To attest or not to attest, this is the question – Provable attestation in FIDO2

Nina Bindel[1], Nicolas Gama[1], **Sandra Guasch[1]**, Eyal Ronen[2]

[1]SandboxAQ, Palo Alto, CA, USA,

[2]Tel Aviv University, Tel Aviv, Israel

Asiacrypt - December 2023

What is FIDO2?

- Standard for **password**less authentication driven by the Fast Identity Online (FIDO) Alliance.
- Widely adopted by browsers, platforms, industry (Amazon, Apple, Google, Intel, Microsoft, RSA, VISA ...).

Classical authentication solutions for web **are not working:**

- **Passwords are** hard to remember or not complex enough; vulnerable to phishing or credential stuffing attacks; difficult to use in multiple devices.
- **Multi-factor authentication / OTPs** present low usability while still vulnerable to phishing, and usually result in extra attack surface (e.g. smishing).

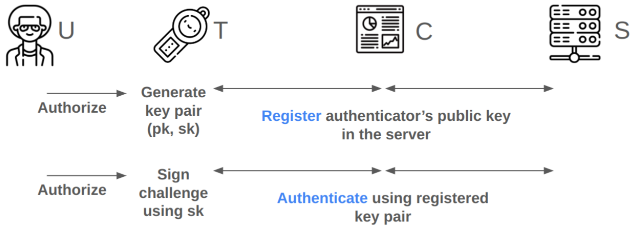
- Standard for **password**less authentication driven by the Fast Identity Online (FIDO) Alliance.
- Widely adopted by browsers, platforms, industry (Amazon, Apple, Google, Intel, Microsoft, RSA, VISA ...).

Classical authentication solutions for web **are not working:**

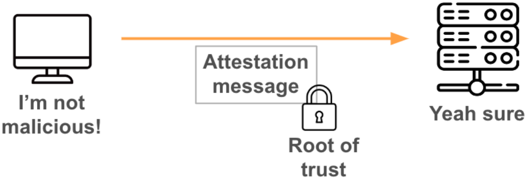
- **Passwords are** hard to remember or not complex enough; vulnerable to phishing or credential stuffing attacks; difficult to use in multiple devices.
- **Multi-factor authentication / OTPs** present low usability while still vulnerable to phishing, and usually result in extra attack surface (e.g. smishing).

Two sub-protocols: CTAP and **WebAuthn**

- CTAP: ensures only an authorized client talks with the authenticator.
- WebAuthn: communication between authenticator, client (or browser), server (or Relying Party).
    - Challenge-response protocol.



Attestation is a way for a system to make statements about itself, so that a 3rd party can make decisions based on that.

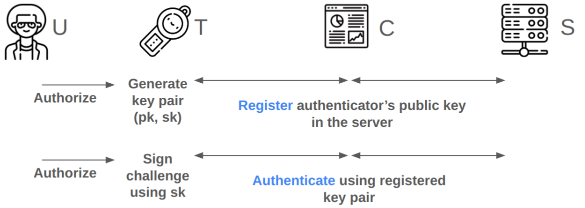


Attestation in FIDO2

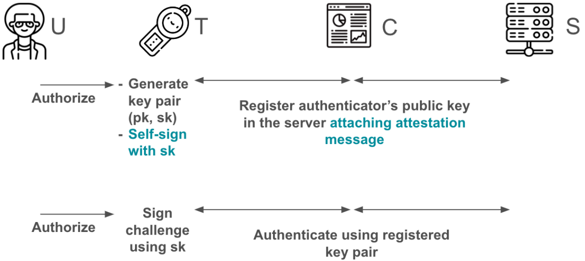
- The goal is to prevent users from using weak or uncertified authenticators. Servers (RPs) can make decisions about which authenticators can be used to authenticate with them.
- FIDO2 supports several attestation modes which different **security** and **privacy** properties.

Attestation Modes

None

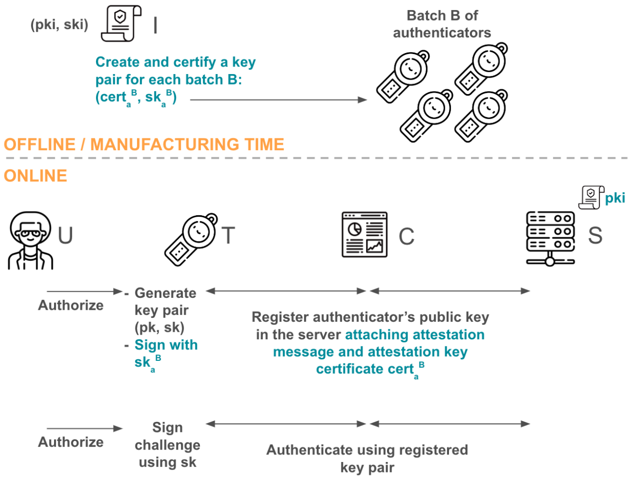


Self



Attestation Modes

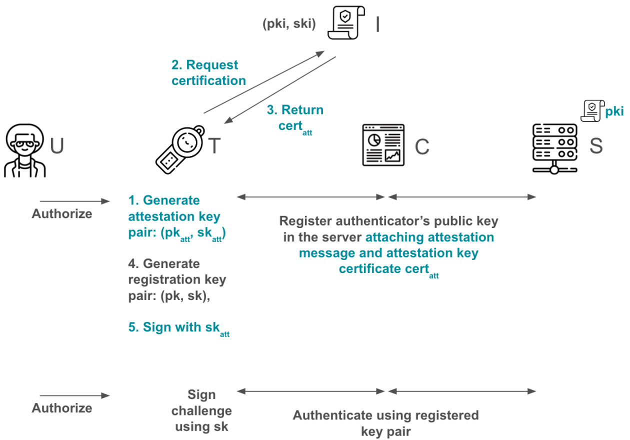
Basic attestation



Attestation Modes

SANDBOXAQ™

Attestation CA / Anonymous attestation



Our contribution

- Model and proofs of the authentication security and privacy properties of FIDO2, including all the supported attestation modes.
- Propose SimpleTW – an attestation mode based on Token Weaver [CJR22] which improves on the properties of existing modes.

Previous works

	[BBCW21]	[HLW23]	[BCZ23]	This work
Properties				
Authentication Security	✓	✓	✓	✓
Unlinkability	✗	✓	✗	✓
PQ-readiness	✗	✗	✓	✓
Post-compromise Security	✗	✗	✗	(✓)
Attestation modes				
None	✗	✗	✓	✓
Self	✗	✓	✗	✓
Basic	(✓)	✗	✗	✓
AttCA	✗	✗	✗	✓
SimpleTW	✗	✗	✗	✓
Adversary type during the protocol phases				
Certification	-	-	-	Active
Registration	Active	Active	Passive	Active
Authentication	Active	Active	Active	Active

Outline (of the rest of the talk)

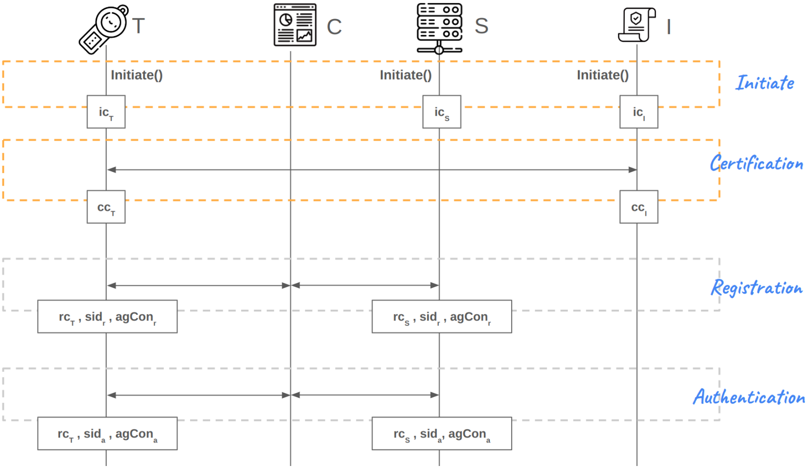
SANDBOXAQ™

1. **Authentication security and Privacy analysis:**
   Adversarial model and results for each attestation mode.
2. **Simple Token Weaver:**
   An attestation protocol leveraging the strongest security and privacy notions while providing additional features.

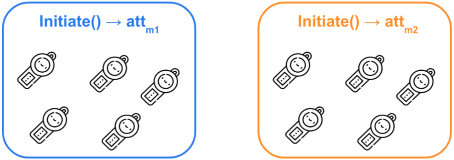
Part I - Authentication security and Privacy

SANDBOXAQ™

Two additional phases to cover additional operations for attestation.



A group G is a set of authenticators that share the same attestation material att_m created in Initiate() and shared with the server during Registration.

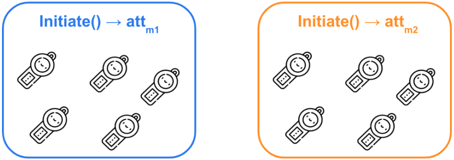


Depending on the attestation mode, a group is:

- *none*, *self*: $\text{att}_\text{m} = \perp, \rightarrow G = \perp$.
- *basic*: a batch of authenticators sharing the same certificate of the attestation public key issued by the same issuer public key.
- *attCA*: a batch of authenticators with attestation keys certified by the same issuer.

SANDBOXAQ™

A group G is a set of authenticators that share the same attestation material att_m created in Initiate() and shared with the server during Registration.



Depending on the attestation mode, a group is:

- *none*, *self*: $\text{att}_\text{m} = \perp, \rightarrow G = \perp$.
- *basic*: a batch of authenticators sharing the same certificate of the attestation public key issued by the same issuer public key.
- *attCA*: a batch of authenticators with attestation keys certified by the same issuer.

SANDBOXAQ™

Authentication security

An adversary shouldn't be able to authenticate *on behalf* of any authenticator from a given group G, provided that it didn't have access to it or to its contents *or to the internal contents of other authenticators from the same group (if $G \neq \perp$).*

Privacy $\rightarrow$ Unlinkability

Group unlinkability: Different registrations in one or many servers can't be linked to the same authenticator *as long as the adversary is restricted to link / distinguish between authenticators of the same group (if $G \neq \perp$).*

SANDBOXAQ™

Authentication security

An adversary shouldn't be able to authenticate *on behalf* of any authenticator from a given group G, provided that it didn't have access to it or to its contents *or to the internal contents of other authenticators from the same group (if* $G \neq \perp$*).*

Privacy → Unlinkability

Group unlinkability: Different registrations in one or many servers can't be linked to the same authenticator *as long as the adversary is restricted to link / distinguish between authenticators of the same group (if* $G \neq \perp$*).*

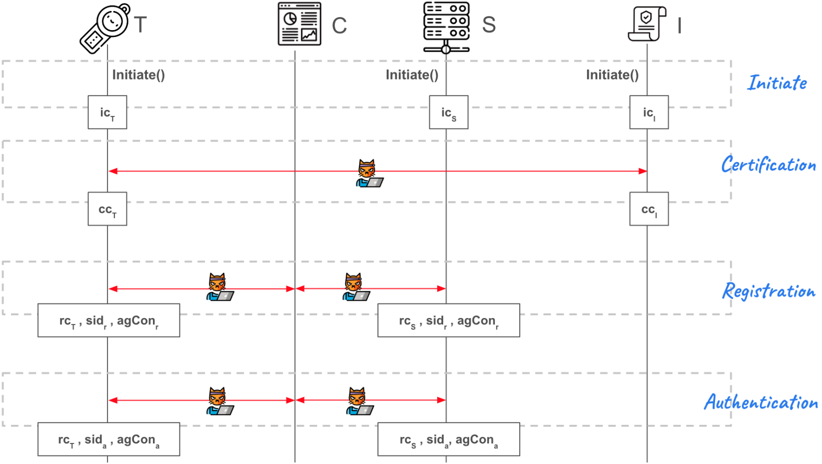
Summary of adversary capabilities

Phase	Authentication Security		Unlinkability	
	$\mathcal{A}$ type	Entities	$\mathcal{A}$ type	Entities
Initialisation I-T	None	I, T	Active	I, T
Initialisation I-S	Passive	I, S	Active	I, S
Certification	Active	I, T^*, C	Active	I, T^*, C^*
Registration	Active	T^*, C, S^*	Active	T^*, C^*, S
Authentication	Active	T^*, C, S^*	Active	T^*, C^*, S

The Adversary can...

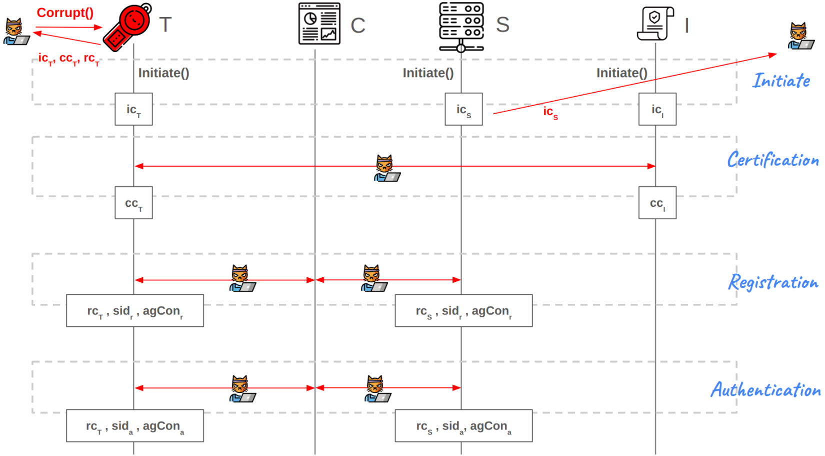
1) Create new authenticators and servers (automatically initialized with the information from an existing issuer)

2) See and modify communications in 3 phases



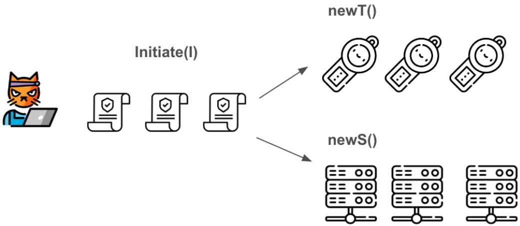
SANDBOXAQ™

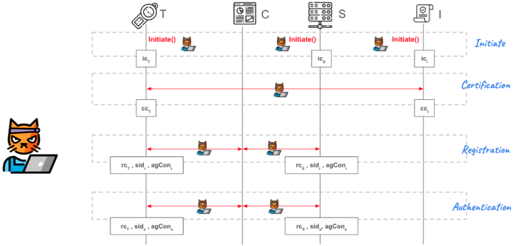
3) Corrupt authenticators and get issuer's public key



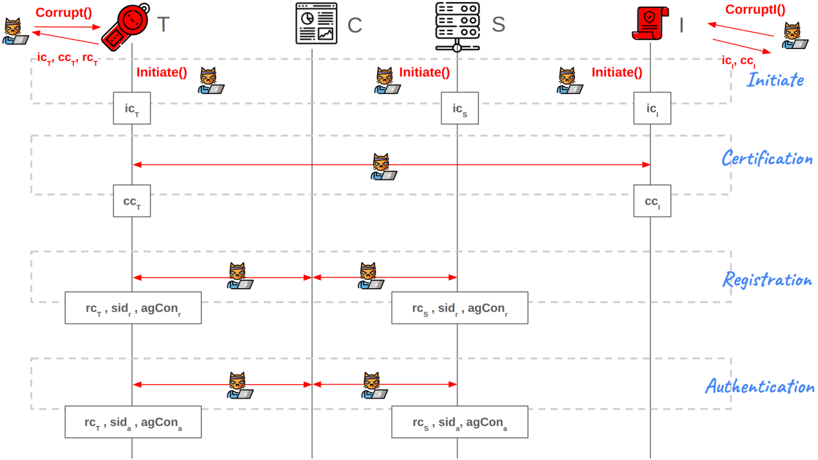
Unlinkability

The adversary can initialize issuers, authenticators and servers, and participate actively in all steps.





Also, the adversary can corrupt authenticators and issuers to get their internal state (including the issued certificates), **except the two authenticators it tries to distinguish from.**



SANDBOXAQ™

Results of the authentication security and privacy analysis

Attestation mode	PAuth-w	PAuth	Unl-w	Unl	$\mathtt{att_m}$
none	✓	✗	✓	✓	$\{\}$
self	✓	✗	✓	✓	$\{\}$
basic	✓	✓	✓	✓	$\mathtt{cert}_a^B$
attCA	✓	✓	✓	✗	pk_I

- PAuth-w: Only a passive adversary during registration.
- Unl-w: The adversary doesn't have access to the issuer internal information (public keys and generated certificates) through *corruptI()*.

Basic is the attestation mode providing best security and privacy capabilities, however a batch of authenticators share the same attestation credentials: compromise 1 $\rightarrow$ compromise all.

Can we do better?

Results of the authentication security and privacy analysis

Attestation mode	PAuth-w	PAuth	Unl-w	Unl	$\mathtt{att_m}$
none	✓	✗	✓	✓	{}
self	✓	✗	✓	✓	{}
basic	✓	✓	✓	✓	$\mathtt{cert}_a^B$
attCA	✓	✓	✓	✗	pk_I

- PAuth-w: Only a passive adversary during registration.
- Unl-w: The adversary doesn't have access to the issuer internal information (public keys and generated certificates) through *corruptI()*.

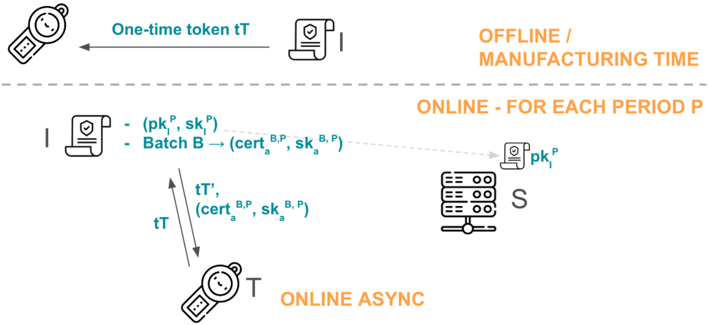
Basic is the attestation mode providing best security and privacy capabilities, however a batch of authenticators share the same attestation credentials: compromise 1 → compromise all.

Can we do better?

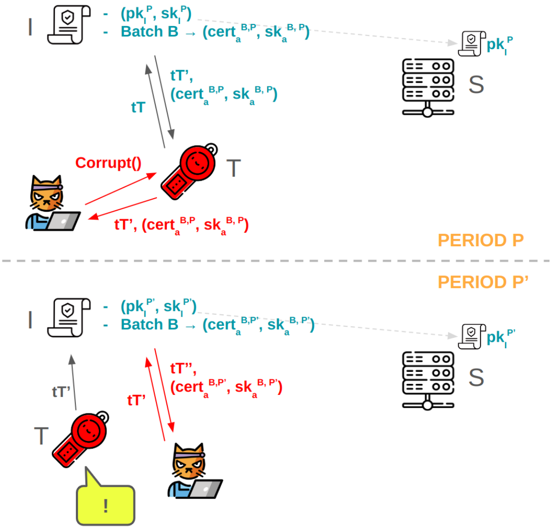
Part II - Simple Token Weaver

SANDBOXAQ™

A batch of authenticators share attestation credentials, like in attestation mode *basic*, but those **credentials are updated periodically**. Authenticators use a one-time token to obtain them.



In case of authenticator compromise, either the adversary is left out or the attack is detected.



Results with simpleTW

Results of the authentication security and privacy analysis

Attestation mode	PAuth-w	PAuth	Unl-w	Unl	$\mathtt{att}_{\mathtt{m}}$
none	✓	✗	✓	✓	$\{\}$
self	✓	✗	✓	✓	$\{\}$
basic	✓	✓	✓	✓	$\mathtt{cert}_a^B$
attCA	✓	✓	✓	✗	pk_I
simpleTW	✓	✓	✓	✓	$\mathtt{cert}_a^{B,P}, pk_I^P$

+

Post-Compromise Security

We can recover the security properties of a batch of authenticators after a compromise without having to replace all of them.

Thank you!

Manuel Barbosa, Alexandra Boldyreva, Shan Chen, and Bogdan Warinschi, *Provable security analysis of FIDO2*, Advances in Cryptology, 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part III (Tal Malkin and Chris Peikert, eds.), Lecture Notes in Computer Science, vol. 12827, Springer, 2021, pp. 125–156.

Nina Bindel, Cas Cremers, and Mang Zhao, *Fido2, CTAP 2.1, and webauthn 2: Provable security and post-quantum instantiation*, 44th IEEE Symposium on Security and Privacy, SP 2023, San Francisco, CA, USA, May 21-25, 2023, IEEE, 2023, pp. 1471–1490.

Cas Cremers, Charlie Jacomme, and Eyal Ronen, *Tokenweaver: Privacy preserving and post-compromise secure attestation*, IACR Cryptol. ePrint Arch. (2022), 1691.

Lucjan Hanzlik, Julian Loss, and Benedikt Wagner, *Token meets wallet: Formalizing privacy and revocation for FIDO2*, 44th IEEE Symposium on Security and Privacy, SP 2023, San Francisco, CA, USA, May 21-25, 2023, IEEE, 2023, pp. 1491–1508.