Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Dipartimento federale della difesa, della protezione della popolazione e dello sport DDPS

Ufficio federale della cibersicurezza UFCS

Settembre 2025

Valutazione tecnologica

Chiavi d'accesso

1 Introduzione

Nonostante le password abbiano molti svantaggi a livello di tecnica di sicurezza, noti e documentati profusamente, esse rimangono ancora molto in voga e vengono impiegate diffusamente come procedura di autenticazione. Fino ad oggi nessuna delle procedure di autenticazione proposte in passato quale alternativa alle password è riuscita ad imporsi. Invece lo standard FIDO2 e le relative chiavi d'accesso sembrano più promettenti. Il presente documento presenta brevemente lo standard FIDO2 e le chiavi d'accesso, approfondendone e valutandone le caratteristiche a livello di tecnica di sicurezza.

2 Standard FIDO2

La «Fast IDentity Online (FIDO) Alliance[1]» raggruppa molti partner industriali animati dall'obiettivo comune di sviluppare e standardizzare procedure di autenticazione senza password quali alternative alle password. Sulla base degli standard FIDO Universal Second Factor (U2F) e FIDO Universal Authentication Framework (UAF), già adottati precedentemente, la FIDO Alliance ha sviluppato un Client to Authenticator Protocol in versione 2 (CTAP2) che, insieme al protocollo Web Authentication (WebAuthn) del «World Wide Web Consortium[2] (W3C)», costituisce il fulcro vero e proprio dello standard FIDO2.

Il funzionamento di un'autenticazione secondo lo standard FIDO2 e l'interazione tra i due protocolli WebAuthn e CTAP2 sono raffigurati schematicamente nella figura 1. Fondamentalmente l'utente può autentificarsi presso un server tramite un client qualsiasi. Al posto di una password si impiega una componente hardware dedicata («authenticator») che viene inserita internamente nel client o collegata al client come dispositivo esterno (di solito sotto forma di un token). Questo collegamento può avvenire in modo cablato tramite un'interfaccia Universal Serial Bus (USB) oppure senza fili tramite bluetooth, rispettivamente Near Field Communication (NFC). In tutti i casi l'autenticazione dell'utente avviene tra l'authenticator e il server, mentre il client si comporta in modo passivo (in altre parole il client trasmette solo i messaggi che devono essere scambiati tra l'authenticator e il server). Il protocollo WebAuthn (raffigurato su

[1] https://fidoalliance.org
[2] https://www.w3.org

sfondo color blu nella figura 1) definisce questo scambio di messaggi tra il client e il server, mentre il protocollo CTAP2 (raffigurato su sfondo color verde nella figura 1) descrive lo scambio di messaggi tra il client e l'authenticator.

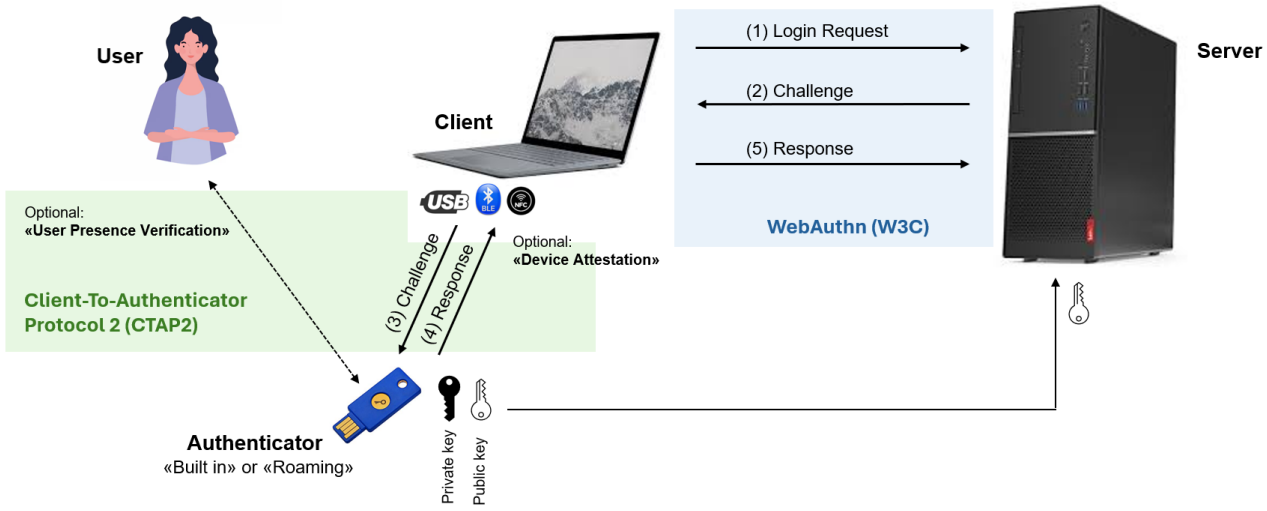

Figura 1: autenticazione secondo lo standard FIDO2

Affinché un authenticator possa essere impiegato per l'autenticazione di un utente presso un server referenziato tramite un URL è necessario che l'authenticator venga dapprima inizializzato. A tale scopo l'authenticator sceglie per l'utente una chiave doppia casuale e asimmetrica che viene associata all'URL del server. La chiave doppia è composta da una chiave privata (con cui si possono generare firme digitali) e una chiave pubblica (con cui è possibile verificare le firme digitali). La chiave privata rimane salvata nell'authenticator mentre la chiave pubblica viene resa nota al server.

Dopo questa inizializzazione l'utente può autentificarsi con l'authenticator presso il server con l'aiuto di una semplice procedura challenge-response, e/o di un semplice protocollo challenge-response. Tale protocollo prevede essenzialmente cinque fasi: nella fase (1) il client si rivolge al server con una login request. Nella fase (2) il server invia al client una challenge, che deve essere firmata con la chiave privata se si vuole ottenere una response valida. Questa firma può essere generata unicamente dall'authenticator con la chiave privata. Nella fase (3) il client invia all'authenticator la challenge e nella fase (4) l'authenticator risponde con una firma idonea come response. Per finire, nella fase (5) il client trasmette questa response al server. Tramite la chiave pubblica il server può quindi verificare la validità della response e, al contempo, anche l'autenticità dell'utente.

Dato che l'URL del server è coinvolto nella challenge e poiché l'authenticator lo verifica prima di generare la firma, la response può essere impiegata unicamente per l'autenticazione presso il server target vero e proprio. In tal modo il protocollo WebAuthn è in grado di proteggere efficacemente l'utente da numerosi attacchi di phishing e attacchi «Mallory-in-the-Middle» (MITM).[3] Inoltre mentre il protocollo è in corso di svolgimento, il client può verificare l'ammissibilità dell'authenticator («device attestation») e l'authenticator può controllare la presenza e

[3] Durante un attacco MITM l'aggressore si inserisce tra il client e il server, vale a dire il client e il server comunicano ognuno con l'MITM credendo invece di comunicare direttamente solo tra di loro. In alternativa o a complemento di questo meccanismo di sicurezza (finalizzato a coinvolgere l'URL del server target) si può anche impiegare il protocollo «token binding», che è specificato come estensione della TLS nei documenti RFC 8471 e 8472. Tuttavia il «token binding» della TLS non è ben supportato dai browser comunemente in commercio.

il consenso dell'utente («user presence verification»). Queste due verifiche sono però opzionali e possono variare da implementazione a implementazione. Nel caso più semplice e più frequente, tramite la device attestation viene imposto l'impiego di un determinato token. Anche la presenza e il consenso dell'utente sono verificati in virtù del fatto che occorre azionare un determinato tasto (nel token o nel client).

3 Chiavi d'accesso

Inizialmente l'obiettivo dello standard FIDO2 consisteva nel salvare in maniera sicura nell'authenticator la chiave privata dell'utente per una determinata applicazione, affinché risultasse non leggibile. Di conseguenza l'autenticità dell'utente è strettamente legata a questo authenticator («device-bound authenticators»). Sebbene ciò sia molto auspicabile nell'ottica della tecnica di sicurezza, tuttavia dal lato pratico fa sorgere alcune difficoltà nell'impiego di più authenticators, anche diversi tra loro, soprattutto se sono inseriti internamente e nei clients. Sarebbe più semplice se gli authenticators dell'utente potessero sincronizzare le chiavi private da essi salvate e gestite, ad esempio tramite un cloud storage («synced authenticators»). In questo modo si potrebbe infatti ottenere una certa indipendenza dal dispositivo. La relativa tecnologia è stata elaborata appunto dalla FIDO Alliance ed è denominata chiavi d'accesso («passkeys»). In effetti oggi il termine chiavi d'accesso è usato per riferirsi sia a questa tecnologia per l'autenticazione sia alle chiavi private che, grazie ad essa, possono essere impiegate al posto delle password.

Per l'implementazione delle chiavi d'accesso esistono molti gradi di libertà e possibilità di semplificazione, che vanno però tutti a detrimento della sicurezza. Come detto le chiavi d'accesso dell'utente possono venire sincronizzate su più dispositivi («across multiple devices»). Inoltre alcune implementazioni supportano anche il formato Credential Exchange Format (CXF) e il relativo protocollo Credential Exchange Protocol (CXP)[4]. In tal modo le chiavi d'accesso possono venire esportate anche singolarmente e reimportate su altri dispositivi. A volte questo scambio è possibile addirittura in modalità multiutente. Per l'utilizzo di authenticators che si trovano fisicamente vicini vi è la possibilità di una Cross Device Authentication (CDA) affinché si possano coinvolgere nell'ecosistema delle chiavi d'accesso anche i dispositivi che da soli non supportano le chiavi d'accesso. Inoltre anche nell'ambito della protezione hardware prevista originariamente sono possibili semplificazioni tali da consentire ad esempio ai password manager di gestire le chiavi d'accesso anche senza una protezione hardware dedicata nonché di proteggere le chiavi d'accesso salvate in maniera basata su software. Di conseguenza in un'implementazione concreta delle chiavi d'accesso occorre studiare molto bene in quale misura ci si discosta dallo standard FIDO2 originario per ottenere una migliore facilità d'uso e quali ne sono le implicazioni a livello di tecnica di sicurezza. Fondamentalmente la FIDO Alliance prevede a tale riguardo tre livelli (di sicurezza) per gli authenticators, rispettivamente per il loro Restricted Operating Environment (ROE),[5] anche se la granularità di questi (tre) livelli non è sufficiente per valutazioni e progettazioni effettive in materia di sicurezza.

[4] https://fidoalliance.org/specs/cx/

[5] Secondo https://fidoalliance.org/certification/authenticator-certification-levels/ nel primo livello (L1) è possibile l'impiego di componenti hardware e software a piacimento. Il secondo livello (L2) richiede l'impiego di un ROE sotto forma di un Secure Element (SE), rispettivamente di un Trusted Execution Environment (TEE). Il terzo livello (L3) richiede addirittura l'impiego di un Allowed ROE (AROE) approvato dalla FIDO Alliance.

4 Valutazione

A livello di tecnica di sicurezza, lo standard FIDO2 merita una valutazione positiva. Al posto delle password vengono impiegate moderne procedure crittografiche che sgravano l'utente. Tuttavia in prospettiva di un'ampia penetrazione sul mercato delle chiavi d'accesso si è rinunciato alla protezione hardware originariamente prevista, e alla non leggibilità del materiale costituito dalle chiavi. Infatti oggi come oggi le chiavi d'accesso consentono un'enorme flessibilità così da coprire quanti più ambiti d'impiego possibile.

Poiché dal punto di vista della sicurezza il diavolo sta sempre nei dettagli e questi dettagli dipendono anche dall'implementazione concreta, non è possibile fare valutazioni valide a titolo generale sulla sicurezza delle chiavi d'accesso. Cionondimeno per almeno tre motivi le chiavi d'accesso sono sempre migliori rispetto alle password:

(1) al contrario delle password, le chiavi d'accesso sono diverse per ogni singolo server target e vanno quindi compromesse una ad una;[6]
(2) al contrario delle password, le chiavi d'accesso sono generate in maniera pseudo-casuale e quindi non sono facili da indovinare;
(3) al contrario delle password, le chiavi d'accesso non vengono trasmesse tramite le linee di comunicazione dati e, di conseguenza, non è possibile leggerle estrapolandole dal traffico dati e abusarne per attacchi «replay».

Purtroppo però molti ambiti d'impiego delle chiavi d'accesso permetteranno di continuare a usare le password o renderanno quest'ultime addirittura obbligatorie (ad esempio come soluzione di backup per ripristinare le password), cosicché password deboli nonché pessimi processi di gestione delle password lasciano verosimilmente la porta aperta a eventuali attacchi. Questo dato di fatto va tenuto in debito conto in sede di valutazione dei sistemi di sicurezza basati sulle chiavi d'accesso.

In considerazione dei suoi vantaggi a livello di tecnica di sicurezza e del suo ampio supporto industriale, l'Ufficio federale della cibersicurezza (UFCS) prevede che lo standard FIDO2 si imporrà sul mercato sotto forma di varie implementazioni delle chiavi d'accesso. I grandi fornitori di tecnologia come Google, Apple e Microsoft si muovono parimenti in questa direzione e stanno sostituendo, laddove possibile, le password con chiavi d'accesso. Si stanno così ponendo le condizioni per l'autenticazione senza password. L'UFCS saluta con favore questo sviluppo e lo sostiene.

Abbreviazioni

AROE	Allowed ROE
UFCS	Ufficio federale della cibersicurezza
CDA	Cross Device Authentication
CTAP	Client to Authenticator Protocol
CXF	Credential Exchange Format
CXP	Credential Exchange Protocol

[6] Al contrario, invece, le password sono concepite e anche impiegate affinché le singole password possano venire usate per molti server o che le singole password possano differenziarsi tra loro anche solo in minima parte.

FIDO	Fast IDentity Online
MITM	Mallory-in-the-Middle
NFC	Near Field Communication
RFC	Request for Comments
ROE	Restricted Operating Environment
SE	Secure Element
TEE	Trusted Execution Environment
TLS	Transport Layer Security
U2F	Universal Second Factor
UAF	Universal Authentication Framework
URL	Uniform Resource Locator
USB	Universal Serial Bus
W3C	World Wide Web Consortium
WebAuthn	Web Authentication