Hybrid Post-Quantum Password Authenticated Key Exchange

draft-vos-cfrg-pqpake

https://eprint.iacr.org/2025/1343.pdf

IETF 124 - CFRG | Jelle Vos

Why an aPAKE?

Recap

Please send the contents of my vault

Why an aPAKE?

Recap

Please send the contents of my vault

Why an aPAKE?

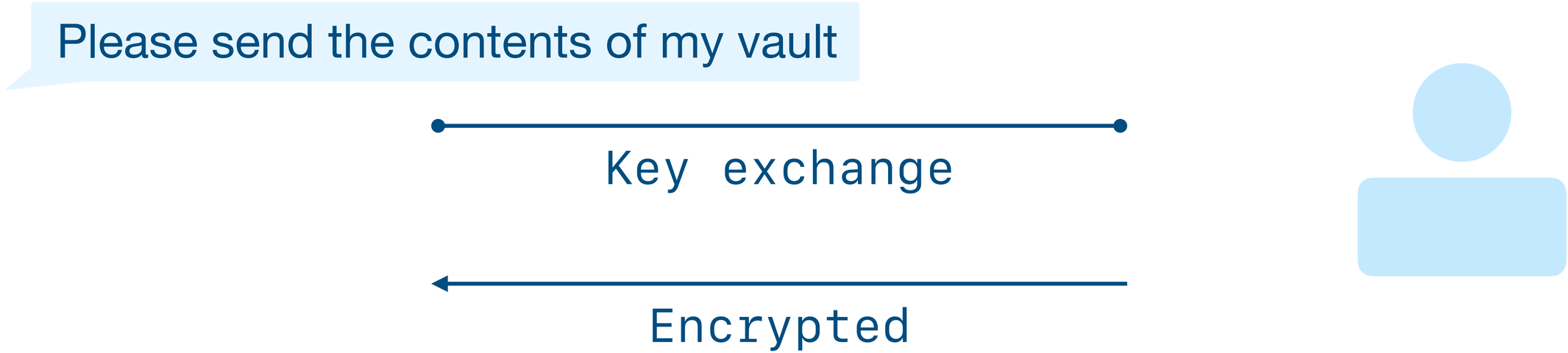
Recap

Please send the contents of my vault

Why an aPAKE?

Recap

Please send the contents of my vault



Why an aPAKE?

Recap

Please send the contents of my vault

No authentication

Why an aPAKE?

Recap

Please send the contents of my vault

Knows password

Password-authenticated key exchange

Encrypted

Knows password

Why an aPAKE?

Recap

Please send the contents of my vault

Knows password

PAKE

Encrypted

Knows password

Why an aPAKE?

Recap

Please send the contents of my vault

Knows password

PAKE

Encrypted

Knows password

Can impersonate the user!

Why an aPAKE?

Recap

Please send the contents of my vault

Knows password

aPAKE

Encrypted

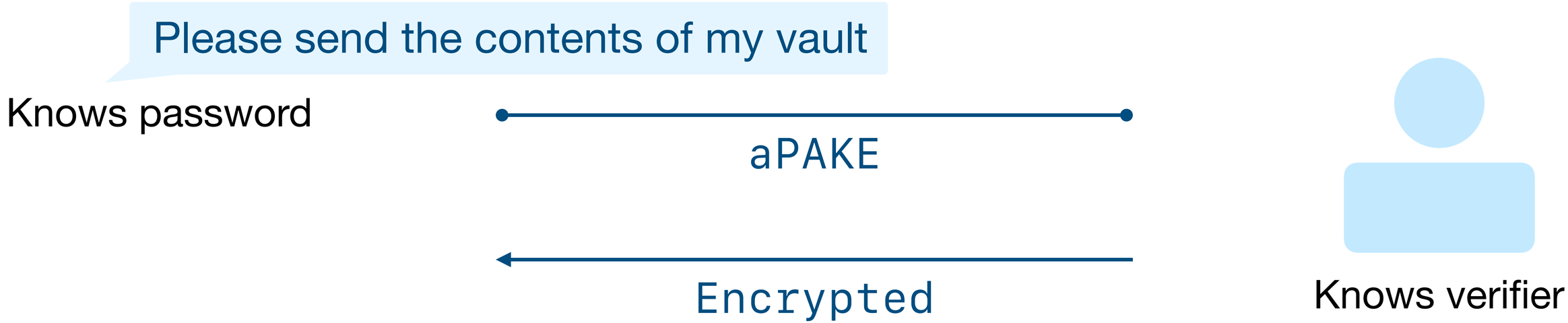
Knows verifier

Why a *post-quantum* aPAKE?

Recap

Please send the contents of my vault

Knows password



Knows verifier

Why a *post-quantum* aPAKE?

Recap

Please send the contents of my vault

Knows password

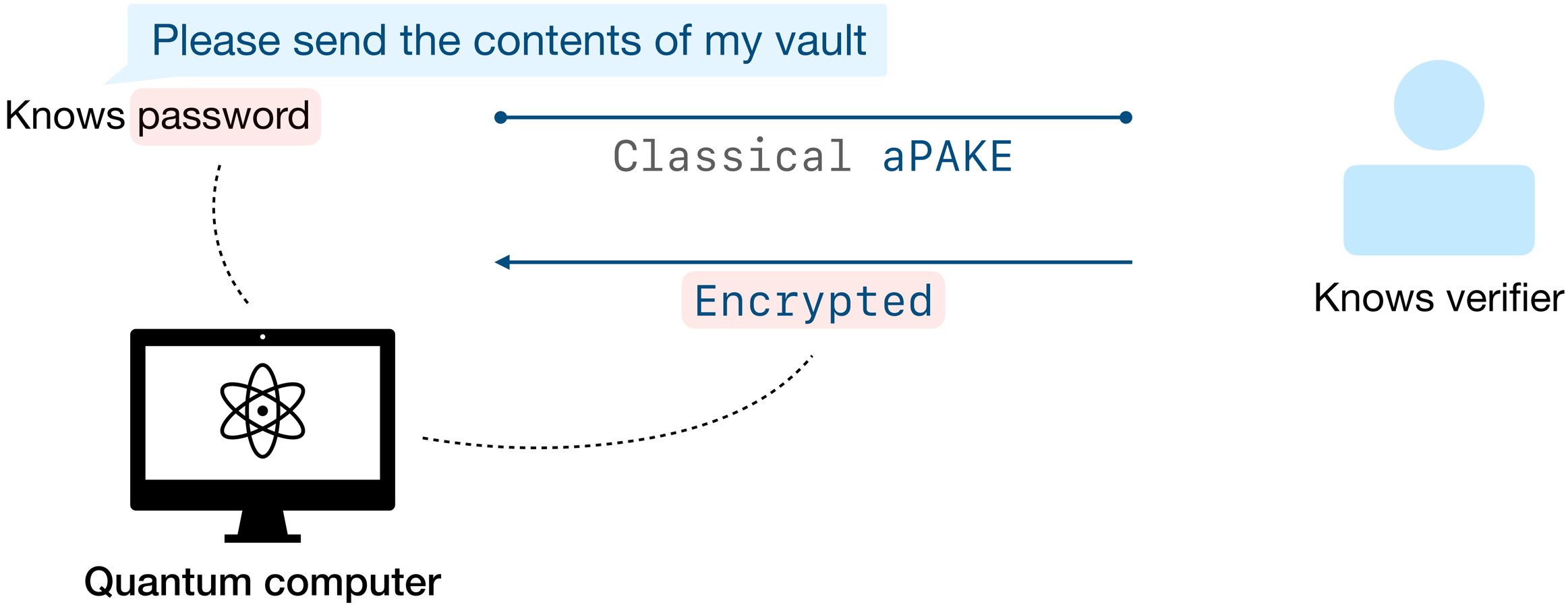
Classical aPAKE

Encrypted

Knows verifier

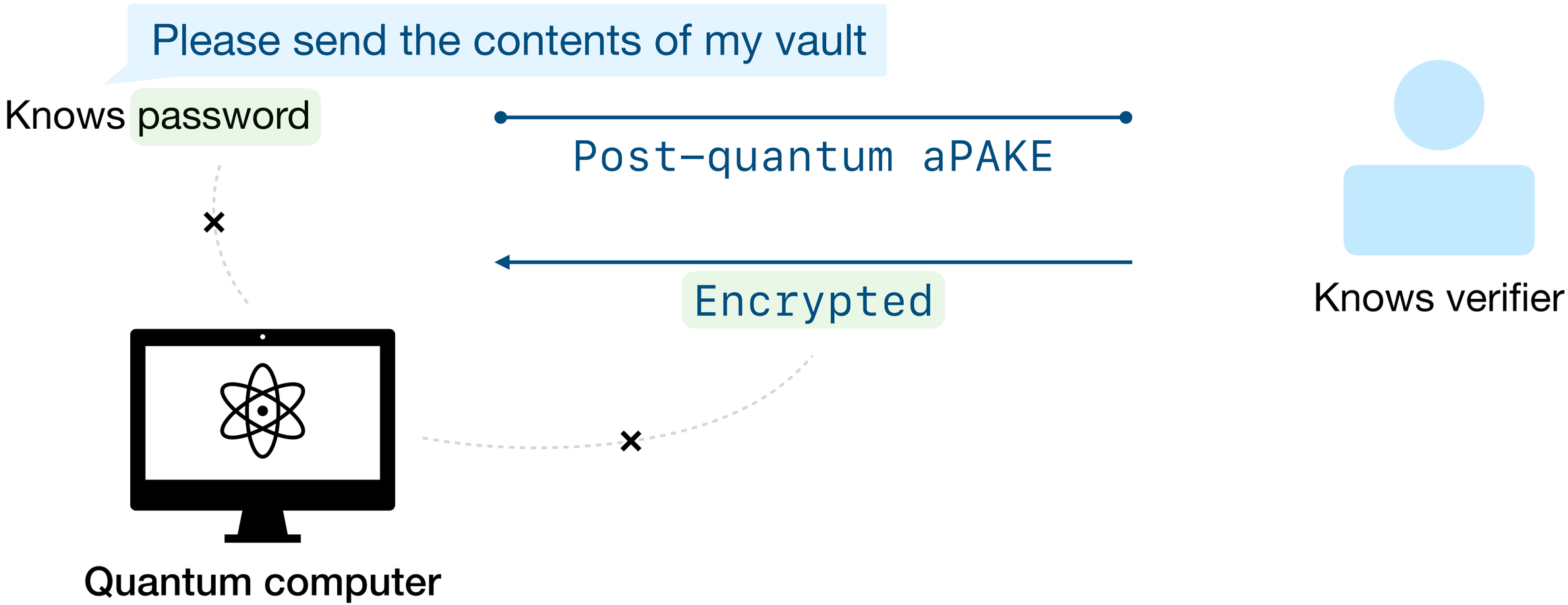
Why a *post-quantum* aPAKE?

Recap



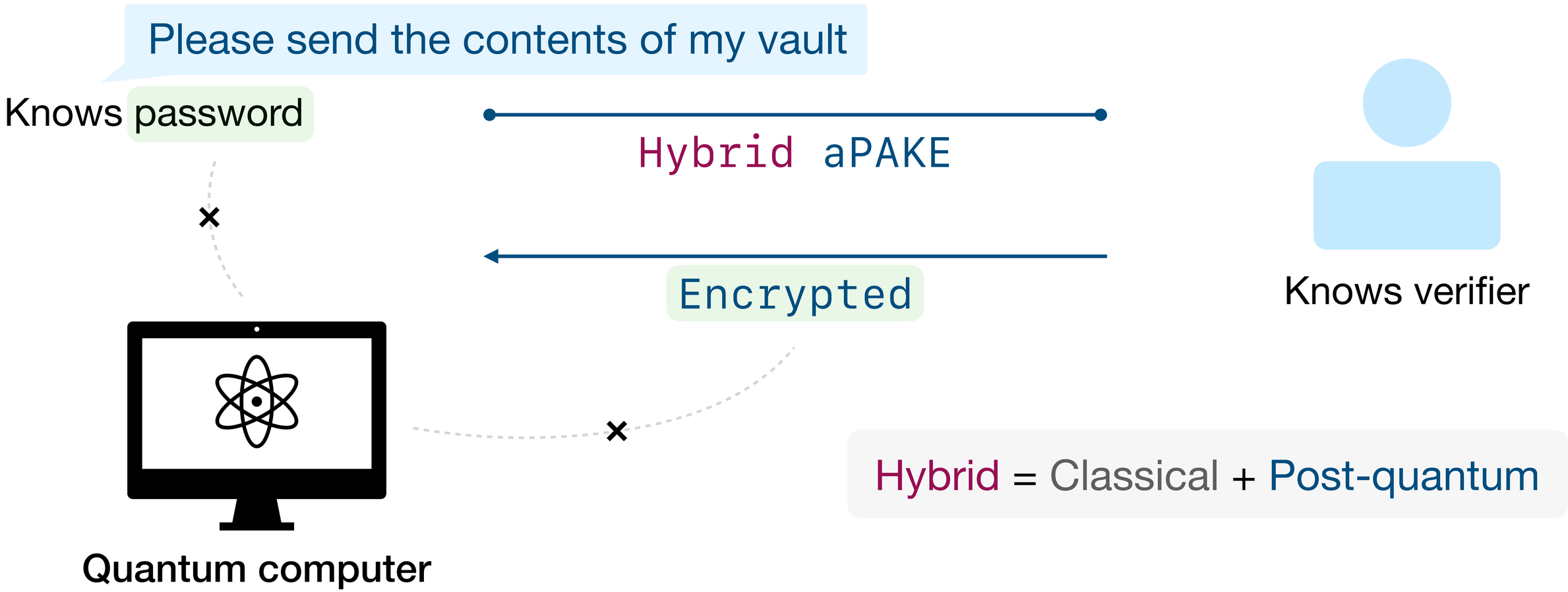
Why a *post-quantum* aPAKE?

Recap



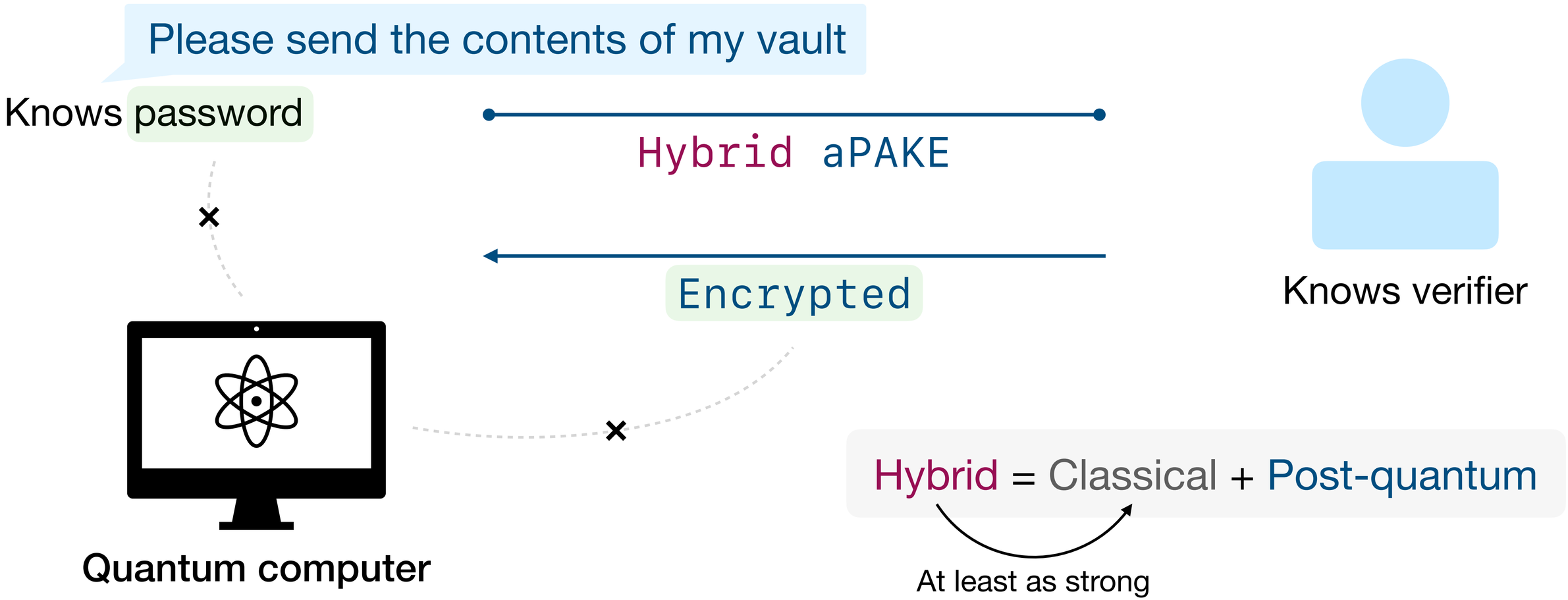
Why a *hybrid* aPAKE?

Recap



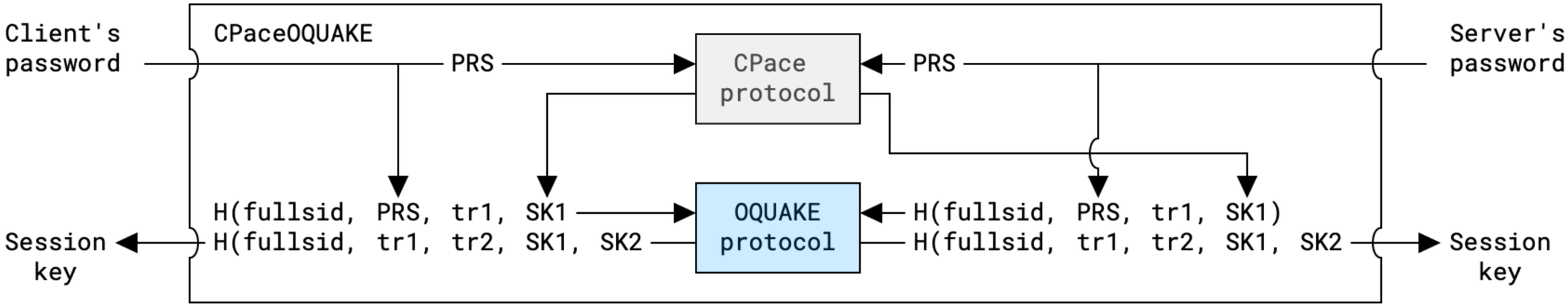
Why a *hybrid* aPAKE?

Recap



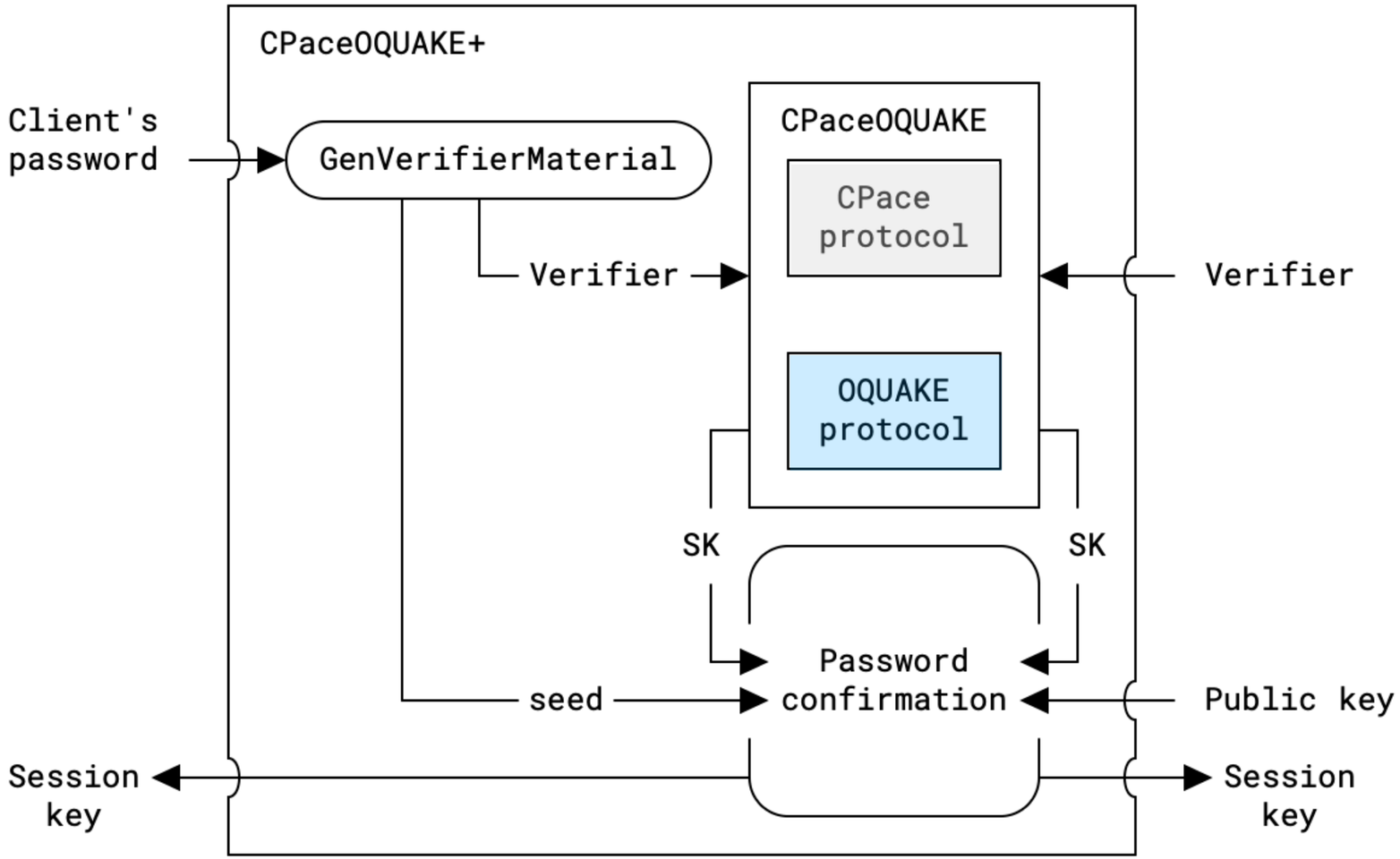
How to design a hybrid PAKE?

Recap



How to design a hybrid aPAKE?

Recap



A post-quantum vs. hybrid aPAKE

Recap

OQUAKE

Post-quantum PAKE

CPaceOQUAKE

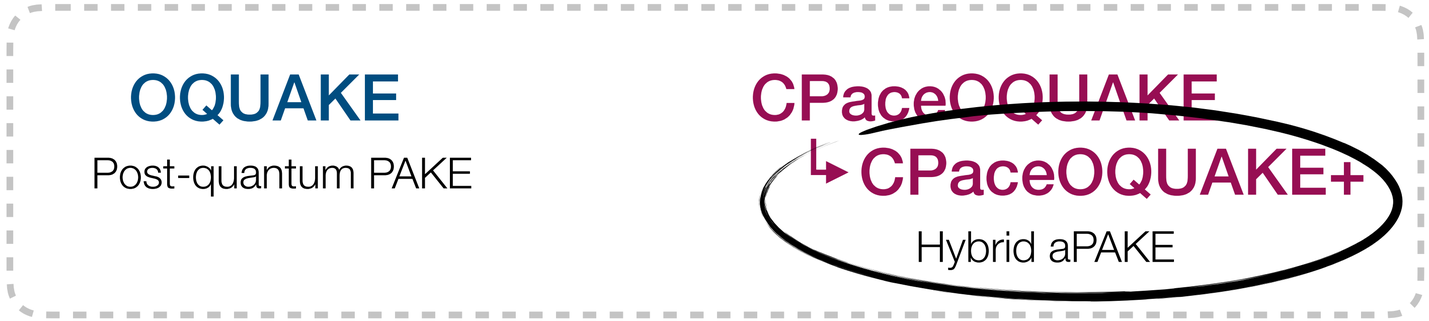
↳ **CPaceOQUAKE+**

Hybrid aPAKE

Current specification

A post-quantum vs. hybrid aPAKE

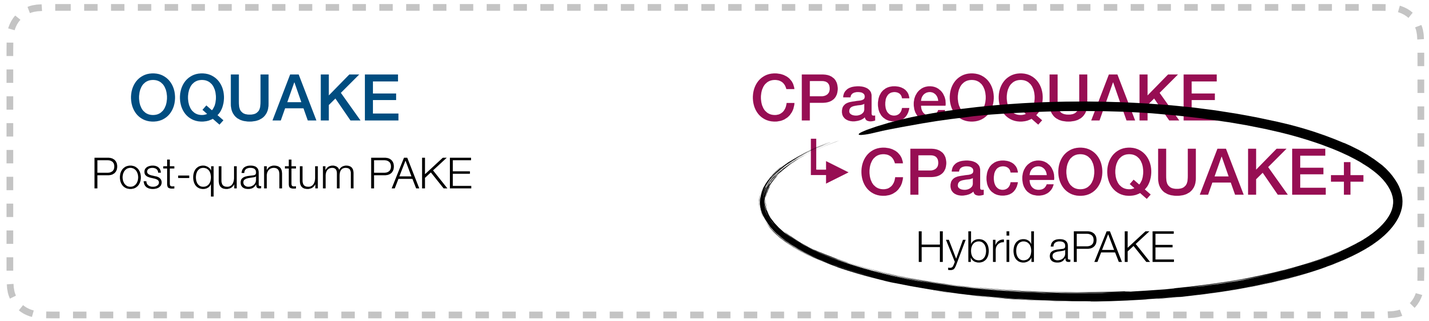
Recap



Current specification

A post-quantum vs. hybrid aPAKE

Recap



Current specification

Should we focus on a hybrid aPAKE, or also offer a post-quantum aPAKE?

A post-quantum vs. hybrid aPAKE

Recap

OQUAKE

Post-quantum PAKE

CPaceOQUAKE

↳ **CPaceOQUAKE+**

Hybrid aPAKE

Should we focus on a hybrid aPAKE, or also offer a post-quantum aPAKE?

A post-quantum vs. hybrid aPAKE

Recap

OQUAKE
↳ **???**

Post-quantum aPAKE

CPaceOQUAKE
↳ **CPaceOQUAKE+**

Hybrid aPAKE

Should we focus on a hybrid aPAKE, or also offer a post-quantum aPAKE?

A post-quantum vs. hybrid aPAKE

Recap

OQUAKE
↳ **OQUAKE+**

Post-quantum aPAKE

CPaceOQUAKE
↳ **CPaceOQUAKE+**

Hybrid aPAKE

Should we focus on a hybrid aPAKE, or also offer a post-quantum aPAKE?

A post-quantum vs. hybrid aPAKE

Recap

	OQUAKE ↳ OQUAKE+ Post-quantum aPAKE	CPaceOQUAKE ↳ CPaceOQUAKE+ Hybrid aPAKE
At least as strong as CPace	✗	✓
Quantum resistant	✓	✓

Should we focus on a hybrid aPAKE, or also offer a post-quantum aPAKE?

A post-quantum vs. hybrid aPAKE

Recap

	OQUAKE ↳ OQUAKE+ Post-quantum aPAKE	CPaceOQUAKE ↳ CPaceOQUAKE+ Hybrid aPAKE
At least as strong as CPace	✗	✓
Quantum resistant	✓	✓
Number of messages	3	5
Bandwidth	~1 ML-KEM key exchange + 64 bytes	~1 ML-KEM key exchange + 128 bytes

Should we focus on a hybrid aPAKE, or also offer a post-quantum aPAKE?

A post-quantum vs. hybrid aPAKE

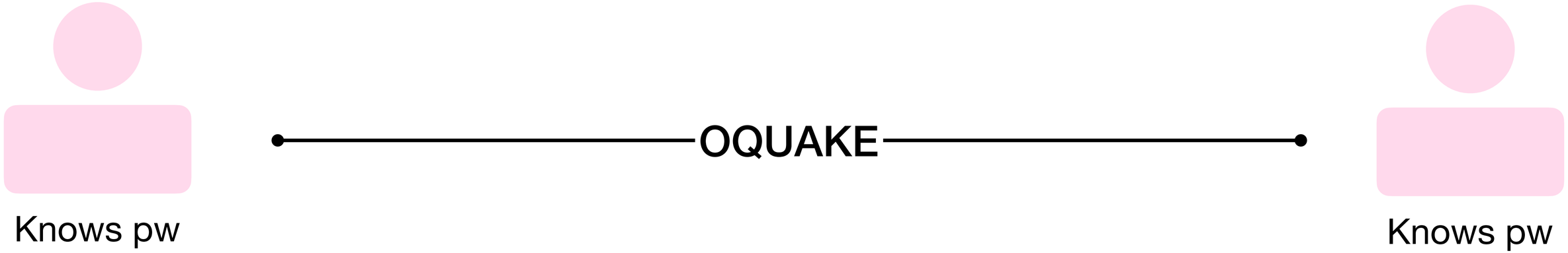
Recap

	OQUAKE ↳ OQUAKE+ Post-quantum aPAKE	CPaceOQUAKE ↳ CPaceOQUAKE+ Hybrid aPAKE
At least as strong as CPace		
Quantum resistant		
Number of messages	3	
Bandwidth	~1 ML-KEM key exchange + 64 bytes	~1 ML-KEM key exchange + 128 bytes

Should we focus on a hybrid aPAKE, or also offer a post-quantum aPAKE?

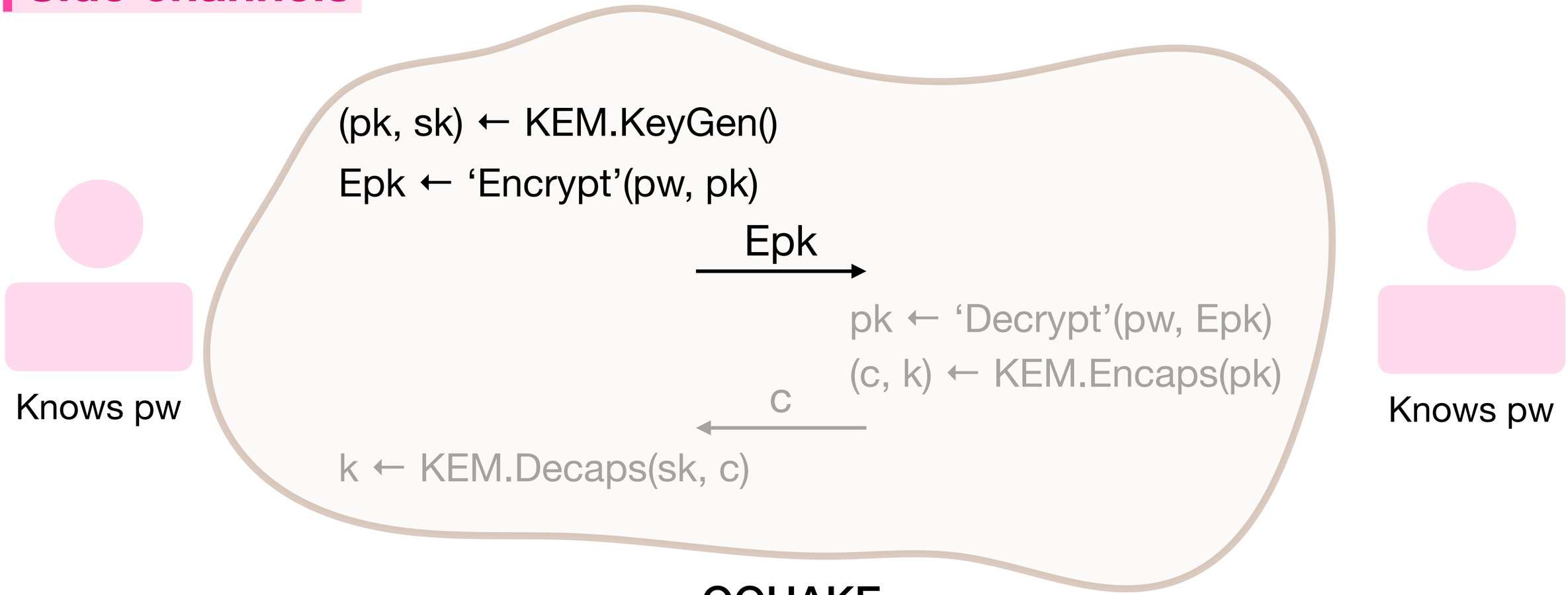
Possible timing side channel in OQUAKE

Side channels



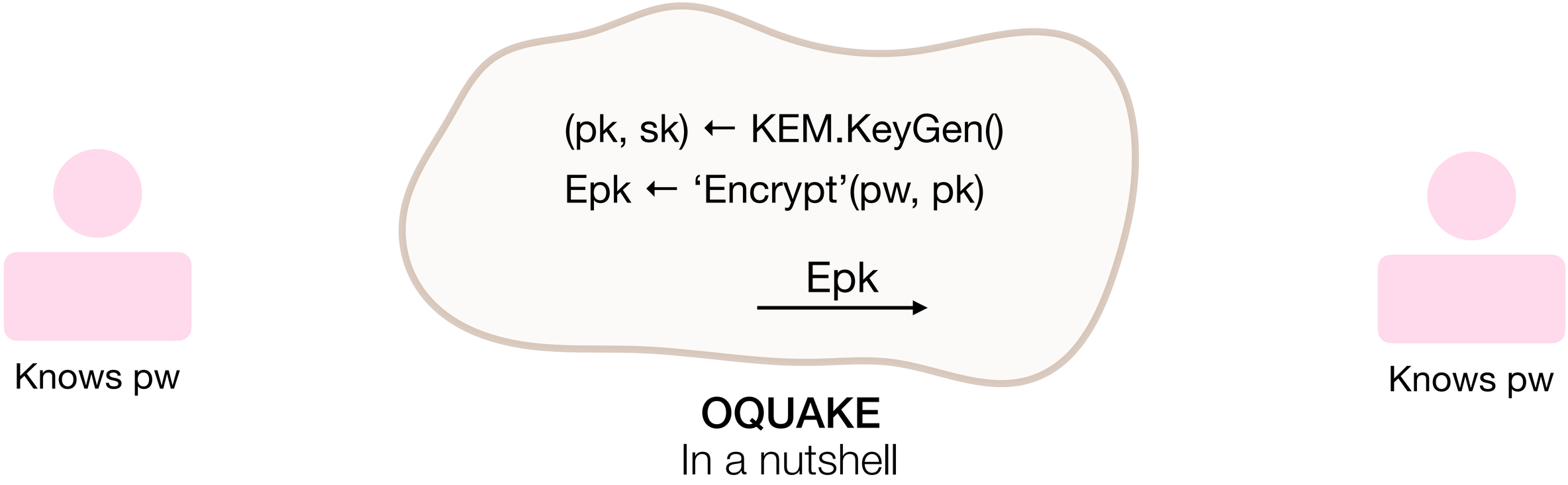
Possible timing side channel in OQUAKE

Side channels



Possible timing side channel in OQUAKE

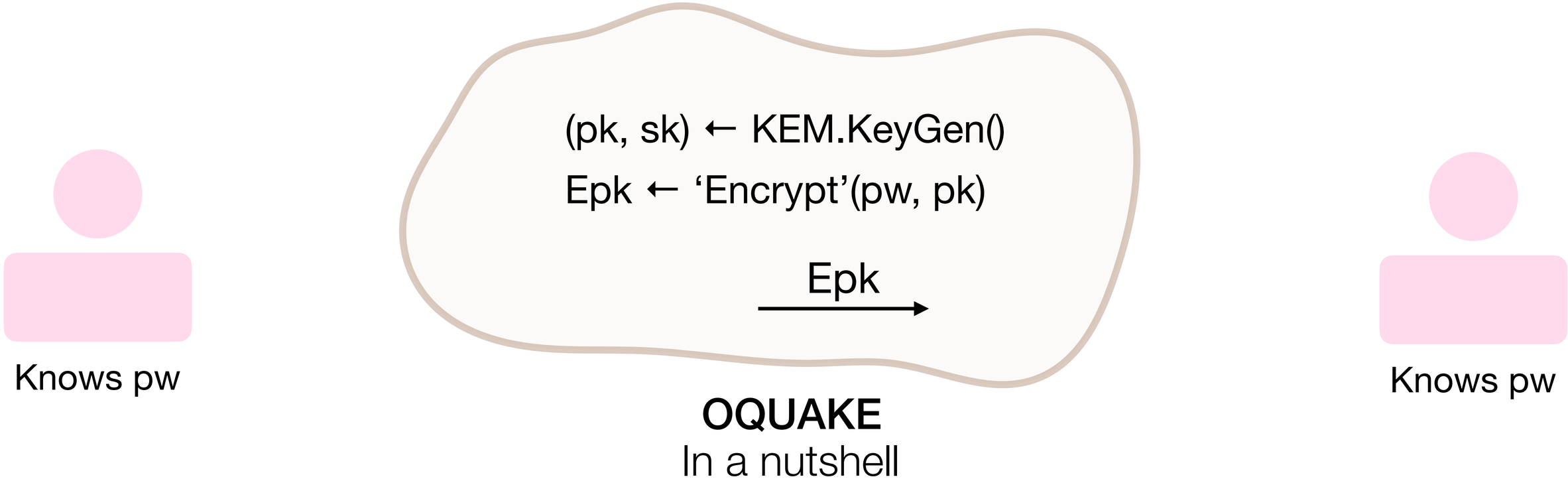
Side channels



OQUAKE
In a nutshell

Possible timing side channel in OQUAKE

Side channels

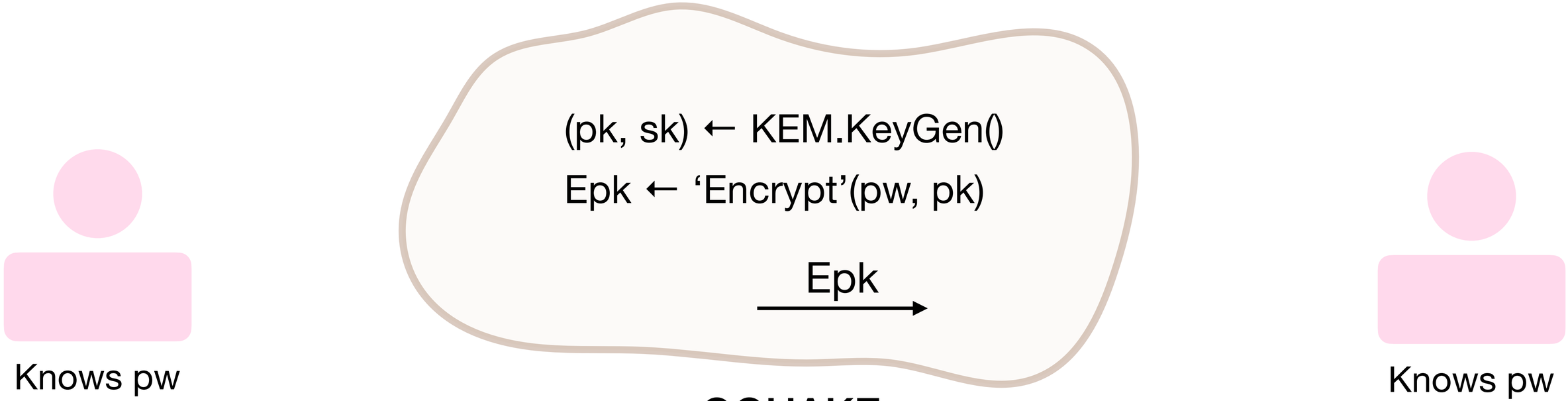


OQUAKE
In a nutshell

If you know pk, you can perform offline pw guessing

Possible timing side channel in OQUAKE

Side channels



OQUAKE
In a nutshell

If you know pk, you can perform offline pw guessing

ML-KEM spec does not require generating pk in constant time

Addressing timing side channels

Side channels

Easiest option

Require ML-KEM to be completely constant time

Addressing timing side channels

Side channels

Easiest option

Require ML-KEM to be completely constant time

A simple fix called TEMPO

Introduce changes to the protocol

Addressing timing side channels

Side channels

Easiest option

Require ML-KEM to be completely constant time

Minor spec changes

A simple fix called TEMPO

Introduce changes to the protocol

Requires small spec changes

Addressing timing side channels

Side channels

Easiest option

Require ML-KEM to be completely constant time

Minor spec changes

Significantly slower

A simple fix called TEMPO

Introduce changes to the protocol

Requires small spec changes

Small performance impact

Addressing timing side channels

Side channels

Easiest option

Require ML-KEM to be completely constant time

Minor spec changes

Significantly slower

FIPS compliant ML-KEM is insufficient

A simple fix called TEMPO

Introduce changes to the protocol

Requires small spec changes

Small performance impact

Wrapper around FIPS-compliant ML-KEM

Timing side channel in ML-KEM PK generation

Side channels

Compact PK = (seed, matrices)

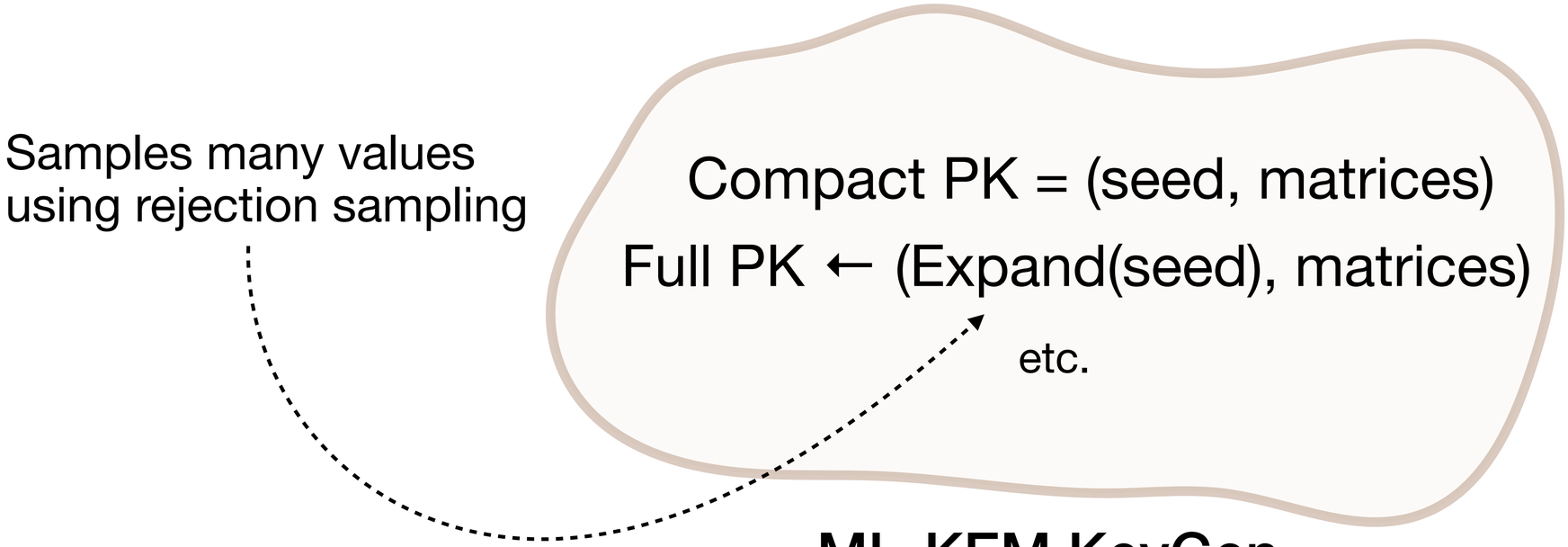
Full PK ← (Expand(seed), matrices)

etc.

ML-KEM.KeyGen
In a nutshell

Timing side channel in ML-KEM PK generation

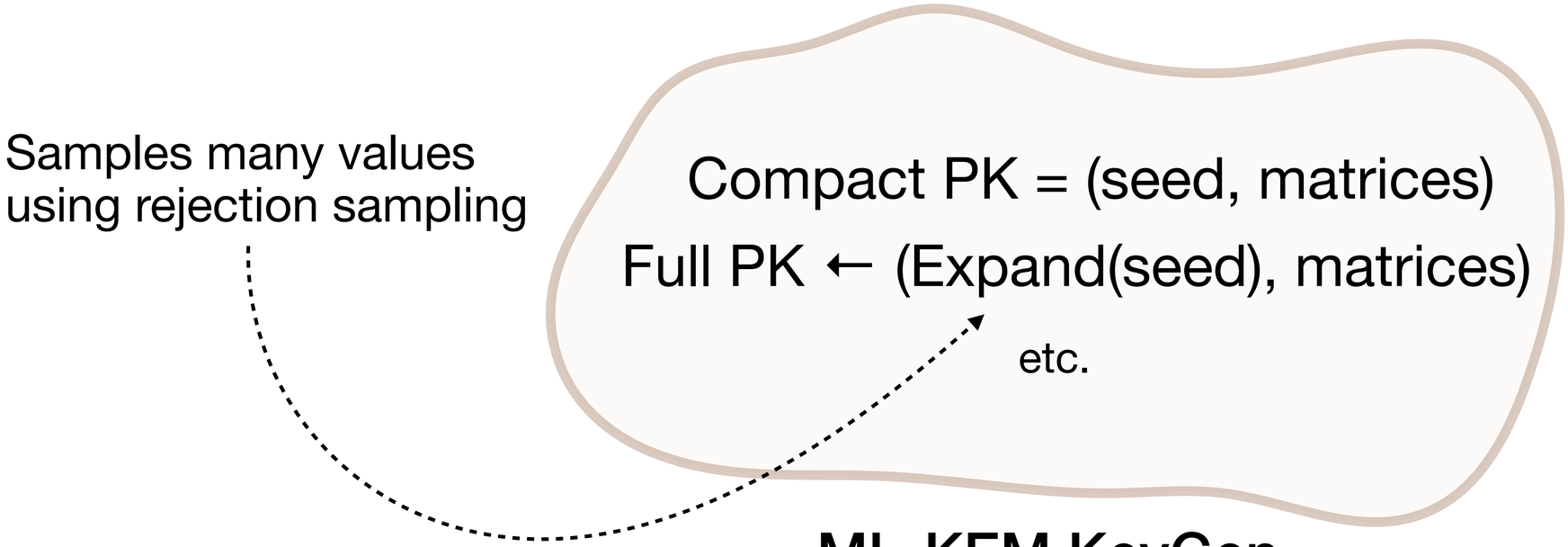
Side channels



ML-KEM.KeyGen
In a nutshell

Timing side channel in ML-KEM PK generation

Side channels



ML-KEM.KeyGen
In a nutshell

The number of rejections leaks information about the 256-bit random seed

A simple fix called TEMPO

Side channels

Before:

Epk ← ‘Encrypt’(pw, Compact PK) ← Compact PK = (seed, matrices)

A simple fix called TEMPO

Side channels

Before:

Epk ← ‘Encrypt’(pw, Compact PK) ← Compact PK = (seed, matrices)

After:

Epk ← seed || ‘Encrypt’(pw, Matrices)

A simple fix called TEMPO

Side channels

Before:

Epk ← ‘Encrypt’(pw, Compact PK) ← Compact PK = (seed, matrices)

After:

Epk ← seed || ‘Encrypt’(pw, Matrices)

Completely random

A simple fix called TEMPO

Side channels

Before:

Epk ← ‘Encrypt’(pw, Compact PK) ← Compact PK = (seed, matrices)

After:

Epk ← seed || ‘Encrypt’(pw, Matrices)

Completely random

Does not require rejection sampling

A simple fix called TEMPO

Side channels

Paper 2025/1399

Tempo: ML-KEM to PAKE Compiler Resilient to Timing Attacks

Afonso Arriaga, University of Luxembourg
Manuel Barbosa, University of Porto, INESC TEC, Max Planck Institute for Security and Privacy
Stanislaw Jarecki, University of California, Irvine

https://eprint.iacr.org/2025/1399

Before:

Epk ← ‘Encrypt’(pw, Compact PK) ← Compact PK = (seed, matrices)

After:

Epk ← seed || ‘Encrypt’(pw, Matrices)

Completely random

Does not require rejection sampling

Summary of other changes

Specification work

Chris: Revised introduction and motivation for a hybrid PAKE

Dan Harkins: Small inconsistency between paper and spec

Stanislaw Jarecki: Several small fixes, e.g. missing inputs

In conclusion

Specification work

Summary

Post-quantum vs. hybrid aPAKE

Should we focus on a hybrid aPAKE,
or also offer a post-quantum aPAKE?

In conclusion

Specification work

Summary

Post-quantum vs. hybrid aPAKE

Should we focus on a hybrid aPAKE, or also offer a post-quantum aPAKE?

A simple fix called TEMPO

Is the TEMPO the right approach against timing side channels in OQUAKE?

Requires small spec changes

Small performance impact

Wrapper around FIPS-compliant ML-KEM

In conclusion

Specification work

Summary

Post-quantum vs. hybrid aPAKE

Should we focus on a hybrid aPAKE, or also offer a post-quantum aPAKE?

A simple fix called TEMPO

Is the TEMPO the right approach against timing side channels in OQUAKE?

Requires small spec changes

Small performance impact

Wrapper around FIPS-compliant ML-KEM

Next steps

Revised version for IETF-125:

Incorporated changes laid out in TEMPO

Addressed current open issues

WIP

https://github.com/chris-wood/draft-pqpake