Post-Quantum

Cryptography Conference

A Sign of the Times: The Transition to Quantum Secure Authentication

Sandra Guasch Castello

Staff Privacy Engineer at SandboxAQ

SANDBOXAQ™

A sign of the times: the transition to quantum-secure authentication

Sandra Guasch, Staff Privacy Engineer
PKI Consortium - Post-Quantum Cryptography Conference
November 7 and 8, 2023

AGENDA

01 **Introduction**
PQC challenges to authentication systems

02 **Use case: FIDO2**
Introduction to the FIDO2 protocol

03 **PQ-readiness of FIDO2**
Is FIDO2 ready for PQC?

04 **Practical limitations and alternatives**

- Storage
- Runtime
- Potential adoption timeline

01

Introduction

PQC challenges to authentication systems

(Some) challenges of PQC to existing systems

- Longer keys, signatures, ciphertexts, certificates…
- Migration to new algorithms requires cryptographic agility
- How do we transition? Hybrid vs pure PQC?
- Interconnected systems, dependencies
- Remote / long-lived systems

(Also some) challenges of PQ authentication

Reliance on hardware

End-user distribution

Low capacity devices (hardware tokens, smartcards…)

We are first focusing on migrating encryption systems due to SNDL attacks

02 Use case: FIDO2

Introduction to the FIDO2 protocol

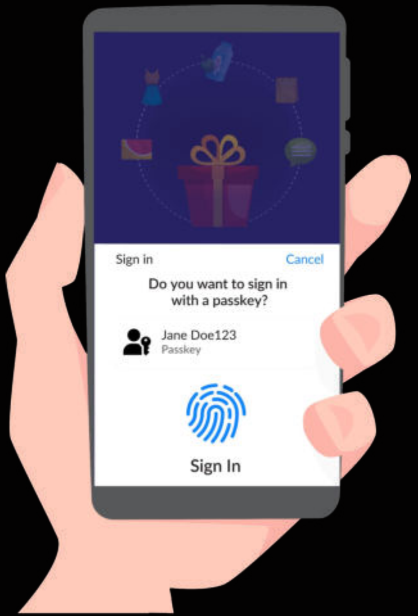
Comprised by more than **40 key companies**, including Amazon, Apple, Google, Intel, Microsoft, RSA, VISA, and Yubico

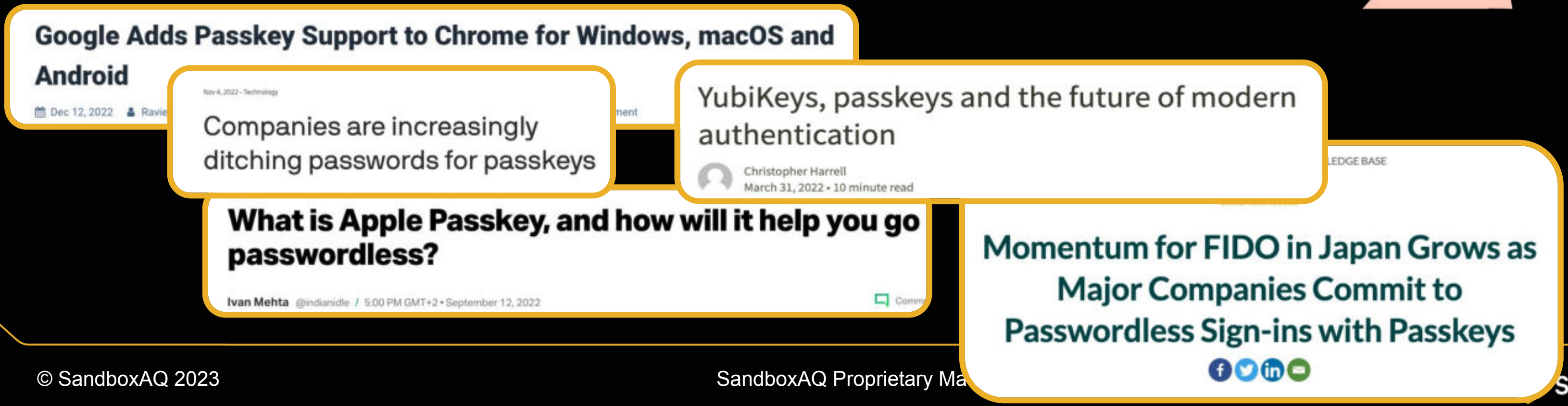
Defined de facto standard for passwordless authentication: **FIDO2 protocol**

What is FIDO2?

Advantages

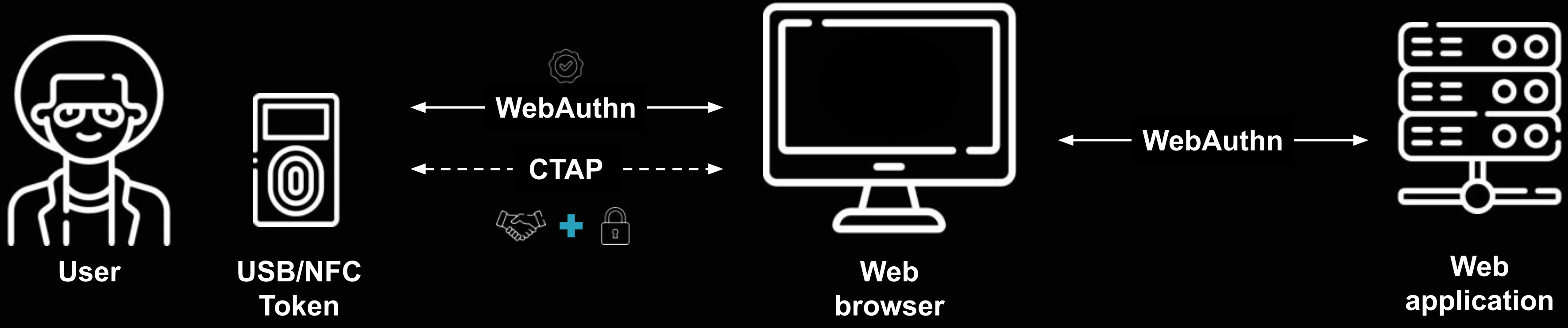
- No need to remember passwords
- Easy to use
- Resistant to phishing attacks
- Widely adopted: FIDO Alliance / W3C standards
  - Supported by all major browsers and platforms
  - Wide range of industry partners
- Constant improvements (e.g., Passkeys)



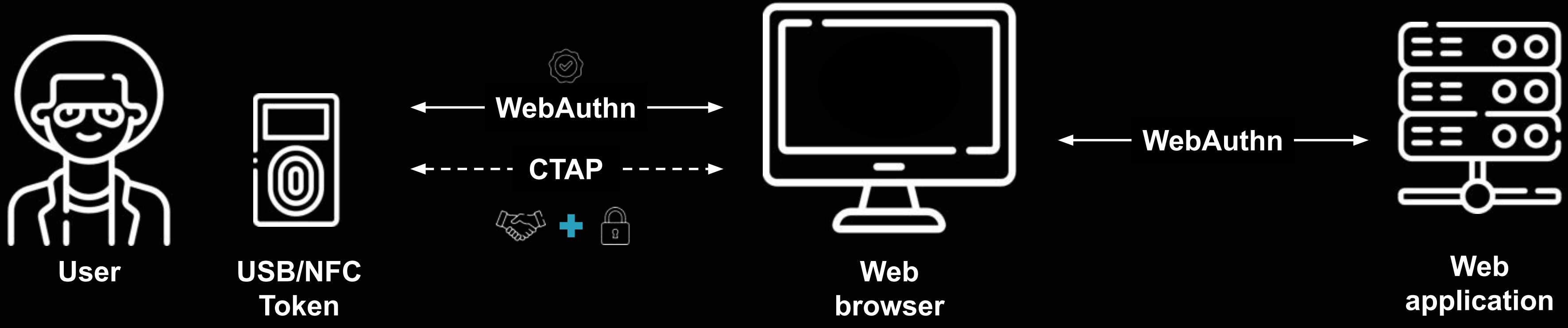


FIDO2 protocol

FIDO2 protocol



FIDO2 protocol

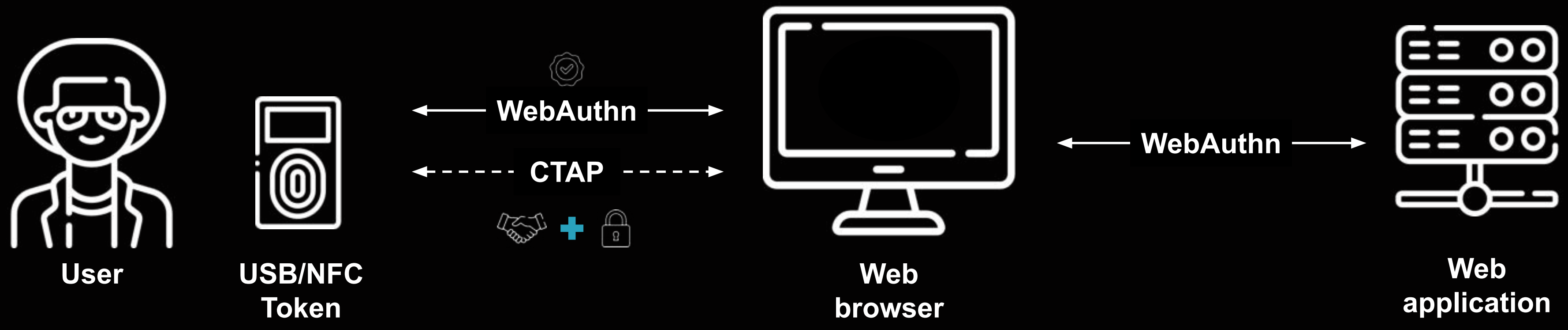


1 **Register** user's token with web application via web browser:

- **Attestation** of token properties
- Generation of **credential keys**

2 **Authenticate/Log in** using credential keys and by pressing button on the token

Token crypto operations



1

Register user's token with web application via web browser:

- Generation of **credential keys**
- **Attestation** of token properties

Generate signing key pair

Sign challenge + pk

Challenge-response WebAuthn

Generate KEM keys

KEM decrypt

Symm encrypt

CTAP session establishment

Remote attestation in FIDO2

Alternative privacy modes used in trusted computing environments (not in FIDO2 now)

None

No attestation signature

Self

Registration credentials are self-signed. No token properties are claimed.

Basic

A group of devices share the same attestation keypair.

Origin of signed attestation records is indistinguishable within the group.

Privacy / Anonymity CA

Multiple attestation keys per device (i.e. one per each server to register with).

Privacy / anonymity CA certifies attestation keys after verifying the device characteristics / identity.

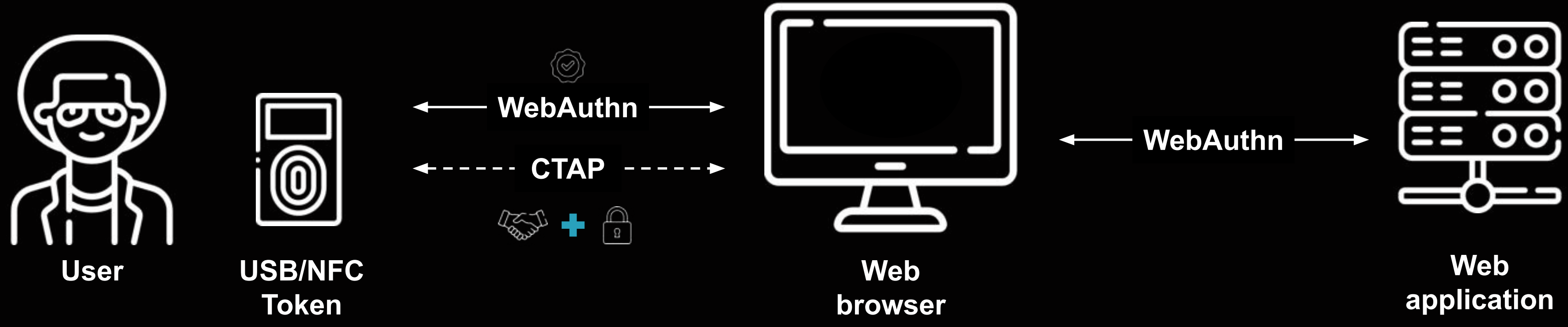
DAA

Device has one certified attestation key that can be "masked". Attestation records signed for different services can't be linked to the same certified key. Used in TC.

EPID

Easier revocation. Used in TC and TEE.

Token crypto operations



2 **Authenticate/Log in** using credential keys and by pressing button on the token

~~Generate signing key pair~~

Sign challenge + pk

Challenge-response WebAuthn

Generate KEM keys

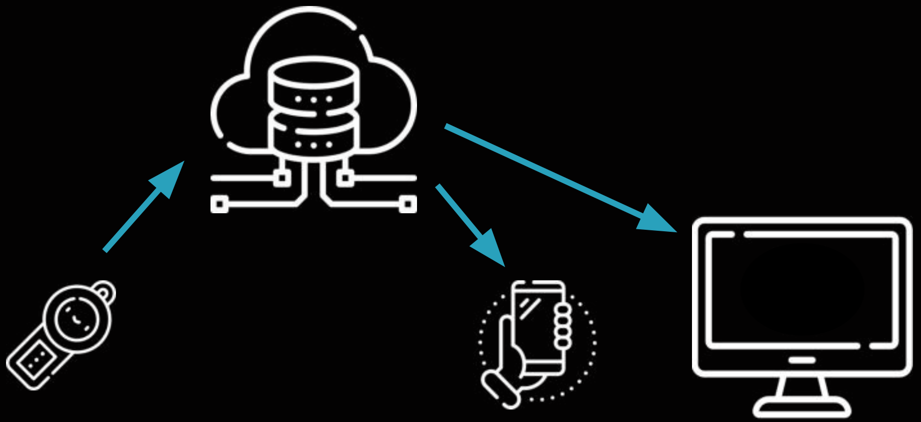
KEM decrypt

Symm encrypt

CTAP session establishment

Passkeys

New name for FIDO2 credentials, enables credential synchronisation among different devices:



- Credentials are encrypted end-to-end
- Device-bound credentials can still be enforced for critical applications
- Attestation becomes crucial to understand how a credential is managed

03

PQ-readiness of FIDO2

Is FIDO2 ready for PQC?

“PQ-adaptability” in FIDO2

Registration and Authentication

Standard digital signatures and KEMs → PQ / PQ-Classical hybrid schemes *

Attestation types: none, self, basic, AttCA, AnonCA

* Discussed in FIDO2, CTAP 2.1, and WebAuthn 2: Provable Security and Post-Quantum Instantiation. Bindel, Cremers, Zhao. IACR eprint 2022/1029

“PQ-adaptability” in FIDO2

Registration and Authentication

Standard digital signatures and KEMs → PQ / PQ-Classical hybrid schemes *

Attestation types: none, self, basic, AttCA, AnonCA

Attestation types: DAA / EPID**

Zero-knowledge proofs and randomisable credentials / signatures → No PQ primitives standardised yet

DAA Several lattice-based constructions (Ring-LWE, Ring-SIS, NTRU assumptions)

EPID Lattice-based proposal, also symmetric cryptography EPID: symmetric ciphers, hash-based signatures (like SPHINCS) and zero-knowledge proofs using the MPCitH paradigm.

Preliminary results emulated in standard processors, times and signature sizes larger than in traditional algs.

* Discussed in FIDO2, CTAP 2.1, and WebAuthn 2: Provable Security and Post-Quantum Instantiation. Bindel, Cremers, Zhao. IACR eprint 2022/1029
** These modes are currently not in the FIDO2 standard but could eventually be included due to their privacy features

04

Practical limitations and alternatives

- Storage
- Runtime
- Potential adoption timeline

FIDO2 tokens capabilities

Solo1 / Nitrokey FIDO2 → *using* → STM32L432 (Cortex M4):

- 256Kb Flash, 64Kb SRAM

New generation Solo2 / Nitrokey 3 → *based on* → NXP LPC55S69 (Cortex M33):

- Up to 640Kb Flash and 320Kb RAM
- Solo2 uses Cortex M4f for production

FIDO2 tokens capabilities

Solo1 / Nitrokey FIDO2 — *using* → STM32L432 (Cortex M4):

- 256Kb Flash, 64Kb SRAM

New generation Solo2 / Nitrokey 3 — *based on* → NXP LPC55S69 (Cortex M33):

- Up to 640Kb Flash and 320kb RAM
- Solo2 uses Cortex M4f for production

Storage constraints

Base firmware	Additional PQ algorithms *	PQ algorithms	Type
~90-120Kb	~15-16.5Kb	Kyber L1, L3, L5	KEM
~90-120Kb	~18-19.5Kb	Dilithium L2, L3, L5	Signature
~90-120Kb	~108-160Kb	Falcon L1-tree, L1, L5	Signature
~90-120Kb	~4.2-4.7Kb	SPHINCS+-shake256 L1-L5	Signature

(*) From pqm4 project

FIDO2 tokens capabilities

Solo1 / Nitrokey FIDO2 — *using* → STM32L432 (Cortex M4):
- 256Kb Flash, 64Kb SRAM

New generation Solo2 / Nitrokey 3 — *based on* → NXP LPC55S69 (Cortex M33):
- Up to 640Kb Flash and 320Kb RAM
- Solo2 uses Cortex M4f for production

Storage constraints

It could be too big for old devices

Base firmware	Additional PQ algorithms *	PQ algorithms	Type
~90-120Kb	~15-16.5Kb	Kyber L1, L3, L5	KEM
~90-120Kb	~18-19.5Kb	Dilithium L2, L3, L5	Signature
~90-120Kb	~108-160Kb	Falcon L1-tree, L1, L5	Signature
~90-120Kb	~4.2-4.7Kb	SPHINCS+-shake256 L1-L5	Signature

(*) From pqm4 project

FIDO2 tokens capabilities

Solo1 / Nitrokey FIDO2 — *using* → STM32L432 (Cortex M4):

- 256Kb Flash, 64Kb SRAM

New generation Solo2 / Nitrokey 3 — *based on* → NXP LPC55S69 (Cortex M33):

- Up to 640Kb Flash and 320kb RAM
- Solo2 uses Cortex M4f for production

Storage constraints

Code reuse

Base firmware	Additional PQ algorithms *	PQ algorithms	Type
~90-120Kb	~15-16.5Kb	Kyber L1, L3, L5	KEM
~90-120Kb	~18-19.5Kb	Dilithium L2, L3, L5	Signature
~90-120Kb	~108-160Kb	Falcon L1-tree, L1, L5	Signature
~90-120Kb	~4.2-4.7Kb	SPHINCS+-shake256 L1-L5	Signature

(*) From pqm4 project

FIDO2 tokens capabilities

Solo1 / Nitrokey FIDO2 → *using* → STM32L432 (Cortex M4):

- 256Kb Flash, 64Kb SRAM

New generation Solo2 / Nitrokey 3 → *based on* → NXP LPC55S69 (Cortex M33):

- Up to 640Kb Flash and 320kb RAM
- Solo2 uses Cortex M4f for production

Storage constraints

Base firmware	Additional PQ algorithms *	PQ algorithms	Type	Credentials (private key)**	50 creds (Solokey)***	25 creds (Yubikey)****
~90-120Kb	~15-16.5Kb	Kyber L1, L3, L5	KEM	~1.6-3.1Kb	~80-155Kb	~40-77.5Kb
~90-120Kb	~18-19.5Kb	Dilithium L2, L3, L5	Signature	~2.5-4.8Kb	~125-240Kb	~62.5-120Kb
~90-120Kb	~108-160Kb	Falcon L1-tree, L1, L5	Signature	~7.4-13.7Kb	~370-685Kb	~185-343Kb
~90-120Kb	~4.2-4.7Kb	SPHINCS+-shake256 L1-L5	Signature	64-128b	~3.2-6.3Kb	~1.6-3.2Kb

(*) From pqm4 project
(**) From NIST report
(***) From Solokey
(****) From Yubikey

FIDO2 tokens capabilities

Solo1 / Nitrokey FIDO2 → *using* → STM32L432 (Cortex M4):

- 256Kb Flash, 64Kb SRAM

New generation Solo2 / Nitrokey 3 → *based on* → NXP LPC55S69 (Cortex M33):

- Up to 640Kb Flash and 320kb RAM
- Solo2 uses Cortex M4f for production

Storage constraints

Can we store the keys somewhere else?

Base firmware	Additional PQ algorithms *	PQ algorithms	Type	Credentials (private key)**	50 creds (Solokey)***	25 creds (Yubikey)****
~90-120Kb	~15-16.5Kb	Kyber L1, L3, L5	KEM	~1.6-3.1Kb	~80-155Kb	~40-77.5Kb
~90-120Kb	~18-19.5Kb	Dilithium L2, L3, L5	Signature	~2.5-4.8Kb	~125-240Kb	~62.5-120Kb
~90-120Kb	~108-160Kb	Falcon L1-tree, L1, L5	Signature	~7.4-13.7Kb	~370-685Kb	~185-343Kb
~90-120Kb	~4.2-4.7Kb	SPHINCS+-shake256 L1-L5	Signature	64-128b	~3.2-6.3Kb	~1.6-3.2Kb

(*) From pqm4 project
(**) From NIST report
(***) From Solokey
(****) From Yubikey

FIDO2 discoverable vs non-discoverable credentials

Discoverable credentials

Passkeys

Private keys are stored in the token.

# of servers to register with is limited by token storage space.

Non-discoverable credentials

Private keys are stored in the remote servers, encrypted with a token master key.

# of servers to register with is potentially unlimited.

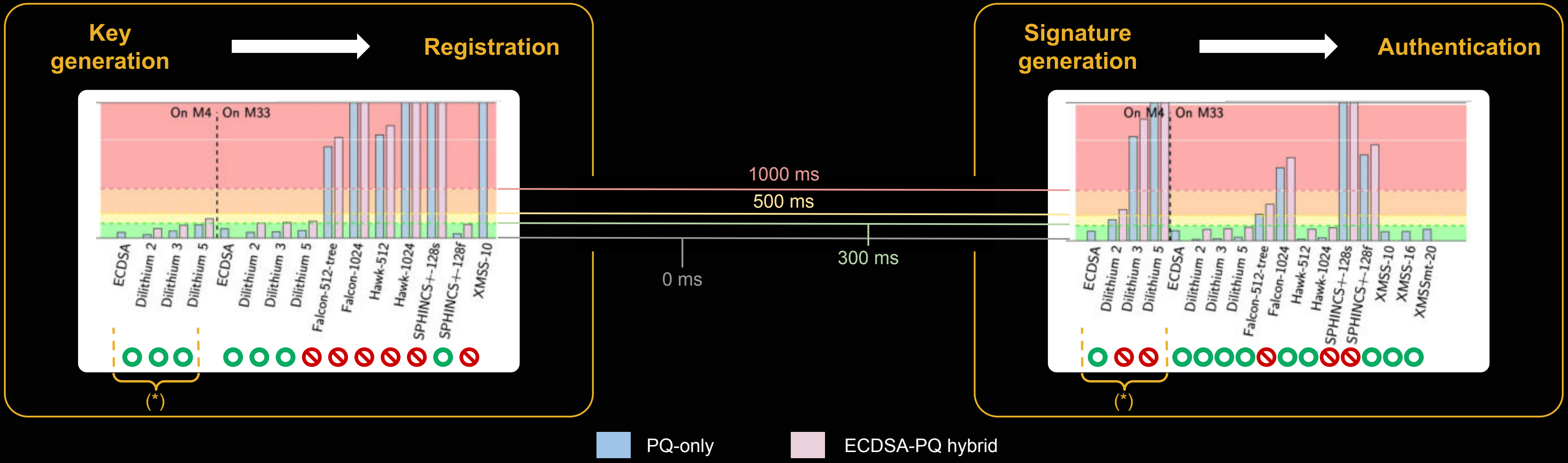
04

Practical limitations and alternatives

- Storage
- Runtime
- Potential adoption timeline

Runtime: Preliminary Experimental Results

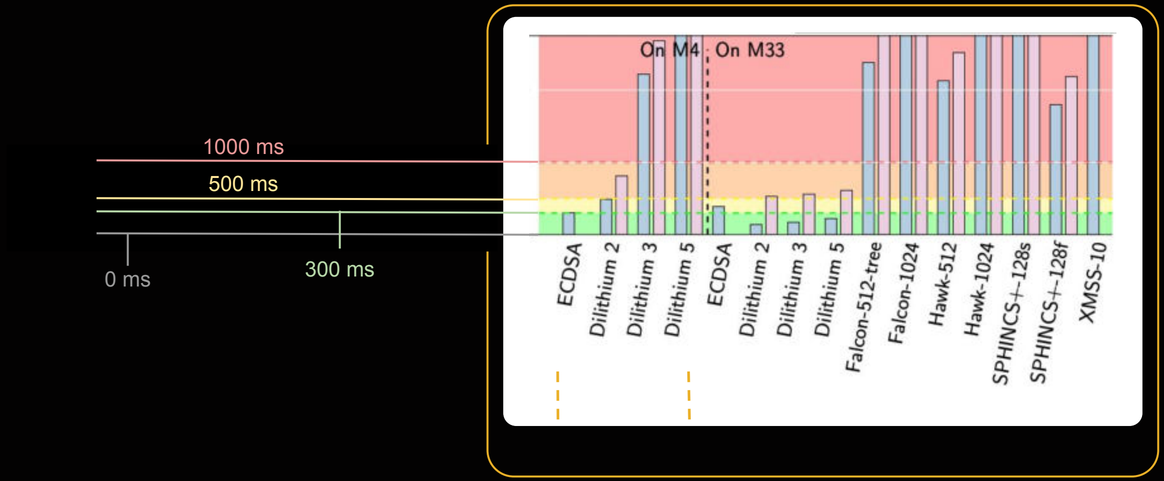
Signature or Key Generation



(*) Results from Ghinea, Kaczmarczyck, Pullman, Cretin, Kölbl, Misoczki, Picod, Invernizzi, and Bursztein. Hybrid Post-Quantum Signatures in Hardware Security Keys. IACR ePrint 2022/1225.
The rest of the experimental data is done in cooperation with Duc Nguyen and James Howe.

Runtime: Preliminary Experimental Results

Signature or Key Generation + Attestation



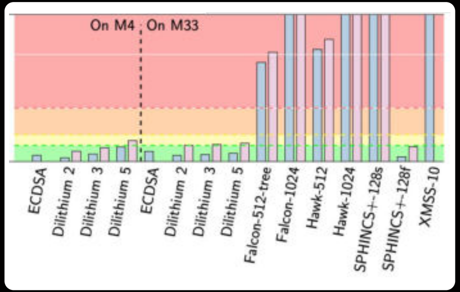
PQ-only ECDSA-PQ hybrid

(*) Results from Ghinea, Kaczmarczyck, Pullman, Cretin, Kölbl, Misoczki, Picod, Invernizzi, and Bursztein. Hybrid Post-Quantum Signatures in Hardware Security Keys. IACR ePrint 2022/1225.
The rest of the experimental data is done in cooperation with Duc Nguyen and James Howe.

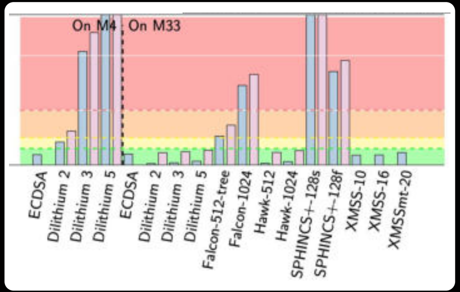
Different Signature Schemes for Attestation and Authentication

Example

Key generation



Signature generation



PQ-only ECDSA-PQ hybrid

Attestation mode Basic

- No attestation key generation on hardware token (pre-generated)

Attestation scheme: Hawk

Authentication scheme: Dilithium

- **Registration:**
  Dilithium key generation + Hawk signing **reasonably fast**
- **Authentication:**
  Dilithium signature generation **fast**

Advantage compared to Dilithium-only:

- Attestation signature only 546 B instead of 2420 B
- Particularly beneficial as during registration other information needs to be sent (e.g., public credential key)

New NIST call for alternative signature schemes may bring additional good options!

(*) Results from Ghinea, Kaczmarczyck, Pullman, Cretin, Kölbl, Misoczki, Picod, Invernizzi, and Bursztein. Hybrid Post-Quantum Signatures in Hardware Security Keys. IACR ePrint 2022/1225.
The rest of the experimental data is done in cooperation with Duc Nguyen and James Howe.

SANDBOXAQ

New open-source library!

Post-quantum secure (Dilithium, Kyber)

End-to-end

Token firmware based on Trussed framework, used by SoloKeys and Nitrokey

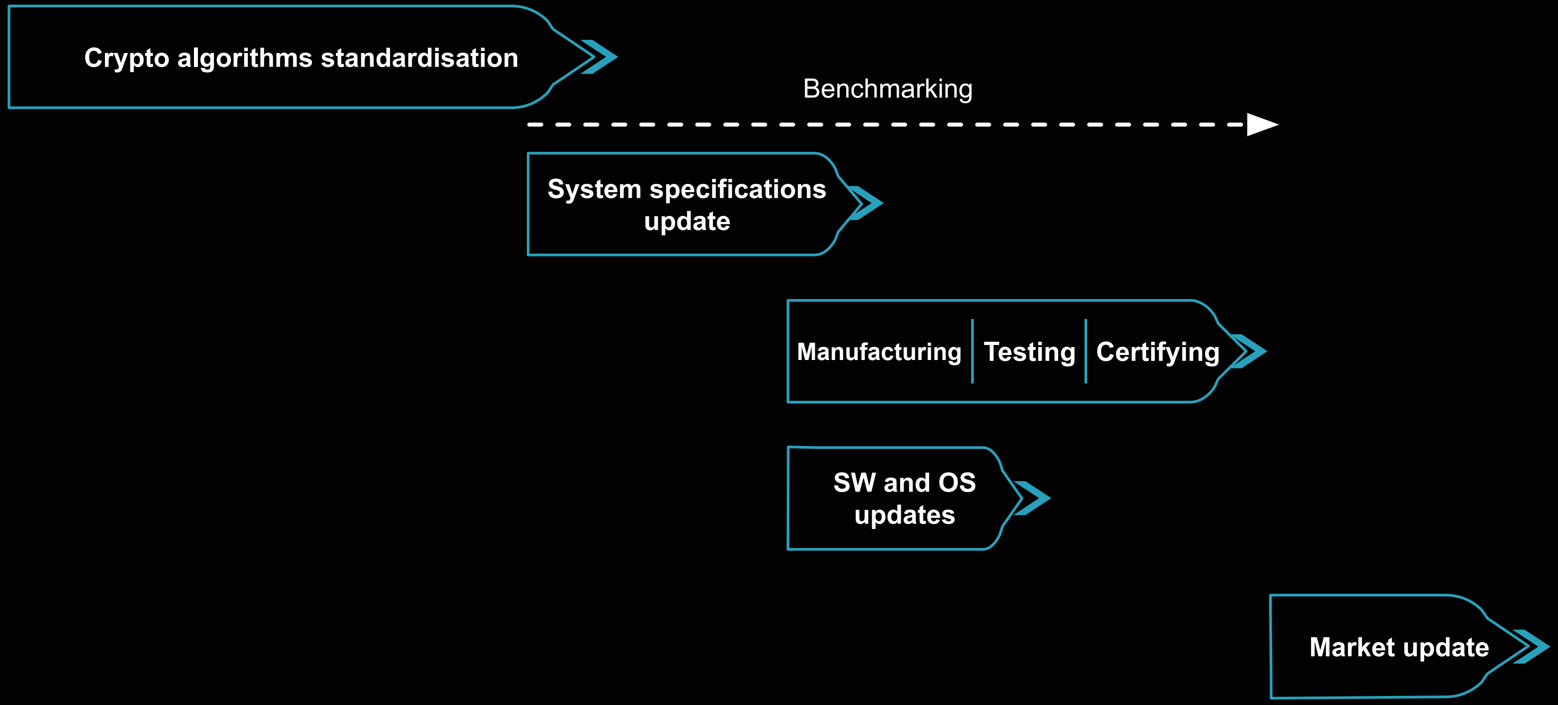
https://github.com/sandbox-quantum/pqc-fido2-impl

04 Practical limitations and alternatives

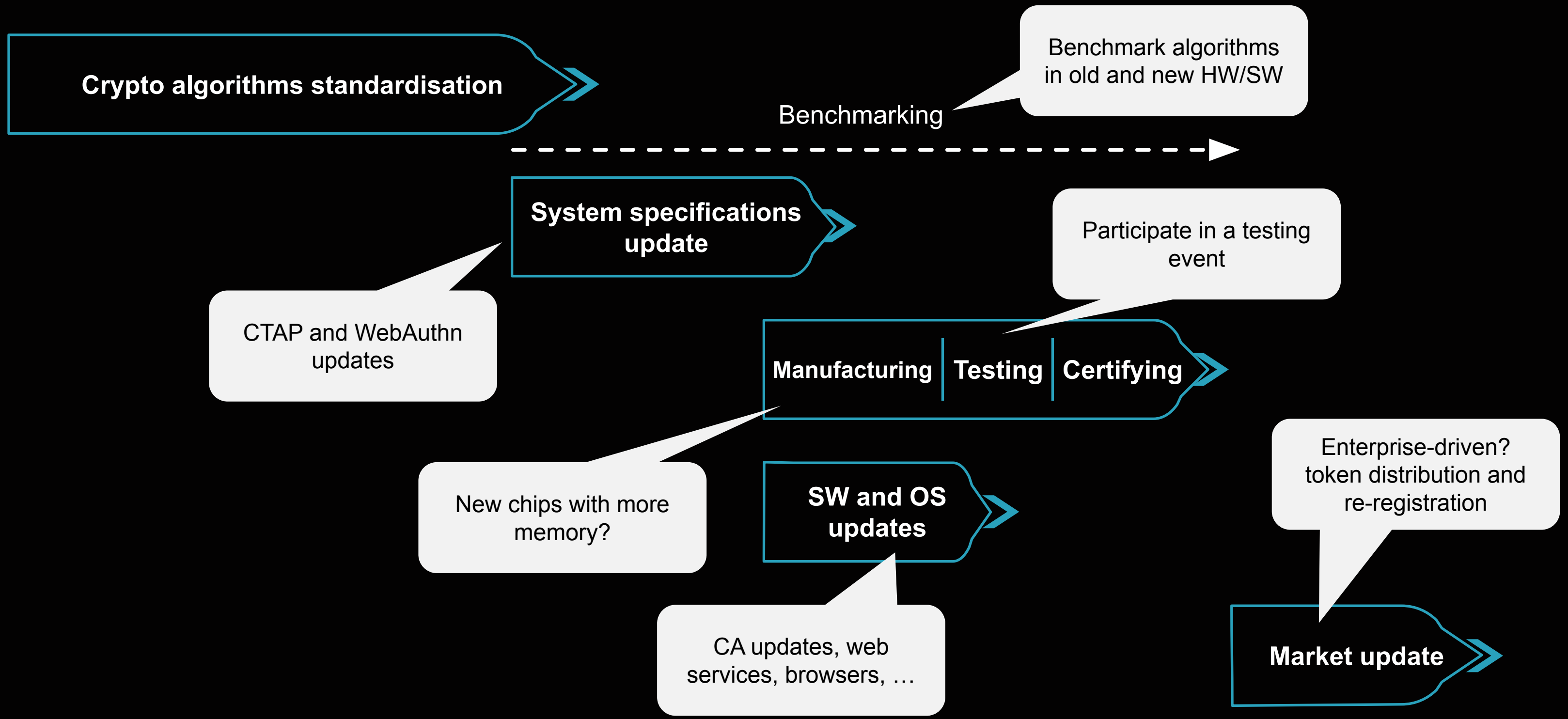
- Storage
- Runtime
- Potential adoption timeline

1. Update FIDO2 specifications (which algorithms, backwards compatibility, etc)
2. Slower key and/or signature generation
3. Potential needed changes in hardware because of 02
4. Re-certification of hardware tokens
5. Distribute new tokens
6. Update CAs, browsers and web applications

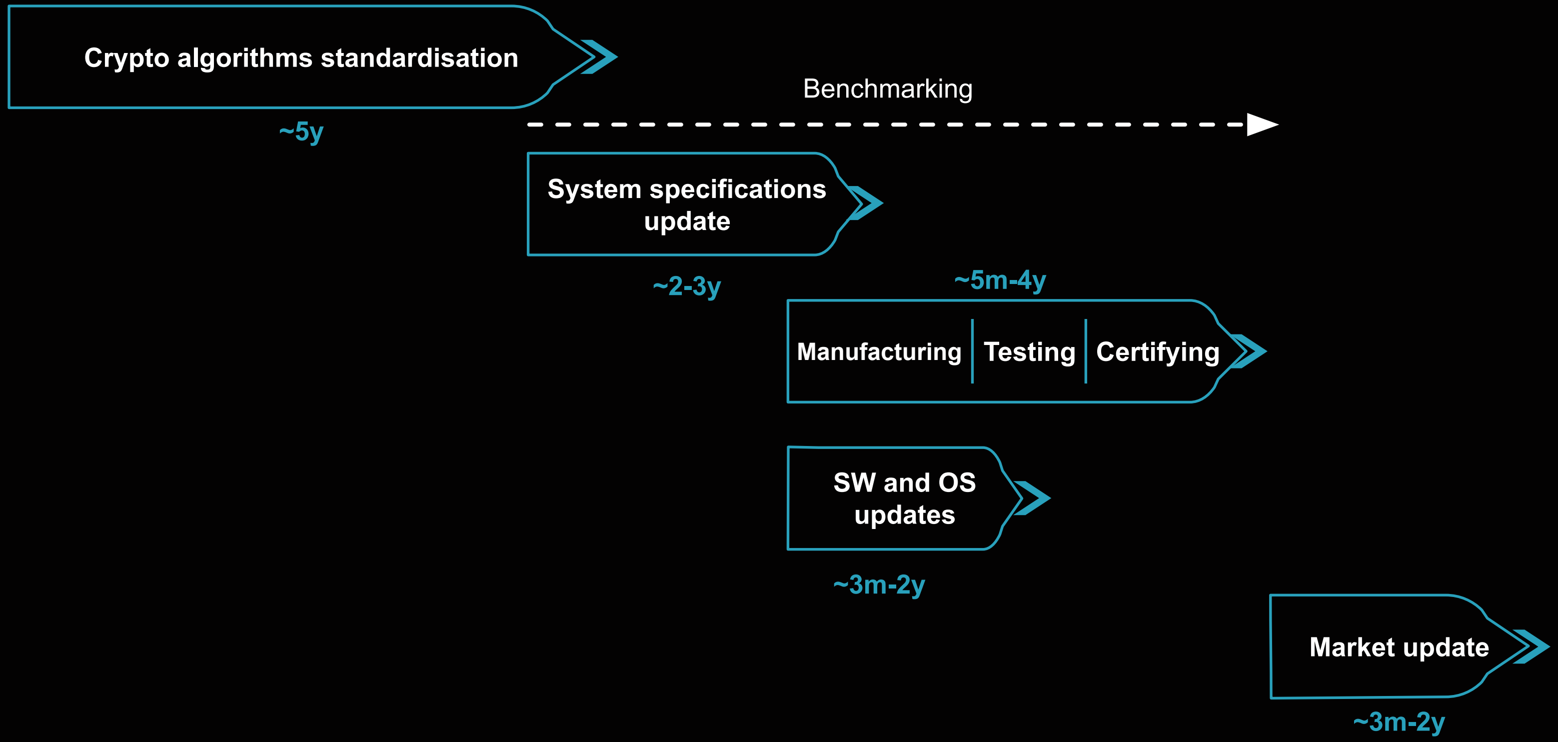
Potential adoption timeline: phases



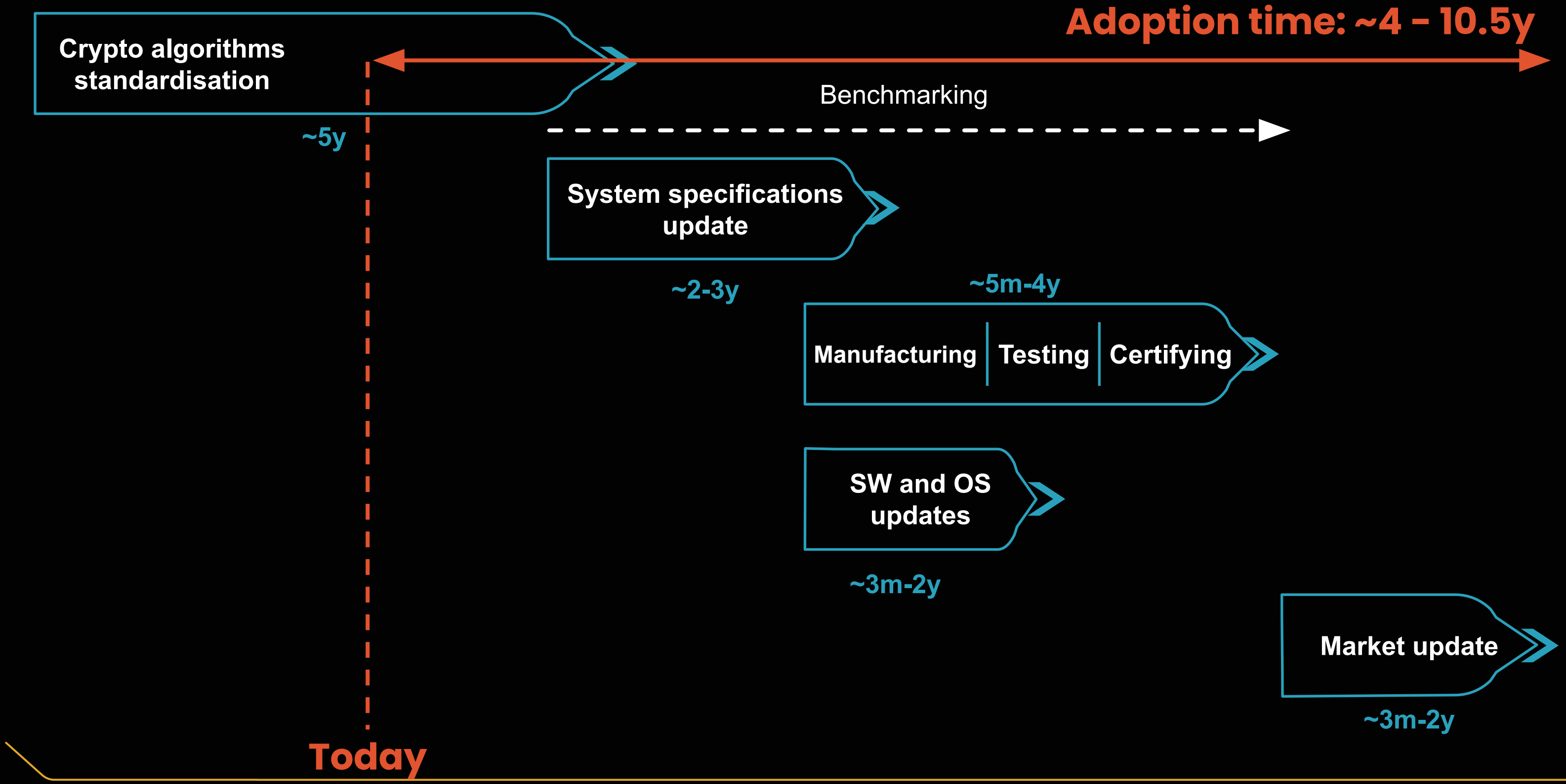
Potential adoption timeline: FIDO2 example



Potential adoption timeline: FIDO2 example



Potential adoption timeline: FIDO2 example



Takeaways

1. Research gaps regarding (efficient) PQ algorithms for some attestation modes - wait or pay the price!

2. New PQ end-to-end FIDO2 implementation

3. Migrations take (surprisingly) long, start preparing!

Nina Bindel*

ninabindel.de

nbindel

@NinaBindel

Sandra Guasch*

sandraguasch

James Howe*

jameshowe.eu

jameshowe1729

@JamesHowe1729

Duc Tri Nguyen*

https://cothan.blog

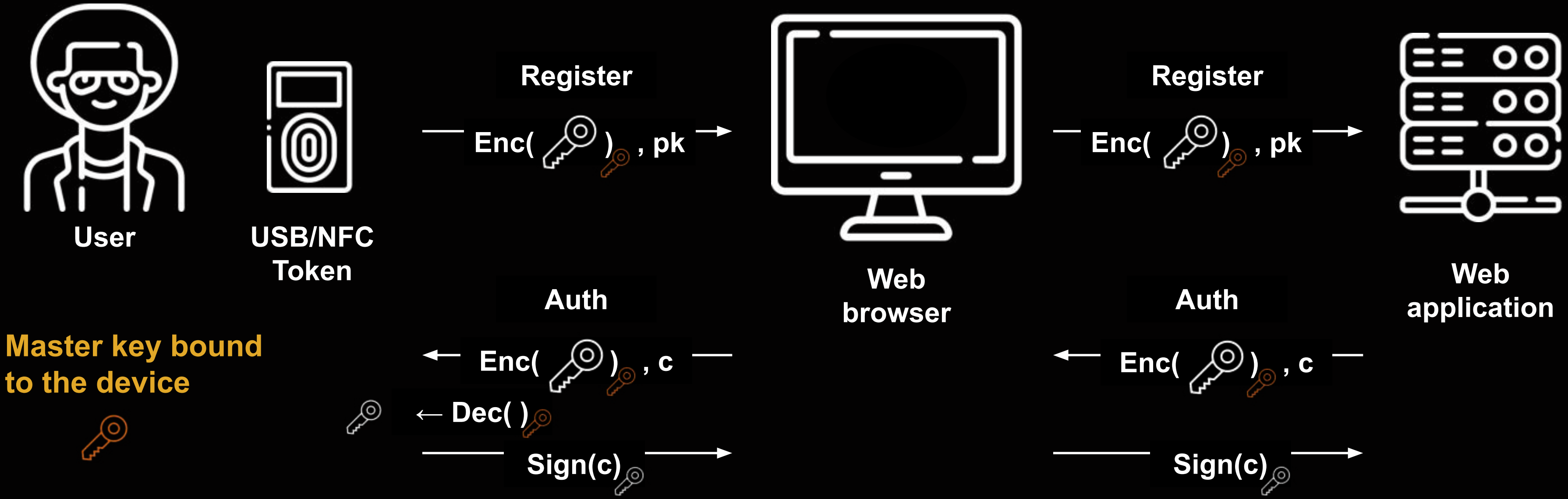
cothan

(*) name.surname@sandboxaq.com

N

Extra materials

FIDO2 non-discoverable credentials



Post-Quantum Cryptography Conference

PKI Consortium