draft-janfred-eap-fido
A new EAP Method based on FIDO keys

IETF 118 in Prague – emu WG | 07.11.2023

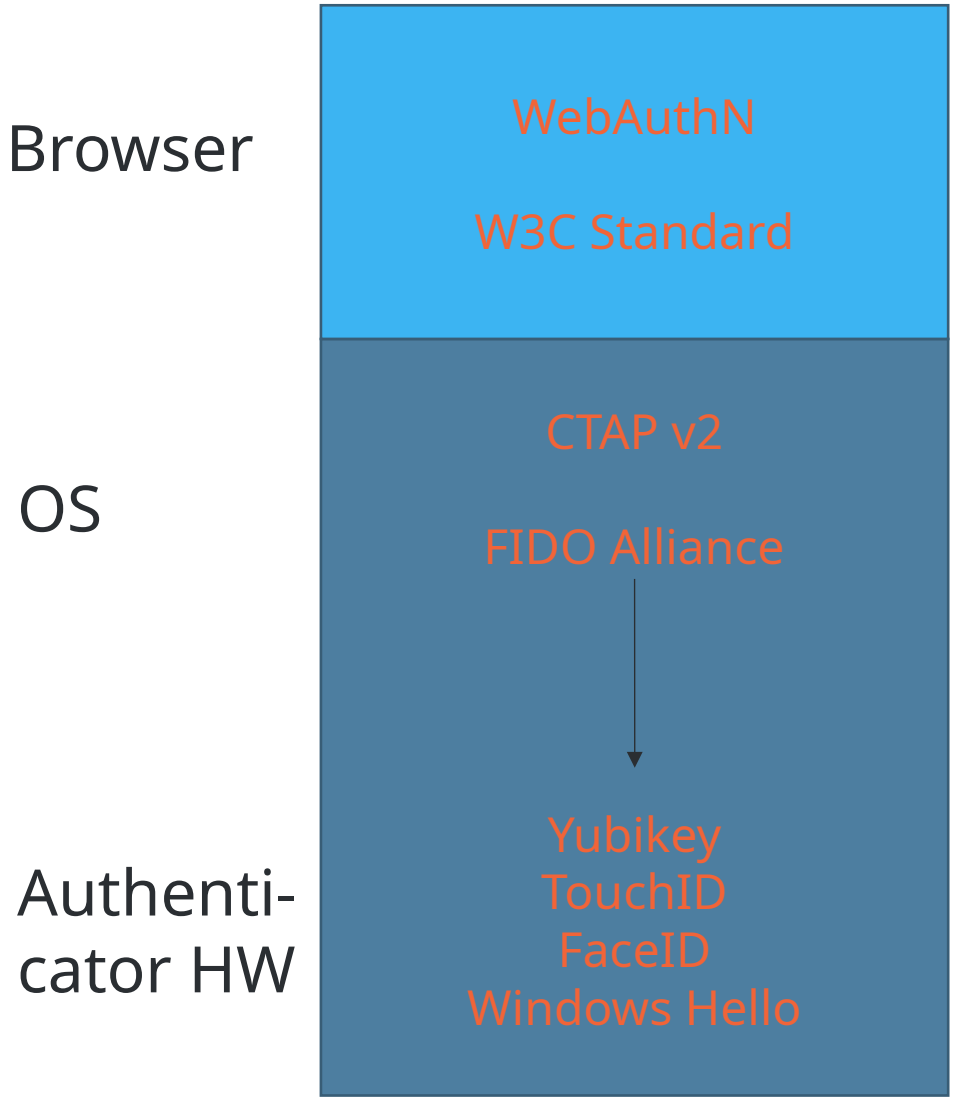
Janfred Rieckers | DFN-Verein
Stefan Winter | restena

Rewind @IETF115

Yours truly: „EAP and FIDO don’t go together”

FIDO2 almost exclusively used in web authentication contexts, and always requiring user action for every auth.

Underlying CTAP v2 ***can*** **be used in other contexts** though, **and without user action**!



Note: For brevity, individual option keys are often referred to as simply an "option", below.

Option Key	Default value	Definition
up	true	user presence: Instructs the authenticator to require user consent to complete the operation.

Within the "flags bits" of the authenticator data structure returned, the authenticator will report what was actually done within the authenticator boundary. The meanings of the combinations of the User Present (UP) and User Verified (UV) bit flags are as follows:

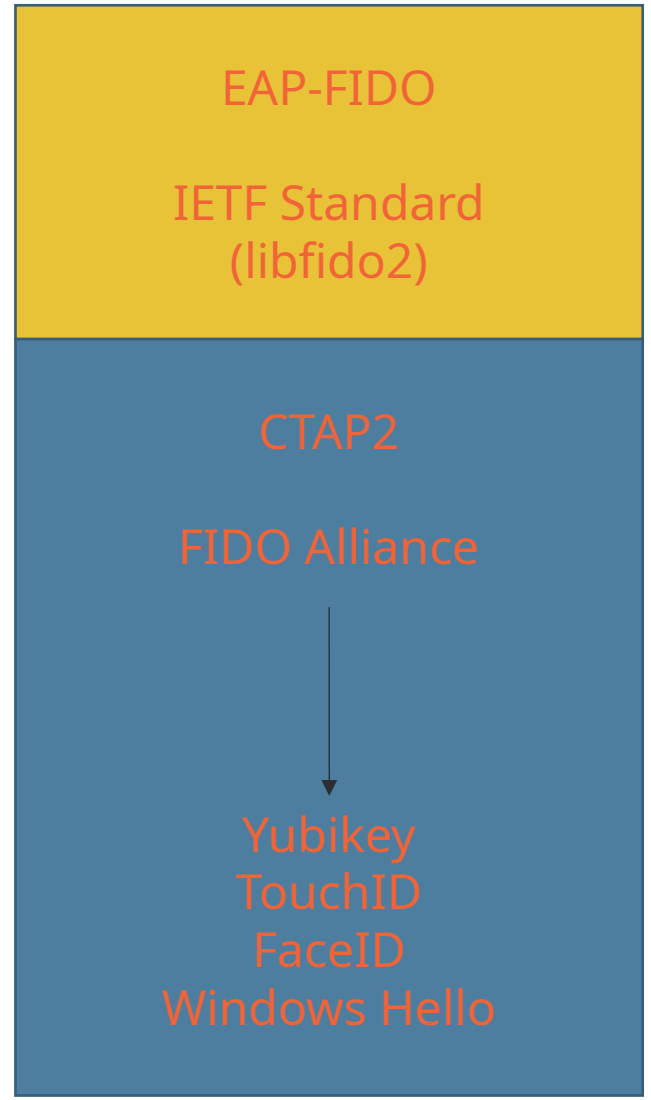
Flags	Meaning
"up"=0 "uv"=0	Silent authentication
"up"=1 "uv"=0	Physical user presence verified, but no user verification
"up"=0 "uv"=1	User verification performed, but physical user presence not verified. Note: Returning an assertion with the "up" bit set to `false` is not considered valid at the WebAuthn API layer [WebAuthn-2], and typically is only used for "pre-flighting".
"up"=1 "uv"=1	User verification performed and physical user presence verified

3.

Enterprise Wi-Fi

One can synthesize a private/public keypair on the device „for free“ (out of scope here/for now; done with WebAuthn).

Supplicant can interface via CTAP v2 to that keypair, and do signing transactions → client authentication „a bit like EAP-TLS“ (but not under the yoke of NotAfter!).



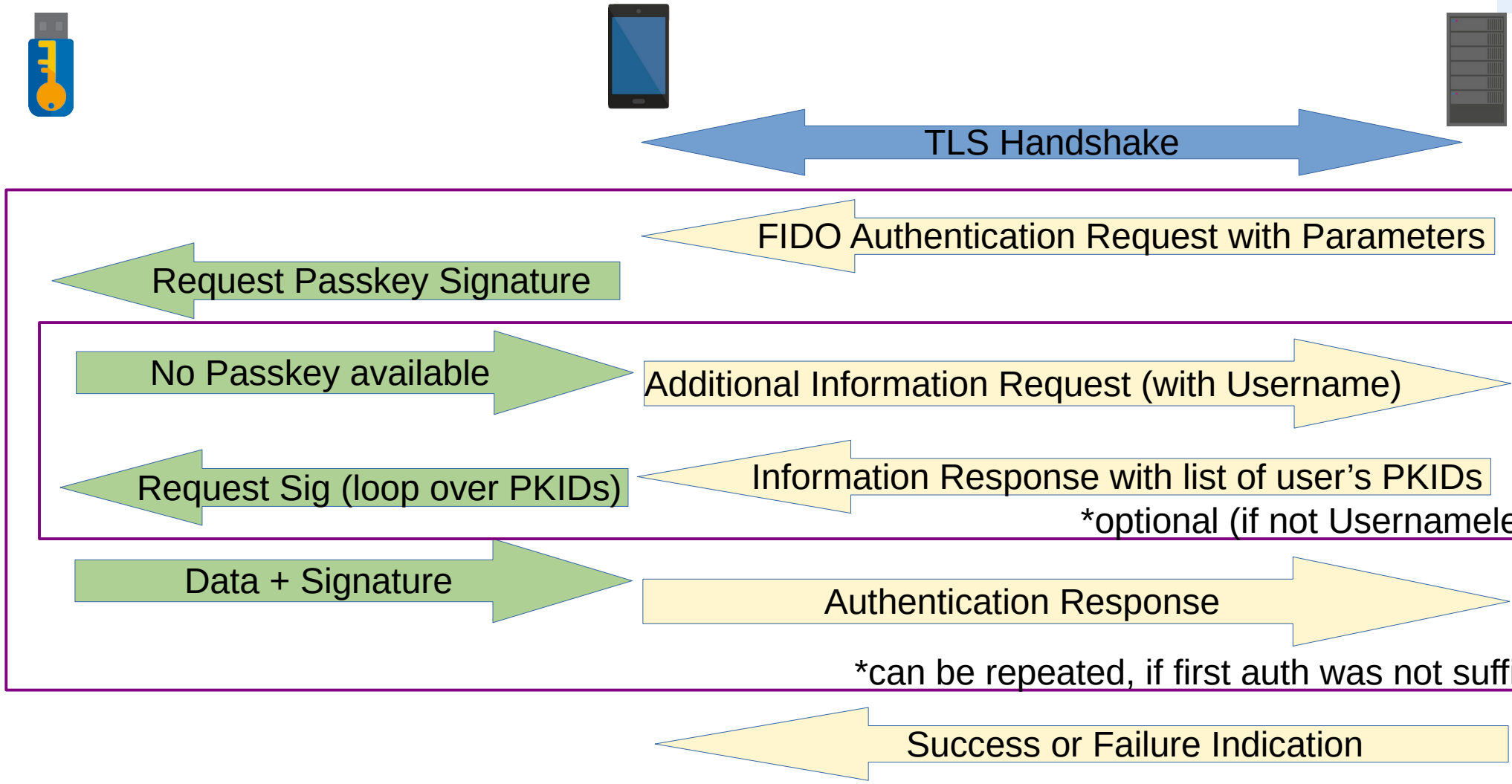
Background on problems with EAP

- Secure TLS configuration is not easy, users get it wrong constantly
  - „Do not validate“ is the most evil culprit. If you select it, it just works.
- Passwords are bad
  - Authentication through Knowledge by giving away the Knowledge is the root cause of phishing attacks. If the TLS cert check is broken, a rogue AP can intercept the password
- TLS Client Certificates need to be bootstrapped and they expire.
  - We need a device-specific provisioning mechanism
  - Users need to be reminded that they need to renew their certificate

Overview of the EAP-FIDO Protocol

- EAP-TLS based protocol with 2 phases
  - Phase 1: TLS Handshake
    - TLSv1.3
    - Server authenticates to the client through Certificate
  - Phase 2: FIDO authentication
    - Server sends authentication parameters (up/uv required, RPID?, …)
    - Supplicant requests signature from FIDO token through CTAP (v2+)
    - Supplicant sends signature back to the server
- Configuration: „One string to rule them all“
  - Aim to have only one string (ideally the institutions registered domain) that the user can be expected to know, everything else follows that.

EAP-FIDO Protocol Flow



EAP-FIDO Protocol Features

- Usernameless authentication
  - With Passkeys / residential Keys / discoverable Credentials a username does not need to be entered
- Dynamic authentication requirements
  - Server can request user presence / user verification either for all authentication flows (i.e. for VPN access) or after the initial authentication if the last up/uv auth for this specific key was too long ago.
- Minimal configuration
  - Users need to input only minimal configuration items, can’t get it wrong

Assumptions and not yet specified features

- EAP-FIDO does not provide provisioning
  - The EAP server must have access to a database with the PKIDs
  - Esp. Useful if the FIDO tokens are used in other contexts as well (i.e. for all logins for the home institution)
- De-provisioning of credentials not yet specified
  - See Discussion on EAP-DIE: We want to signal the supplicant to back off. (Either permanently or temporarily)
- Error handling
  - We need different errors than only returning „Failure“
  - Challenge: FIDO authentication does not fail verbosly, but for everything else we can send a more detailed error message

Results of the EAP-FIDO Side meeting

- Which configuration item is the „one string“?
  - NAI („anonymous identity“), expected Server Name or FIDO RPID
- Different ideas:
  - Have a static host prefix for cert (i.e. *eap-fido-authentication.dfn.de*)
    - Will the DNS people bite our head off over this?
  - SRV lookup for host name at time of provisioning
    - Requires internet connection at that time. (Captive Portal WiFi? Use LTE?)
    - Maybe with fallback to static host prefix?

Questions

- Feedback on the Protocol Design?
- Other opinions?

- Is this something that emu should/wants to work on?

Discussion/Questions?

- Contact
  - Jan-Frederik Rieckers

    Mail: rieckers@dfn.de
    Phone: 0049 30 884299-339
    Fax: 0049 30 884299-370

    Address:
    DFN-Verein, Geschäftsstelle
    Alexanderplatz1
    10178 Berlin

- Contact
  - Stefan Winter

    Mail: stefan.winter@restena.lu
    Phone: 00352 424409 1

    Address:
    Fondation Restena
    2, avenue de l'Université
    L-4365 Esch-sur-Alzette