Non-interactive Blind Signatures: Post-quantum and Stronger Security

Foteini Baldimtsi, Jiaqi Cheng, Rishab Goyal, **Aayush Yadav**

ASIACRYPT 2024

Blind signatures

[Chaum83]

(vk_S, μ)

$(\mathsf{sk}_S, \mathsf{vk}_S)$

Blind signatures

[Chaum83]

(vk_S, μ)

$\beta \leftarrow \mathsf{Blind}(\mu)$

$(\mathsf{sk}_S, \mathsf{vk}_S)$

(vk_S, μ)

$(\mathsf{sk}_S, \mathsf{vk}_S)$

$\beta \leftarrow \mathsf{Blind}(\mu) \xrightarrow{\quad \beta \quad}$

Blind signatures

[Chaum83]

(vk_S, μ)

$(\mathsf{sk}_S, \mathsf{vk}_S)$

$\beta \leftarrow \mathsf{Blind}(\mu)$ $\xrightarrow{\beta}$

$\overline{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\beta)$

Blind signatures

[Chaum83]

(vk_S, μ)

$(\mathsf{sk}_S, \mathsf{vk}_S)$

$\beta \leftarrow \mathsf{Blind}(\mu) \quad \xrightarrow{\quad\beta\quad}$

$\underbrace{\overline{\sigma}}_{\text{pre-signature}} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\beta)$

Blind signatures

[Chaum83]

(vk_S, μ)

$(\mathsf{sk}_S, \mathsf{vk}_S)$

$\beta \leftarrow \mathsf{Blind}(\mu)$ $\xrightarrow{\quad\beta\quad}$

$\xleftarrow{\quad\overline{\sigma}\quad}$ $\underbrace{\overline{\sigma}}_{\text{pre-signature}} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\beta)$

Blind signatures

[Chaum83]

(vk_S, μ)

$(\mathsf{sk}_S, \mathsf{vk}_S)$

$\beta \leftarrow \mathsf{Blind}(\mu)$ $\xrightarrow{\quad\beta\quad}$

$\xleftarrow{\quad\bar{\sigma}\quad}$ $\underbrace{\bar{\sigma}}_{\text{pre-signature}} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\beta)$

Blind signatures

[Chaum83]

(vk_S, μ)

$(\mathsf{sk}_S, \mathsf{vk}_S)$

$\beta \leftarrow \mathsf{Blind}(\mu)$ $\xrightarrow{\quad\beta\quad}$

$\xleftarrow{\quad\overline{\sigma}\quad}$ $\underbrace{\overline{\sigma}}_{\text{pre-signature}} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\beta)$

$\sigma \leftarrow \mathsf{Unblind}(\mathsf{vk}_S, \beta, \overline{\sigma})$

Blind signatures

[Chaum83]

(vk_S, μ) $\qquad\qquad\qquad\qquad$ $(\mathsf{sk}_S, \mathsf{vk}_S)$

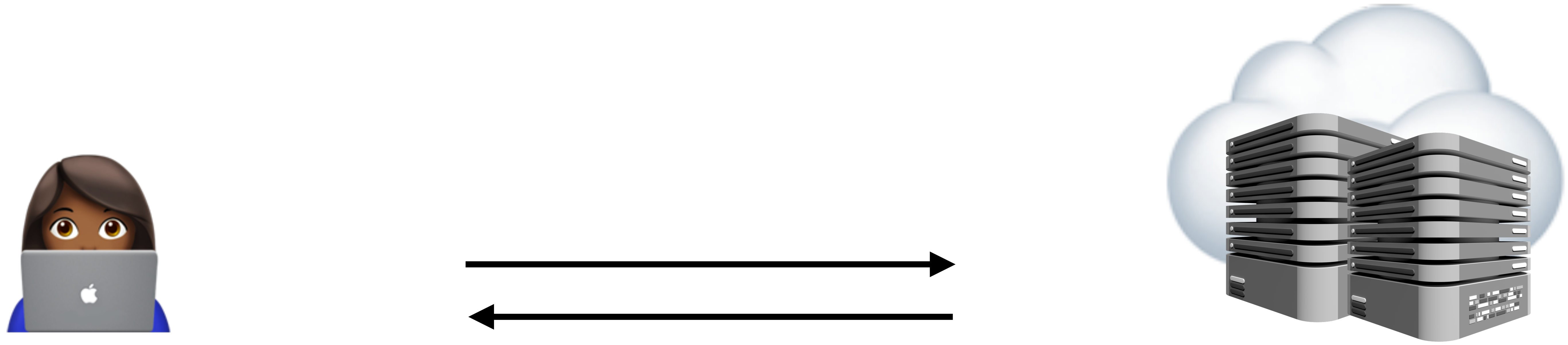
$\beta \leftarrow \mathsf{Blind}(\mu)$ $\quad \xrightarrow{\beta} \quad$

$\quad \xleftarrow{\overline{\sigma}} \quad$ $\underbrace{\overline{\sigma}}_{\text{pre-signature}} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\beta)$

$\sigma \leftarrow \mathsf{Unblind}(\mathsf{vk}_S, \beta, \overline{\sigma})$ $\quad \xrightarrow{\mu, \sigma} \quad$ $\mathsf{Verify}_{\mathsf{vk}_S}(\mu, \sigma)$

(vk_S, μ)

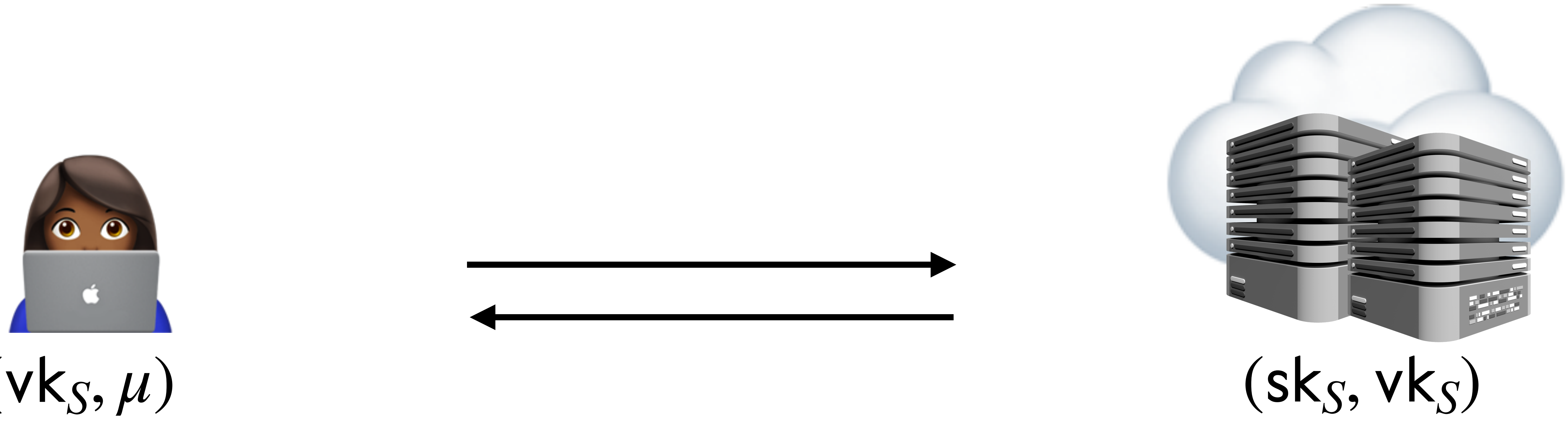
$(\mathsf{sk}_S, \mathsf{vk}_S)$



(vk_S, μ) $(\mathsf{sk}_S, \mathsf{vk}_S)$

(One-more) Unforgeability

User can only obtain valid signatures on chosen messages by interacting with the signer.



(One-more) Unforgeability

User can only obtain valid signatures on chosen messages by interacting with the signer.

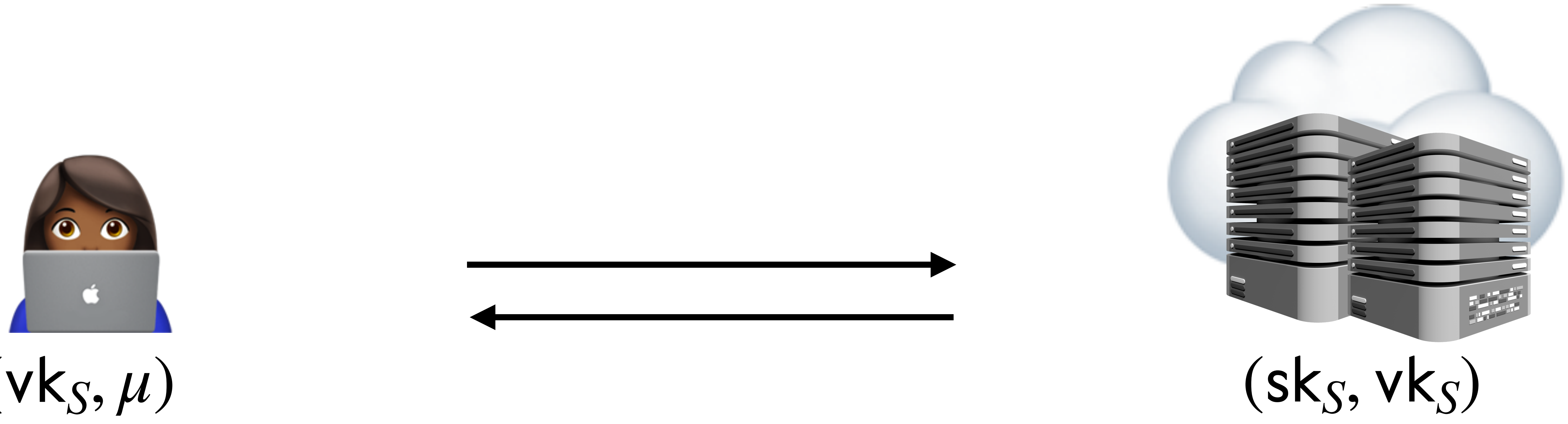
Blindness/Unlinkability

Signer cannot link a message and signature pair to any specific signing session.

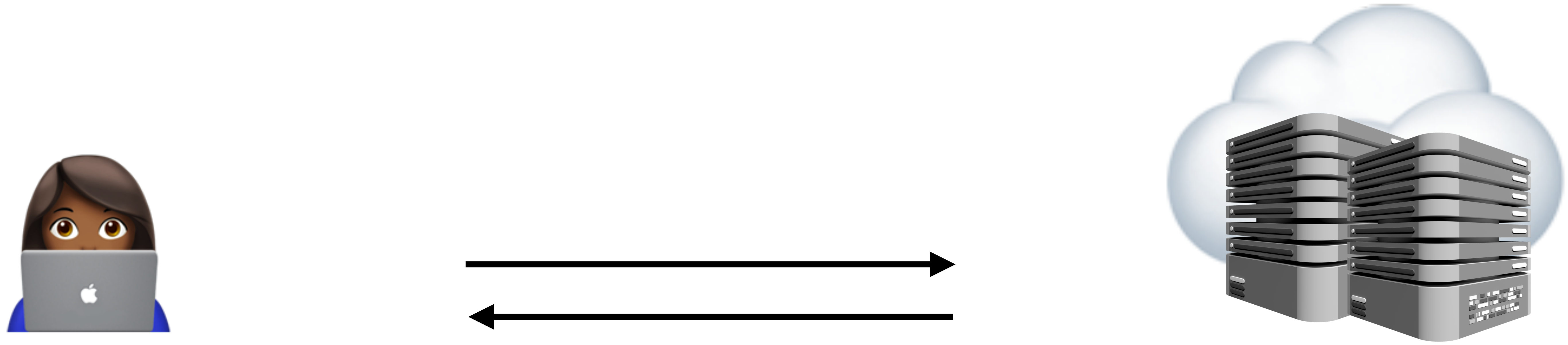
Applications

- Electronic cash [Chaum83]
- Electronic voting [Canard-Gaud-Traoré06]
- Cryptographic tumblers [Heilman-Alshenibr-Baldimtsi-Goldberg17]
- Anonymous credential schemes [Baldimtsi-Lysyanskaya13, Fuchsbauer-Hanser-Slamanig19]
- Authentication tokens/Anonymous web-browsing [Davidson-Goldberg-Sullivan-Tankersley-Valsorda18]

A major limitation of interactive schemes



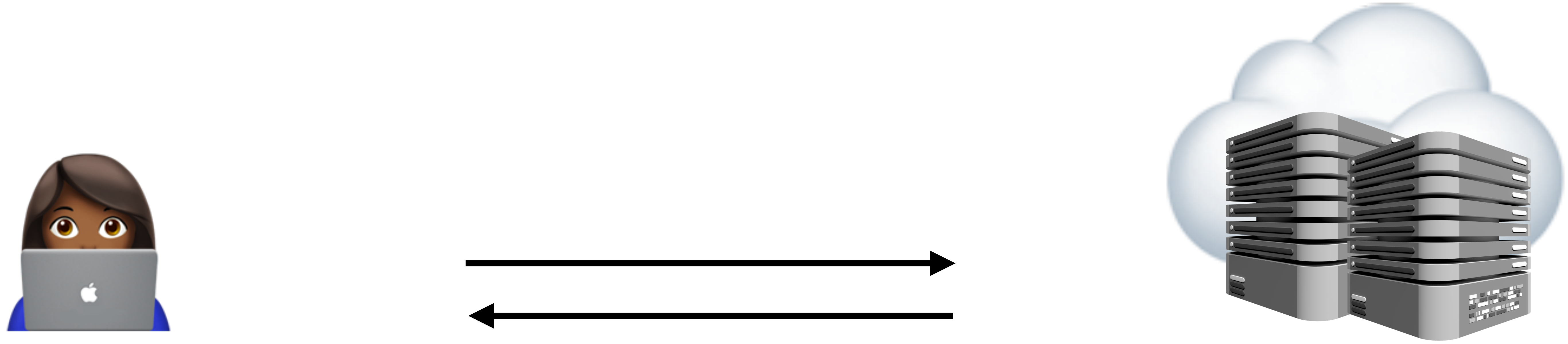
A major limitation of interactive schemes



(vk_S, μ) $(\mathsf{sk}_S, \mathsf{vk}_S)$

The interactive signature **issuance** protocol requires both parties to be *online*.

A major limitation of interactive schemes



(vk_S, μ) $(\mathsf{sk}_S, \mathsf{vk}_S)$

The interactive signature **issuance** protocol requires both parties to be *online*.

Can this protocol be made non-interactive?

Fact

Any blind signature scheme on user specified messages requires an interactive signature generation algorithm.

Fact

Any blind signature scheme on **user specified messages** requires an interactive signature generation algorithm.

Fact

Any blind signature scheme on **user specified messages** requires an interactive signature generation algorithm.

Observation [Hanzlik23]

The blindly signed message is **randomly chosen** by the user in many modern applications.

Fact

Any blind signature scheme on **user specified messages** requires an interactive signature generation algorithm.

Observation [Hanzlik23]

The blindly signed message is **randomly chosen** by the user in many modern applications.

Proposition [Hanzlik23]

If the user does not require a specific distribution or structure to the message, then a non-interactive signature generation algorithm exists.

Non-interactive blind signatures

on random messages (NIBS)

$(\mathsf{vk}_S, \mathsf{sk}_U, \mathsf{pk}_U)$

$(\mathsf{sk}_S, \mathsf{vk}_S, \mathsf{pk}_U)$

Non-interactive blind signatures

on random messages (NIBS)

$(\mathsf{vk}_S, \mathsf{sk}_U, \mathsf{pk}_U)$

$(\mathsf{sk}_S, \mathsf{vk}_S, \mathsf{pk}_U)$

$r \leftarrow \{0,1\}^\lambda$

Non-interactive blind signatures

on random messages (NIBS)

$(\mathsf{vk}_S, \mathsf{sk}_U, \mathsf{pk}_U)$

$(\mathsf{sk}_S, \mathsf{vk}_S, \mathsf{pk}_U)$

$$r \leftarrow \{0,1\}^\lambda$$

$$\underbrace{\bar{\sigma}}_{\text{pre-signature}} \leftarrow \mathsf{Issue}_{\mathsf{sk}_S}(\mathsf{pk}_U \,||\, r)$$

Non-interactive blind signatures

on random messages (NIBS)

$(\mathsf{vk}_S, \mathsf{sk}_U, \mathsf{pk}_U)$

$(\mathsf{sk}_S, \mathsf{vk}_S, \mathsf{pk}_U)$

$$r \leftarrow \{0,1\}^\lambda$$

$$\xleftarrow{\bar{\sigma},\, r} \qquad \underbrace{\bar{\sigma}}_{\text{pre-signature}} \leftarrow \mathsf{Issue}_{\mathsf{sk}_S}(\mathsf{pk}_U \,||\, r)$$

Non-interactive blind signatures

on random messages (NIBS)

$(\mathsf{vk}_S, \mathsf{sk}_U, \mathsf{pk}_U)$

$(\mathsf{sk}_S, \mathsf{vk}_S, \mathsf{pk}_U)$

$r \leftarrow \{0,1\}^{\lambda}$

$\underbrace{\bar{\sigma}}_{\text{pre-signature}} \leftarrow \mathsf{Issue}_{\mathsf{sk}_S}(\mathsf{pk}_U \,||\, r)$

$\xleftarrow{\quad \bar{\sigma},\, r \quad}$

Non-interactive blind signatures

on random messages (NIBS)

$(\mathsf{vk}_S, \mathsf{sk}_U, \mathsf{pk}_U)$

$(\mathsf{sk}_S, \mathsf{vk}_S, \mathsf{pk}_U)$

$r \leftarrow \{0,1\}^\lambda$

$\underbrace{\bar{\sigma}}_{\text{pre-signature}} \leftarrow \mathsf{Issue}_{\mathsf{sk}_S}(\mathsf{pk}_U \,||\, r)$

$\xleftarrow{\bar{\sigma},\, r}$

$(\mu, \sigma) \leftarrow \mathsf{Obtain}_{\mathsf{sk}_U}(\mathsf{vk}_S, \bar{\sigma}, r)$

Non-interactive blind signatures

on random messages (NIBS)

$(\mathsf{vk}_S, \mathsf{sk}_U, \mathsf{pk}_U)$

$(\mathsf{sk}_S, \mathsf{vk}_S, \mathsf{pk}_U)$

$r \leftarrow \{0,1\}^\lambda$

$\xleftarrow{\overline{\sigma},\, r}$ $\quad \underbrace{\overline{\sigma}}_{\text{pre-signature}} \leftarrow \mathsf{Issue}_{\mathsf{sk}_S}(\mathsf{pk}_U \,||\, r)$

$(\mu, \sigma) \leftarrow \mathsf{Obtain}_{\mathsf{sk}_U}(\mathsf{vk}_S, \overline{\sigma}, r)$ $\quad \xrightarrow{\mu,\, \sigma}$

Non-interactive blind signatures

on random messages (NIBS)

$(\mathsf{vk}_S, \mathsf{sk}_U, \mathsf{pk}_U)$

$(\mathsf{sk}_S, \mathsf{vk}_S, \mathsf{pk}_U)$

$r \leftarrow \{0,1\}^\lambda$

$\xleftarrow{\quad \bar{\sigma}, r \quad}$ $\quad \underbrace{\bar{\sigma}}_{\text{pre-signature}} \leftarrow \mathsf{Issue}_{\mathsf{sk}_S}(\mathsf{pk}_U \,||\, r)$

$(\mu, \sigma) \leftarrow \mathsf{Obtain}_{\mathsf{sk}_U}(\mathsf{vk}_S, \bar{\sigma}, r) \quad \xrightarrow{\quad \mu, \sigma \quad} \quad \mathsf{Verify}_{\mathsf{vk}_S}(\mu, \sigma)$

NIBS

Security properties

$(\mathsf{vk}_S, \mathsf{sk}_U, \mathsf{pk}_U)$

$(\mathsf{sk}_S, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{vk}_S, \color{red}{\mathsf{sk}_U}, \color{blue}{\mathsf{pk}_U})$

$(\mathsf{sk}_S, \mathsf{vk}_S, \color{blue}{\mathsf{pk}_U})$

(One-more) Unforgeability

User can only obtain valid signatures on (random) messages from the signer.

$(\mathsf{vk}_S, \mathsf{sk}_U, \mathsf{pk}_U)$

$(\mathsf{sk}_S, \mathsf{vk}_S, \mathsf{pk}_U)$

(One-more) Unforgeability

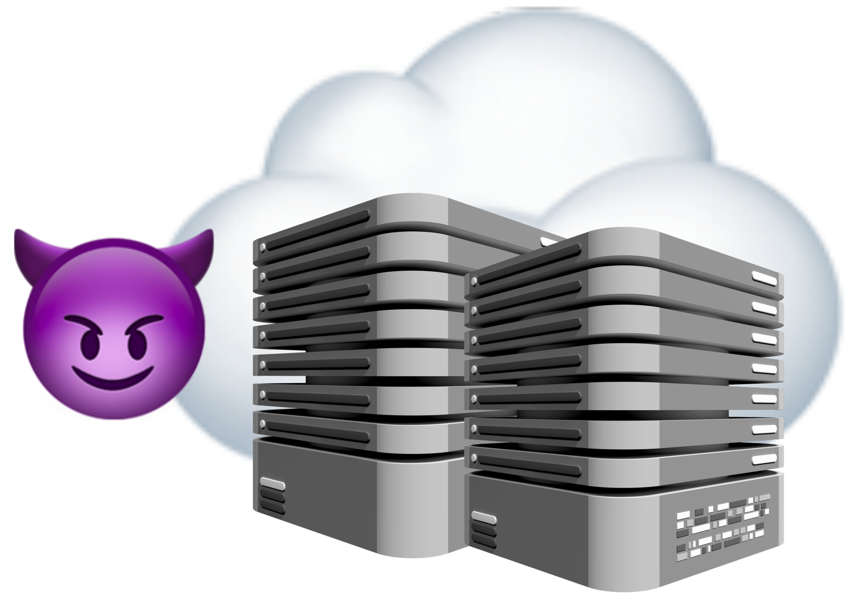
User can only obtain valid signatures on (random) messages from the signer.

Blindness

Signer cannot link a message and signature pair to any specific issuance.

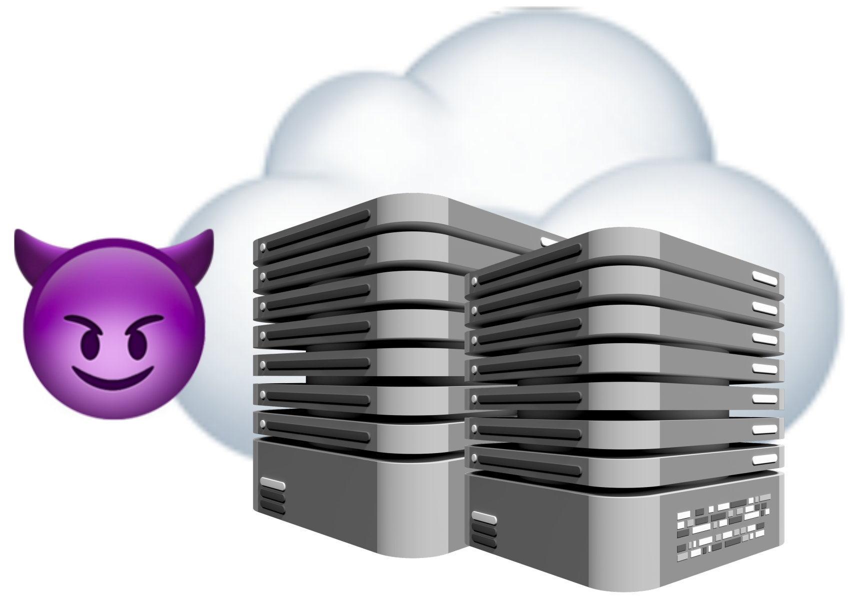
NIBS blindness experiment

[Han23]



NIBS blindness experiment

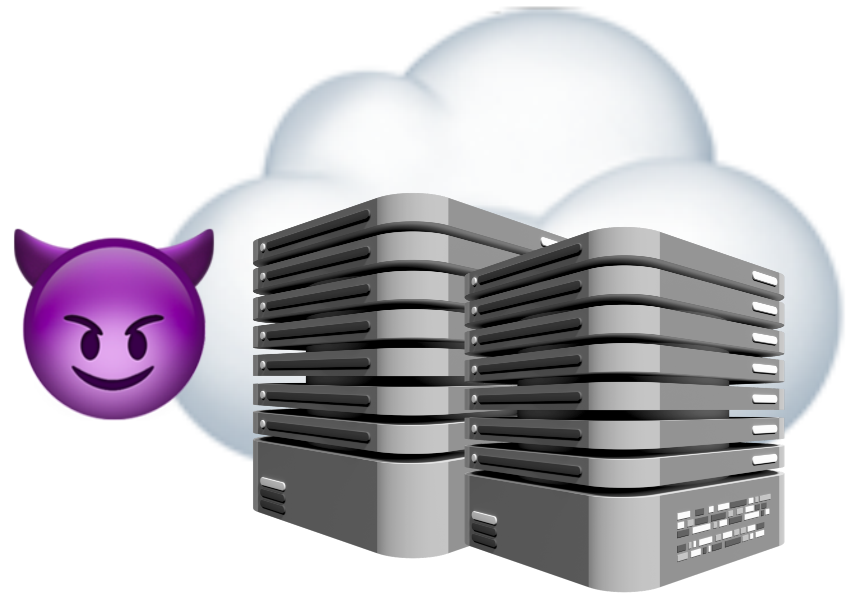
[Han23]



$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$ $\longrightarrow$

NIBS blindness experiment

[Han23]

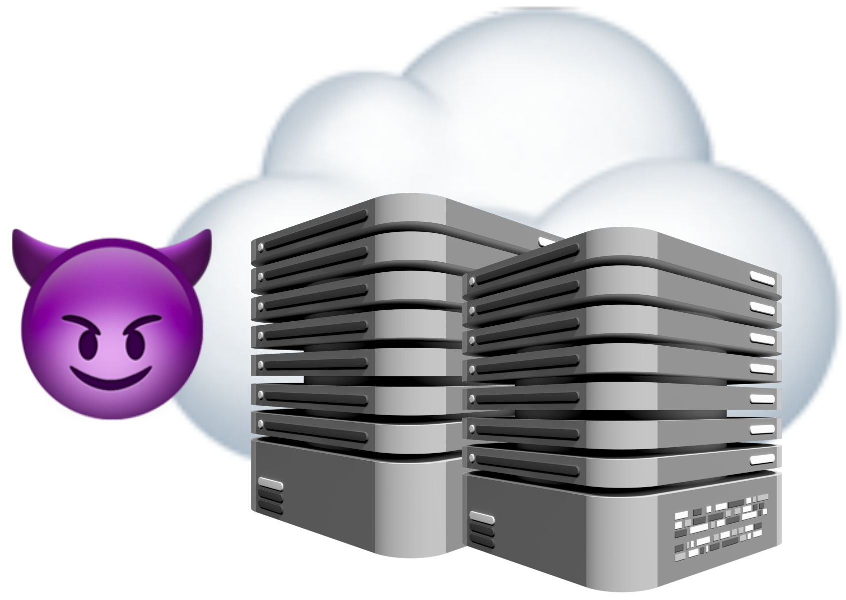


$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$ $\longrightarrow$

$\longleftarrow$ $\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$

NIBS blindness experiment

[Han23]



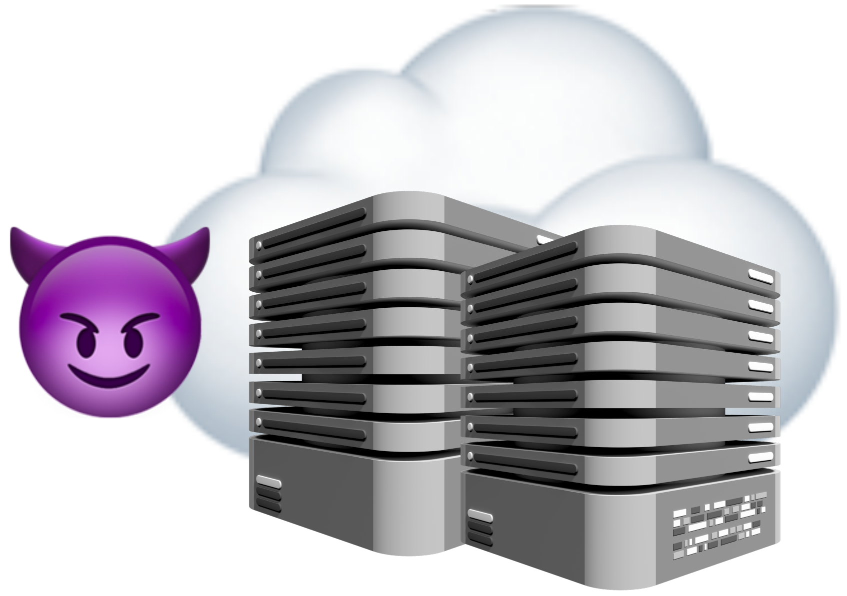
$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$ $\longrightarrow$

$\longleftarrow$ $\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

NIBS blindness experiment

[Han23]



$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$

$\longrightarrow$

$\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$

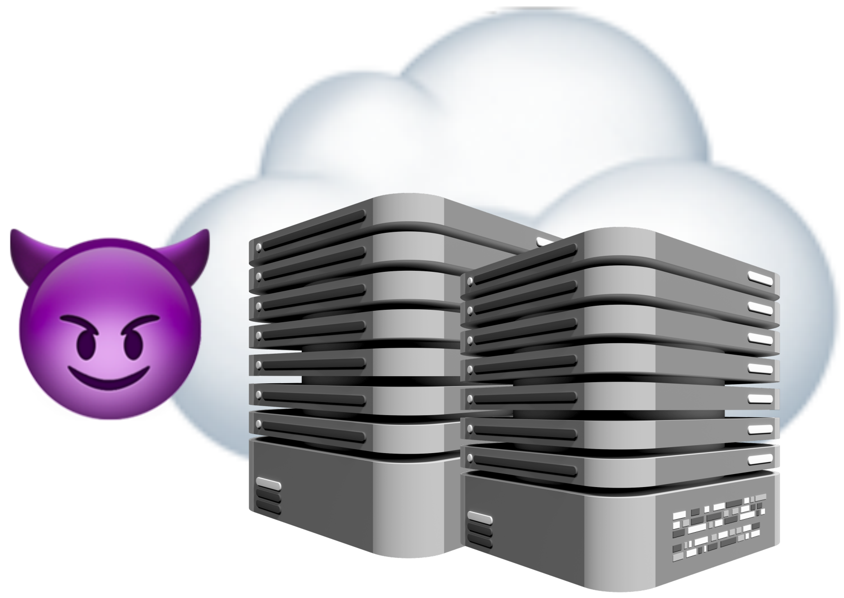
$\longleftarrow$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

NIBS blindness experiment

[Han23]



$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$

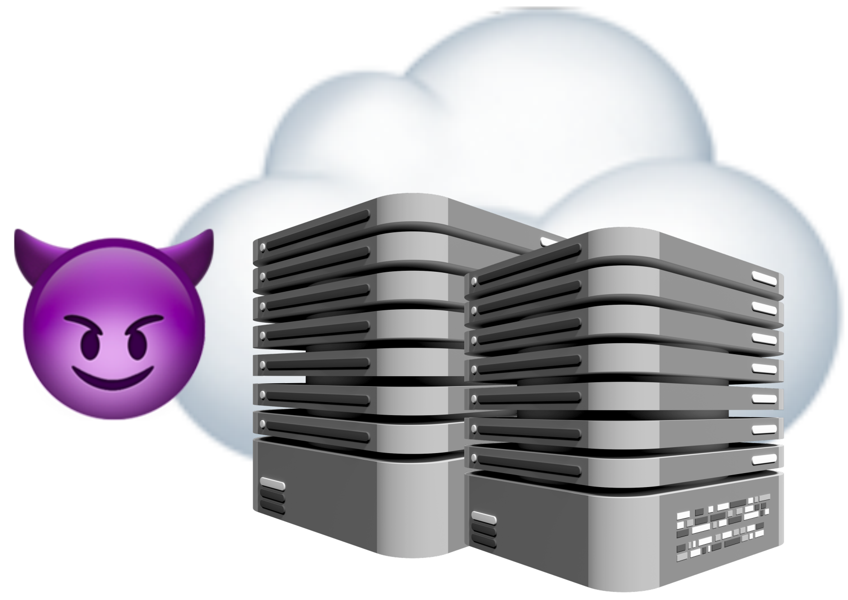
$\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$

$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

NIBS blindness experiment

[Han23]



$\xrightarrow{\mathsf{pk}_U^{(0)},\ \mathsf{pk}_U^{(1)}}$

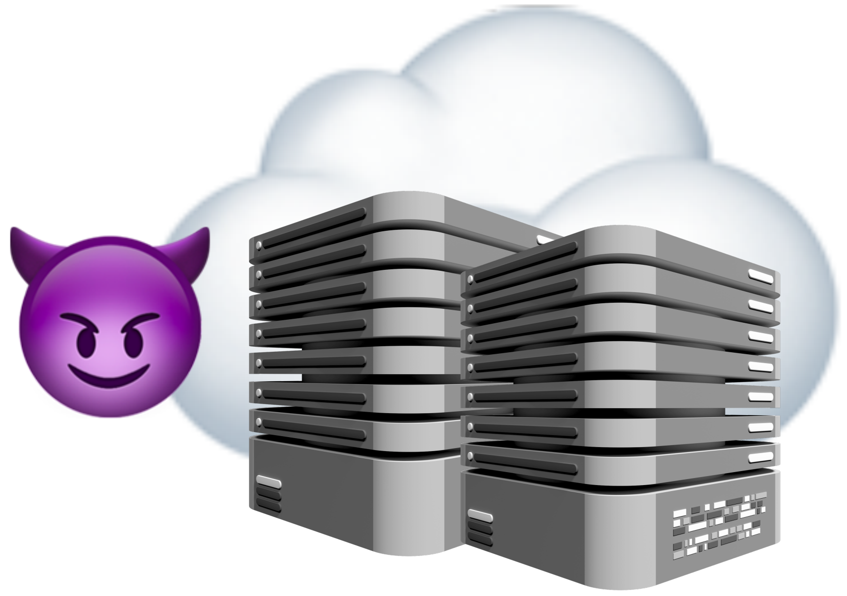
$\xleftarrow{\mathsf{vk}_S,\ \overline{\sigma}_0,\ \overline{\sigma}_1}$

$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

NIBS blindness experiment

[Han23]



$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$ $\longrightarrow$

$\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$ $\longleftarrow$

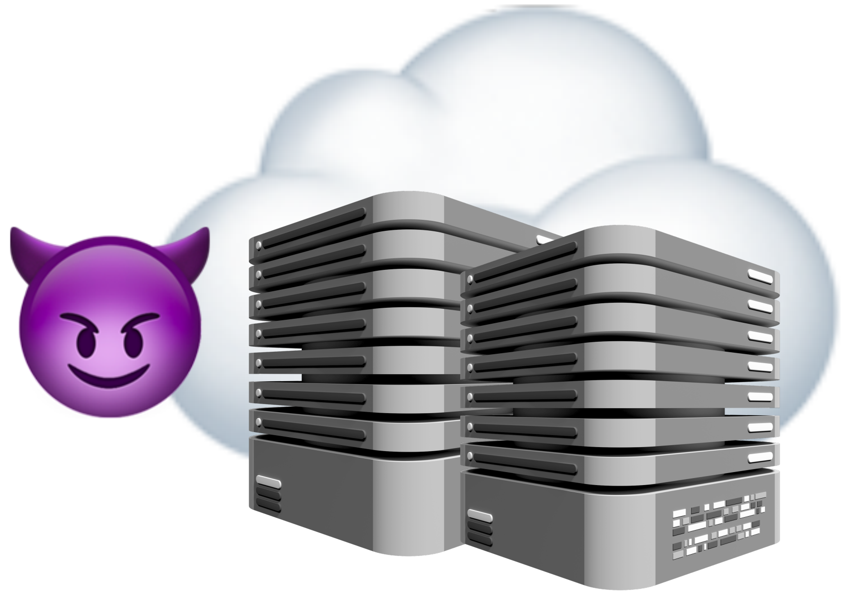
$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

$\longrightarrow$

NIBS blindness experiment

[Han23]



$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$ $\longrightarrow$

$\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$ $\longleftarrow$

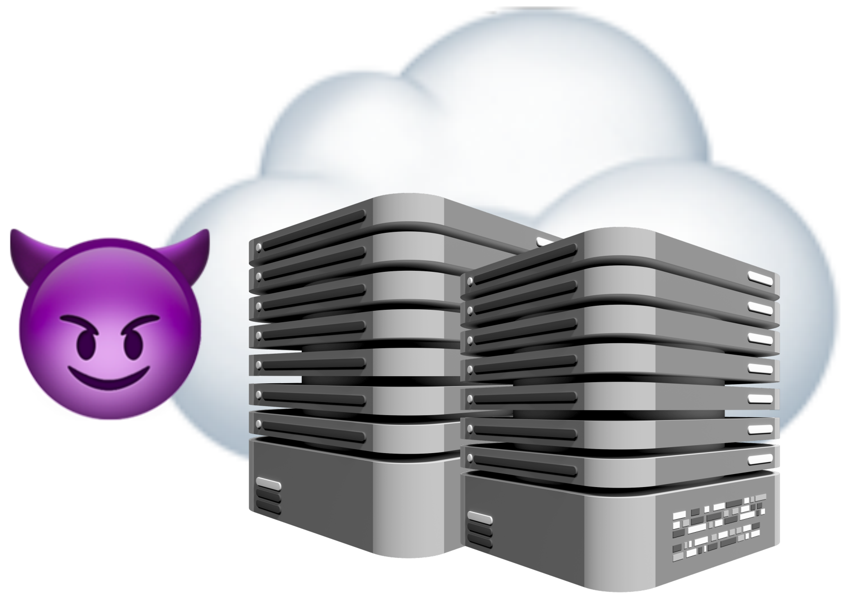
 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

 $\longrightarrow$

NIBS blindness experiment

[Han23]



$\color{red}{\mathsf{pk}_U^{(0)}}, \color{green}{\mathsf{pk}_U^{(1)}}$ $\longrightarrow$

$\longleftarrow$ $\mathsf{vk}_S, \color{red}{\overline{\sigma}_0}, \color{green}{\overline{\sigma}_1}$

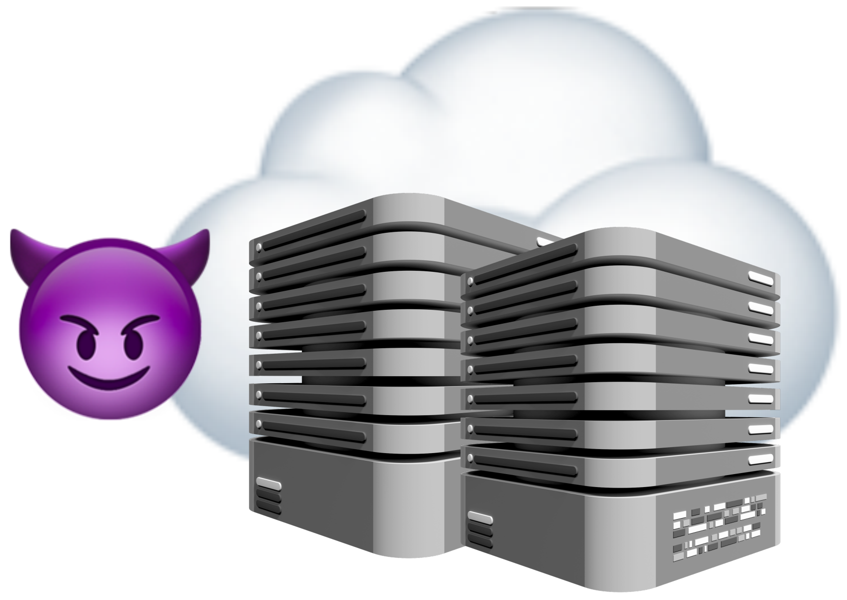
 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

 $\longrightarrow$

NIBS blindness experiment

[Han23]



$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$

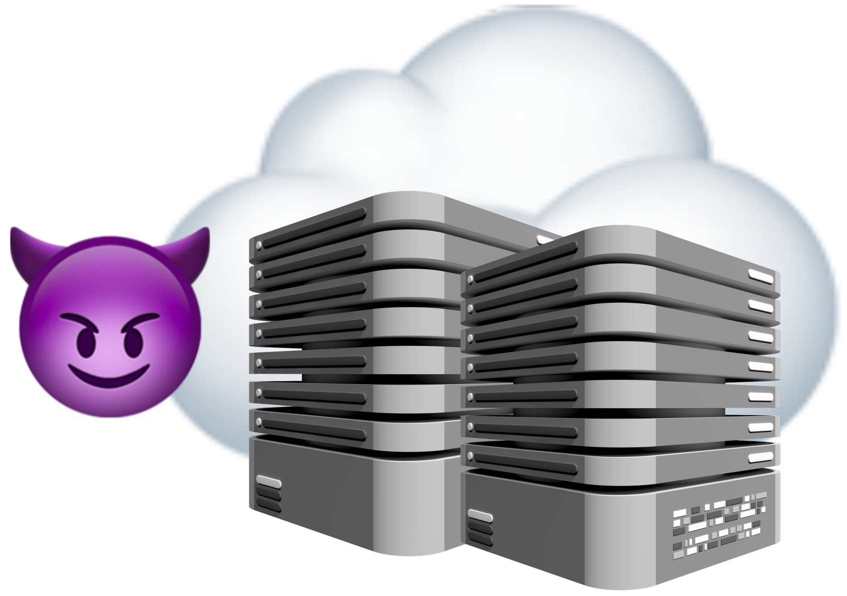
$\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

NIBS blindness experiment

[Han23]



$\color{red}{\mathsf{pk}_U^{(0)}}, \color{green}{\mathsf{pk}_U^{(1)}}$ $\longrightarrow$

$\mathsf{vk}_S, \color{red}{\overline{\sigma}_0}, \color{green}{\overline{\sigma}_1}$ $\longleftarrow$

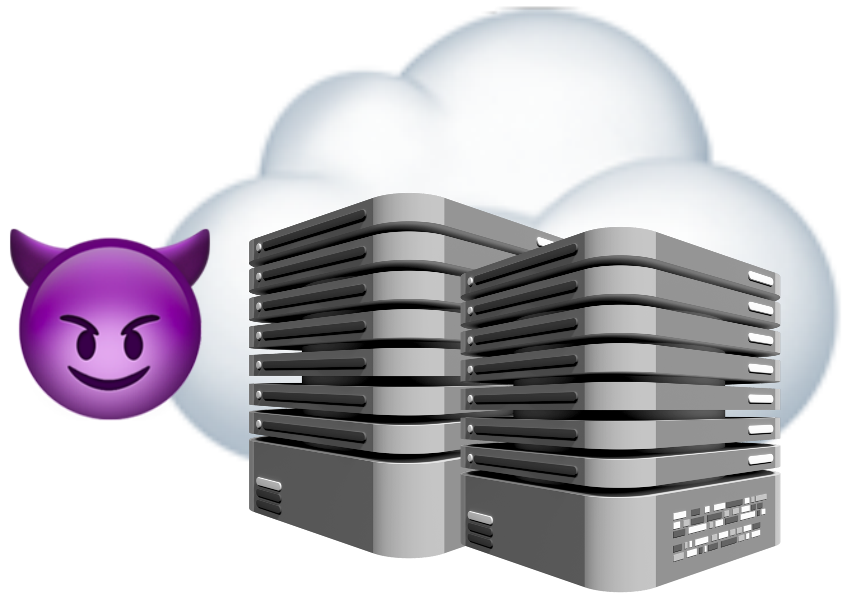
$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

$\longrightarrow$

NIBS blindness experiment

[Han23]



$$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)} \longrightarrow$$

$$\longleftarrow \mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$$

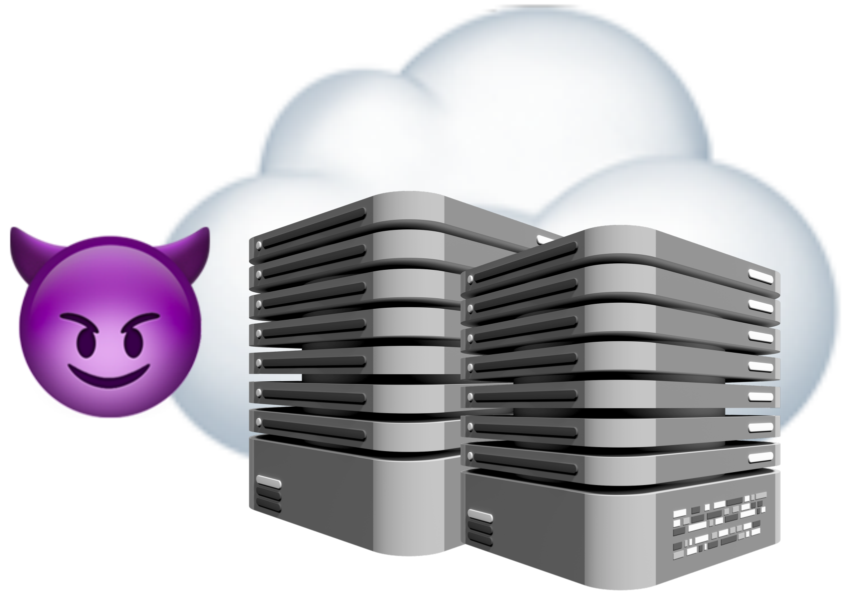
 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

$$\longrightarrow$$

NIBS blindness experiment

[Han23]



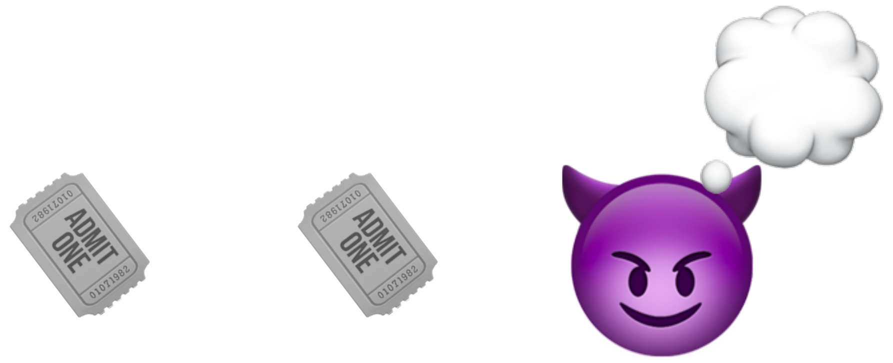
$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$ $\longrightarrow$

$\longleftarrow$ $\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

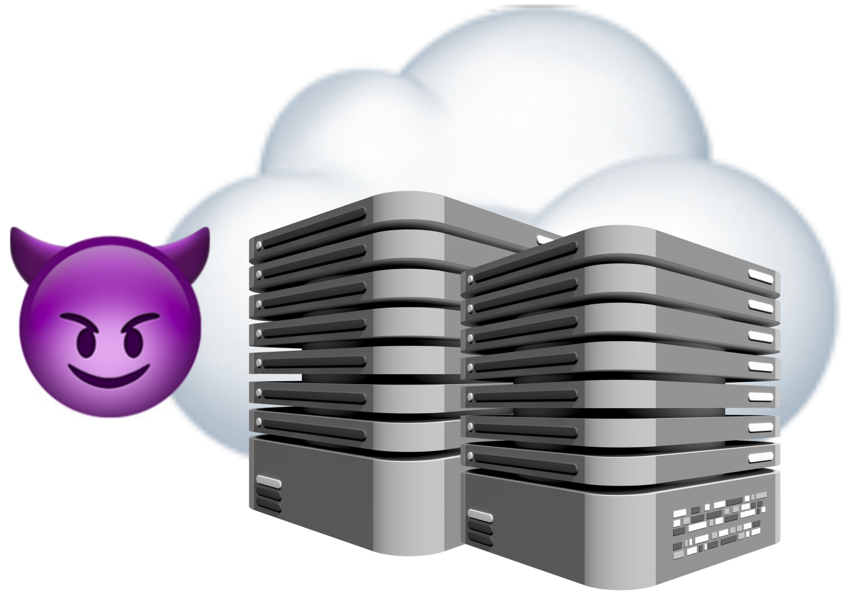
 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

$\longrightarrow$



NIBS blindness experiment

[Han23]

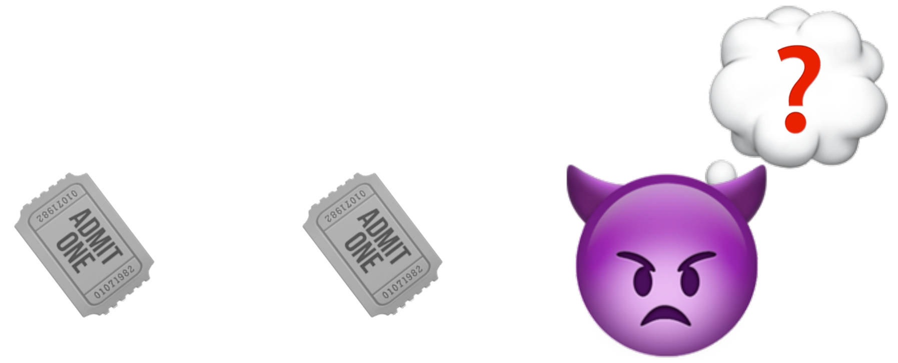


$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$

$\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$



Recall

Recall

Blindness requires that the signer should be unable to link a message and signature pair to any signing session.

Recall

Blindness requires that the signer should be unable to link a message and signature pair to any signing session.

Observation

Recall

Blindness requires that the signer should be unable to link a message and signature pair to any signing session.

Observation

There exist NIBS protocols that are provably secure under the blindness definitions of [Hanzlik23], but are easily **broken under very mild assumptions**.

Observation

There exist NIBS protocols that are provably secure under the blindness definitions of [Hanzlik23], but are easily **broken under very mild assumptions.**

Cause

Observation

There exist NIBS protocols that are provably secure under the blindness definitions of [Hanzlik23], but are easily **broken under very mild assumptions.**

Cause

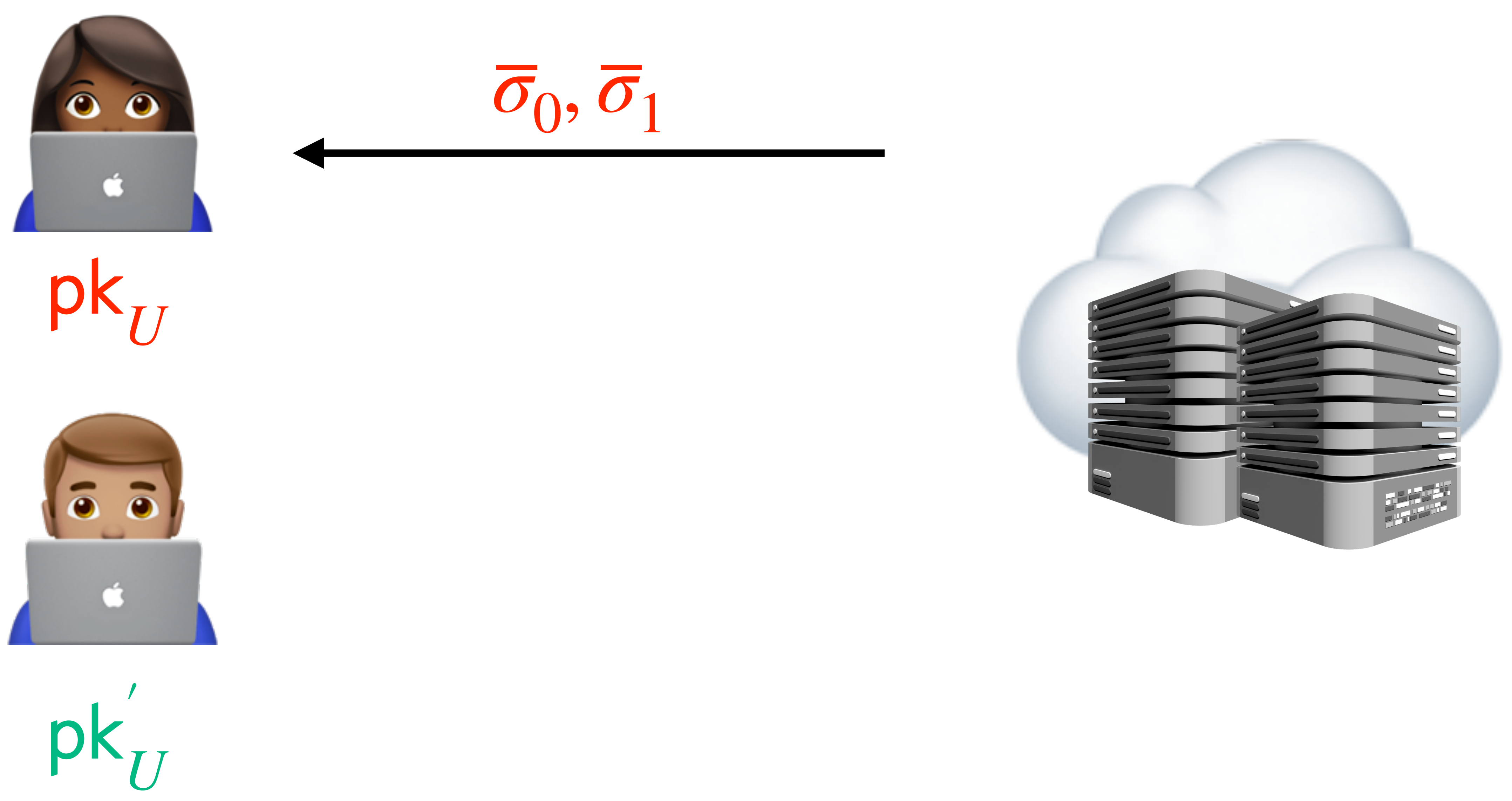
Previous definition restricts to the case where the adversary receives exactly *two* message and signature pairs from the challenger. In general, this need not be the case.

The issue with NIBS blindness

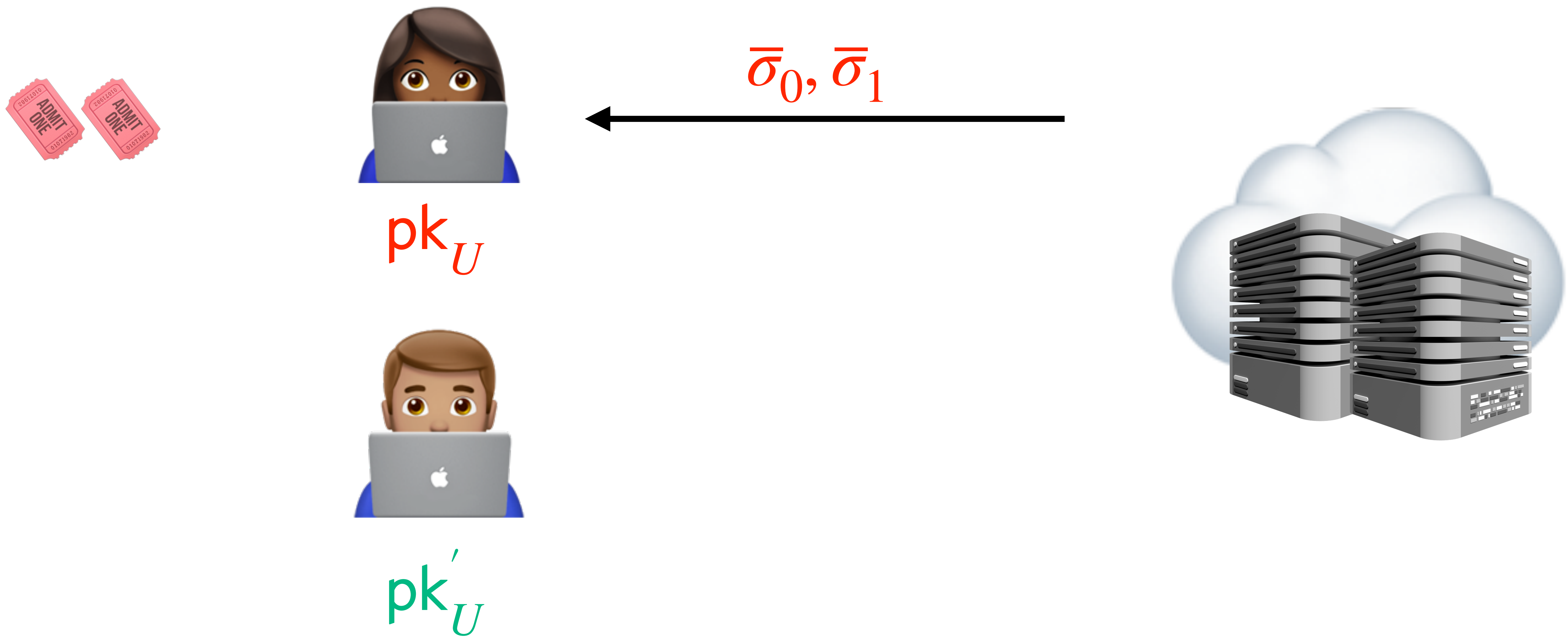
pk_U

$\mathsf{pk}_U^{'}$

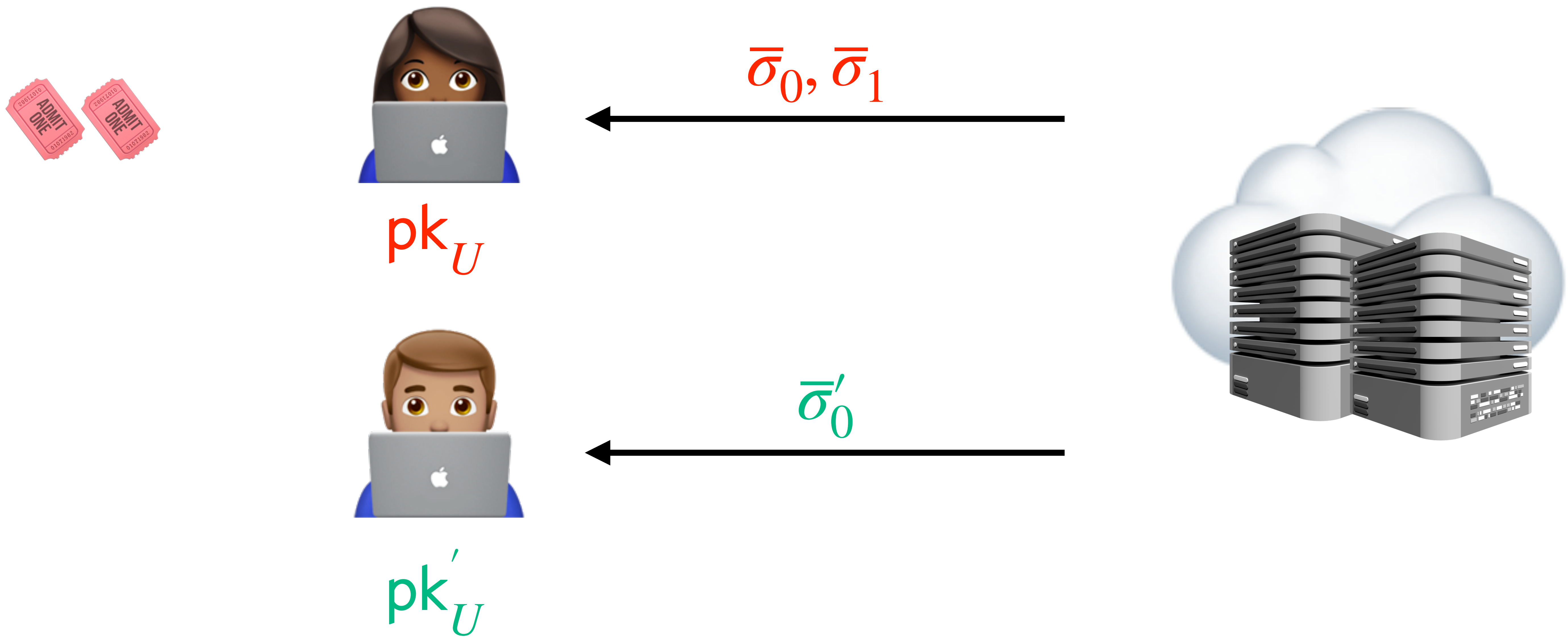
The issue with NIBS blindness



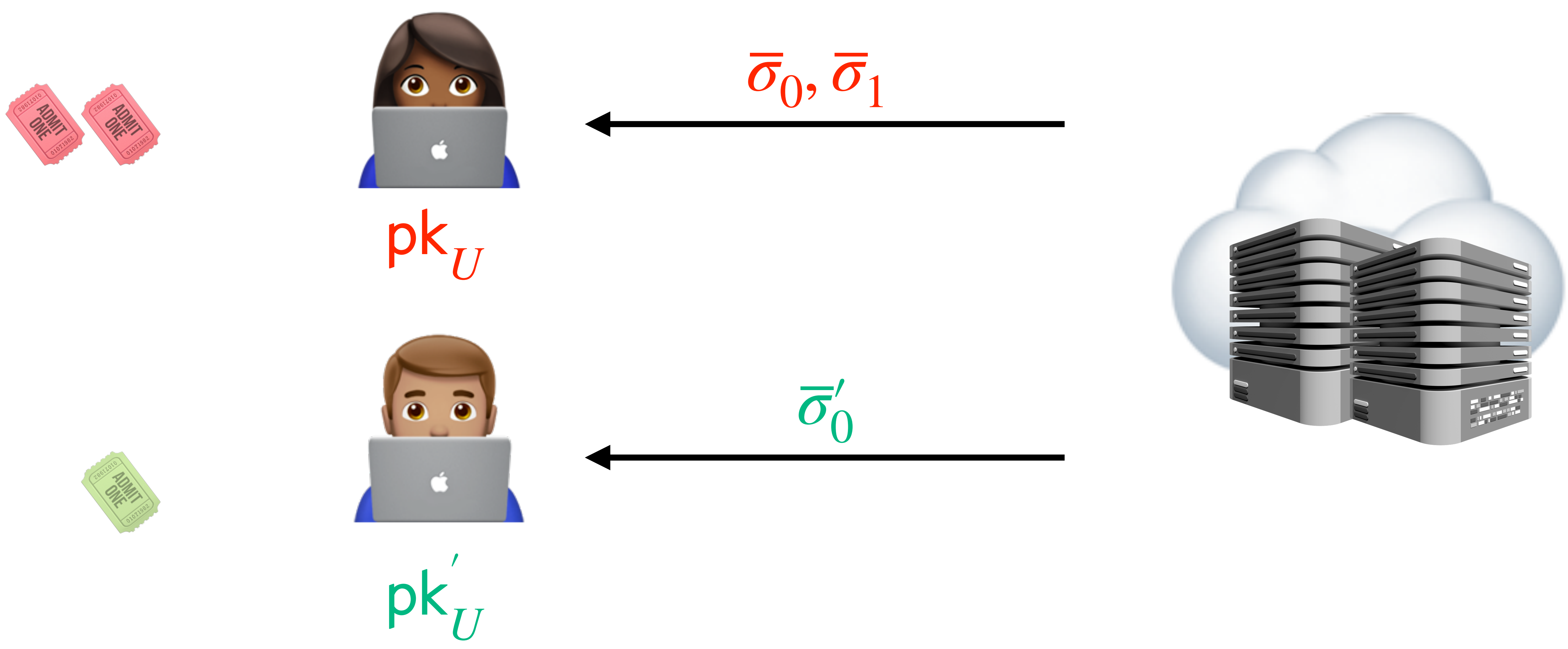
The issue with NIBS blindness

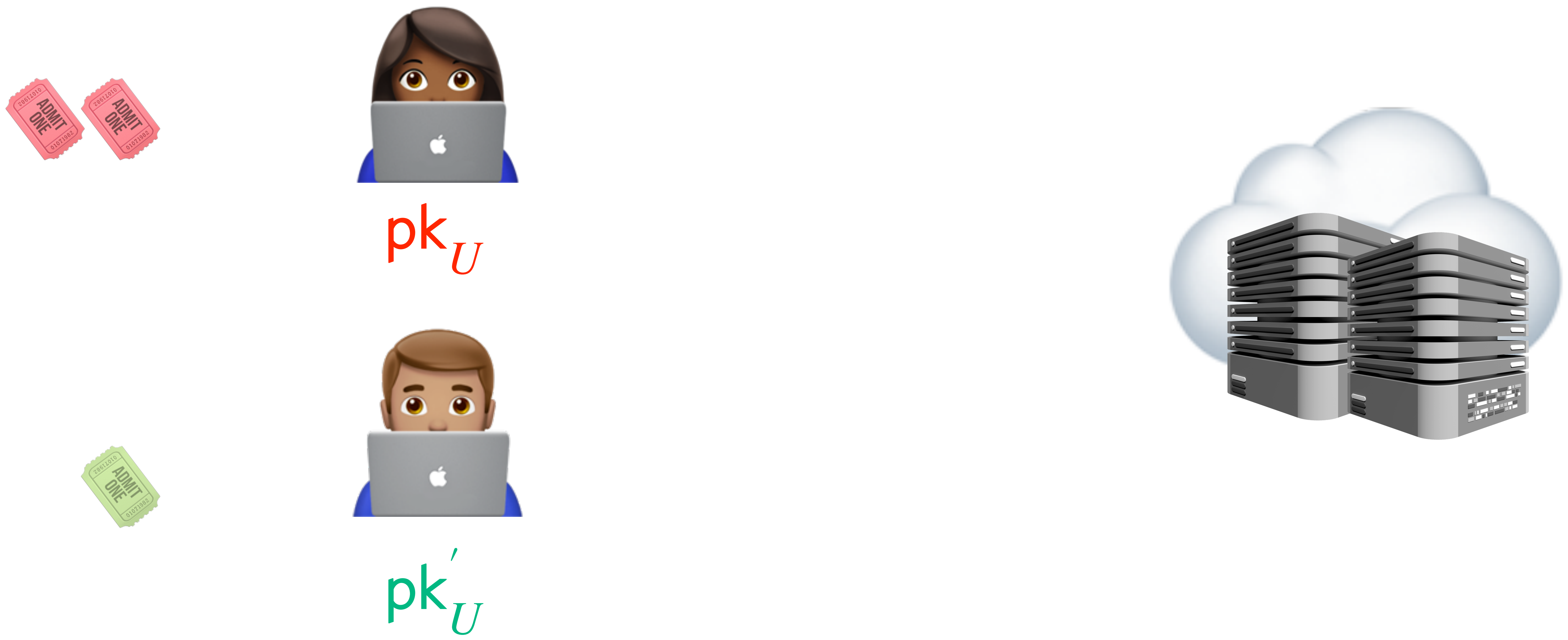


The issue with NIBS blindness

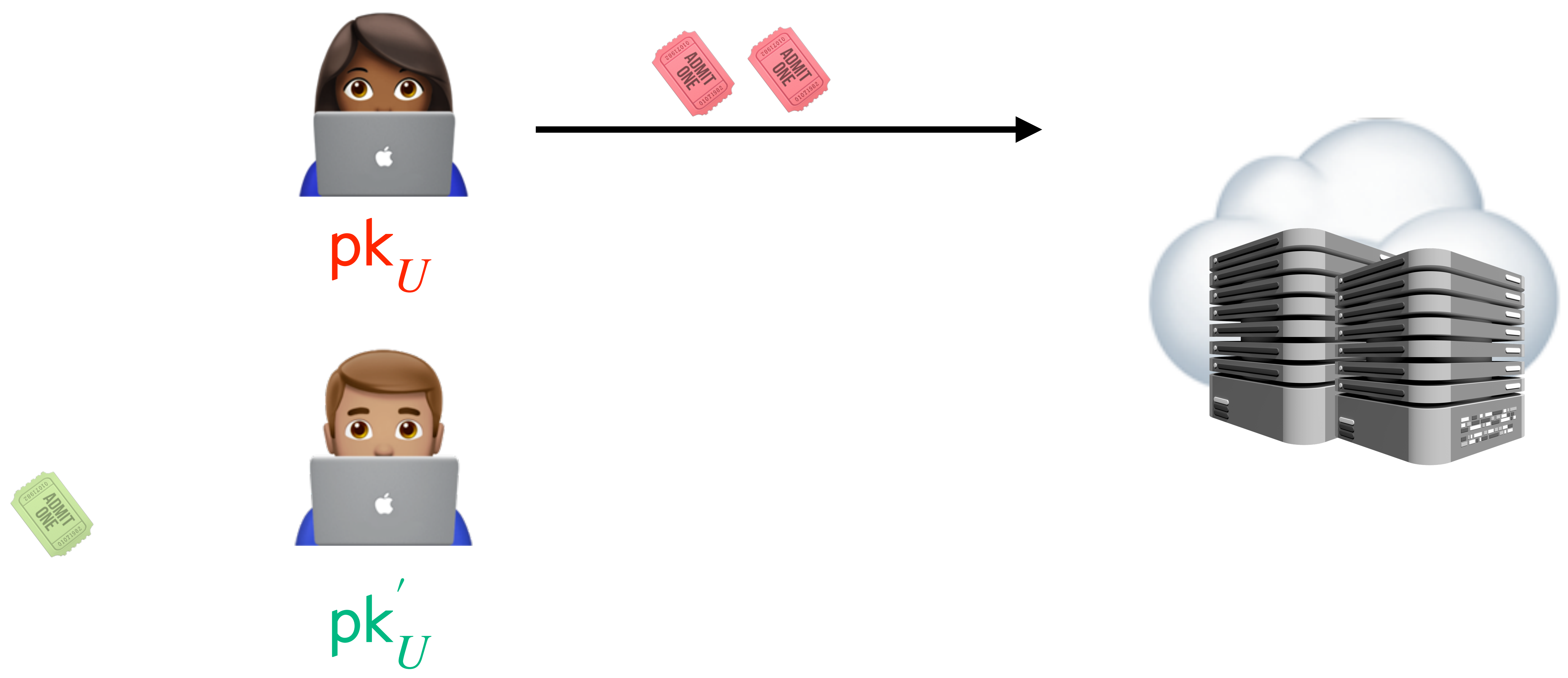


The issue with NIBS blindness

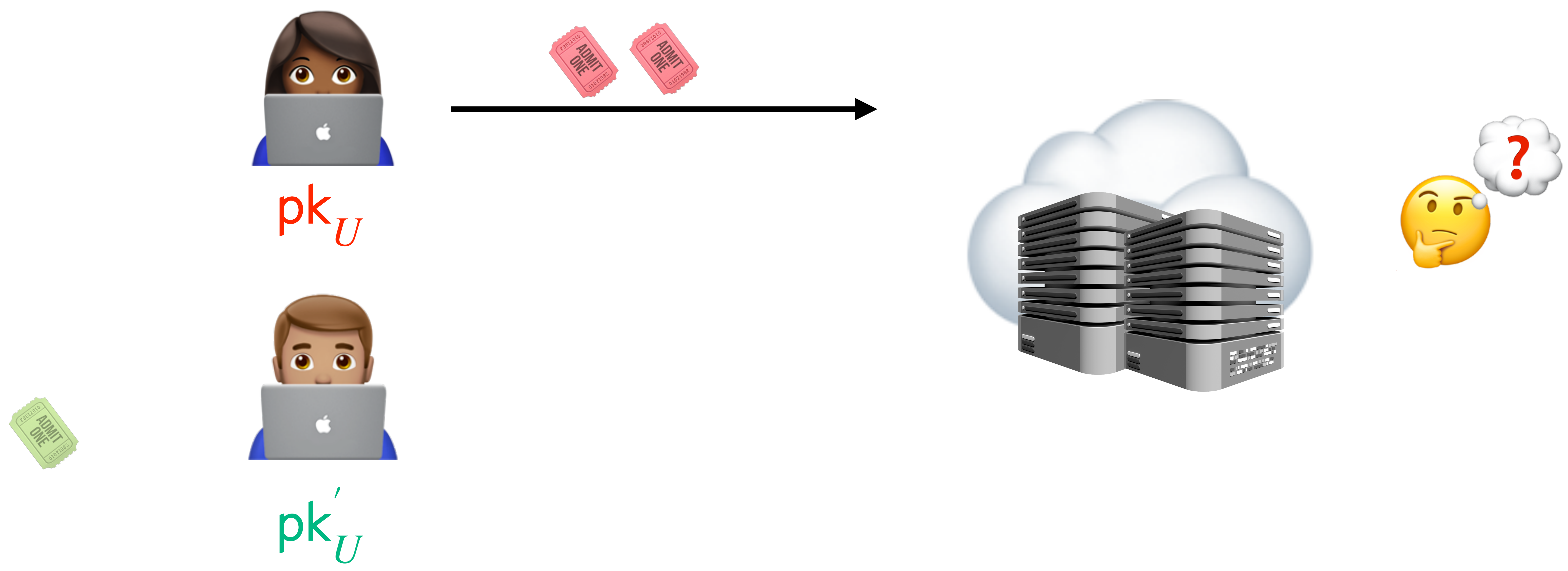



pk_U
pk'_U

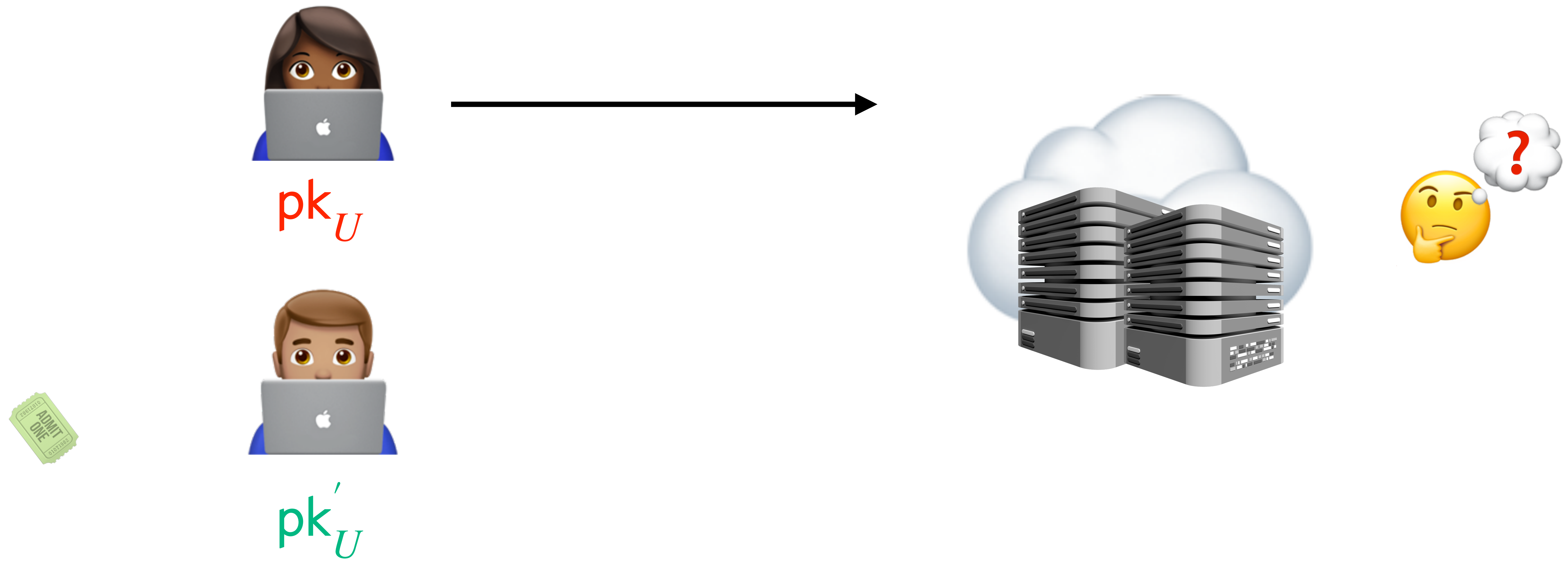
The issue with NIBS blindness



The issue with NIBS blindness



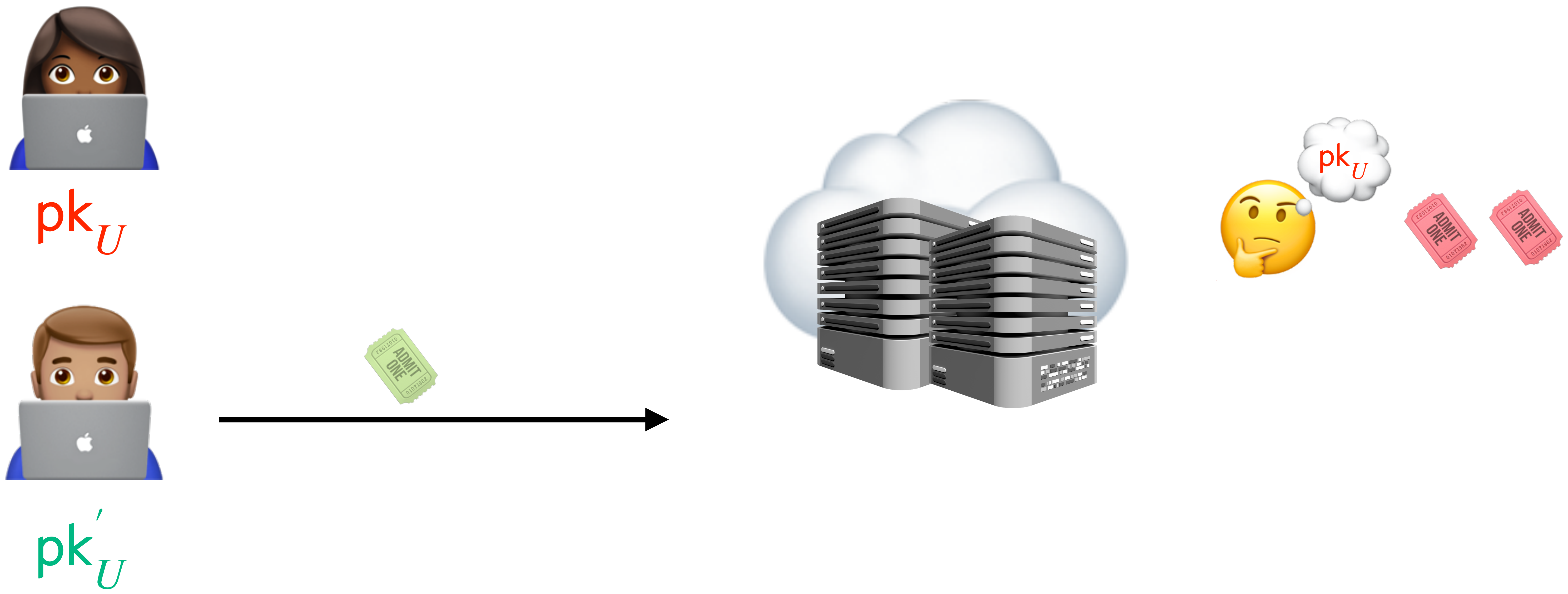
The issue with NIBS blindness



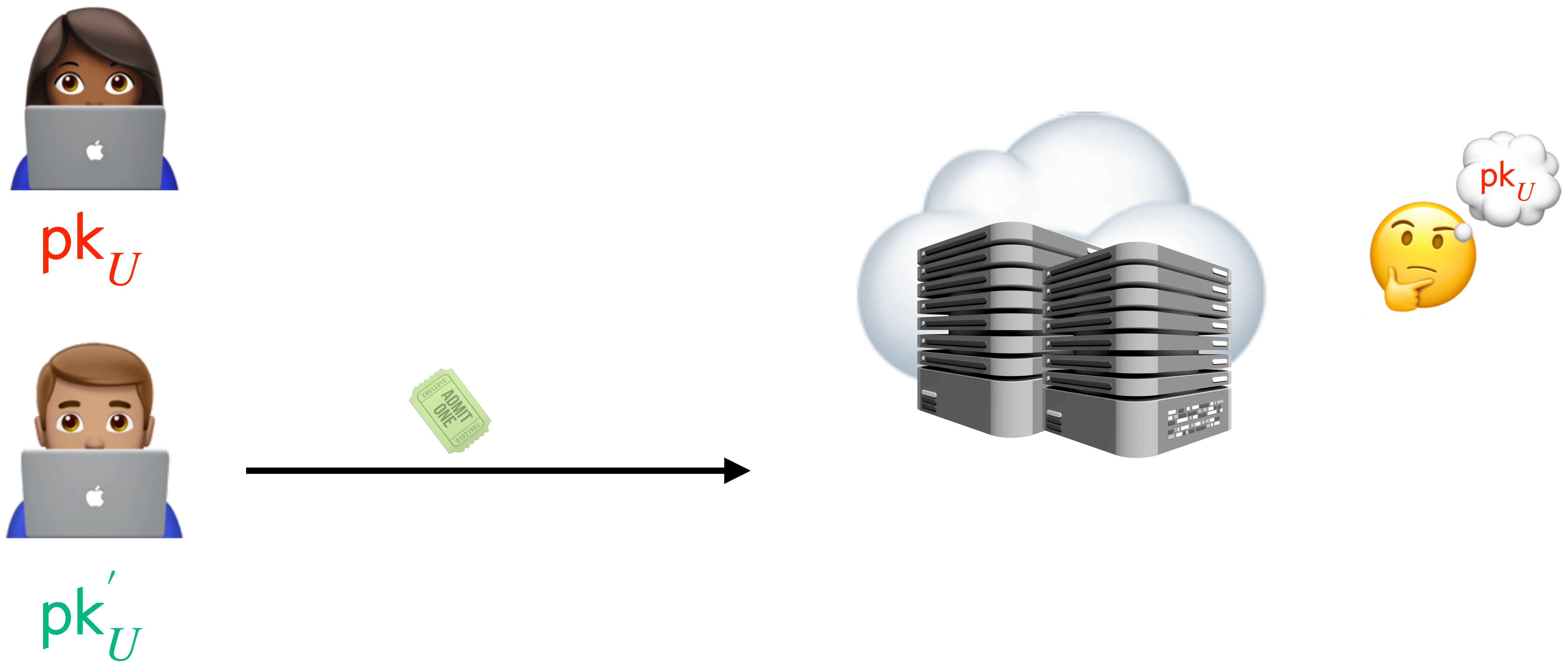
The issue with NIBS blindness

The issue with NIBS blindness

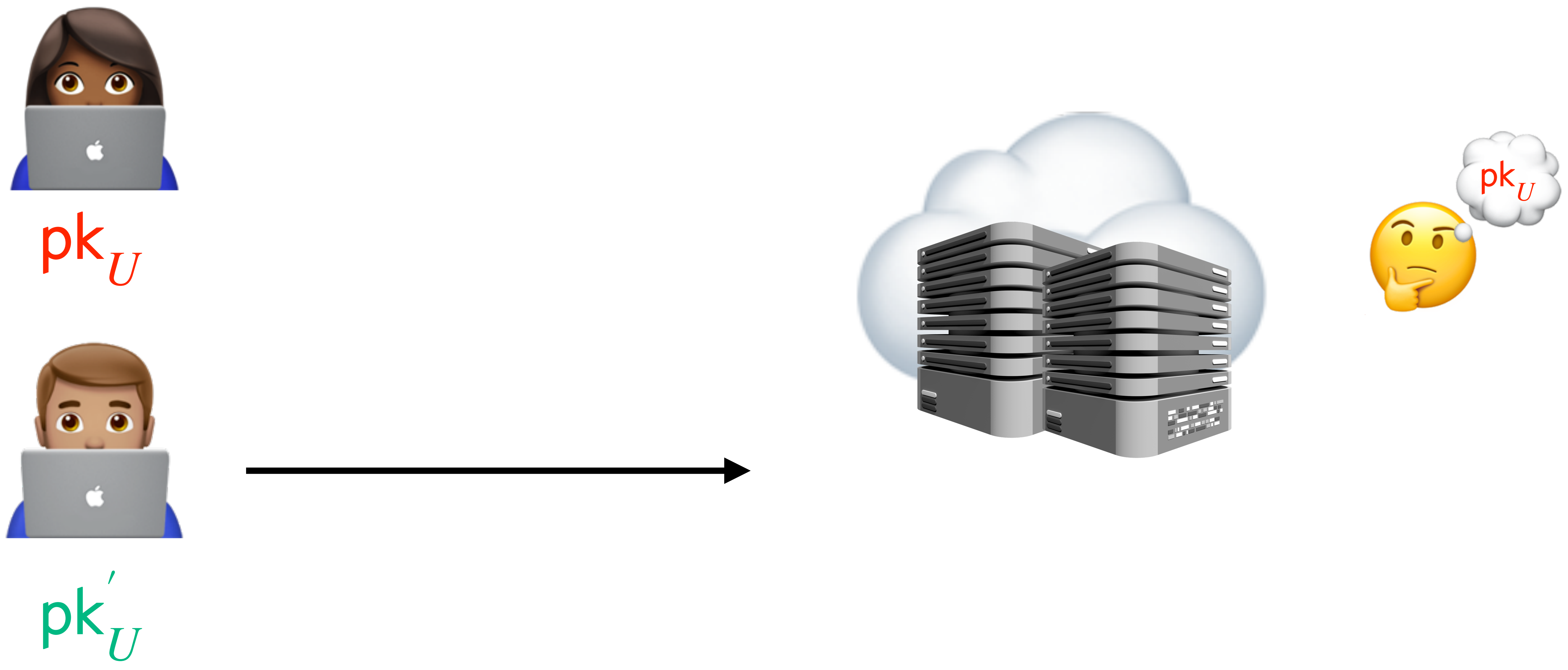
The issue with NIBS blindness



The issue with NIBS blindness



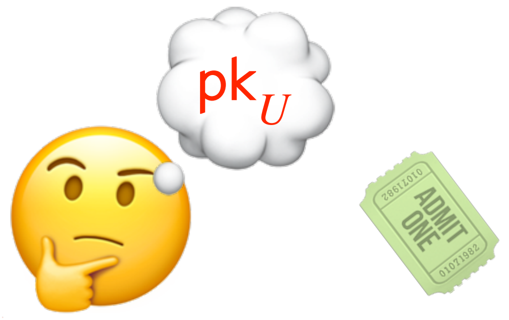
The issue with NIBS blindness



The issue with NIBS blindness

pk_U

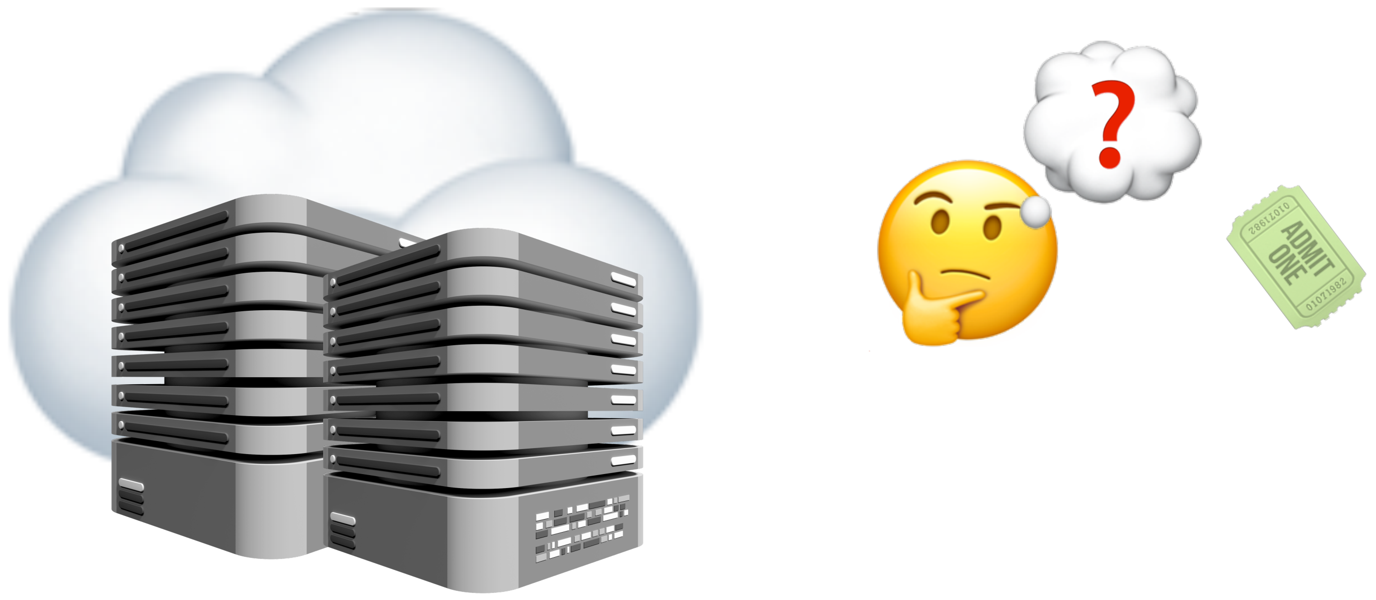
pk'_U



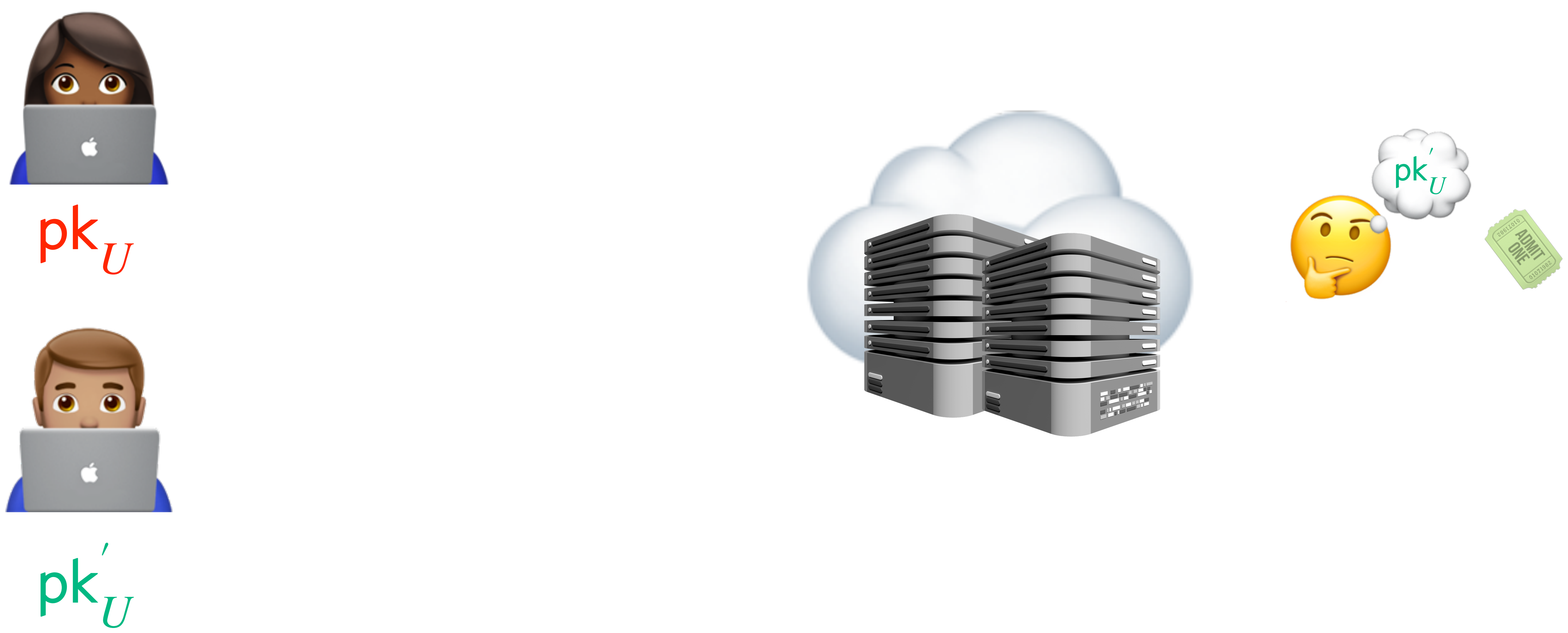
The issue with NIBS blindness

pk_U

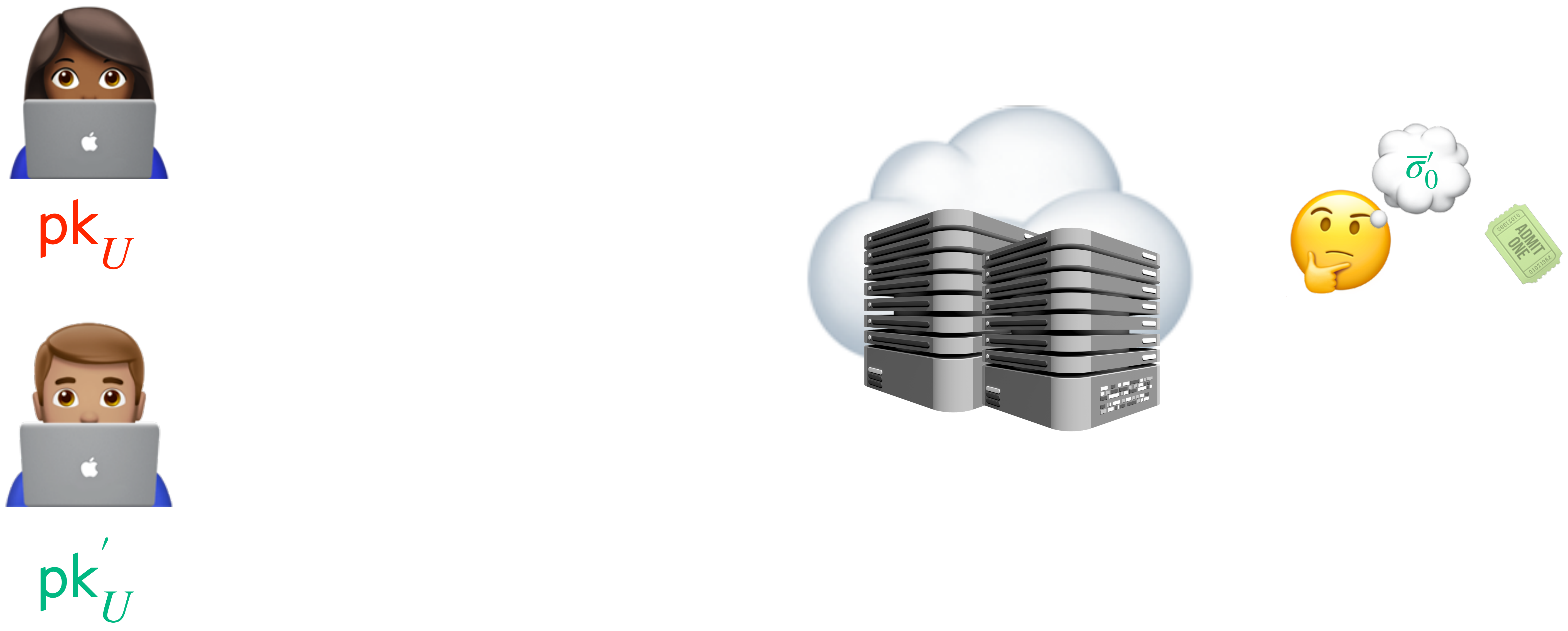
pk_U'



The issue with NIBS blindness



The issue with NIBS blindness



Cause

Previous definition restricts to the case where the adversary receives exactly *two* message and signature pairs from the challenger. In general, this need not be the case.

Our solution

- Give a new definition of blindness.

Cause

Previous definition restricts to the case where the adversary receives exactly *two* message and signature pairs from the challenger. In general, this need not be the case.

Our solution

- Give a new definition of blindness.
- Facilitated by providing the adversary with access to an oracle for the Obtain algorithm.

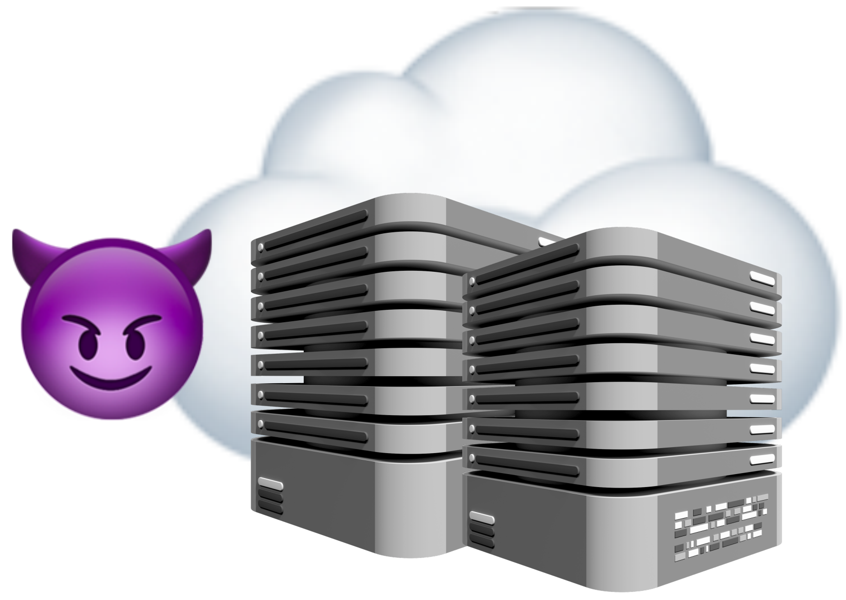
Cause

Previous definition restricts to the case where the adversary receives exactly *two* message and signature pairs from the challenger. In general, this need not be the case.

Our solution

- Give a new definition of blindness.
- Facilitated by providing the adversary with access to an oracle for the Obtain algorithm.
- Holds for an unbounded number of message and signature showings.

The issue with NIBS blindness



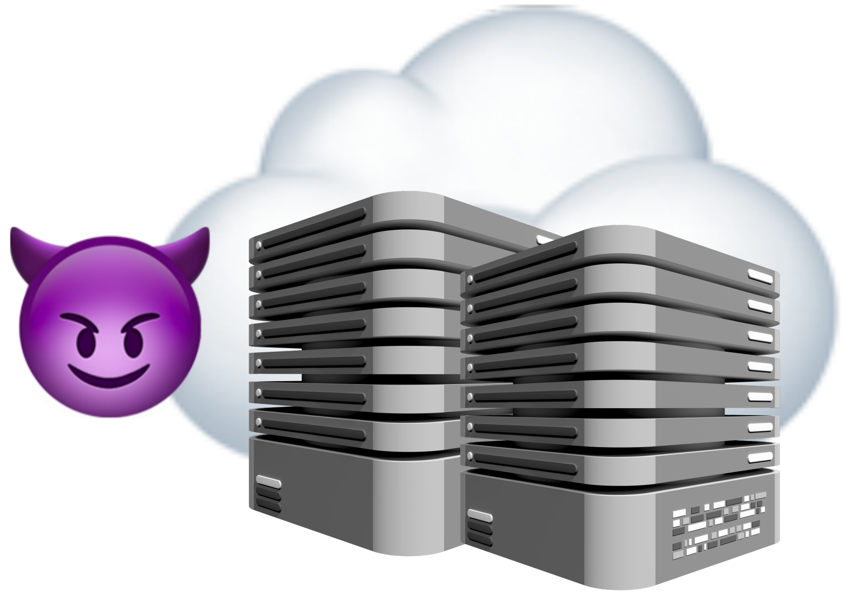
$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$

$\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

 $\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

The issue with NIBS blindness



$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$ $\longrightarrow$

$\longleftarrow$ $\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$

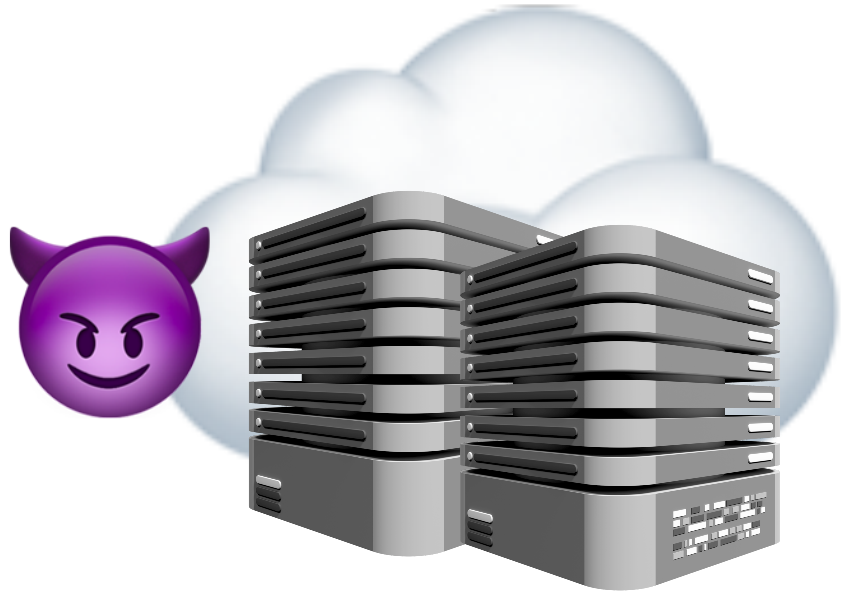
$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

$\longrightarrow$

The issue with NIBS blindness

$\overline{\sigma}$



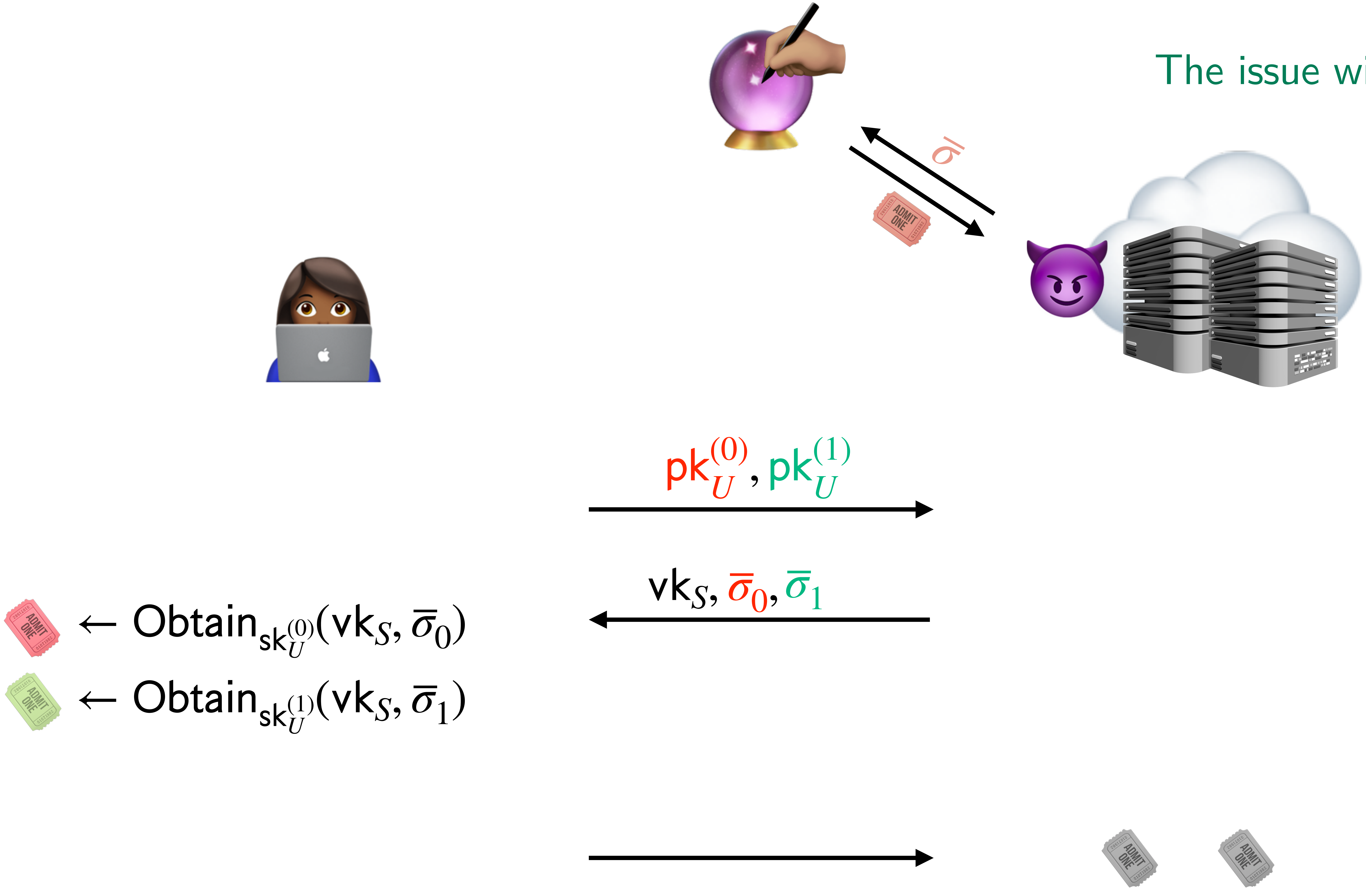
$\mathsf{pk}_U^{(0)}, \mathsf{pk}_U^{(1)}$

$\mathsf{vk}_S, \overline{\sigma}_0, \overline{\sigma}_1$

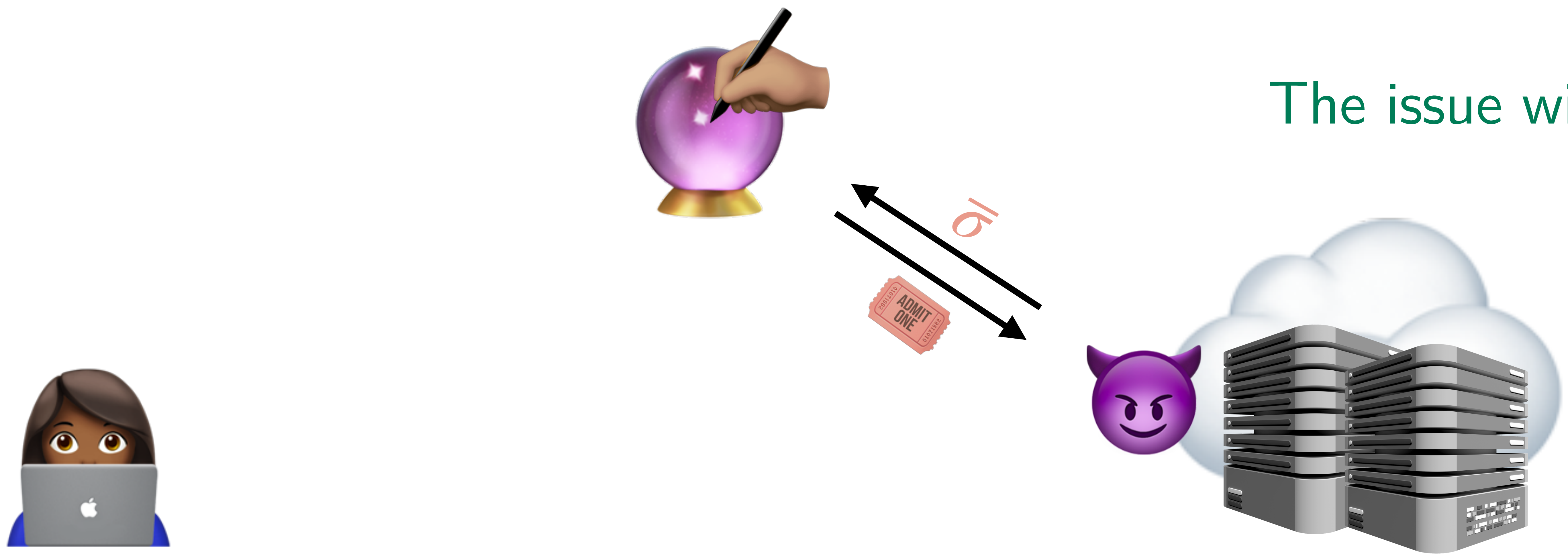
$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(0)}}(\mathsf{vk}_S, \overline{\sigma}_0)$

$\leftarrow \mathsf{Obtain}_{\mathsf{sk}_U^{(1)}}(\mathsf{vk}_S, \overline{\sigma}_1)$

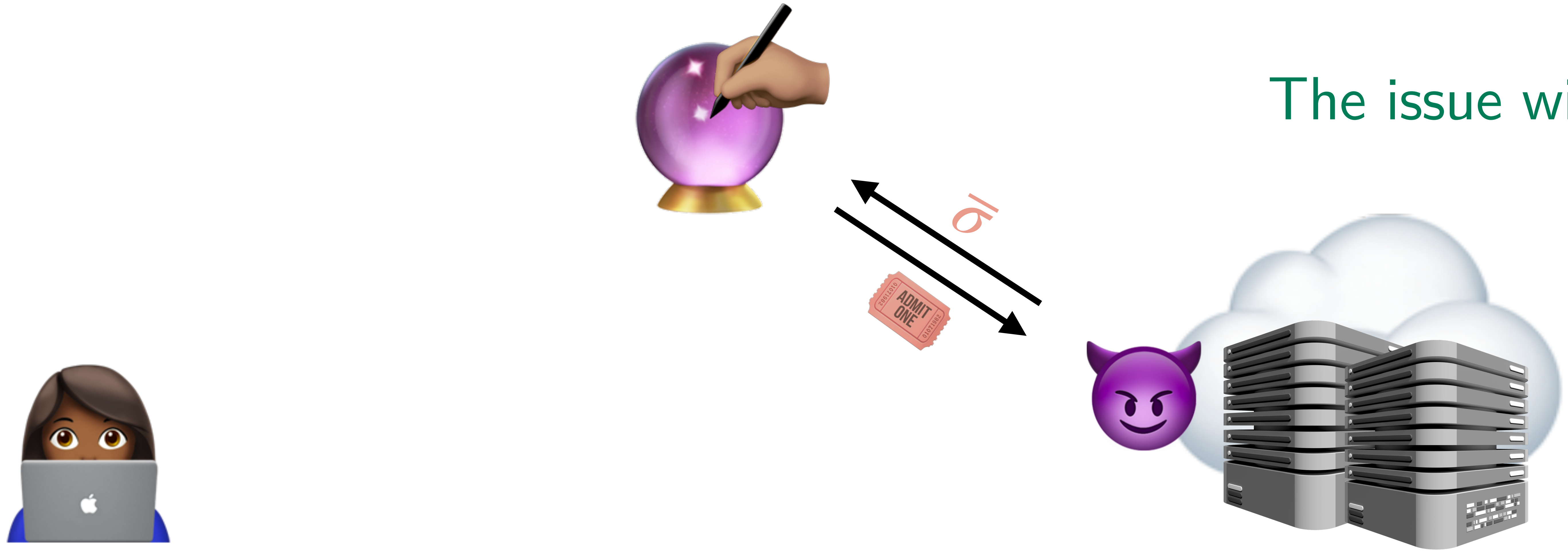
The issue with NIBS blindness



The issue with NIBS blindness



Claim



The issue with NIBS blindness

Claim

The stronger blindness definition captures the actual baseline requirements for NIBS blindness.

Interlude: the landscape of NIBS constructions

- Generic compiler from dual-mode witness indistinguishable proofs [Groth-Sahai08] and verifiable random functions*.
- Pairing-based construction* (secure in the generic group model) [Han23]

Interlude: the landscape of NIBS constructions

👍 Generic compiler from dual-mode witness indistinguishable proofs [Groth-Sahai08] and verifiable random functions*.

- Pairing-based construction* (secure in the generic group model) [Han23]

Interlude: the landscape of NIBS constructions

- 👍 Generic compiler from dual-mode witness indistinguishable proofs [Groth-Sahai08] and verifiable random functions*.
- 👍 Pairing-based construction* (secure in the generic group model) [Han23]

Interlude: the landscape of NIBS constructions

👍 Generic compiler from dual-mode witness indistinguishable proofs [Groth-Sahai08] and verifiable random functions*.

👍 Pairing-based construction* (secure in the generic group model) [Han23]

*We believe that both constructions satisfy our (stronger) baseline blindness definitions.

Interlude: the landscape of NIBS constructions

- Generic compiler from dual-mode witness indistinguishable proofs [Groth-Sahai08] and verifiable random functions*.
- Pairing-based construction* (secure in the generic group model) [Han23]
  - Complicated generic construction with large signatures.
  - No post-quantum secure construction.

Interlude: the landscape of NIBS constructions

👍 Generic compiler from dual-mode witness indistinguishable proofs [Groth-Sahai08] and verifiable random functions*.

👍 Pairing-based construction* (secure in the generic group model) [Han23]

👎 Complicated generic construction with large signatures.

- No post-quantum secure construction.

Interlude: the landscape of NIBS constructions

- Generic compiler from dual-mode witness indistinguishable proofs [Groth-Sahai08] and verifiable random functions*.
- Pairing-based construction* (secure in the generic group model) [Han23]
- Complicated generic construction with large signatures.
- No post-quantum secure construction.

Fischlin's paradigm

A generic approach to blind signatures [Fischlin06]

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_{\mathsf{COM}}, \mathsf{vk}_{\Sigma})$

$(\mathsf{sk}_{\mathsf{COM}}, \mu)$

(sk_{Σ})

$r, \nu \leftarrow \{0,1\}^{\lambda}$

$\beta \leftarrow \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(\mu; r) \quad \xrightarrow{\quad\beta\quad}$

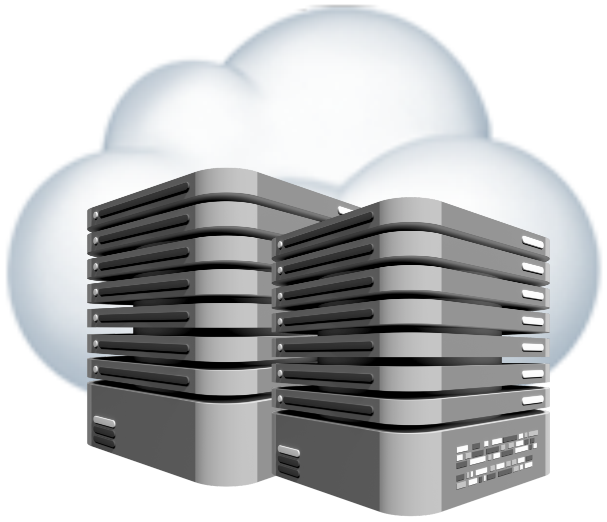
$\psi \leftarrow \mathsf{Enc}_{\mathsf{pk}_{\mathsf{PKE}}}(\overline{\sigma} \,||\, \beta; \nu) \quad \xleftarrow{\quad\overline{\sigma}\quad} \quad \overline{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_{\Sigma}}(\beta)$

$$\pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}\begin{pmatrix} \mathfrak{x} := (\mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_{\Sigma}, \mu, \psi), \\ \mathfrak{w} := (\mathsf{sk}_{\mathsf{COM}}, r, \nu, \overline{\sigma}) \end{pmatrix} \quad \xrightarrow{\mu, \sigma := (\psi, \pi)} \quad \mathsf{Verify}_{\mathsf{crs}}(\mathfrak{x} := (\mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_{\Sigma}, \mu, \psi), \pi)$$

Fischlin's paradigm

[Fischlin06]

(μ)



(sk_Σ)

$r \leftarrow \{0,1\}^\lambda$

$\beta \leftarrow \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(\mu; r)$ $\xrightarrow{\beta}$

$\xleftarrow{\overline{\sigma}}$ $\overline{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_\Sigma}(\beta)$

$$\pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}\begin{pmatrix} \mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \\ \mathfrak{w} := (\mathsf{sk}_{\mathsf{COM}}, r, \overline{\sigma}) \end{pmatrix} \xrightarrow{\mu,\, \sigma := \pi} \mathsf{Verify}_{\mathsf{crs}}(\mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \pi)$$

Fischlin's paradigm

[Fischlin06]

(μ)

(sk_Σ)

$r \leftarrow \{0,1\}^\lambda$

$\beta \leftarrow$ Commitment to μ

$\beta \leftarrow \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(\mu; r)$

$\xrightarrow{\beta}$

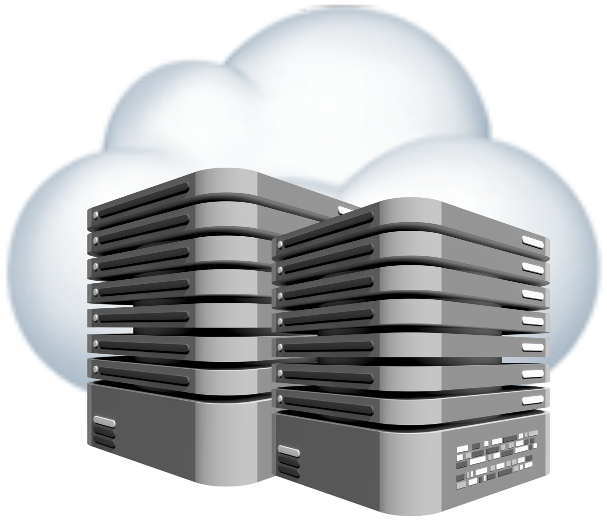
$\xleftarrow{\overline{\sigma}}$ $\qquad \overline{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_\Sigma}(\beta)$

$$\pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}\begin{pmatrix} \mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \\ \mathfrak{w} := (\mathsf{sk}_{\mathsf{COM}}, r, \overline{\sigma}) \end{pmatrix} \xrightarrow{\mu,\ \sigma := \pi} \mathsf{Verify}_{\mathsf{crs}}(\mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \pi)$$

Fischlin's paradigm

[Fischlin06]

(μ)



(sk_Σ)

$\beta \leftarrow$ Commitment to μ

$\xrightarrow{\quad\beta\quad}$

$\xleftarrow{\quad\overline{\sigma}\quad}$

$\overline{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_\Sigma}(\beta)$

$$\pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}\begin{pmatrix} \mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \\ \mathfrak{w} := (\mathsf{sk}_{\mathsf{COM}}, r, \overline{\sigma}) \end{pmatrix}$$

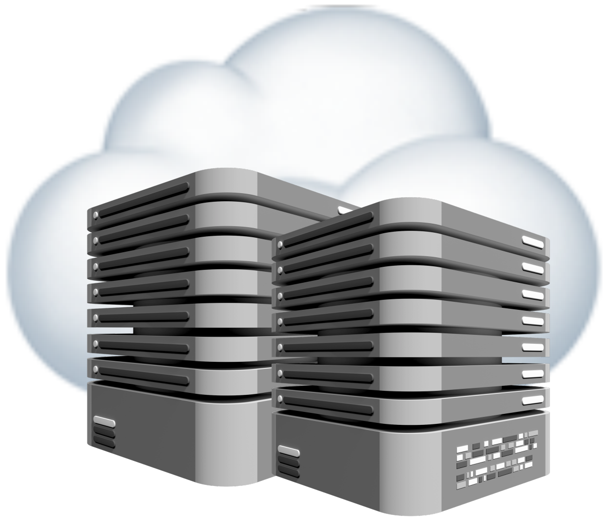
$\xrightarrow{\quad\mu,\, \sigma := \pi\quad}$

$$\mathsf{Verify}_{\mathsf{crs}}(\mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \pi)$$

Fischlin's paradigm

[Fischlin06]

(μ)



(sk_Σ)

$\beta \leftarrow$ Commitment to μ $\xrightarrow{\quad\beta\quad}$

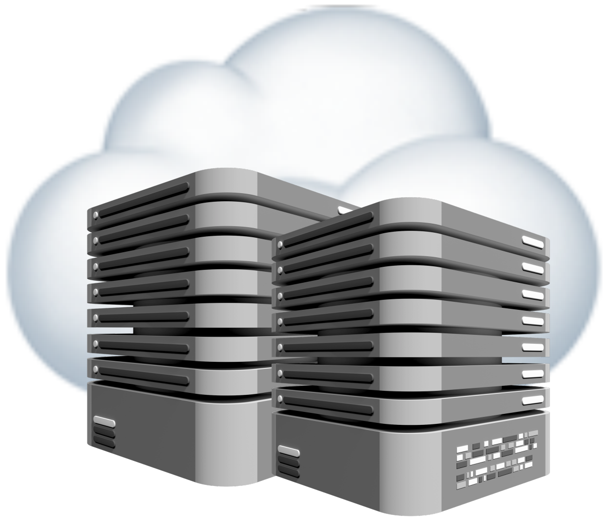
$\xleftarrow{\quad\overline{\sigma}\quad}$ $\overline{\sigma} \leftarrow$ Signature on β

$$\pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}\begin{pmatrix} \mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \\ \mathfrak{w} := (\mathsf{sk}_{\mathsf{COM}}, r, \overline{\sigma}) \end{pmatrix} \xrightarrow{\quad \mu, \sigma := \pi \quad} \mathsf{Verify}_{\mathsf{crs}}(\mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \pi)$$

Fischlin's paradigm

[Fischlin06]

(μ)



(sk_Σ)

$\beta \leftarrow$ Commitment to μ $\xrightarrow{\beta}$

$\xleftarrow{\overline{\sigma}}$ $\overline{\sigma} \leftarrow$ Signature on β

$$\pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}\begin{pmatrix} \mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \\ \mathfrak{w} := (\mathsf{sk}_{\mathsf{COM}}, r, \overline{\sigma}) \end{pmatrix} \xrightarrow{\mu, \sigma := \pi} \mathsf{Verify}_{\mathsf{crs}}(\mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \pi)$$

[Fischlin06]

(μ)

(sk_Σ)

$\beta \leftarrow$ Commitment to μ $\xrightarrow{\beta}$

$\xleftarrow{\overline{\sigma}}$ $\overline{\sigma} \leftarrow$ Signature on β

$\pi \leftarrow$ Proof of knowledge for $\mathscr{R}$

$\pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}\left(\begin{matrix}\mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \\ \mathfrak{w} := (\mathsf{sk}_{\mathsf{COM}}, r, \overline{\sigma})\end{matrix}\right)$

$\xrightarrow{\mu, \sigma := \pi}$ $\mathsf{Verify}_{\mathsf{crs}}(\mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \pi)$

Fischlin's paradigm

[Fischlin06]

(μ)

(sk_Σ)

$\beta \leftarrow$ Commitment to μ

$\xrightarrow{\beta}$

$\xleftarrow{\overline{\sigma}}$

$\overline{\sigma} \leftarrow$ Signature on β

$\pi \leftarrow$ Proof of knowledge for $\mathscr{R}$

$\pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}\left(\begin{array}{l}\mathfrak{x} := (\mathsf{vk}_\Sigma, \mu),\\ \mathfrak{w} := (\mathsf{sk}_{\mathsf{COM}}, r, \overline{\sigma})\end{array}\right)$

$\xrightarrow{\mu, \sigma := \pi}$

$\mathsf{Verify}_{\mathsf{crs}}(\mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \pi)$

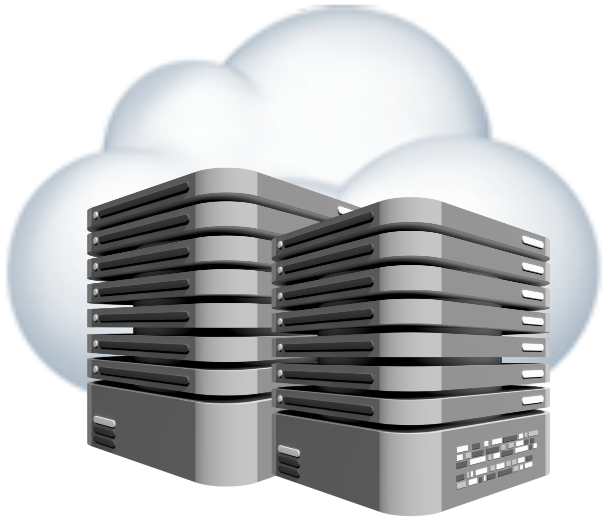
Relation $\mathscr{R}$

$\overline{\sigma}$ is a verifying signature on β

Fischlin's paradigm

[Fischlin06]

(μ)



(sk_Σ)

$\beta \leftarrow$ Commitment to μ $\xrightarrow{\beta}$

$\xleftarrow{\overline{\sigma}}$ $\overline{\sigma} \leftarrow$ Signature on β

$\pi \leftarrow$ Proof of knowledge for $\mathscr{R}$ $\xrightarrow{\mu, \sigma := \pi}$ $\mathsf{Verify}_{\mathsf{crs}}(\mathfrak{x} := (\mathsf{vk}_\Sigma, \mu), \pi)$

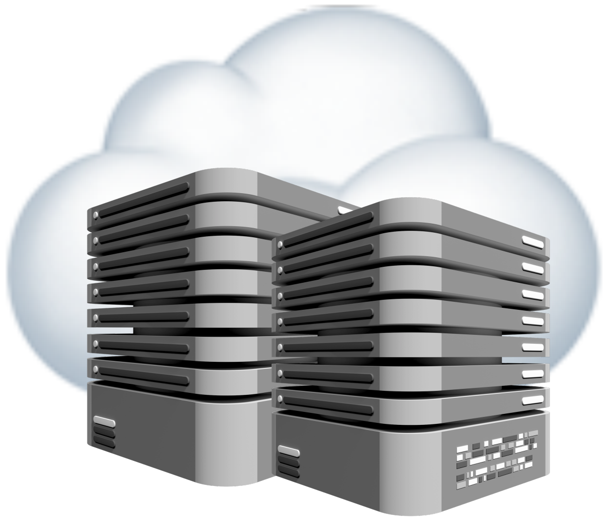
Relation $\mathscr{R}$

$\overline{\sigma}$ is a verifying signature on β

Fischlin's paradigm

[Fischlin06]

(μ)



(sk_Σ)

$\beta \leftarrow$ Commitment to μ $\xrightarrow{\beta}$

$\xleftarrow{\overline{\sigma}}$ $\overline{\sigma} \leftarrow$ Signature on β

$\pi \leftarrow$ Proof of knowledge for $\mathscr{R}$ $\xrightarrow{\mu, \sigma := \pi}$ Verify π $\mathcal{V}_{\mathsf{crs}}(\mathbb{x} := (\mathsf{vk}_\Sigma, \mu), \pi)$

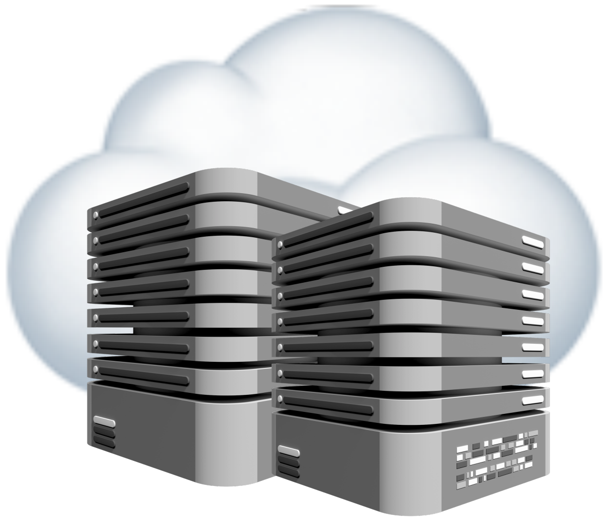
Relation $\mathscr{R}$

$\overline{\sigma}$ is a verifying signature on β

Fischlin's paradigm

[Fischlin06]

(μ)



(sk_Σ)

$\beta \leftarrow$ Commitment to μ $\quad \xrightarrow{\quad\beta\quad}$

$\xleftarrow{\quad\overline{\sigma}\quad} \quad$ $\overline{\sigma} \leftarrow$ Signature on β

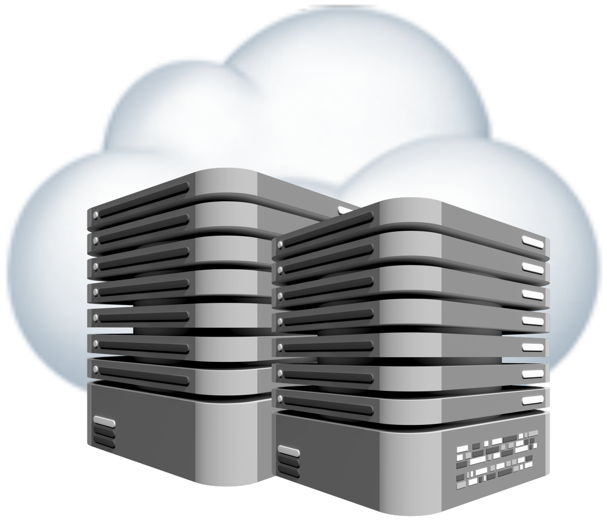
$\pi \leftarrow$ Proof of knowledge for $\mathscr{R}$ $\quad \xrightarrow{\mu, \sigma := \pi} \quad$ Verify π

Relation $\mathscr{R}$

$\overline{\sigma}$ is a verifying signature on β

Adapting Fischlin's paradigm to NIBS

(μ)



(sk_Σ)

$\beta \leftarrow$ Commitment to μ

$\xrightarrow{\quad \beta \quad}$

$\xleftarrow{\quad \overline{\sigma} \quad}$

$\overline{\sigma} \leftarrow$ Signature on β

$\pi \leftarrow$ Proof of knowledge for $\mathscr{R}$

$\xrightarrow{\quad \mu, \sigma := \pi \quad}$

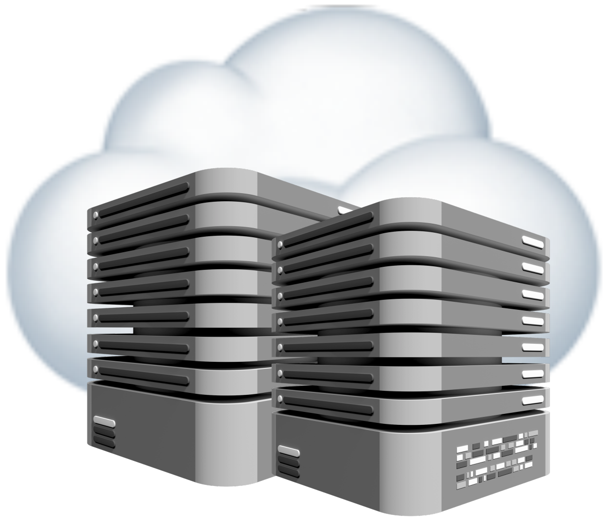
Verify π

Want

- The public key must act as a succinct **commitment of an exponential number of messages** at once.

Adapting Fischlin's paradigm to NIBS

(μ)



(sk_Σ)

$\beta \leftarrow$ Commitment to μ $\xrightarrow{\beta}$

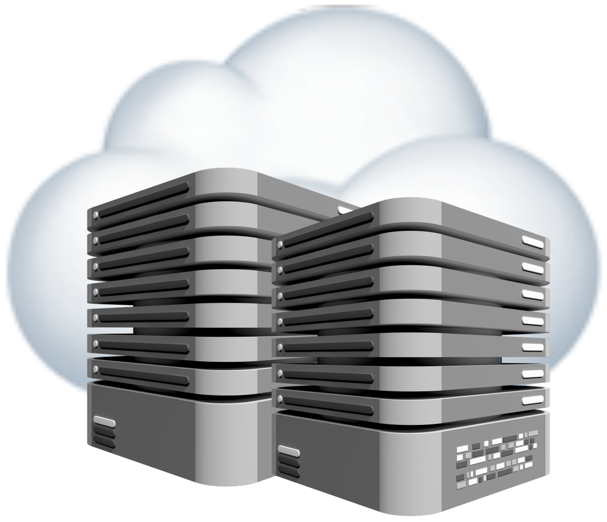
$\xleftarrow{\overline{\sigma}}$ $\overline{\sigma} \leftarrow$ Signature on β

$\pi \leftarrow$ Proof of knowledge for $\mathscr{R}$ $\xrightarrow{\mu, \sigma := \pi}$ Verify π

Want

- The public key must act as a succinct **commitment of an exponential number of messages** at once.
- Signer must sign the commitment in a way such that the **signature obliviously and randomly binds to exactly one of those messages**.

(μ)



(sk_Σ)

$\beta \leftarrow$ Commitment to μ $\qquad \xrightarrow{\beta}$

$\xleftarrow{\overline{\sigma}} \qquad \overline{\sigma} \leftarrow$ Signature on β

$\pi \leftarrow$ Proof of knowledge for $\mathscr{R}$ $\qquad \xrightarrow{\mu, \sigma := \pi} \qquad$ Verify π

Challenges

- How to commit to an exponential number of messages efficiently?
- How to have the signer obliviously select one of those messages?

How to commit to an exponential number of messages efficiently?

How to have the signer obliviously select one of those messages?

How to commit to an exponential number of messages efficiently?

User sets pk_U to be a commitment to some PRF key K.

How to have the signer obliviously select one of those messages?

How to commit to an exponential number of messages efficiently?

User sets pk_U to be a commitment to some PRF key K.

How to have the signer obliviously select one of those messages?

The signer samples the messages by selecting a random input r and signing it along with pk_U.

How to commit to an exponential number of messages efficiently?

User sets pk_U to be a commitment to some PRF key K.

How to have the signer obliviously select one of those messages?

The signer samples the messages by selecting a random input r and signing it along with pk_U.

The **final message is then** $\mathsf{F}_K(r)$ and the **signature is a proof of knowledge of the (pre)signature on** pk_U **and** r corresponding to this message.

Our generic NIBS compiler

(sk_S)

Our generic NIBS compiler

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{COM}}, \mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K))$

(sk_S)

Our generic NIBS compiler

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{COM}}, \mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K))$

(sk_S)

Our generic NIBS compiler

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{COM}}, \mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K))$

(sk_S)

$\}$ Issue

Our generic NIBS compiler

(crs, $\mathsf{pk}_{\mathsf{PKE}}$, vk_S, pk_U)

($\mathsf{sk}_U := K$, $\mathsf{sk}_{\mathsf{COM}}$, $\mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K)$)

(sk_S)

$$\left.r \leftarrow \{0,1\}^\lambda \right\} \text{Issue}$$

Our generic NIBS compiler

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{COM}}, \mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K))$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^{\lambda} \\ \overline{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\mathsf{pk}_U || r) \end{array}\right\} \text{Issue}$$

Our generic NIBS compiler

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{COM}}, \mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K))$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^{\lambda} \\ \overline{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\mathsf{pk}_U \,||\, r) \end{array}\right\} \text{Issue}$$

$$\xleftarrow{\overline{\sigma}, r}$$

Our generic NIBS compiler

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{COM}}, \mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K))$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^\lambda \\ \overline{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\mathsf{pk}_U || r) \end{array}\right\} \text{Issue}$$

$\overline{\sigma}, r$

Obtain $\{$

Our generic NIBS compiler

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{COM}}, \mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K))$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^{\lambda} \\ \overline{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\mathsf{pk}_U \,||\, r) \end{array}\right\} \text{Issue}$$

$\xleftarrow{\overline{\sigma}, r}$

$$\text{Obtain} \left\{ \quad \mu := \mathsf{F}_K(r) \right.$$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{COM}}, \mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K))$

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^\lambda \\ \overline{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\mathsf{pk}_U || r) \end{array}\right\} \text{Issue}$$

$\xleftarrow{\overline{\sigma}, r}$

$$\text{Obtain} \left\{\begin{array}{r} \mu := \mathsf{F}_K(r) \\ \pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}(\cdots) \end{array}\right.$$

Our generic NIBS compiler

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{COM}}, \mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K))$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^{\lambda} \\ \bar{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\mathsf{pk}_U \,||\, r) \end{array}\right\} \text{Issue}$$

$\xleftarrow{\bar{\sigma}, r}$

$$\text{Obtain} \left\{\begin{array}{r} \mu := \mathsf{F}_K(r) \\ \pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}(\cdots) \end{array}\right.$$

$\xrightarrow{\mu, \sigma := \pi}$

Our generic NIBS compiler

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{COM}}, \mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K))$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^{\lambda} \\ \bar{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\mathsf{pk}_U \,||\, r) \end{array}\right\} \text{Issue}$$

$\xleftarrow{\bar{\sigma}, r}$

$$\text{Obtain} \left\{\begin{array}{r} \mu := \mathsf{F}_K(r) \\ \pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}(\cdots) \end{array}\right.$$

$\xrightarrow{\mu, \sigma := \pi}$

$\mathsf{Verify}_{\mathsf{crs}}(\cdots)$

(One-more) Unforgeability

Strong blindness

(One-more) Unforgeability

From AoK property of the NIZK and existential unforgeability of the signature scheme under chosen messages.

Strong blindness

(One-more) Unforgeability

From AoK property of the NIZK and existential unforgeability of the signature scheme under chosen messages.

Strong blindness

Zero-knowledge of the NIZK, hiding of the commitment scheme and pseudo-randomness of F.

Our generic NIBS compiler

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{COM}}, \mathsf{pk}_U := \mathsf{Com}_{\mathsf{sk}_{\mathsf{COM}}}(K))$

(sk_S)

$$\left.\begin{array}{r} r \leftarrow \{0,1\}^\lambda \\ \bar{\sigma} \leftarrow \mathsf{Sign}_{\mathsf{sk}_S}(\mathsf{pk}_U \,||\, r) \end{array}\right\} \text{Issue}$$

$\xleftarrow{\quad \bar{\sigma}, r \quad}$

$$\text{Obtain} \left\{\begin{array}{r} \mu := \mathsf{F}_K(r) \\ \pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}(\cdots) \end{array}\right.$$

$\xrightarrow{\quad \mu, \sigma := \pi \quad}$

$\mathsf{Verify}_{\mathsf{crs}}(\cdots)$

Observation

Homomorphic encryption enables arbitrary homomorphic operations on the receiver's commitment.

An alternate construction from homomorphic encryption

How to commit to an exponential number of messages efficiently?

How to have the signer obliviously select one of those messages?

How to commit to an exponential number of messages efficiently?

User sets pk_U to be an (homomorphic) encryption of the PRF key K.

How to have the signer obliviously select one of those messages?

How to commit to an exponential number of messages efficiently?

User sets pk_U to be an (homomorphic) encryption of the PRF key K.

How to have the signer obliviously select one of those messages?

The signer homomorphically evaluates a signature on the message $\mathsf{F}_K(r)$ for some randomness r of its choice.

How to commit to an exponential number of messages efficiently?

User sets pk_U to be an (homomorphic) encryption of the PRF key K.

How to have the signer obliviously select one of those messages?

The signer homomorphically evaluates a signature on the message $\mathsf{F}_K(r)$ for some randomness r of its choice.

The **final message is then** $\mathsf{F}_K(r)$ and the **signature σ is an actual signature**, obtained by decrypting the (pre)signature.

(sk_S)

NIBS from HE

$$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$$

(sk_S)

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$

(sk_S)

NIBS from HE

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K), \mathsf{ek}_{\mathsf{HE}}}_{\psi})$

(sk_S)

$\}$ Issue

NIBS from HE

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

(sk_S)

$r \leftarrow \{0,1\}^\lambda$

$\left.\right\}$ Issue

NIBS from HE

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^{\lambda} \\ \bar{\sigma} \leftarrow \mathsf{Eval}_{\mathsf{ek}_{\mathsf{HE}}}(\psi, \mathsf{C}_{\mathsf{sk}_S, r}) \end{array}\right\} \text{Issue}$$

NIBS from HE

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^{\lambda} \\ \bar{\sigma} \leftarrow \mathsf{Eval}_{\mathsf{ek}_{\mathsf{HE}}}(\psi, \mathsf{C}_{\mathsf{sk}_S, r}) \end{array}\right\} \text{Issue}$$

where $\mathsf{C}_{\mathsf{sk}_S, r} := \mathsf{Sign}_{\mathsf{sk}_S}\left(\mathsf{F}_\circ(r)\right)$

NIBS from HE

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^\lambda \\ \bar{\sigma} \leftarrow \mathsf{Eval}_{\mathsf{ek}_{\mathsf{HE}}}(\psi, \mathsf{C}_{\mathsf{sk}_S, r}) \end{array}\right\} \text{Issue}$$

$\xleftarrow{\quad \bar{\sigma}, r \quad}$

where $\mathsf{C}_{\mathsf{sk}_S, r} := \mathsf{Sign}_{\mathsf{sk}_S}\left(\mathsf{F}_\circ(r)\right)$

NIBS from HE

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$

(sk_S)

$$\left.\begin{aligned} r &\leftarrow \{0,1\}^\lambda \\ \overline{\sigma} &\leftarrow \mathsf{Eval}_{\mathsf{ek}_{\mathsf{HE}}}(\psi, \mathsf{C}_{\mathsf{sk}_S, r}) \end{aligned}\right\} \text{Issue}$$

$\xleftarrow{\overline{\sigma},\, r}$

where $\mathsf{C}_{\mathsf{sk}_S, r} := \mathsf{Sign}_{\mathsf{sk}_S}\left(\mathsf{F}_\circ(r)\right)$

Obtain $\Big\{$

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^\lambda \\ \overline{\sigma} \leftarrow \mathsf{Eval}_{\mathsf{ek}_{\mathsf{HE}}}(\psi, \mathsf{C}_{\mathsf{sk}_S, r}) \end{array}\right\} \text{Issue}$$

$$\xleftarrow{\overline{\sigma}, r}$$

where $\mathsf{C}_{\mathsf{sk}_S, r} := \mathsf{Sign}_{\mathsf{sk}_S}\big(\mathsf{F}_\circ(r)\big)$

$$\text{Obtain} \left\{ \quad \mu := \mathsf{F}_K(r) \right.$$

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$

(sk_S)

$$\left.\begin{aligned} r &\leftarrow \{0,1\}^\lambda \\ \overline{\sigma} &\leftarrow \mathsf{Eval}_{\mathsf{ek}_{\mathsf{HE}}}(\psi, \mathsf{C}_{\mathsf{sk}_S, r}) \end{aligned}\right\} \text{Issue}$$

$\xleftarrow{\overline{\sigma}, r}$

where $\mathsf{C}_{\mathsf{sk}_S, r} := \mathsf{Sign}_{\mathsf{sk}_S}\big(\mathsf{F}_\circ(r)\big)$

$$\text{Obtain} \left\{\begin{aligned} \mu &:= \mathsf{F}_K(r) \\ \sigma &\leftarrow \mathsf{Dec}_{\mathsf{sk}_{\mathsf{HE}}}(\overline{\sigma}) \end{aligned}\right.$$

NIBS from HE

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^\lambda \\ \overline{\sigma} \leftarrow \mathsf{Eval}_{\mathsf{ek}_{\mathsf{HE}}}(\psi, \mathsf{C}_{\mathsf{sk}_S, r}) \end{array}\right\} \text{Issue}$$

where $\mathsf{C}_{\mathsf{sk}_S, r} := \mathsf{Sign}_{\mathsf{sk}_S}\big(\mathsf{F}_\circ(r)\big)$

$$\xleftarrow{\quad \overline{\sigma}, r \quad}$$

$$\text{Obtain} \left\{\begin{array}{r} \mu := \mathsf{F}_K(r) \\ \sigma \leftarrow \mathsf{Dec}_{\mathsf{sk}_{\mathsf{HE}}}(\overline{\sigma}) \end{array}\right.$$

$$\xrightarrow{\quad \mu, \sigma \quad}$$

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$

(sk_S)

$$\left.\begin{array}{l} r \leftarrow \{0,1\}^\lambda \\ \overline{\sigma} \leftarrow \mathsf{Eval}_{\mathsf{ek}_{\mathsf{HE}}}(\psi, \mathsf{C}_{\mathsf{sk}_S, r}) \end{array}\right\} \text{Issue}$$

$\xleftarrow{\overline{\sigma}, r}$

where $\mathsf{C}_{\mathsf{sk}_S, r} := \mathsf{Sign}_{\mathsf{sk}_S}\left(\mathsf{F}_\circ(r)\right)$

$$\text{Obtain} \left\{\begin{array}{r} \mu := \mathsf{F}_K(r) \\ \sigma \leftarrow \mathsf{Dec}_{\mathsf{sk}_{\mathsf{HE}}}(\overline{\sigma}) \end{array}\right.$$

$\xrightarrow{\mu, \sigma}$

$\mathsf{Verify}_{\mathsf{vk}_S}(\mu, \sigma)$

NIBS from HE

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := K, \mathsf{sk}_{\mathsf{HE}}, \mathsf{pk}_U := \underbrace{\mathsf{Enc}_{\mathsf{ek}_{\mathsf{HE}}}(K)}_{\psi}, \mathsf{ek}_{\mathsf{HE}})$

(sk_S)

Issue

$$r \leftarrow \{0,1\}^{\lambda}$$

$$\widehat{\psi} \leftarrow \mathsf{Eval}_{\mathsf{ek}_{\mathsf{HE}}}(\psi, \mathsf{C}_{\mathsf{sk}_S, r})$$

where $\mathsf{C}_{\mathsf{sk}_S, r} := \mathsf{Sign}_{\mathsf{sk}_S}\left(\mathsf{F}_{\circ}(r)\right)$

$$\pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}(\cdots)$$

$$\xleftarrow{\ \bar{\sigma} := (\widehat{\psi}, \pi), r\ }$$

Obtain

$$\mathsf{Verify}_{\mathsf{crs}}(\cdots)$$

$$\mu := \mathsf{F}_K(r)$$

$$\sigma \leftarrow \mathsf{Dec}_{\mathsf{sk}_{\mathsf{HE}}}(\widehat{\psi})$$

$$\xrightarrow{\ \mu, \sigma\ }$$

$$\mathsf{Verify}_{\mathsf{vk}_S}(\mu, \sigma)$$

(One-more) Unforgeability

Strong blindness

(One-more) Unforgeability

Zero-knowledge of the NIZK, existential unforgeability of the signature scheme under chosen messages and circuit-privacy of HE.

Strong blindness

(One-more) Unforgeability

Zero-knowledge of the NIZK, existential unforgeability of the signature scheme under chosen messages and circuit-privacy of HE.

Strong blindness

From AoK property of the NIZK, CPA security of HE and pseudo-randomness of F.

(One-more) Unforgeability

Zero-knowledge of the NIZK, existential unforgeability of the signature scheme under chosen messages and circuit-privacy of HE.

Strong blindness

From AoK property of the NIZK, CPA security of HE and pseudo-randomness of F.

Remark

Can be instantiated from standard lattice assumptions, giving a first theoretical construction for post-quantum secure NIBS.

Construction	$\lvert \mathsf{pk}_U \rvert$	$\lvert \overline{\sigma} \rvert$	$\lvert \sigma \rvert$	Blindness
Circuit-private LHE	poly(λ)	poly(λ)	$<$ 1 KB	Strong
General-purpose NIZK	poly(λ)	$<$ 1 KB	poly(λ)	Strong
Lattice-based (rOM-ISIS)	1.6 KB	$<$ 1 KB	68 KB	Weak/one-time

Table. Public key, transcript and signature sizes of our constructions.

Identify an issue with the existing definition and give the **right definition for blindness of NIBS** (and a new correctness notion).

Identify an issue with the existing definition and give the **right definition for blindness of NIBS** (and a new correctness notion).

A **Fishclin-like compiler for NIBS** and prove security in our baseline setting.

Identify an issue with the existing definition and give the **right definition for blindness of NIBS** (and a new correctness notion).

A **Fishclin-like compiler for NIBS** and prove security in our baseline setting.

A **generic construction for NIBS from leveled homomorphic encryption** and prove security in our baseline setting.

Summary of our results

Identify an issue with the existing definition and give the **right definition for blindness of NIBS** (and a new correctness notion).

A **Fishclin-like compiler for NIBS** and prove security in our baseline setting.

A **generic construction for NIBS from leveled homomorphic encryption** and prove security in our baseline setting.

Construction from a (non-standard) lattice assumption called rOM-ISIS which satisfies the weaker one-time blindness.

Future work

- Efficient post-quantum secure NIBS with baseline security.
- Formal cryptanalysis of the rOM-ISIS assumption.
- NIBS from pairing free assumptions.
- Other models for non-interactive signing.

Solved in an upcoming work—NIBS from MLWE/MSIS + ISIS_f

- Formal cryptanalysis of the **rOM-ISIS** assumption.
- NIBS from pairing free assumptions.
- Other models for non-interactive signing.

Thank you.

Full version: ia.cr/2024/614

Lattice-based NIBS

from our generic compiler

- An **efficient lattice-based NIBS scheme** that is secure (blind) under the definition of [Han23]*.
- The **final signature size is 68 KB** (total communication $\sim$70 KB). The current state-of-the-art (interactive) lattice-based blind signature scheme [BLNS23] has signature around 22 KB (but total communication is 100+ KB).
- Our security proof relies on a new lattice assumption that we call the randomized one-more ISIS assumption (rOM-ISIS). rOM-ISIS is a more robust variant of the one-more ISIS assumption due to Agrawal, et al. [AKSY22]. We also provide some high-level cryptanalysis to show that rOM-ISIS is (likely) at least as hard as OM-ISIS.

Lattice-based NIBS

from our generic compiler

$(\mathsf{crs}, \mathsf{pk}_{\mathsf{PKE}}, \mathbf{A}, \mathbf{B}, \mathsf{vk}_S, \mathsf{pk}_U)$

$(\mathsf{sk}_U := (\mathbf{x}, \delta), \mathsf{pk}_U := \mathbf{A} \cdot \mathbf{x} + \mathsf{H}(\delta))$

$(\mathsf{sk}_S := \mathbf{T}_{\mathbf{C}}, \mathsf{vk}_S := \mathbf{C})$

$$\left.\begin{array}{l} \mathbf{y} \leftarrow \pm 1 \\ \mathbf{z} \leftarrow \mathbf{C}^{-1}(\mathsf{pk}_U - \mathbf{B} \cdot \mathbf{y}) \end{array}\right\} \text{Issue}$$

$\xleftarrow{\mathbf{z}, \mathbf{y}}$

$$\text{Obtain} \left\{\begin{array}{r} \mathbf{m} := \mathsf{A}_{\mathsf{L}} \cdot \mathbf{x}_{\perp} + \mathsf{A}_{\mathsf{R}} \cdot \mathbf{z}_{\perp} \\ \psi \leftarrow \mathsf{Enc}_{\mathsf{pk}_{\mathsf{PKE}}}(\mathbf{x} \,||\, \mathbf{y} \,||\, \mathbf{z}) \\ \pi \leftarrow \mathsf{Prove}_{\mathsf{crs}}(\cdots) \end{array}\right.$$

$\xrightarrow{\mu := (\mathbf{m}, \delta), \sigma := (\psi, \pi)}$

$\mathsf{Verify}_{\mathsf{crs}}(\cdots)$

CP-LHE

3.4.6 Leveled homomorphic encryption

Let $\mathcal{C}_d$ denote the class of boolean valued circuits of depth d. A leveled homomorphic encryption scheme $\mathcal{LHE}$ with message space $\{0,1\}$ for circuit class $\{\mathcal{C}_d\}_{d\in\mathbb{N}}$ consists of the following polynomial time algorithms:

$\mathsf{Setup}(1^\lambda, 1^d) \to (\mathsf{sk}, \mathsf{ek})$ The setup algorithm takes as input the security parameter λ, bound on circuit depth d and outputs a secret key sk and evaluation key ek.

$\mathsf{Enc}(\mathsf{sk}, m \in \{0,1\}) \to \mathsf{ct}$ The encryption algorithm takes as input a secret key sk, message $m \in \{0,1\}$ and outputs a ciphertext ct.

$\mathsf{Eval}(\mathsf{ek}, C \in \mathcal{C}_d, \mathbf{ct}) \to \mathsf{ct}'$ The evaluation algorithm takes as input an evaluation key ek, a circuit $C \in \mathcal{C}_d$, a sequence of ciphertexts $\mathbf{ct} = (\mathsf{ct}_1, \ldots, \mathsf{ct}_\ell)$ for some $\ell > 0$ and outputs a ciphertext $\widehat{\mathsf{ct}}$. Here ℓ denotes the input length of C.

$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \to x$ The decryption algorithm takes as input a secret key sk and ciphertext ct and outputs $x \in \{0,1\} \cup \{\perp\}$.

Correctness. The scheme $\mathcal{LHE}$ is said to be (perfectly) correct if for all security parameter λ, circuit-depth bound d, $(\mathsf{sk}, \mathsf{ek}) \leftarrow \mathsf{Setup}(1^\lambda, 1^d)$, circuit $C \in \mathcal{C}_d$ and messages $m_1, \ldots, m_\ell \in \{0,1\}$, every ciphertext $\mathsf{ct}_i \leftarrow\!\!\$\ \mathsf{Enc}(\mathsf{sk}, m_i)$ where ℓ denotes input length of C, the following holds:

$$\Pr\left[\mathsf{Dec}(\mathsf{sk}, \mathsf{Eval}(\mathsf{ek}, C, (\mathsf{ct}_1, \ldots, \mathsf{ct}_\ell)) = C(m_1, \ldots, m_\ell)\right] = 1.$$

Definition 3.19 (Circuit privacy)**.** An $\mathcal{LHE}$ scheme is said to be circuit private if there exists a PPT algorithm Sim such that for every $d \in \mathbb{N}$ any circuit $C \in \mathcal{C}_d$ with input length $\ell = \mathsf{poly}(\lambda)$, and any sequence of message bits $m_1, \ldots, m_\ell \in \{0,1\}$, the following holds:

$$(\mathsf{ek}, \mathsf{Eval}(\mathsf{ek}, C, (\mathsf{ct}_1, \ldots, \mathsf{ct}_\ell)), \mathsf{ct}_1, \ldots, \mathsf{ct}_\ell) \approx_c \left(\mathsf{ek}, \widehat{\mathsf{ct}}, \mathsf{ct}_1, \ldots, \mathsf{ct}_\ell\right)$$

where $(\mathsf{sk}, \mathsf{ek}) \leftarrow\!\!\$\ \mathsf{Setup}(1^\lambda, 1^d)$, $\mathsf{ct}_i \leftarrow\!\!\$\ \mathsf{Enc}(\mathsf{sk}, m_i)\ \forall i \in [\ell]$, $\widehat{\mathsf{ct}} = \mathsf{Sim}(\mathsf{ek}, C(m_1, \ldots, m_\ell), \mathsf{ct}_1, \ldots, \mathsf{ct}_\ell)$.

rOM-ISIS

1. The challenger samples a challenge matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and a randomization matrix $\mathbf{B} \in \mathbb{Z}_q^{n \times m}$ along with a large set of random target vectors $T \subset \mathbb{Z}_q^n$. It provides the attacker with $\mathbf{A}$, $\mathbf{B}$ and the vector set T.

2. $\mathcal{A}$ can make preimage queries for any target vector $\widehat{\mathbf{t}} \in \mathbb{Z}_q^n$ such that the challenger replies with a short vector $\widehat{\mathbf{x}}$ and a ± 1 vector $\widehat{\mathbf{y}} \in \{\pm 1\}^m$ such that $\mathbf{A} \cdot \widehat{\mathbf{x}} + \mathbf{B} \cdot \widehat{\mathbf{y}} = \widehat{\mathbf{t}}$.

3. rOM-ISIS assumption says that $\mathcal{A}$ cannot output $\ell + 1$ distinct vector tuples $\left\{(\mathbf{x}_j, \mathbf{y}_j, \mathbf{t}_j)\right\}_{j \in [\ell+1]}$ such that $\mathbf{A} \cdot \mathbf{x}_j + \mathbf{B} \cdot \mathbf{y}_j = \mathbf{t}_j$, $\mathbf{t}_j \in T$, $\mathbf{x}_j$ is sufficiently short, $\mathbf{y}_j$ is a ± 1 vector, *and* $\mathcal{A}$ made at most ℓ preimage queries.

Intuitively, the attacker now cannot truly select the preimage vector arbitrarily since the challenger randomizes the *actual* target vector as $(\mathbf{t} - \mathbf{B} \cdot \mathbf{y})$, where $\mathbf{y}$ is a random ± 1 vector. Since the attacker receives the vector $\widehat{\mathbf{y}}$ used for randomization, it is unclear whether we can reduce it to the standard ISIS assumption.[9] However, our preliminary cryptanalysis (cf. § 6.1) shows that it is more robust when compared with the OM-ISIS assumption. We believe that this new formulation could serve as a better lattice analogue of the one-more RSA assumption [BNPS03]. For example, we can also prove that a mild adaptation of the Agrawal et al. [AKSY22] two-round blind signature scheme is still secure under rOM-ISIS assumption, and now we no longer have set the parameters as carefully to avoid simple attacks as was done in [AKSY22]. This further illustrates the flexibility of our new assumption. Later, in Section 6, we describe the assumption in full detail and also provide some preliminary cryptanalysis.

NIBS

Definition 4.2 (Reusability). A NIBS scheme $\mathcal{S}$ satisfies the reusability property, if there exists a negligible function $\mathsf{negl}(\cdot)$ such that for every $\lambda \in \mathbb{N}$, the following holds:

$$\Pr\left[\begin{array}{l} \mathsf{nonce}_0 = \mathsf{nonce}_1 \\ \vee\ \mu_0 = \mu_1 \end{array} : \begin{array}{r} \mathsf{pp} \leftarrow\!\!\$\ \mathsf{Setup}(1^\lambda) \\ (\mathsf{sk}, \mathsf{vk}) \leftarrow\!\!\$\ \mathsf{KeyGen}_S(\mathsf{pp}), (\mathsf{sk}_R, \mathsf{pk}_R) \leftarrow\!\!\$\ \mathsf{KeyGen}_R(\mathsf{pp}) \\ \forall b \in \{0,1\} : (\mathsf{psig}_b, \mathsf{nonce}_b) \leftarrow\!\!\$\ \mathsf{Issue}(\mathsf{sk}, \mathsf{pk}_R) \\ \forall b \in \{0,1\} : (\mu_b, \sigma_b) \leftarrow\!\!\$\ \mathsf{Obtain}(\mathsf{sk}_R, \mathsf{vk}, (\mathsf{psig}_b, \mathsf{nonce}_b)) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Definition 4.3 (One-more unforgeability). A NIBS scheme $\mathcal{S}$ satisfies one-more unforgeability, if for every *stateful admissible* PPT adversary $\mathcal{A}$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for every $\lambda \in \mathbb{N}$, the following holds:

$$\Pr\left[\begin{array}{l} \bigwedge_{i \in [\ell+1]} \mathsf{Verify}(\mathsf{vk}, \mu_i, \sigma_i) = 1 \\ \wedge \left(\bigwedge_{i \neq j \in [\ell+1]} \mu_i \neq \mu_j\right) \end{array} : \begin{array}{r} \mathsf{pp} \leftarrow\!\!\$\ \mathsf{Setup}(1^\lambda) \\ (\mathsf{sk}, \mathsf{vk}) \leftarrow\!\!\$\ \mathsf{KeyGen}_S(\mathsf{pp}) \\ \{(\mu_i, \sigma_i)\}_{i=1}^{\ell+1} \leftarrow\!\!\$\ \mathcal{A}^{O_{\mathsf{sk}}(\cdot)}(\mathsf{vk}) \end{array}\right] \leq \mathsf{negl}(\lambda),$$

where $O_{\mathsf{sk}}(\cdot)$ takes as input a receiver's public key pk_{R_i}, and outputs a presignature-nonce pair $(\mathsf{psig}_i, \mathsf{nonce}_i)$ by running $\mathsf{Issue}(\mathsf{sk}, \mathsf{pk}_{R_i})$, and $\mathcal{A}$ is an admissible adversary iff $\mathcal{A}$ makes at most ℓ queries to O_{sk}.

Definition 4.4 (Strong receiver blindness). A NIBS scheme $\mathcal{S}$ satisfies *strong* receiver blindness, if for every *stateful admissible* PPT adversary $\mathcal{A}$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for every $\lambda \in \mathbb{N}$, the following holds:

$$\Pr\left[\begin{array}{l} \mathcal{A}^{O_{\mathsf{sk}_{R_0},\mathsf{sk}_{R_1}}(\cdot,\cdot,\cdot)}(\mu_{\hat{b}},\sigma_{\hat{b}},\mu_{1-\hat{b}},\sigma_{1-\hat{b}}) = \hat{b} : \\ \hfill \mathsf{pp} \leftarrow\$ \mathsf{Setup}(1^\lambda),\ \hat{b} \leftarrow\$ \{0,1\}, \\ \hfill \forall b \in \{0,1\}:\ (\mathsf{sk}_{R_b},\mathsf{pk}_{R_b}) \leftarrow\$ \mathsf{KeyGen}_R(\mathsf{pp}) \\ \hfill (\mathsf{vk},(\mathsf{psig}_b,\mathsf{nonce}_b)_b) \leftarrow\$ \mathcal{A}^{O_{\mathsf{sk}_{R_0},\mathsf{sk}_{R_1}}(\cdot,\cdot,\cdot)}(\mathsf{pk}_{R_0},\mathsf{pk}_{R_1}) \\ \hfill \forall b \in \{0,1\}:\ (\mu_b,\sigma_b) \leftarrow\$ \mathsf{Obtain}(\mathsf{sk}_{R_b},\mathsf{vk},(\mathsf{psig}_b,\mathsf{nonce}_b)) \end{array}\right] \le \frac{1}{2} + \mathsf{negl}(\lambda),$$

where oracle $O_{\mathsf{sk}_{R_0},\mathsf{sk}_{R_1}}$, on the i-th query $(b^{(i)},\mathsf{vk}^{(i)},(\mathsf{psig}^{(i)},\mathsf{nonce}^{(i)}))$, outputs $\mathsf{Obtain}(\mathsf{sk}_{R_{b^{(i)}}},\mathsf{vk}^{(i)},(\mathsf{psig}^{(i)},\mathsf{nonce}^{(i)}))$. That is, $O_{\mathsf{sk}_{R_0},\mathsf{sk}_{R_1}}$ provides $\mathcal{A}$ oracle access to the Obtain algorithm w.r.t. $\mathsf{sk}_{R_0},\mathsf{sk}_{R_1}$. We say that $\mathcal{A}$ is an admissible adversary iff:

- $\sigma_0,\sigma_1 \neq \perp$ (i.e., Obtain algorithm does not abort), and
- $\mathsf{nonce}_0 \neq \mathsf{nonce}^{(i)}$ and $\mathsf{nonce}_1 \neq \mathsf{nonce}^{(i)}$ for all i. (That is, $\mathcal{A}$ cannot make an Obtain query with nonce value to be either of the challenge nonce values.)

Definition 4.5 (Strong nonce blindness). A NIBS scheme $\mathcal{S}$ satisfies nonce blindness, if for every *stateful admissible* PPT adversary $\mathcal{A}$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for every $\lambda \in \mathbb{N}$, the following holds:

$$\Pr\left[\begin{array}{r} \mathcal{A}^{O_{\mathsf{sk}_R}(\cdot,\cdot)}(\mu_{\hat{b}}, \sigma_{\hat{b}}, \mu_{1-\hat{b}}, \sigma_{1-\hat{b}}) = \hat{b} : \hspace{6cm} \\ \mathsf{pp} \leftarrow\!\!\$\ \mathsf{Setup}(1^\lambda),\ (\mathsf{sk}_R, \mathsf{pk}_R) \leftarrow\!\!\$\ \mathsf{KeyGen}_R(\mathsf{pp}) \\ (\mathsf{vk}, (\mathsf{psig}_b, \mathsf{nonce}_b)_b) \leftarrow\!\!\$\ \mathcal{A}^{O_{\mathsf{sk}_R}(\cdot,\cdot)}(\mathsf{pk}_R),\ \hat{b} \leftarrow\!\!\$\ \{0,1\} \\ \forall b \in \{0,1\}:\ (\mu_b, \sigma_b) \leftarrow\!\!\$\ \mathsf{Obtain}(\mathsf{sk}_R, \mathsf{vk}, (\mathsf{psig}_b, \mathsf{nonce}_b)) \end{array}\right] \le \frac{1}{2} + \mathsf{negl}(\lambda),$$

where oracle O_{sk_R}, on the i-th query $(\mathsf{vk}^{(i)}, (\mathsf{psig}^{(i)}, \mathsf{nonce}^{(i)}))$, outputs $\mathsf{Obtain}(\mathsf{sk}_R, \mathsf{vk}^{(i)}, (\mathsf{psig}^{(i)}, \mathsf{nonce}^{(i)}))$. That is, O_{sk_R} provides $\mathcal{A}$ oracle access to the Obtain algorithm w.r.t. sk_R. We say that $\mathcal{A}$ is an admissible adversary iff:

- $\sigma_0, \sigma_1 \neq \perp$ (i.e., Obtain algorithm does not abort), and
- $\mathsf{nonce}_0 \neq \mathsf{nonce}^{(i)}$ and $\mathsf{nonce}_1 \neq \mathsf{nonce}^{(i)}$ for all i. (That is, $\mathcal{A}$ cannot make an Obtain query with nonce value to be either of the challenge nonce values.)