Robust Quantum Public-Key Encryption

(With Applications to Quantum Key Distribution)

Giulio Malavolta (Bocconi University & MPI-SP)
Michael Walter (Ruhr University Bochum)

European Research Council
Established by the European Commission

https://arxiv.org/pdf/2304.02999.pdf

Classical Key Exchange

Quantum Key Exchange

Classical Key Exchange

- Security against polynomial-time attacker

Quantum Key Exchange

Classical Key Exchange

- Security against polynomial-time attacker
- Requires computational assumptions (e.g., DDH, LWE, LPN…)

Quantum Key Exchange

Classical Key Exchange

- Security against polynomial-time attacker
- Requires computational assumptions (e.g., DDH, LWE, LPN…)
- Two-message protocol (minimal)

Quantum Key Exchange

Classical Key Exchange

- Security against polynomial-time attacker
- Requires computational assumptions (e.g., DDH, LWE, LPN…)
- Two-message protocol (minimal)

Quantum Key Exchange

- Unconditional (?) security

Classical Key Exchange

- Security against polynomial-time attacker
- Requires computational assumptions (e.g., DDH, LWE, LPN…)
- Two-message protocol (minimal)

Quantum Key Exchange

- Unconditional (?) security
- Requires sending qubits

Classical Key Exchange

- Security against polynomial-time attacker
- Requires computational assumptions (e.g., DDH, LWE, LPN…)
- Two-message protocol (minimal)

Quantum Key Exchange

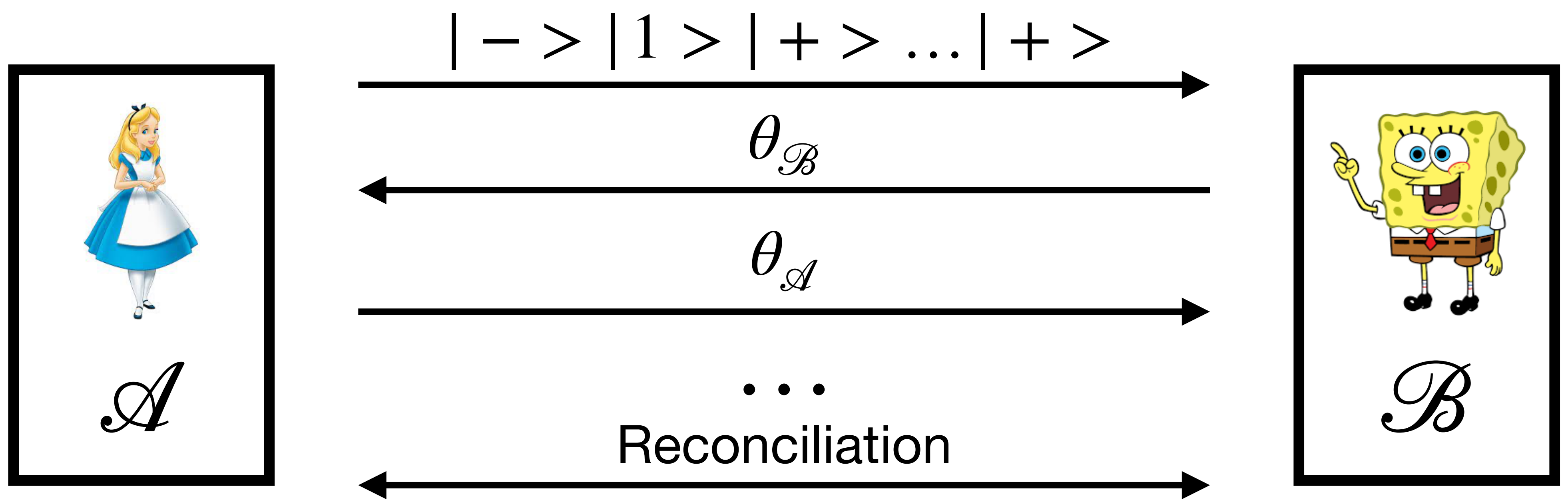
- Unconditional (?) security
- Requires sending qubits
- Multiple rounds of interaction

Classical Key Exchange

- Security against polynomial-time attacker
- Requires computational assumptions (e.g., DDH, LWE, LPN…)
- Two-message protocol (minimal)

Quantum Key Exchange

- Unconditional (?) security
- Requires sending qubits
- Multiple rounds of interaction



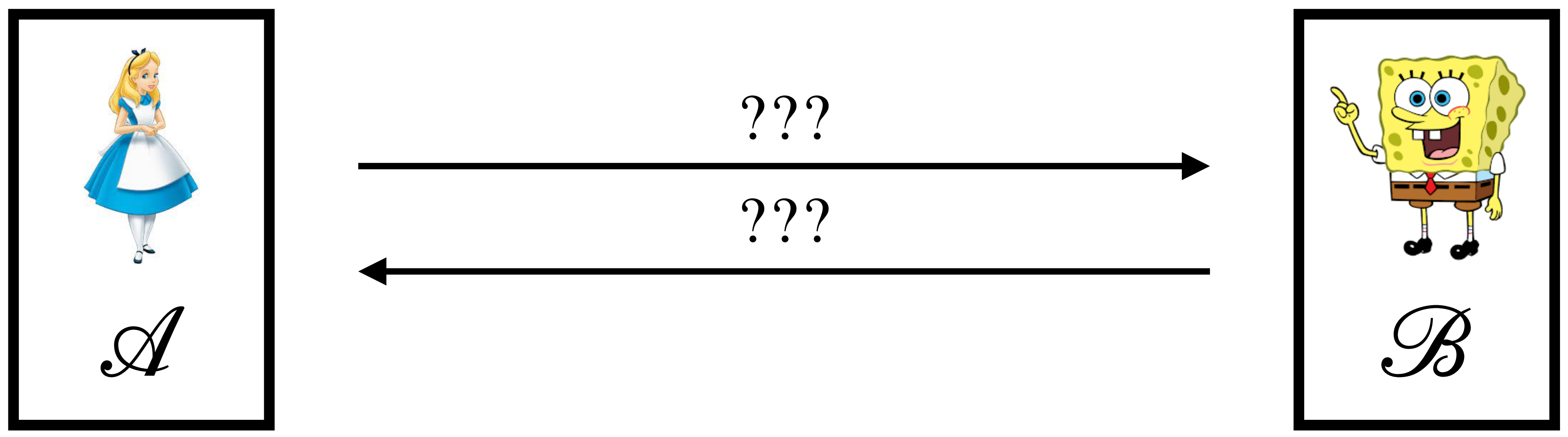
[BB84]

Classical Key Exchange

- Security against polynomial-time attacker
- Requires computational assumptions (e.g., DDH, LWE, LPN…)
- Two-message protocol (minimal)

Quantum Key Exchange

- Unconditional (?) security
- Requires sending qubits
- Multiple rounds of interaction



Theorem I: If quantum-secure *one-way functions* exists, there exists one-time QPKE with *everlasting* security.

Theorem I: If quantum-secure *one-way functions* exists, there exists one-time QPKE with *everlasting* security.

- Implies 2-message key exchange with **everlasting** security

Theorem I: If quantum-secure *one-way functions* exists, there exists one-time QPKE with *everlasting* security.

- Implies 2-message key exchange with **everlasting** security
- Classically, everlasting security is impossible

Theorem I: If quantum-secure *one-way functions* exists, there exists one-time QPKE with *everlasting* security.

- Implies 2-message key exchange with **everlasting** security
- Classically, everlasting security is impossible
- Conceptually different from BB84 (simple analysis!)

Theorem I: If quantum-secure *one-way functions* exists, there exists one-time QPKE with *everlasting* security.

- Implies 2-message key exchange with **everlasting** security
- Classically, everlasting security is impossible
- Conceptually different from BB84 (simple analysis!)

Theorem II: If quantum-secure *one-way functions* exists, there exists standard QPKE with *computational* security.

Theorem I: If quantum-secure *one-way functions* exists, there exists one-time QPKE with *everlasting* security.

- Implies 2-message key exchange with **everlasting** security
- Classically, everlasting security is impossible
- Conceptually different from BB84 (simple analysis!)

Theorem II: If quantum-secure *one-way functions* exists, there exists standard QPKE with *computational* security.

- Classically, one-way functions are (widely believed to be) insufficient to construct PKE

Everlasting Security

Unconditional Security

Everlasting Security

- Computational assumptions, *only during* the protocol

Unconditional Security

Everlasting Security

- Computational assumptions, *only during* the protocol

- Authenticated classical channels

Unconditional Security

Everlasting Security

- Computational assumptions, *only during* the protocol
- ~~Authenticated classical channels~~

Unconditional Security

Everlasting Security

- Computational assumptions, *only during* the protocol
- ~~Authenticated classical channels~~

Unconditional Security

- No computational assumptions!

Everlasting Security

- Computational assumptions, *only during* the protocol
- ~~Authenticated classical channels~~

Unconditional Security

- No computational assumptions!
- Authenticated classical channels

Everlasting Security

- Computational assumptions, *only during* the protocol
- ~~Authenticated classical channels~~

Unconditional Security

- No computational assumptions!
- Authenticated classical channels
  - Computational assumptions…

Everlasting Security

- Computational assumptions, *only during* the protocol
- ~~Authenticated classical channels~~

Unconditional Security

- No computational assumptions!
- Authenticated classical channels
  - Computational assumptions…
  - … but *only during* the protocol!

Everlasting Security

- Computational assumptions, *only during* the protocol
- ~~Authenticated classical channels~~

Unconditional Security

- No computational assumptions!
- Authenticated classical channels
  - Computational assumptions…
  - … but *only during* the protocol!

Roadmap

Roadmap

- Part I: Definitions

Roadmap

- Part I: Definitions

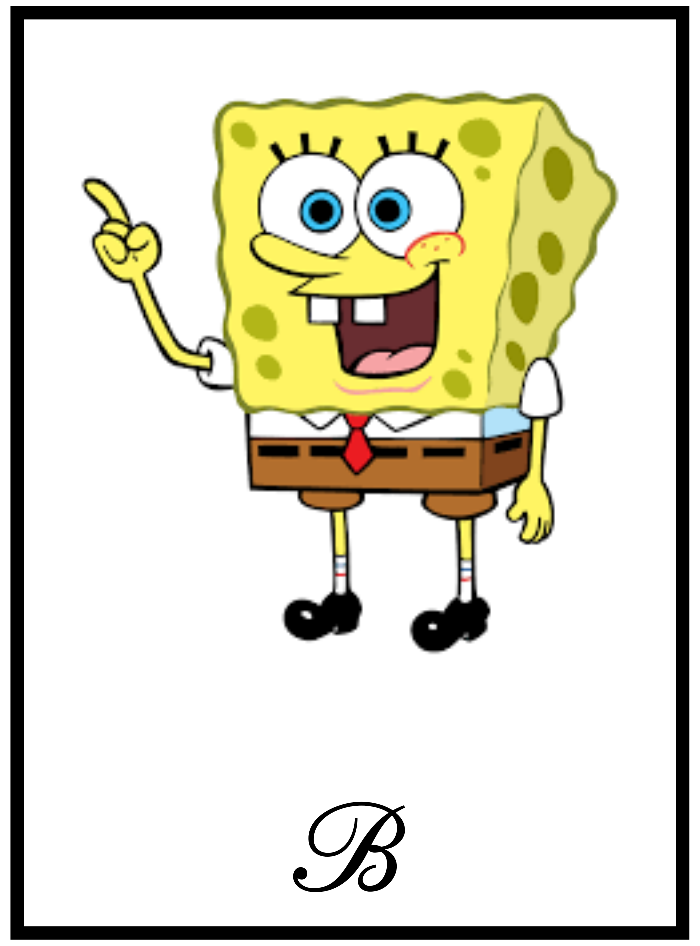
- Part II: The Protocol

Roadmap

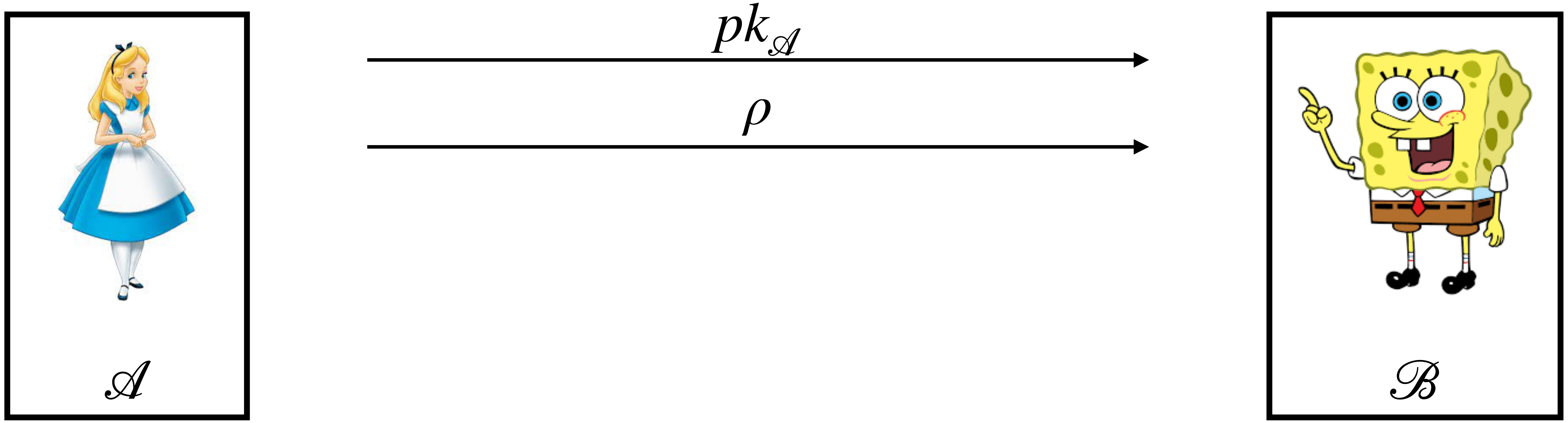
- Part I: Definitions
- Part II: The Protocol
- Part III: Conclusions

Part I: Definitions

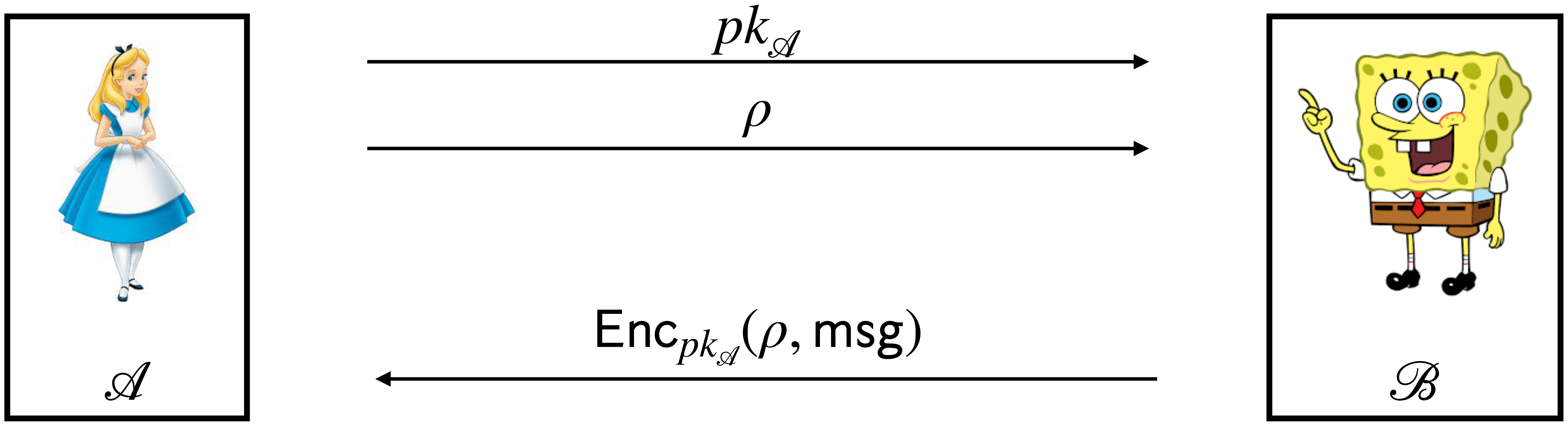
Quantum PKE*



Quantum PKE*

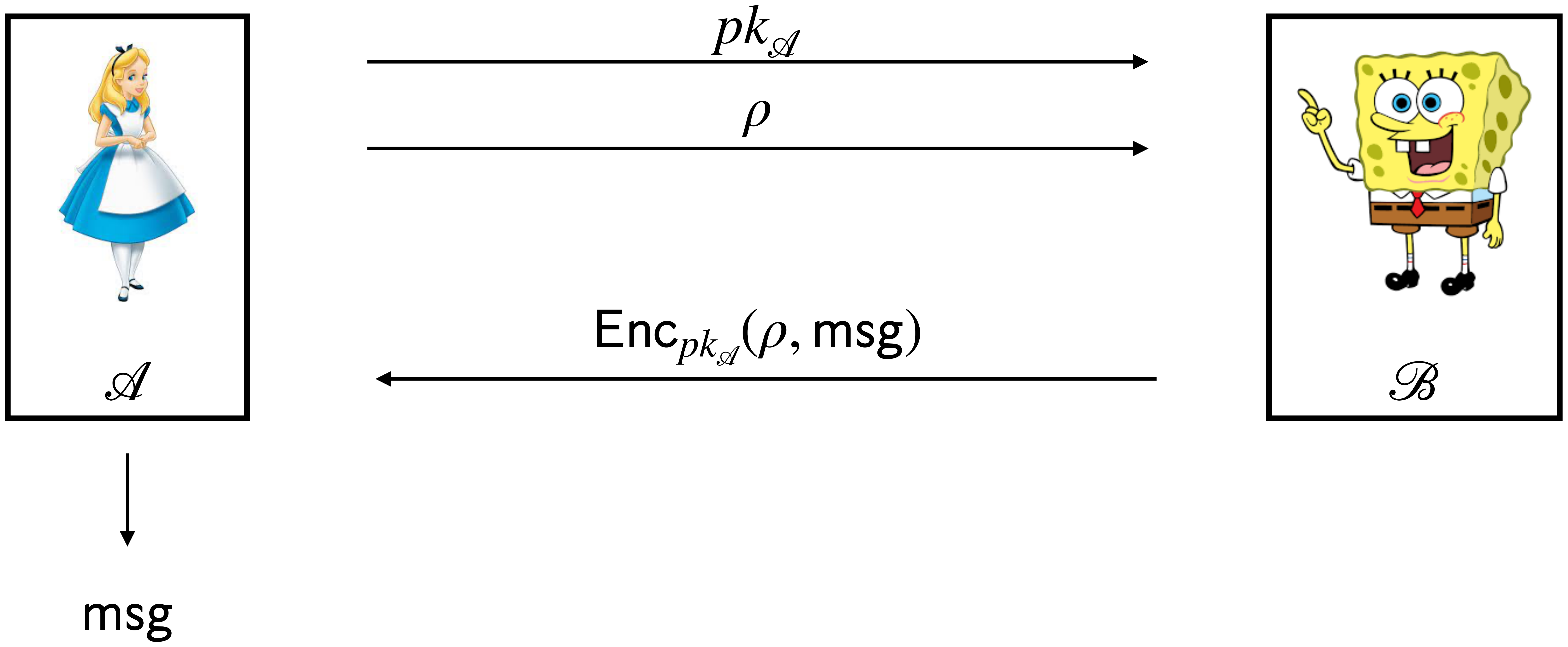


Quantum PKE*



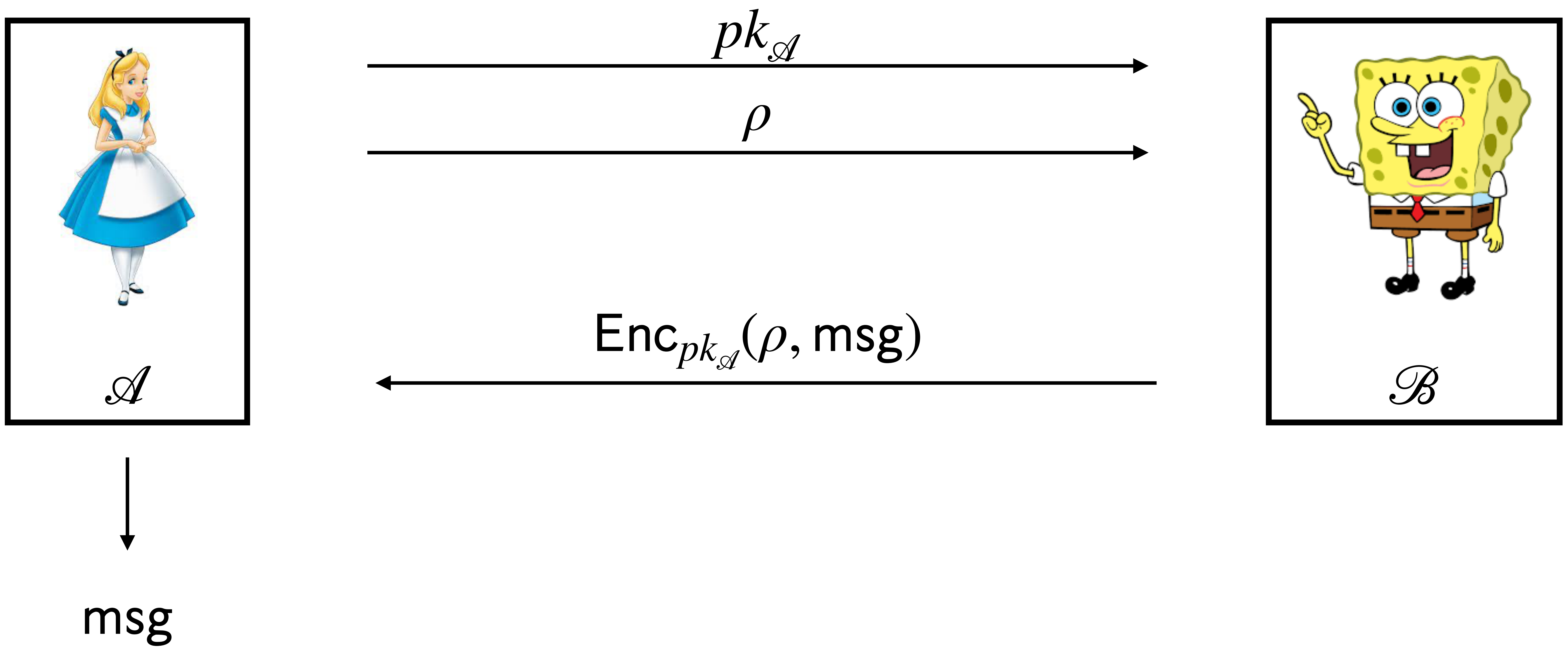
$pk_{\mathscr{A}}$
ρ
$\mathsf{Enc}_{pk_{\mathscr{A}}}(\rho, \mathsf{msg})$
$\mathscr{A}$
$\mathscr{B}$

Quantum PKE*



$pk_{\mathscr{A}}$
ρ
$\mathsf{Enc}_{pk_{\mathscr{A}}}(\rho, \mathsf{msg})$
$\mathscr{A}$
$\mathscr{B}$
msg

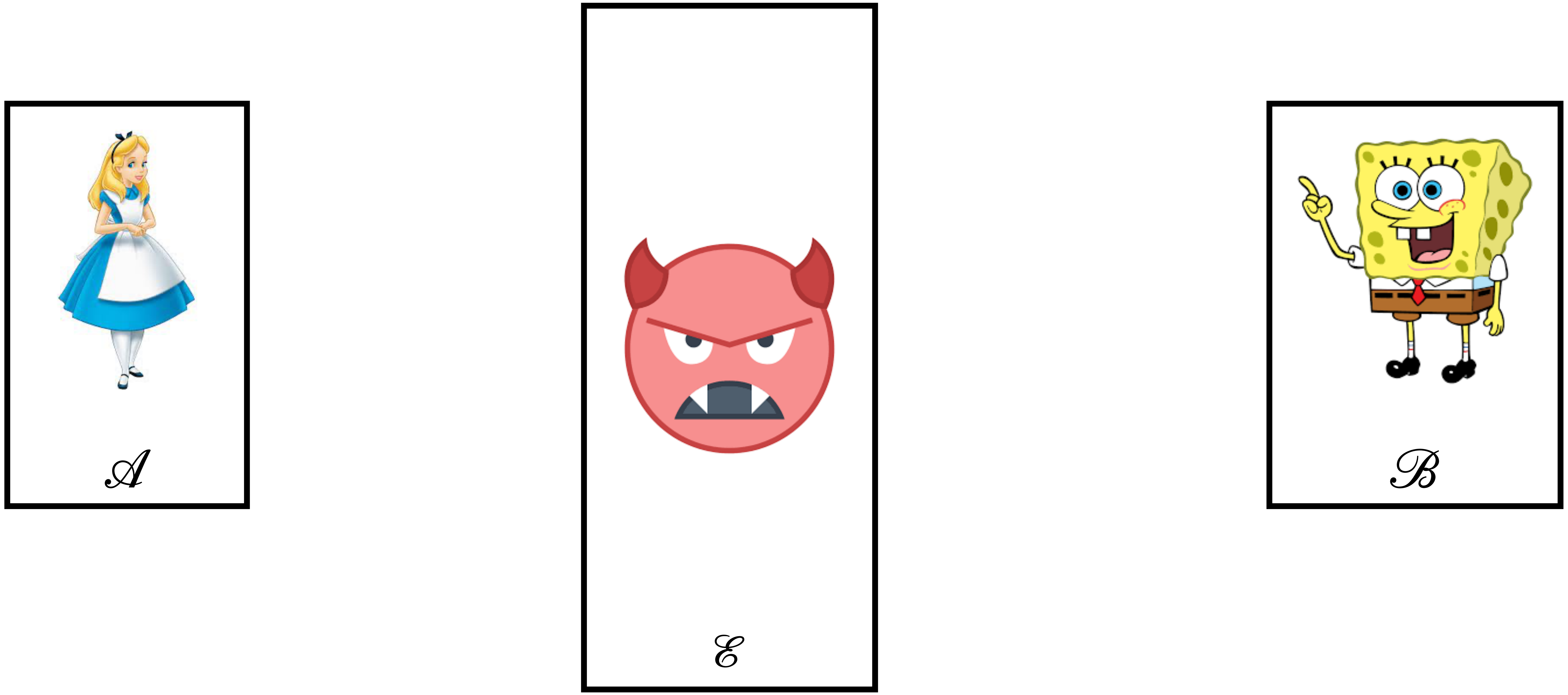
Quantum PKE*



*Suffices for QKD (see paper)

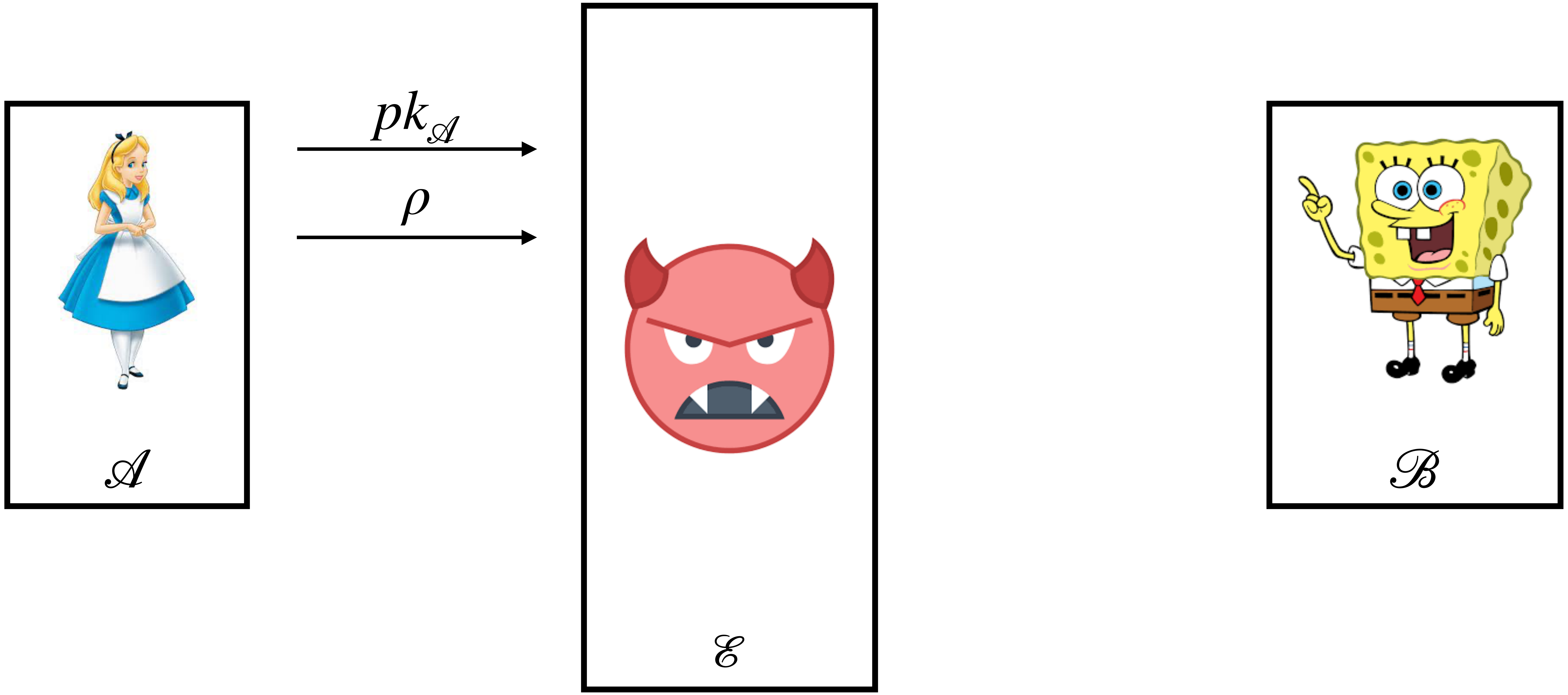
Security Definition

$\forall$ QPT $\mathscr{E}, \forall\ (\mathrm{msg}_0, \mathrm{msg}_1)$:



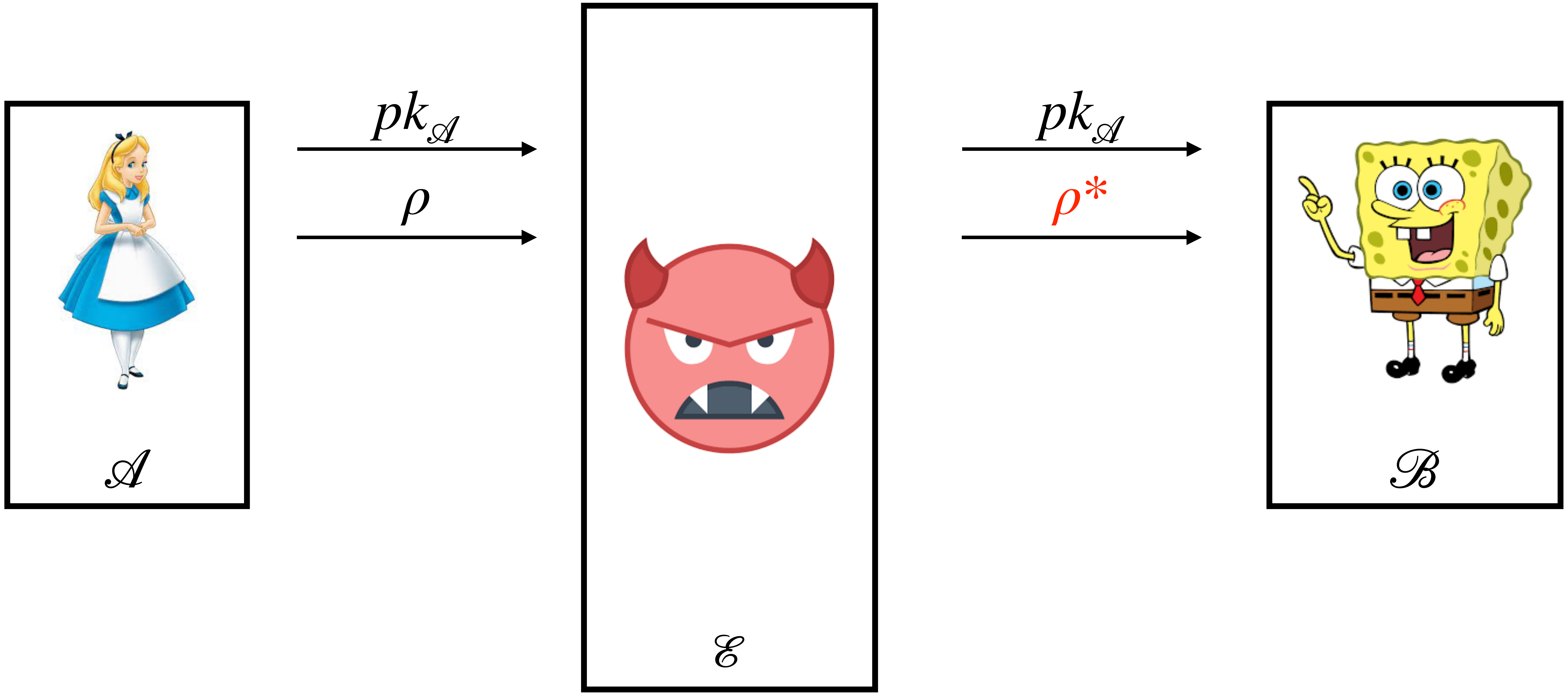
Security Definition

$\forall$ QPT $\mathscr{E}, \forall\ (\text{msg}_0, \text{msg}_1)$:



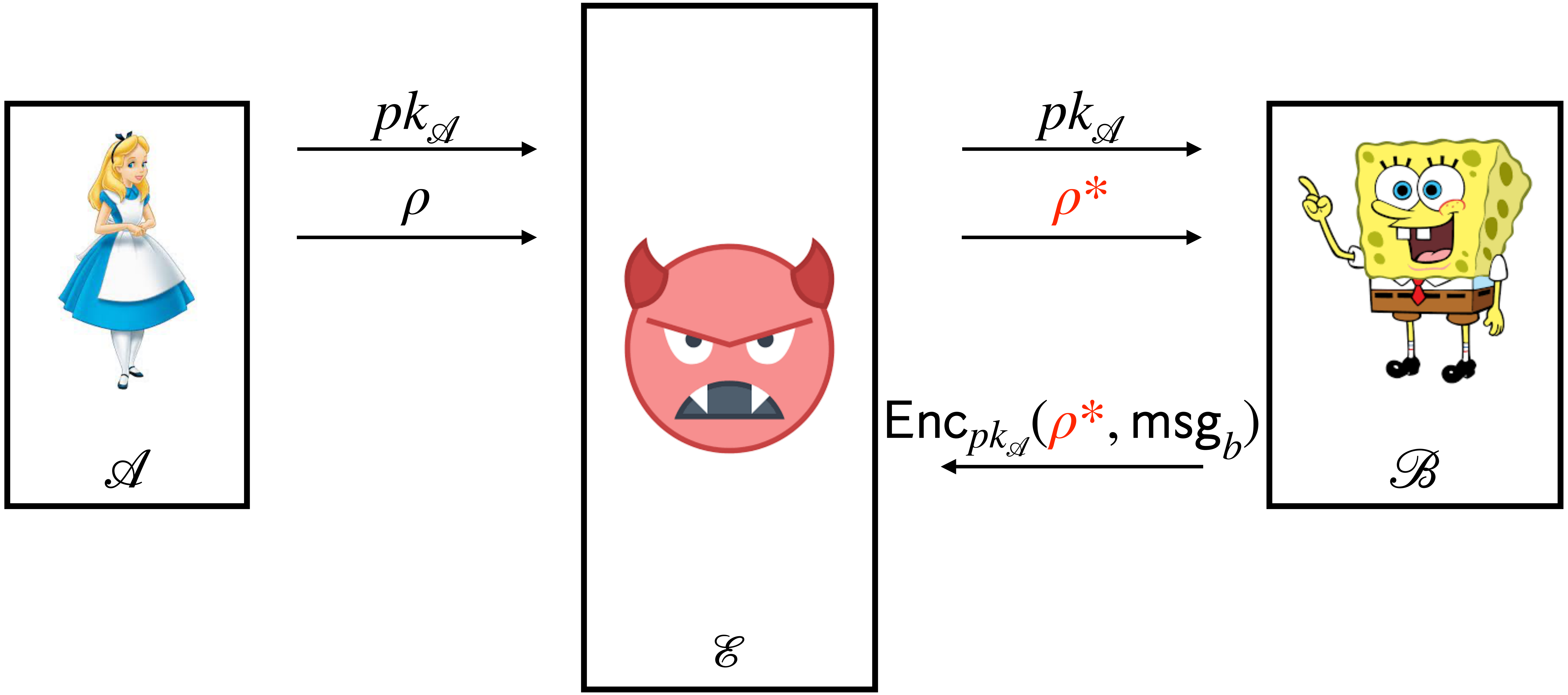
Security Definition

$\forall$ QPT $\mathscr{E}$, $\forall$ $(\mathsf{msg}_0, \mathsf{msg}_1)$:



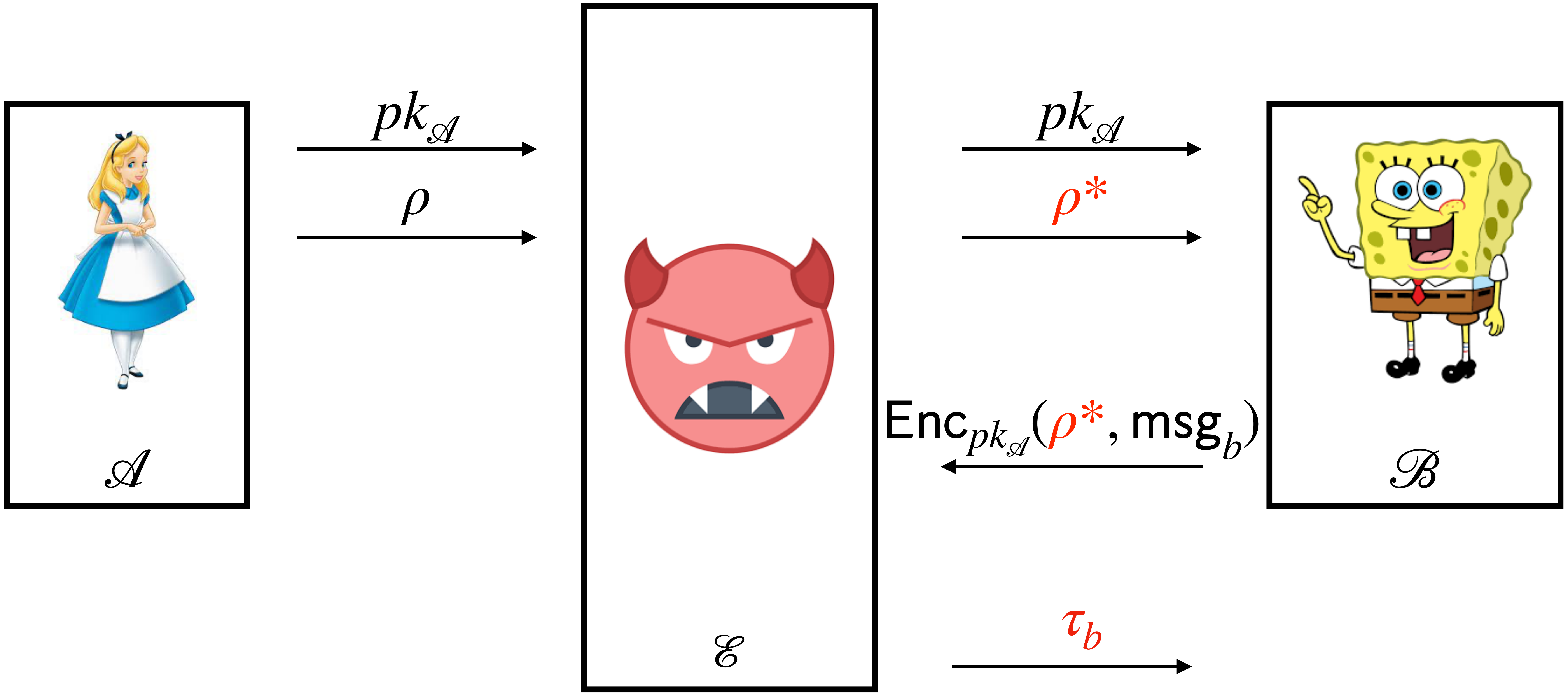
Security Definition

$\forall$ QPT $\mathscr{E}, \forall\ (\mathsf{msg}_0, \mathsf{msg}_1)$:



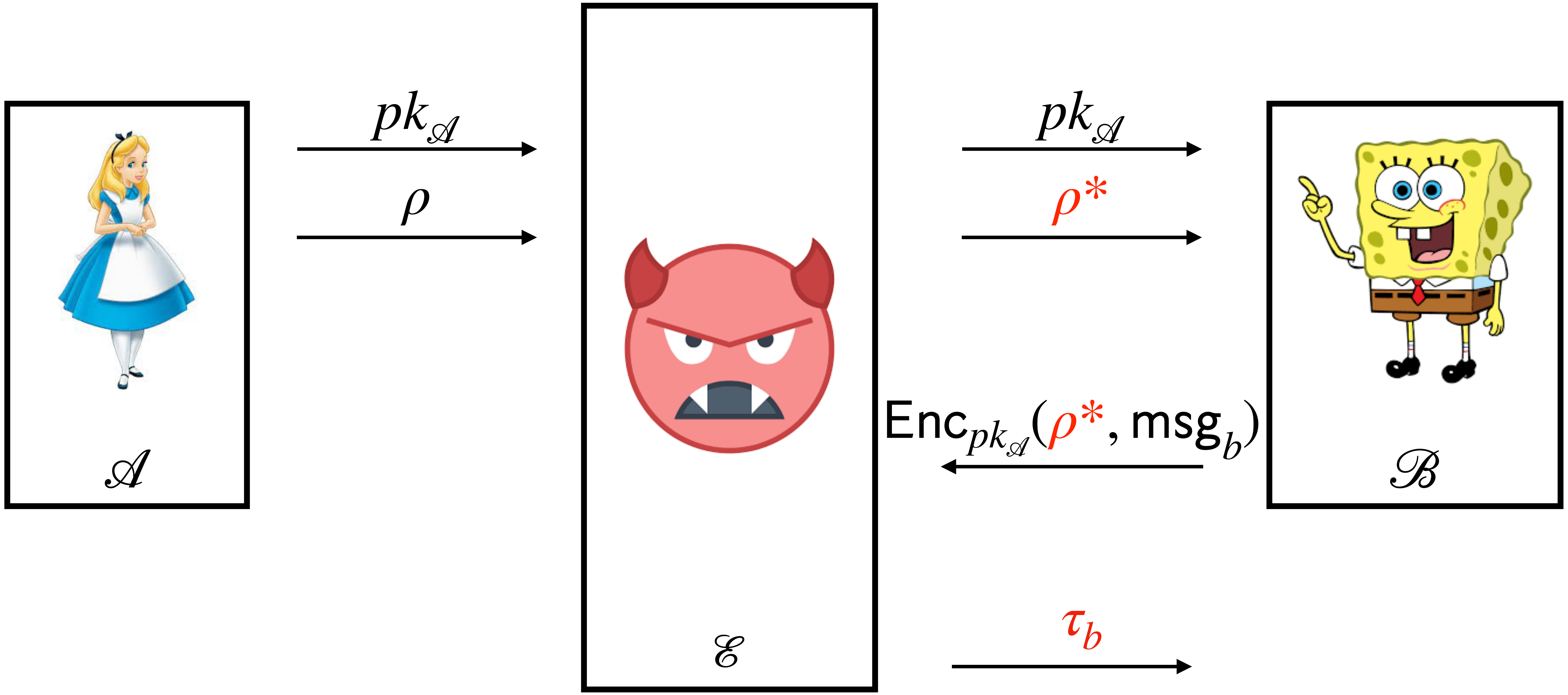
Security Definition

$\forall$ QPT $\mathscr{E}$, $\forall$ $(\mathsf{msg}_0, \mathsf{msg}_1)$:



Security Definition

$\forall$ QPT $\mathscr{E}, \forall$ $(\text{msg}_0, \text{msg}_1)$:



$\text{TD}(\tau_0, \tau_1) \approx 0$

Part II: The Protocol*

*Everlasting variant (see paper for the computational one)

One-Time Digital Signatures

One-Time Digital Signatures

- Consist of three algorithms:

One-Time Digital Signatures

- Consist of three algorithms:
  - $\text{Gen} \to (\text{sk}, \text{vk})$

One-Time Digital Signatures

- Consist of three algorithms:
  - $\mathsf{Gen} \rightarrow (\mathsf{sk}, \mathsf{vk})$
  - $\mathsf{Sign}(\mathsf{sk}, \mathsf{msg}) \rightarrow \sigma$

One-Time Digital Signatures

- Consist of three algorithms:
  - $\mathsf{Gen} \to (\mathsf{sk}, \mathsf{vk})$
  - $\mathsf{Sign}(\mathsf{sk}, \mathsf{msg}) \to \sigma$
  - $\mathsf{Verify}(\mathsf{vk}, \mathsf{msg}, \sigma) \to \{0,1\}$

One-Time Digital Signatures

- Consist of three algorithms:
  - $\text{Gen} \to (\text{sk}, \text{vk})$
  - $\text{Sign}(\text{sk}, \text{msg}) \to \sigma$
  - $\text{Verify}(\text{vk}, \text{msg}, \sigma) \to \{0,1\}$
- Security (existential unforgeability):

One-Time Digital Signatures

- Consist of three algorithms:
    - $\mathsf{Gen} \to (\mathsf{sk}, \mathsf{vk})$
    - $\mathsf{Sign}(\mathsf{sk}, \mathsf{msg}) \to \sigma$
    - $\mathsf{Verify}(\mathsf{vk}, \mathsf{msg}, \sigma) \to \{0,1\}$
- Security (existential unforgeability):
    - Given a *single query* to a signing oracle $\mathsf{Sign}(\mathsf{sk}, \cdot\,)$

One-Time Digital Signatures

- Consist of three algorithms:
  - $\mathsf{Gen} \to (\mathsf{sk}, \mathsf{vk})$
  - $\mathsf{Sign}(\mathsf{sk}, \mathsf{msg}) \to \sigma$
  - $\mathsf{Verify}(\mathsf{vk}, \mathsf{msg}, \sigma) \to \{0,1\}$
- Security (existential unforgeability):
  - Given a *single query* to a signing oracle $\mathsf{Sign}(\mathsf{sk}, \cdot\,)$
  - It is (computationally) hard to forge a *new valid* signature

One-Time Digital Signatures

- Consist of three algorithms:
  - $\mathsf{Gen} \to (\mathsf{sk}, \mathsf{vk})$
  - $\mathsf{Sign}(\mathsf{sk}, \mathsf{msg}) \to \sigma$
  - $\mathsf{Verify}(\mathsf{vk}, \mathsf{msg}, \sigma) \to \{0,1\}$
- Security (existential unforgeability):
  - Given a *single query* to a signing oracle $\mathsf{Sign}(\mathsf{sk}, \cdot\,)$
  - It is (computationally) hard to forge a *new valid* signature
- Exists iff one-way functions exist [Lam79]

Key Generation (Alice)

Key Generation (Alice)

- Sample a key pair $(\mathsf{sk}, \mathsf{vk})$

Key Generation (Alice)

- Sample a key pair $(\mathsf{sk}, \mathsf{vk})$
- Compute the state

$$|\Psi> = \frac{|0,\ 0,\ \sigma_0 > + |1,\ 1,\ \sigma_1 >}{\sqrt{2}}$$

Key Generation (Alice)

- Sample a key pair $(\mathsf{sk}, \mathsf{vk})$
- Compute the state

$$|\Psi> = \frac{|0,\ 0,\ \sigma_0 > + |1,\ 1,\ \sigma_1 >}{\sqrt{2}}$$

- Define a projective measurement $\{\Pi_{\mathsf{vk}}, \mathtt{I} - \Pi_{\mathsf{vk}}\}$ where

$$\Pi_{\mathsf{vk}} = \sum_{\sigma:\mathsf{Verify}(vk,0,\sigma)=1} |0,\ \sigma > < 0,\ \sigma| + \sum_{\sigma:\mathsf{Verify}(vk,1,\sigma)=1} |1,\ \sigma > < 1,\ \sigma|$$

Key Generation (Alice)

- Sample a key pair $(\mathsf{sk}, \mathsf{vk})$
- Compute the state

$$|\Psi> = \frac{|0,\ 0,\ \sigma_0 > + |1,\ 1,\ \sigma_1 >}{\sqrt{2}}$$

- Define a projective measurement $\{\Pi_{\mathsf{vk}}, \mathtt{I} - \Pi_{\mathsf{vk}}\}$ where

$$\Pi_{\mathsf{vk}} = \sum_{\sigma:\mathsf{Verify}(vk,0,\sigma)=1} |0,\ \sigma > < 0,\ \sigma| + \sum_{\sigma:\mathsf{Verify}(vk,1,\sigma)=1} |1,\ \sigma > < 1,\ \sigma|$$

- Measure the first register in the Hadamard basis to obtain $s \in \{0,1\}$

Key Generation (Alice)

- Sample a key pair $(\mathsf{sk}, \mathsf{vk})$
- Compute the state

$$|\Psi> = \frac{|0,\ 0,\ \sigma_0 > + |1,\ 1,\ \sigma_1 >}{\sqrt{2}}$$

- Define a projective measurement $\{\Pi_{\mathsf{vk}}, \mathbb{I} - \Pi_{\mathsf{vk}}\}$ where

$$\Pi_{\mathsf{vk}} = \sum_{\sigma:\mathsf{Verify}(vk,0,\sigma)=1} |0,\ \sigma > < 0,\ \sigma| + \sum_{\sigma:\mathsf{Verify}(vk,1,\sigma)=1} |1,\ \sigma > < 1,\ \sigma|$$

- Measure the first register in the Hadamard basis to obtain $s \in \{0,1\}$
- Let ρ be the residual quantum state; return (ρ, vk)

Encryption (Bob)

Encryption (Bob)

- Project the state ρ onto the image of Π_{vk}

Encryption (Bob)

- Project the state ρ onto the image of Π_{vk}
- Abort if the above projection fails

Encryption (Bob)

- Project the state ρ onto the image of Π_{vk}
- Abort if the above projection fails
  - This guarantees that

$$\rho \in \mathsf{Img}(\Pi_{\mathsf{vk}}) = \mathsf{Span}(\{\,|\,b,\ \sigma_b >: \mathsf{Verify}(\mathsf{vk}, b, \sigma_b) = 1\})$$

Encryption (Bob)

- Project the state ρ onto the image of Π_{vk}
- Abort if the above projection fails
  - This guarantees that

$$\rho \in \mathtt{Img}(\Pi_{\mathsf{vk}}) = \mathsf{Span}(\{ | b, \sigma_b >: \mathsf{Verify}(\mathsf{vk}, b, \sigma_b) = 1\})$$

- Measure the residual state in the Hadamard basis to obtain

$$(d_1, d_2) \in \{0,1\} \times \{0,1\}^n$$

Encryption (Bob)

- Project the state ρ onto the image of Π_{vk}
- Abort if the above projection fails
  - This guarantees that

$$\rho \in \mathtt{Img}(\Pi_{\mathsf{vk}}) = \mathsf{Span}(\{\,|\,b,\ \sigma_b >: \mathsf{Verify}(\mathsf{vk}, b, \sigma_b) = 1\})$$

- Measure the residual state in the Hadamard basis to obtain

$$(d_1, d_2) \in \{0,1\} \times \{0,1\}^n$$

- Return msg $\oplus\, d_1, d_2$

Decryption (Alice)

Decryption (Alice)

- We pretend to delay the measurement of Alice (does not affect correctness)

Decryption (Alice)

- We pretend to delay the measurement of Alice (does not affect correctness)
- The rotated state corresponds to

$$H|\Psi > = \sum_{d} (-1)^{d\cdot(0,0,\sigma_0)}|d> + (-1)^{d\cdot(1,1,\sigma_1)}|d> = \sum_{d: d\cdot(1,1,\sigma_0 \oplus \sigma_1)=0} |d>$$

Decryption (Alice)

- We pretend to delay the measurement of Alice (does not affect correctness)
- The rotated state corresponds to

$$H|\Psi> = \sum_{d} (-1)^{d\cdot(0,0,\sigma_0)}|d> + (-1)^{d\cdot(1,1,\sigma_1)}|d> = \sum_{d:d\cdot(1,1,\sigma_0 \oplus \sigma_1)=0} |d>$$

- Thus measuring the rotated state returns

$$d = (s, d_0, d_1) \qquad \text{s.t.} \qquad d_1 \oplus d_2 \cdot (\sigma_0, \sigma_1) = s$$

Decryption (Alice)

- We pretend to delay the measurement of Alice (does not affect correctness)
- The rotated state corresponds to

$$H|\Psi> = \sum_{d} (-1)^{d\cdot(0,0,\sigma_0)}|d> + (-1)^{d\cdot(1,1,\sigma_1)}|d> = \sum_{d:d\cdot(1,1,\sigma_0\oplus\sigma_1)=0} |d>$$

- Thus measuring the rotated state returns

$$d = (s, d_0, d_1) \qquad \text{s.t.} \qquad d_1 \oplus d_2 \cdot (\sigma_0, \sigma_1) = s$$

- Recall that Bob sends msg $\oplus\, d_1, d_2$

Decryption (Alice)

- We pretend to delay the measurement of Alice (does not affect correctness)
- The rotated state corresponds to

$$H|\Psi> = \sum_{d} (-1)^{d\cdot(0,0,\sigma_0)}|d> + (-1)^{d\cdot(1,1,\sigma_1)}|d> = \sum_{d: d\cdot(1,1,\sigma_0\oplus\sigma_1)=0} |d>$$

- Thus measuring the rotated state returns

$$d = (s, d_0, d_1) \qquad \text{s.t.} \qquad d_1 \oplus d_2 \cdot (\sigma_0, \sigma_1) = s$$

- Recall that Bob sends msg $\oplus\, d_1, d_2$
- Alice can recover d_1, and consequently msg, since she knows s and (σ_0, σ_1)

Proof Sketch

Proof Sketch

- From the point of view of the attacker, the residual state ρ is a classical mixture

$|0, \sigma_0>$ with prob. $1/2$ $\qquad$ $|1, \sigma_1>$ with prob. $1/2$

Proof Sketch

- From the point of view of the attacker, the residual state ρ is a classical mixture

$|0, \sigma_0>$ with prob. $1/2$ $|1, \sigma_1>$ with prob. $1/2$

- Since Bob projects the state onto $\mathsf{Span}(\{|b, \sigma_b>: \mathsf{Verify}(\mathsf{vk}, b, \sigma_b) = 1\})$, the attacker must have either:

Proof Sketch

- From the point of view of the attacker, the residual state ρ is a classical mixture

$|0, \sigma_0 >$ with prob. $1/2$ $\qquad$ $|1, \sigma_1 >$ with prob. $1/2$

- Since Bob projects the state onto $\mathsf{Span}(\{ |b, \sigma_b >: \mathsf{Verify}(\mathsf{vk}, b, \sigma_b) = 1\})$, the attacker must have either:
  - Passed along the state

Proof Sketch

- From the point of view of the attacker, the residual state ρ is a classical mixture

$|0, \sigma_0>$ with prob. $1/2$ $\qquad$ $|1, \sigma_1>$ with prob. $1/2$

- Since Bob projects the state onto $\mathsf{Span}(\{|b, \sigma_b>: \mathsf{Verify}(\mathsf{vk}, b, \sigma_b) = 1\})$, the attacker must have either:
    - Passed along the state
    - Put a non-trivial amplitude on another signature (breaks unforgeability!)

Proof Sketch

- From the point of view of the attacker, the residual state ρ is a classical mixture

$|0, \sigma_0>$ with prob. $1/2$ $\qquad$ $|1, \sigma_1>$ with prob. $1/2$

- Since Bob projects the state onto $\mathsf{Span}(\{|b, \sigma_b>: \mathsf{Verify}(\mathsf{vk}, b, \sigma_b) = 1\})$, the attacker must have either:
    - Passed along the state
    - ~~Put a non-trivial amplitude on another signature (breaks unforgeability!)~~

Proof Sketch

- From the point of view of the attacker, the residual state ρ is a classical mixture

$|0, \sigma_0>$ with prob. $1/2$ $\qquad\qquad$ $|1, \sigma_1>$ with prob. $1/2$

- Since Bob projects the state onto $\mathsf{Span}(\{|b, \sigma_b>: \mathsf{Verify}(\mathsf{vk}, b, \sigma_b) = 1\})$, the attacker must have either:
    - Passed along the state
    - ~~Put a non-trivial amplitude on another signature (breaks unforgeability!)~~
- Measuring the basis state in the Hadamard basis, gives

$$d \sim \text{Uniform} : \{0,1\}^{n+1}$$

Proof Sketch

- From the point of view of the attacker, the residual state ρ is a classical mixture

$|0, \sigma_0>$ with prob. $1/2$ $\qquad$ $|1, \sigma_1>$ with prob. $1/2$

- Since Bob projects the state onto $\mathsf{Span}(\{|b, \sigma_b>: \mathsf{Verify}(\mathsf{vk}, b, \sigma_b) = 1\})$, the attacker must have either:
  - Passed along the state
  - ~~Put a non-trivial amplitude on another signature (breaks unforgeability!)~~
- Measuring the basis state in the Hadamard basis, gives

$$d \sim \mathsf{Uniform} : \{0,1\}^{n+1}$$

□

Part III: Conclusions

Concurrent & Follow-up

Open Problems

Concurrent & Follow-up

- Concurrent work by [KMNY23]

Open Problems

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction

Open Problems

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!

Open Problems

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!
- Follow-up works:

Open Problems

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!
- Follow-up works:
  - Cryptography with certified deletion from minimal assumptions

Open Problems

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!
- Follow-up works:
  - Cryptography with certified deletion from minimal assumptions
  - Revocable digital signatures

Open Problems

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!
- Follow-up works:
  - Cryptography with certified deletion from minimal assumptions
  - Revocable digital signatures

Open Problems

- Key-rate?

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!
- Follow-up works:
  - Cryptography with certified deletion from minimal assumptions
  - Revocable digital signatures

Open Problems

- Key-rate?
- Noise tolerance?

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!

- Follow-up works:
  - Cryptography with certified deletion from minimal assumptions
  - Revocable digital signatures

Open Problems

- Key-rate?
- Noise tolerance?
- Qubit-by-qubit?

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!
- Follow-up works:
  - Cryptography with certified deletion from minimal assumptions
  - Revocable digital signatures

Open Problems

- Key-rate?
- Noise tolerance?
- Qubit-by-qubit?
- Assumptions?

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!
- Follow-up works:
  - Cryptography with certified deletion from minimal assumptions
  - Revocable digital signatures

Open Problems

- Key-rate?
- Noise tolerance?
- Qubit-by-qubit?
- Assumptions?
  - OWFs are not minimal for quantum crypto

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!
- Follow-up works:
  - Cryptography with certified deletion from minimal assumptions
  - Revocable digital signatures

Open Problems

- Key-rate?
- Noise tolerance?
- Qubit-by-qubit?
- Assumptions?
  - OWFs are not minimal for quantum crypto
- **Experiments?**

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!
- Follow-up works:
  - Cryptography with certified deletion from minimal assumptions
  - Revocable digital signatures

Open Problems

- Key-rate?
- Noise tolerance?
- Qubit-by-qubit?
- Assumptions?
  - OWFs are not minimal for quantum crypto
- **Experiments?**
  - Reach out if interested!

giulio.malavolta@hotmail.it

Concurrent & Follow-up

- Concurrent work by [KMNY23]
  - Computationally secure construction
  - CCA-secure!
- Follow-up works:
  - Cryptography with certified deletion from minimal assumptions
  - Revocable digital signatures

Open Problems

- Key-rate?
- Noise tolerance?
- Qubit-by-qubit?
- Assumptions?
  - OWFs are not minimal for quantum crypto
- **Experiments?**
  - Reach out if interested!

giulio.malavolta@hotmail.it

THANK YOU!