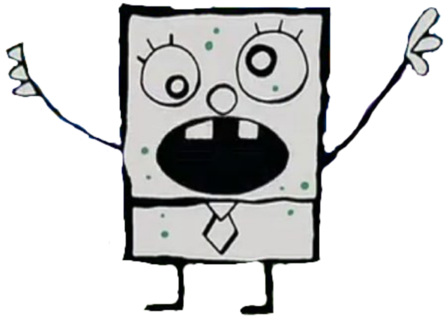
Certified deletion security: Game 2

msg_0



Certified deletion security: Game 2

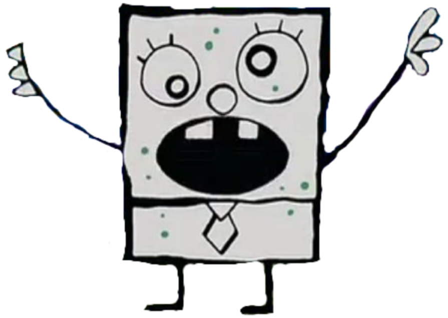
msg_0



Certified deletion security: Game 2

msg_0

A
B
B'

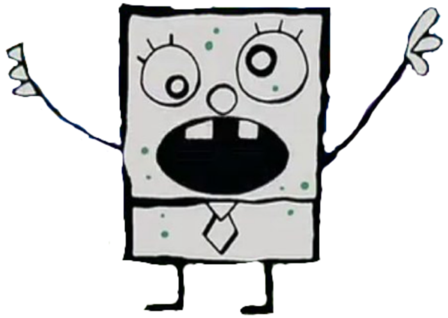


Certified deletion security: Game 2

msg_0

A

B
B'

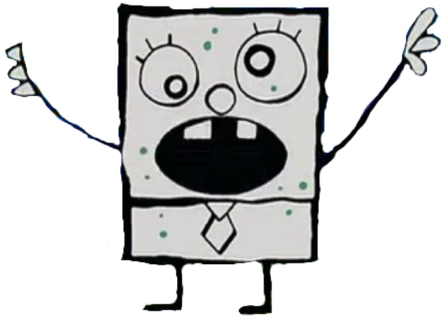


Certified deletion security: Game 2

msg_0

A

B'



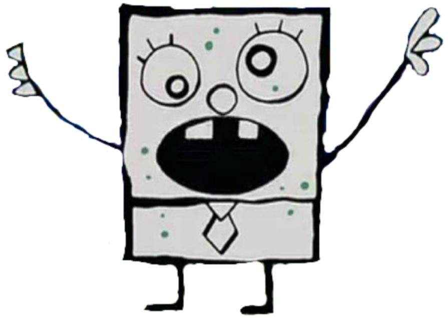
Certified deletion security: Game 2

msg_0

A

y

B'

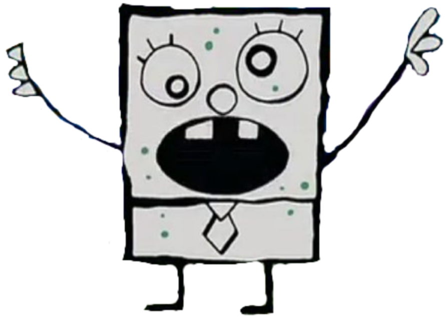


Certified deletion security: Game 2

msg_0

A
y

B'



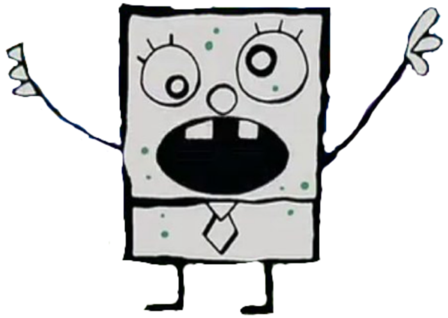
Certified deletion security: Game 2

msg_0

A
y

$\theta,\ u,\ H$

B'



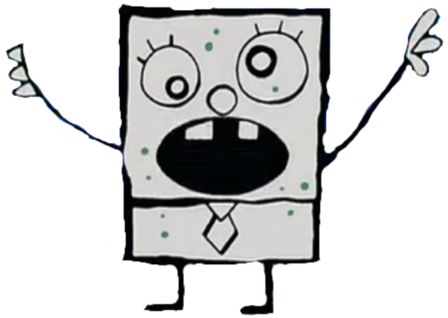
Certified deletion security: Game 2

msg_0

y

$\theta,\ u,\ H$

B'



Certified deletion security: Game 2

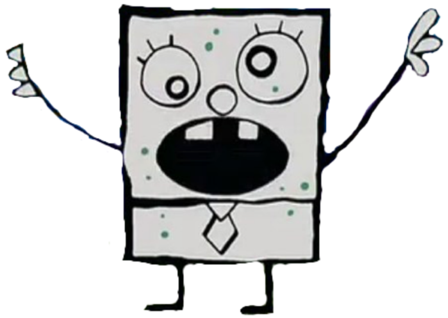
msg_0

y

$\theta,\ u,\ H$

r

B'



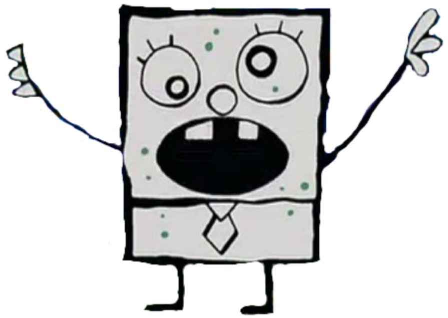
Certified deletion security: Game 2

msg_0

y

$\theta,\ u,\ H$

B'



Certified deletion security: Game 2

msg_0

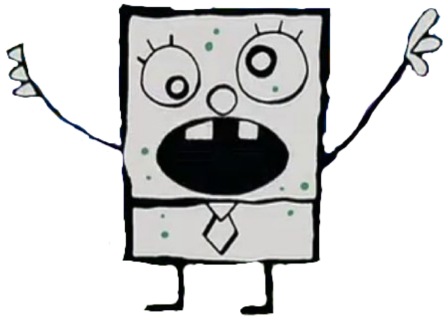
y

$\theta,\ u,\ H$

r_{comp}

r_{diag}

B'



Certified deletion security: Game 2

msg_0

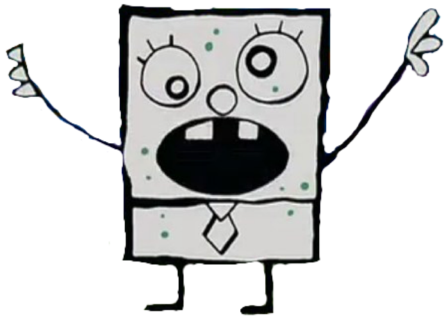
y

$\theta,\ u,\ H$

r_{comp} x

r_{diag}

B'



Certified deletion security: Game 2

msg_0

y

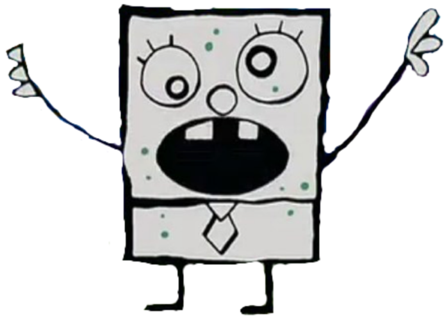
$\theta,\ u,\ H$

r_{comp} x

r_{diag}

b

B'



Certified deletion security: Game 2

msg_0

y

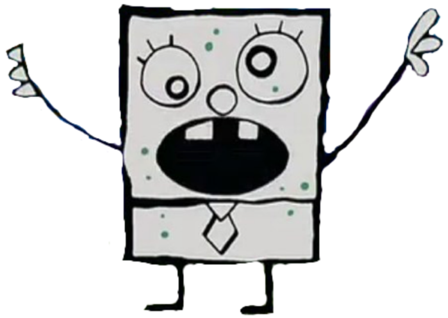
$\theta,\ u,\ H$

r_{comp} x

r_{diag}

b ok

B'



Certified deletion security: Game 2

msg_0

y

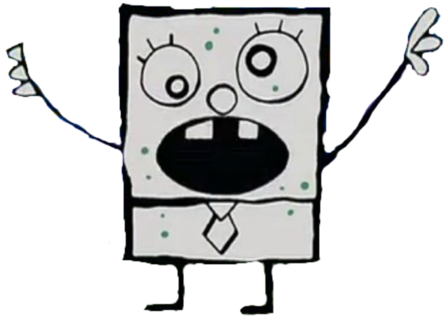
B'

$\theta,\ u,\ H$

r_{comp} x

r_{diag}

b ok $msg \bigoplus x \bigoplus u \bigoplus x \bigoplus u$



Certified deletion security: Game 2

msg_0

y

B'

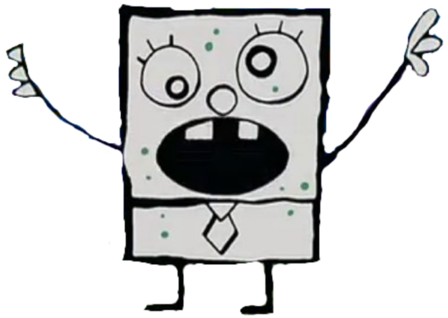
$\theta,\ u,\ H$

r_{comp} x

r_{diag}

b ok

$msg \bigoplus x \bigoplus u \bigoplus x \bigoplus u$



Certified deletion security: Game 2

msg_0

y

B'

$\theta, \ u, \ H$

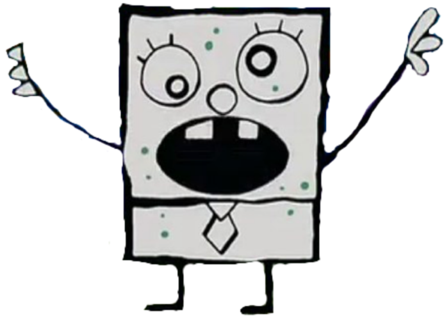
r_{comp} x

b'

r_{diag}

b ok

$msg \oplus x \oplus u \oplus x \oplus u$



Certified deletion security: similarity

Certified deletion security: similarity

- Entanglement in Game 2 corresponds to Bob's measurement in Game 1
  - Measuring everything in the Hadamard basis in Game 1 is like fully entangling A and B in Game 2 – this will give him r_{diag}
  - Measuring everything in the computational basis in Game 1 is like fully entangling A and B' in Game 2, and then measuring B' in the computational basis – this will give him r_{comp}

Entropic uncertainty relation

Entropic uncertainty relation

- Entanglement-based setting allows use of entropic uncertainty relations

Entropic uncertainty relation

- Entanglement-based setting allows use of entropic uncertainty relations
- We use one from work by Tomamichel (arXiv: 1203.2142)

Entropic uncertainty relation

- Entanglement-based setting allows use of entropic uncertainty relations
- We use one from work by Tomamichel (arXiv: 1203.2142)
- Here, it can be used to describe the information trade-off that Bob is making in Game 2 using smooth min- and max-entropies.

Entropic uncertainty relation

- Entanglement-based setting allows use of entropic uncertainty relations
- We use one from work by Tomamichel (arXiv: 1203.2142)
- Here, it can be used to describe the information trade-off that Bob is making in Game 2 using smooth min- and max-entropies.
- Takeaway: if the verification test is passed: the information that Bob has access to about r_{comp} is low with high probability

Privacy amplification

Privacy amplification

- The hash function accomplishes the task of privacy amplification

Privacy amplification

- The hash function accomplishes the task of privacy amplification
- Formalized using the Leftover Hashing Lemma from Renner
  - Lower bound on Bob's uncertainty about r_{comp} tells us how close x is to a uniformly random string from Bob's perspective
  - Bob is blocked from getting information about msg

Applications and Next Steps

Applications and Next Steps

- Protection against key leakage

Applications and Next Steps

- Protection against key leakage
- Protection against data retention
  - EU regulation 2016/679

Applications and Next Steps

- Protection against key leakage
- Protection against data retention
  - EU regulation 2016/679
- Everlasting security
  - Transform long-term computational assumption into a temporary one

Applications and Next Steps

- Protection against key leakage
- Protection against data retention
  - EU regulation 2016/679
- Everlasting security
  - Transform long-term computational assumption into a temporary one
- Homomorphic encryption

Thank you!