How to Sign Quantum Messages

Mohammed Barhoush and Louis Salvail

University of Montreal

June 20, 2025

What are digital signatures?

A digital signature (DS) scheme on classical messages consists of the following algorithms:

- KeyGen(1^λ): Generates a secret key sk and a verification key vk.
- Sign(sk, μ) : Outputs a signature σ for μ using sk.
- Verify(vk, μ', σ'): Verifies whether σ' is a valid signature for μ' using vk and correspondingly outputs $\top$/ $\bot$.

[1] Rompel, J., 1990, April. One-way functions are necessary and sufficient for secure signatures.

What are digital signatures?

A digital signature (DS) scheme on classical messages consists of the following algorithms:

- KeyGen(1^λ): Generates a secret key sk and a verification key vk.
- Sign(sk, μ) : Outputs a signature σ for μ using sk.
- Verify(vk, μ', σ'): Verifies whether σ' is a valid signature for μ' using vk and correspondingly outputs $\top$/ $\bot$.

Digital signatures have many crucial applications such as in email certification, online transactions, and software distribution.

[1] Rompel, J., 1990, April. One-way functions are necessary and sufficient for secure signatures.

What are digital signatures?

A digital signature (DS) scheme on classical messages consists of the following algorithms:

- KeyGen(1^λ): Generates a secret key sk and a verification key vk.
- Sign(sk, μ) : Outputs a signature σ for μ using sk.
- Verify(vk, μ', σ'): Verifies whether σ' is a valid signature for μ' using vk and correspondingly outputs $\top$/ $\bot$.

Digital signatures have many crucial applications such as in email certification, online transactions, and software distribution. Fortunately, DS for classical messages can be constructed from OWFs [1].

[1] Rompel, J., 1990, April. One-way functions are necessary and sufficient for secure signatures.

Can We Sign Quantum Messages?

[2]Barnum, H., Crépeau, C., Gottesman, D., Smith, A. and Tapp, A., 2002, November. Authentication of quantum messages.

[3]Alagic, G., Gagliardoni, T. and Majenz, C., 2021. Can you sign a quantum state?.

Can We Sign Quantum Messages?

- Unlike in the classical setting, authentication of quantum message necessitates encryption [2].

[2] Barnum, H., Crépeau, C., Gottesman, D., Smith, A. and Tapp, A., 2002, November. Authentication of quantum messages.

[3] Alagic, G., Gagliardoni, T. and Majenz, C., 2021. Can you sign a quantum state?.

Can We Sign Quantum Messages?

- Unlike in the classical setting, authentication of quantum message necessitates encryption [2].
- Any verification algorithm should obtain the message.
- This implies public-verifiability is impossible!

[2] Barnum, H., Crépeau, C., Gottesman, D., Smith, A. and Tapp, A., 2002, November. Authentication of quantum messages.

[3] Alagic, G., Gagliardoni, T. and Majenz, C., 2021. Can you sign a quantum state?.

Can We Sign Quantum Messages?

- Unlike in the classical setting, authentication of quantum message necessitates encryption [2].
- Any verification algorithm should obtain the message.
- This implies public-verifiability is impossible!
- In other words, signing quantum messages is impossible [3] (even under computational assumptions).

[2] Barnum, H., Crépeau, C., Gottesman, D., Smith, A. and Tapp, A., 2002, November. Authentication of quantum messages.

[3] Alagic, G., Gagliardoni, T. and Majenz, C., 2021. Can you sign a quantum state?.

Informal Idea

$$|\sigma\rangle \xrightarrow{\mathsf{Verify}_{\mathsf{vk}}} |\mu\rangle \sim |\mu'\rangle \xrightarrow{\mathsf{Verify}^{\dagger}_{\mathsf{vk}}} |\sigma'\rangle.$$

A Partial Solution: Signcryption

- Every user generates a pair of public-key encryption keys.
- If Alice wants to sign a message to Bob, she uses Bob's public encryption key to first encrypt the message.
- No public-verifiability as only Bob can validate the signature.

A Partial Solution: Signcryption

- Every user generates a pair of public-key encryption keys.
- If Alice wants to sign a message to Bob, she uses Bob's public encryption key to first encrypt the message.
- No public-verifiability as only Bob can validate the signature.

Is quantum authentication with public-verifiablity achievable?

A Partial Solution: Signcryption

- Every user generates a pair of public-key encryption keys.
- If Alice wants to sign a message to Bob, she uses Bob's public encryption key to first encrypt the message.
- No public-verifiability as only Bob can validate the signature.

Is quantum authentication with public-verifiablity achievable?

Yes, it is!

Our Solution

Our solutions is to add a *time-dependence* to the signature scheme.

Our Solution

Our solutions is to add a *time-dependence* to the signature scheme. We sample a key pair $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{C.KeyGen}(1^\lambda)$ for a classical DS.

Our Solution

Our solutions is to add a *time-dependence* to the signature scheme.
We sample a key pair $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{C.KeyGen}(1^\lambda)$ for a classical DS.
Signing a quantum message is as follows $\mathsf{Q.Sign}(\mathsf{sk}, |\mu\rangle)$:

Our Solution

Our solutions is to add a *time-dependence* to the signature scheme.
We sample a key pair $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{C.KeyGen}(1^\lambda)$ for a classical DS.
Signing a quantum message is as follows $\mathsf{Q.Sign}(\mathsf{sk}, |\mu\rangle)$:

1. Sample a key k for a one-time *symmetric* authenticated encryption scheme on quantum messages $(\mathsf{Q.Enc}, \mathsf{Q.Dec})$.

Our Solution

Our solutions is to add a *time-dependence* to the signature scheme. We sample a key pair $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{C.KeyGen}(1^\lambda)$ for a classical DS. Signing a quantum message is as follows $\mathsf{Q.Sign}(\mathsf{sk}, |\mu\rangle)$:

1. Sample a key k for a one-time *symmetric* authenticated encryption scheme on quantum messages $(\mathsf{Q.Enc}, \mathsf{Q.Dec})$.
2. Classically sign the current time and key $\sigma \leftarrow \mathsf{C.Sign}(\mathsf{sk}, (\mathsf{t}, k))$.

Our Solution

Our solutions is to add a *time-dependence* to the signature scheme. We sample a key pair $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{C.KeyGen}(1^\lambda)$ for a classical DS. Signing a quantum message is as follows $\mathsf{Q.Sign}(\mathsf{sk}, |\mu\rangle)$:

1. Sample a key k for a one-time *symmetric* authenticated encryption scheme on quantum messages $(\mathsf{Q.Enc}, \mathsf{Q.Dec})$.
2. Classically sign the current time and key $\sigma \leftarrow \mathsf{C.Sign}(\mathsf{sk}, (\mathtt{t}, k))$.
3. Output $\rho := \mathsf{Q.Enc}_k(|\mu\rangle)$ and a time-lock puzzle $Z := \mathsf{TLP}(1, (k, \mathtt{t}, \sigma))$.

Verify$(\mathsf{vk}, (\rho, Z))$:

1. Take note of the current time $\mathtt{t}'$.
2. Compute $(k, \mathtt{t}, \sigma) \leftarrow \mathsf{Solve}(Z)$.
3. Check that $\mathtt{t} + 0.5 \geq \mathtt{t}'$ and $\mathsf{CS.Verify}(\mathsf{vk}, (\mathtt{t}, k), \sigma) = \top$.
4. Output $\mathsf{Q.Dec}_k(\rho')$.

Why the Impossibility Does Not Apply

$$|\sigma\rangle \xrightarrow{\mathsf{Verify}_{\mathsf{vk}}} |\mu\rangle \sim |\mu'\rangle \xrightarrow{\mathsf{Verify}^{\dagger}_{\mathsf{vk}}} |\sigma'\rangle.$$

Why the Impossibility Does Not Apply

$$|\sigma\rangle \xrightarrow{\mathsf{Verify}_{\mathsf{vk}}} |\mu\rangle \sim |\mu'\rangle \xrightarrow{\mathsf{Verify}^{\dagger}_{\mathsf{vk}}} |\sigma'\rangle.$$

Disadvantage

The problems with our scheme:

1. The signature expires i.e. you cannot reuse it after a while.
2. Time-lock puzzles are a heavy computational assumption that have only been constructed in the QROM.

Solution: Dynamic Verification Keys

By utilizing verification keys that evolve over time, we eliminate the need for TLPs in our construction. The idea is quite simple:

Solution: Dynamic Verification Keys

By utilizing verification keys that evolve over time, we eliminate the need for TLPs in our construction. The idea is quite simple:

- For each time interval $[t_{i-1}, t_i)$, we use a different key k_i.

Solution: Dynamic Verification Keys

By utilizing verification keys that evolve over time, we eliminate the need for TLPs in our construction. The idea is quite simple:

- For each time interval $[t_{i-1}, t_i)$, we use a different key k_i.
- To sign $|\mu\rangle$ at time $\textcircled{\scriptsize 🕒} \in [t_{i-1}, t_i)$, simply output $\mathsf{Q.Enc}_{k_i}(|\mu\rangle)$.

Solution: Dynamic Verification Keys

By utilizing verification keys that evolve over time, we eliminate the need for TLPs in our construction. The idea is quite simple:

- For each time interval $[t_{i-1}, t_i)$, we use a different key k_i.
- To sign $|\mu\rangle$ at time $\textcircled{\scriptsize 🕒} \in [t_{i-1}, t_i)$, simply output $\mathsf{Q.Enc}_{k_i}(|\mu\rangle)$.
- At time $\textcircled{\scriptsize 🕒} : t_i$, we announce k_i.

Solution: Dynamic Verification Keys

By utilizing verification keys that evolve over time, we eliminate the need for TLPs in our construction. The idea is quite simple:

- For each time interval $[t_{i-1}, t_i)$, we use a different key k_i.
- To sign $|\mu\rangle$ at time $\in [t_{i-1}, t_i)$, simply output $\mathsf{Q.Enc}_{k_i}(|\mu\rangle)$.
- At time $: t_i$, we announce k_i.
- To verify a message received at time $\in [t_{i-1}, t_i)$, we store the signature state and wait for announcement of k_i. (During this time k_i is still hidden).

Solution: Dynamic Verification Keys

By utilizing verification keys that evolve over time, we eliminate the need for TLPs in our construction. The idea is quite simple:

- For each time interval $[t_{i-1}, t_i)$, we use a different key k_i.
- To sign $|\mu\rangle$ at time $\in [t_{i-1}, t_i)$, simply output $\mathsf{Q.Enc}_{k_i}(|\mu\rangle)$.
- At time $: t_i$, we announce k_i.
- To verify a message received at time $\in [t_{i-1}, t_i)$, we store the signature state and wait for announcement of k_i. (During this time k_i is still hidden).
- This leads to signatures from OWFs with dynamic verification keys.

Applications

We leverage time-dependent signatures with dynamic keys to achieve the following objectives, relying solely on OWFs:

- *Authenticated Quantum Public Keys:* We design a public-key encryption scheme featuring authenticated quantum public-keys that resist adversarial tampering.

Applications

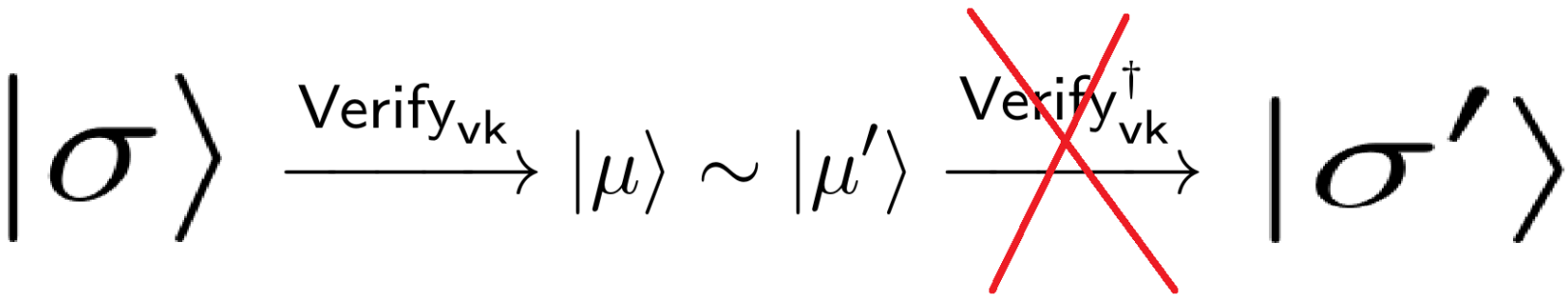
We leverage time-dependent signatures with dynamic keys to achieve the following objectives, relying solely on OWFs:

- *Authenticated Quantum Public Keys:* We design a public-key encryption scheme featuring authenticated quantum public-keys that resist adversarial tampering.
- *Public-Key Quantum Money:* We construct a *time-dependent* public-key quantum money scheme.

Alternative Solution: Bounded Quantum Storage Model

- In this model, an adversary $\mathcal{A}$ is limited with respect to its quantum memory.
- $\mathcal{A}$ is never restricted with respect to its computational power or classical memory.
- **Our result**: We build information-theoretically secure signatures for quantum messages in this model (no time dependence or computational assumptions required).

Why the Impossibility Does Not Apply



Thanks for listening!