A-SIT

Secure Information Technology Center – Austria

(Web) APIs for Identity Management

Web APIs for Identity Management

Author:
Stefan More
Mail: stefan.more@a-sit.at
Datum: October 2025

Abstract:

Browsers and the web community are continuously introducing new web browser interfaces for various use cases. Of particular interest here are APIs (Application Programming Interfaces) for identity management (IDM).

Up to now, identity management in the browser has primarily been based on generic web technologies (HTML, cookies, redirects, ...) and is now being more deeply integrated into the browser through the new interfaces. On the one hand, this ensures the desired functionality even as privacy requirements increase and web browsers become more restrictive (e.g. restricting "bounce redirects" and third-party cookies). Furthermore, IDM-specific browser APIs enable new use cases, such as deeper integration with the operating system or access to hardware.

Examples of such IDM-specific web APIs include the Digital Credentials API (DC API), Federated Credential Management (FedCM), and the Web Authentication API (WebAuthn).

Within the scope of this project, current APIs are analyzed and their state of development evaluated. In addition, current drafts and discussions in the area of IDM-specific web standards are examined.

Contents

1. Introduction

1.1. Motivation and scope

Identity management (IDM) on the web is moving from ad hoc page flows to explicit web platform APIs. Historically, single sign on (SSO) and eGovernment login solutions have used protocol specific redirects, cookies, and hidden iframes to implement SAML and OpenID Connect (OIDC) flows. These mechanisms are now stressed by privacy protections such as tracking prevention and the deprecation of third-party cookies. Further, new identity management paradigm (e.g., SSSI, wallets) necessitate deeper browser integration.

In response, browser vendors and standards bodies are defining identity specific web APIs, e.g.

- Federated Credential Management API (FedCM): browser mediated auth federation
- Web Authentication API (WebAuthn) and FIDO2 or passkeys for phishing resistant authentication
- Digital Credentials API (DC API): presentation and issuance of digital credentials from wallets

This report focuses on these APIs as web standards, rather than on specific vendor products or national implementations, and analyses their suitability as building blocks for public sector identity systems.

1.2. Regulatory and eGovernment context

In April 2024, the European Union adopted Regulation (EU) 2024/1183 (eIDAS 2), which amends Regulation (EU) No 910/2014 (eIDAS) and establishes a European Digital Identity Framework [5]. This framework introduces the European Digital Identity Wallet (EUDI Wallet) as a harmonised means for electronic identification and the sharing of attributes across Member States. Starting from late 2026, each Member State must provide at least one wallet for natural and legal persons.

The technical implementation of this framework is described in the European Digital Identity Architecture and Reference Framework (EUDI ARF). The ARF references several web standards, notably:

- Verifiable Credentials Data Model [7]
- OpenID for Verifiable Credential Issuance (OID4VCI) [8] and OpenID for Verifiable Presentations (OID4VP) [9]
- the Digital Credentials API, as a candidate wallet browser interface for remote, browser based flows [6,15]

Austria's national (eIDAS 1) eID system, ID Austria is already in production and uses OpenID Connect and SAML 2.0 to integrate service providers [16]. It further provides a qualified electronic signature (QES) that is legally equivalent to a handwritten signature throughout the EU [17]. ID Austria builds on a smartphone-backed authentication factor, but also supports FIDO2 tokens with WebAuthn as a second factor [18].

This report uses ID Austria as a concrete example of an existing eID solution, and EUDI wallets as the forward looking architecture under eIDAS 2.

2. Background: Identity Management on the Web

The Internet was built without an identity layer.
--- Kim Cameron, (Microsoft Chief Identity Architect, The Laws of Identity, May 2005)

2.1. Classical web single sign on

Classical web single sign on (SSO) is typically implemented by combining:

- HTTP redirects between the relying party (RP) and the identity provider (IdP)
- browser managed cookies at both sites to maintain sessions
- HTML or script based flows (including hidden iframes) to exchange protocol messages

The dominant protocols are Security Assertion Markup Language (SAML) 2.0 and OpenID Connect Core 1.0 (OIDC). OIDC standardizes authentication on top of OAuth 2.0, using signed JSON Web Tokens (ID Tokens) and supports discovery and dynamic client registration [1]. SAML 2.0 is widely deployed in the public sector and is considered mature and relatively static.

While this model is flexible and does not require explicit browser support, it has several limitations:

- Privacy and tracking: Cross site cookies and embedded resources can be used for tracking users, not just for authentication. This is especially problematic if big IdPs are used by many RPs (e.g., Google, or the federal government).
- Complex flows: Application developers must handle protocol specific redirects and error handling explicitly, and facilitate communication with each IdP separately.
- Cross device flows: Redirect based flows are cumbersome when the user initiates a transaction on one device and completes it on another. This is a challenge for use cases that build on desktop usage but require the smartphone as an second factor or for managing identity credentials (wallet).

2.2. Web standards building blocks

To address these issues, several web platform standards have emerged.

Credential Management API
The Credential Management API defines generic interfaces for credential storage and retrieval in browsers. It supports password credentials, federated credentials, and public key credentials and is the base upon which WebAuthn and the Digital Credentials API build.

Web Authentication API (WebAuthn) and FIDO2
WebAuthn defines an API for creating and using public key-based credentials bound to authenticators, with attestation and origin scoping [10]. Together with the FIDO Alliance's Client to Authenticator Protocol (CTAP), it forms the FIDO2 standard, and underpins passkeys. Passkeys are discoverable WebAuthn credentials integrated into operating systems and synchronized across devices.

Federated Credential Management API (FedCM)
FedCM is a W3C Federated Identity Working Group specification that defines a browser mediated API for federated sign in. It aims to preserve the benefits of identity federation while removing the dependency on third party cookies and opaque redirects [1,22].

Digital Credentials API[1] (DC API)
The Digital Credentials API extends the Credential Management API to allow user agents to mediate presentation and issuance of digital credentials (such as ID cards or driving licenses) from underlying platforms or wallets [3].

These APIs are complemented by data and protocol standards:
Verifiable Credentials Data Model v2.0, a W3C Recommendation for expressing verifiable credentials in a format agnostic way [7].
OpenID for Verifiable Credential Issuance (OID4VCI) and OpenID for Verifiable Presentations (OID4VP), which specify OAuth based APIs for issuing and presenting credentials and explicitly support transport over the Digital Credentials API [8,9].

2.3. eIDAS 2 and the European Digital Identity Wallet

Regulation (EU) 2024/1183 amends the eIDAS 1 regulation to establish a European Digital Identity Framework, under which Member States must offer at least one European Digital Identity Wallet. The wallet is a software product issued (or recognized) by a Member State. A wallet allows the user to identify online and offline, authenticate towards relying parties (RPs), and present attributes and credentials, issued by both public and private entities [5].

The EUDI Architecture and Reference Framework (ARF) specifies technical building blocks. For web interactions, it identifies:

- VC 2.0 and related W3C standards as the data model for credentials
- OID4VCI and OID4VP as preferred issuance and presentation protocols
- the Digital Credentials API as a candidate platform API between the wallet unit and the browser or operating system for remote flows [6,15]

3. Web Standards for Browser Mediated Identity

3.1. Federated Credential Management (FedCM)

3.1.1. Overview and standardization status

The Federated Credential Management API (FedCM) is defined by the W3C Federated Identity Working Group as a browser mediated alternative to classical federated sign in based on redirects and third-party cookies [1]. As of October 2025, the current specification is a First Public Working Draft on the W3C Recommendation track. The Working Group intends to advance it to Recommendation status once sufficient implementation experience is gathered.

The MDN describes FedCM as providing "a standard mechanism for identity providers to make identity federation services available on the web in a privacy preserving way, without the need for third party cookies and redirects" [22]. Chromium-based browsers already implement FedCM, and major identity providers, including Google, have migrated sign in flows to it.

[1] A-SIT previously evaluated an earlier version of the Digital Credentials API:
Implementation and Evaluation of the Digital Credentials API

3.1.2. Protocol model and architecture

FedCM introduces a JavaScript API (`*navigator.credentials.get()*` with federated options) and a browser-controlled account chooser UI. In a simplified FedCM flow:

1. The relying party (RP) calls the FedCM API with information about supported identity providers (IdPs) and requested scopes.
2. The browser queries the IdP's configuration endpoints defined by FedCM and the underlying protocol, possibly using the Login Status API for account information.
3. The browser presents a native dialog listing accounts and permissions to the user.
4. Upon user consent, the browser retrieves an assertion (for example an OIDC ID Token) from the IdP and returns it to the RP.

FedCM is deliberately protocol agnostic. OIDC is the primary underlying protocol in practice, but the Working Group emphasizes keeping protocol specific logic outside the FedCM specification.

A related proposal, often referred to as Lightweight FedCM, discusses simplifying IdP integration, for example by letting IdPs push account information to the browser via the Login Status API. This is still under discussion in the FedID Community Group and Working Group.

3.1.3. Privacy and security properties

FedCM aims to improve privacy (and security) compared to classic redirect based SSO:

- User consent and privacy: Since the login flow is mediated by the browser, the user is in control of the flow. The IdP does not learn about an user's intentions until the user consented to the login. This flow also enables tracking prevention mechanisms like BISON (see below).
- Browser mediated UI: The account chooser is controlled by the browser, which reduces opportunities for misleading fake login pages.
- Partitioned state: FedCM is designed to work without third party cookies. IdP state is partitioned per RP or per top level site, in line with tracking prevention mechanisms.
- Limited metadata leakage: The FedID Working Group and Community Group discuss timing attacks and caching strategies to avoid leaking which RPs a user logs in to [1].

However, FedCM remains IdP centric. The IdP still sees each authentication event. Preventing IdP based cross service tracking requires careful design of identifiers and policies.

3.1.4. Relevance for eGovernment

For eGovernment systems like ID Austria, FedCM could serve as a modernized front end for existing OIDC based login:

- The ID Austria OIDC endpoints and discovery documents are public [16].
- In principle, ID Austria could expose a FedCM configuration for its OIDC endpoints, allowing browsers to offer a native "Sign in with ID Austria" experience.

This would preserve federation benefits without third party cookies, improve user experience with a consistent browser UI, and reduce reliance on fragile redirect or cookie flows. For cross border use under the legacy eIDAS node model, FedCM could be used similarly with national IdPs.

In a wallet centric EUDI world FedCM is less relevant than the Digital Credentials API.

3.1.5. Privacy Extension: BISON Pseudonyms

A-SIT's BISON protocol ("Blind Identification with Stateless scOped pseudoNyms") proposes a way to derive per-RP pseudonyms using cryptography, using oblivious pseudorandom functions [19]. It is designed to integrate into existing protocols (notably OIDC) and hides the RP identity from the IdP while still producing stable, RP-scoped identifiers.

Further, BISON provides:

- an OIDC extension that uses BISON for pairwise pseudonymous identifiers
- prototype implementations for IdP and RP, plus a Firefox extension that simulates the required browser support [20]

Although BISON is not (yet?) a standard, it illustrates how FedCM plus OIDC deployments could, in the future, combine browser mediated user experience with strong cryptographic privacy guarantees for high sensitivity eGovernment domains.

3.2. Web Authentication API (WebAuthn) and FIDO2/Passkeys

3.2.1. Overview and standardization status

The Web Authentication API (WebAuthn) is a W3C Recommendation that defines an API for web applications to create and use public key credentials for strong authentication [10]. WebAuthn Level 1 and Level 2 are Recommendations. Work on Level 3 continues to refine capabilities and improve alignment with FIDO2 and passkey deployments.

WebAuthn works in conjunction with the Client to Authenticator Protocol (CTAP) defined by the FIDO Alliance. Together they comprise FIDO2, covering both the browser to authenticator interface (CTAP) and the web application API (WebAuthn).

3.2.2. Security properties

WebAuthn credentials have several important properties:

- Origin scoping: Credentials are scoped to a specific RP identifier, which prevents phishing and re-use.
- Public key-based: The server stores public keys. Private keys never leave the authenticator. This turns the authenticator into a factor suitable for single- or multi-factor authentication.
- Attestation: Authenticators can provide attestation statements that allow RPs (and eID schemes) to assess the security properties of the authenticator.
- Phishing resistance: WebAuthn is designed to neutralize classic phishing attacks by binding authentication to the origin shown in the browser [10].

FIDO2 authenticators can be platform authenticators (built into devices) or roaming authenticators (for example USB or NFC tokens). CTAP 2.2 further standardizes hybrid flows, where a mobile device acts as an authenticator for a desktop browser using QR codes and proximity channels such as Bluetooth [11].

3.2.3. Current deployment in eGovernment

In Austria, ID Austria already supports FIDO2 tokens with WebAuthn as a alternativesecond authentication factor. The responsible infrastructure operator A-Trust lists FIDO tokens as supported for triggering ID Austria, with the requirement that they are FIDO2 Level 2 certified and support WebAuthn [18].

This integration illustrates that high assurance eID schemes can incorporate WebAuthn or FIDO2 as part of their multi factor authentication chains, in line with eIDAS assurance level requirements.

3.2.4. Role under eIDAS

Under eIDAS 2 and the EUDI Wallet framework, WebAuthn and FIDO2 play two roles:

1. Authentication to RPs: Like in eIDAS 2, relying parties can use WebAuthn or passkeys directly as a factor for citizen login, including for public sector portals, provided that the overall scheme meets the regulatory assurance requirements.
2. Secure remote wallet flows: EUDI ARF Topic F describes remote same device and cross device flows where the browser interacts with a wallet over the internet, and cross device proximity is secured using CTAP 2.2 hybrid flows (QR and a proximity channel such as Bluetooth) [6,15].

3.3. Digital Credentials API (DC API)

3.3.1. Overview and standardization status

The Digital Credentials specification defines the Digital Credentials API as a web API that allows user agents to mediate the presentation and issuance of digital credentials [3]. The API extends Credential Management Level 1 and introduces a `*DigitalCredential*` credential type. RP websites can:

- request credentials from user agents (via `*navigator.credentials.get()*` with digital credential options)
- create or store credentials using issuance operations

The specification is developed by the W3C Federated Identity Working Group and has been published as a Working Draft on the Recommendation track [3,11,16].

The DC API specification emphasizes that the API is protocol agnostic, relying on a registry of supported protocols managed by the Working Group. The DC API is also subject to a thorough privacy and security review, as it can expose highly sensitive information [3,6].

3.3.2. Implementations in browsers and platforms

Implementation support has progressed rapidly:

- Chrome: An origin trial for the Digital Credentials API started with Chrome 128 and has since led to general availability starting from Chrome 141, enabling websites to verify user information using digital IDs in a privacy preserving way [12,13,21].
- Safari: Apple's developer documentation describes how Safari uses the Digital Credentials API for requesting mobile documents (ISO or IEC 18013 5 or 7 mdocs) from apps on the device, and notes that it uses the W3C Digital Credentials API, ISO standards for request formats, and FIDO CTAP for cross platform flows [14].

- Android: Android's Credential Manager exposes a *DigitalCredential* API which supports OpenID4VP and OpenID4VCI for presentation and issuance against wallet apps [25].

A 2025 overview notes that both Chrome and Safari now support the Digital Credentials API, with the precise feature set differing between platforms [24].

3.3.3. Integration with VC and OID4VCI or OID4VP

The Digital Credentials API is designed as a transport layer for higher level credential protocols and formats. The EUDI ARF and related documentation envisage that:

- credentials will generally conform to the Verifiable Credentials (VC) Data Model v2.0 [7]
- issuance and presentation flows will use OID4VCI and OID4VP as OAuth based APIs

The OpenID for Verifiable Presentations specification explicitly defines a mode where OID4VP messages are carried over the Digital Credentials API instead of HTTPS redirects [9]. This allows wallet RP communication to use protocol messages exchanged through the browser's DC API, with the wallet and the browser coordinating via the operating system.

3.3.4. DC API in EUDI Wallet and eIDAS 2

The EUDI ARF's Discussion Topic F, Digital Credentials API, describes the DC API as the candidate platform API for remote transaction flows [6,15].

Key points include:

- Same device flows: browser and wallet run on the same device. The DC API invokes the wallet through operating system provided interfaces.
- Cross device flows: browser and wallet are on different devices, but in proximity. The DC API is combined with CTAP 2.2 hybrid flows (QR code and a proximity channel) to secure the interaction.
- Wallet sovereignty: the wallet remains responsible for user consent, credential selection, and policy enforcement. The browser acts as a transport and mediation layer, not as an IdP.
- Scope: current work focuses on presentation. Attestation issuance is explicitly out of scope for the first version but may be considered in future updates [6].

In effect, the DC API provides the web standards connector between EUDI wallets and web relying parties.

3.3.5. Maturity and open issues

Although the Digital Credentials API has progressed quickly, moving from draft to origin trials and now deployment in major browsers, several issues remain under discussion:

- Governance of the protocol registry: which protocols and formats should be registered, how to avoid discrimination between credential formats, and how to reflect EU Implementing Acts
- Privacy risks: the spec notes that DC API can expose sensitive attributes and browsing behavior. Mitigations include strict user consent, restricted origin access, and minimization of linkable identifiers [3,6]
- Interoperability between platform level APIs and browser APIs: Android's DigitalCredential API in Credential Manager and platform specific mechanisms (for example to request ISO mdocs) must interoperate with the web level DC API [14,25]

From an eGovernment perspective, the Digital Credentials API is strategically central but still evolving.

4. Mapping Web Standards to eGovernment Use Cases

4.1. Representative eGovernment use cases

For this mapping, we consider six use cases:

- UC1: National citizen login to online services (for example Austrian government portal)
- UC2: Cross border authentication
- UC3: Qualified electronic signatures (QES)
- UC4: Attribute and credential presentation (for example professional licenses, age verification)
- UC5: Private sector use of eID or EUDI wallets (for example banking, telecom)
- UC6: High privacy scenarios (for example health, social services) with strong linkability constraints

4.2. Baseline: OIDC or SAML

ID Austria integrates service providers via OpenID Connect and SAML 2.0. The official documentation lists OIDC discovery and endpoint URLs and describes usage in browser based and native app scenarios [16,14]. ID Austria also provides a qualified electronic signature that is EU wide equivalent to a handwritten signature [17,18].

In this setup:

- UC1 and UC5 are well supported (national and private sector login)
- UC2 is supported via the eIDAS federation architecture
- UC3 is realized by coupling ID Austria authentication to QES services at trust service providers [19]
- UC4 is limited to attributes carried in assertions provided by the ID Austria system
- UC6 depends on operational policies rather than technical mechanisms

4.3. FedCM

For UC1 (national login) and UC5 (private sector login), FedCM can serve as a browser mediated front end to ID Austria's OIDC endpoints:

- The RP integrates FedCM. The browser uses the ID Austria OIDC configuration to interact with the IdP, and shows a native dialog "Sign in with ID Austria".
- This improves user experience and removes the dependency on third party cookies, in line with tracking prevention efforts [22].

For UC2 (cross border) in an IdP centric scenario (before full EUDI wallet adoption), FedCM could similarly front end foreign national IdPs connected to eIDAS nodes.

In high privacy scenarios (UC6), FedCM's privacy properties are better than classical SSO but still constrained by IdP visibility. Combining FedCM with privacy enhancing protocols such as BISON could further reduce tracking, but this requires additional standardization and implementation effort [19,20].

4.4. WebAuthn and FIDO2/Passkeys

WebAuthn/FIDO2 is already used in ID Austria as a second factor, and thus directly supports UC1 and UC3. FIDO2 Level 2 tokens with WebAuthn are accepted as an additional factor to trigger ID Austria [18].

The resulting authentication can be used to authorize QES operations, which are then executed by qualified trust services [17,19].

For UC2 (cross border), WebAuthn can form part of the authentication stack for notified eID schemes, provided that their assurance level profiles recognize the underlying authenticators.

For UC4 (attribute presentation), WebAuthn does not transport attributes itself.

In UC6 (high privacy), WebAuthn's origin bound, public key-based credentials limit cross RP linkability and provide strong phishing resistance, which is beneficial for sensitive eGovernment services.

4.5. Digital Credentials API and wallet-based flows

The Digital Credentials API is central to future wallet based eGovernment interactions:

- For UC1 and UC2, citizens can authenticate by presenting identity credentials (for example Person Identification Data) from their EUDI wallet to RP websites via DC API, using OID4VP over DC API [6,9,15].
- For UC3 (QES), the EUDI framework envisages wallet-based signatures and remote qualified signature creation devices (rQSCDs). DC API can carry the protocol messages needed to bind the signing session to the RP and the browser, although detailed Implementing Acts are still being developed.
- For UC4 (attribute presentation), DC API is explicitly designed to support selective disclosure of attributes from credentials, with cross device flows secured by CTAP 2.2 hybrid mechanisms [6,15].
- For UC5 (private sector reuse), the EUDI ARF foresees wallet-based interactions with banks, telecom operators and other private relying parties, again using web standards such as VC 2.0, OID4VCI or OID4VP, and DC API [6,15].

In UC6 (high privacy), DC API's protocol agnostic design, combined with VC 2.0 and privacy enhancing credential formats such as SD JWT VC, supports minimal disclosure and user-controlled consent [7]. The privacy risk remains that powerful wallets and browser or operating system platforms could be leveraged for surveillance. The EUDI framework attempts to mitigate this through governance and technical constraints.

5. Discussion

5.1. Design tradeoffs in browser centric identity

The move from ad hoc flows to browser centric identity APIs redistributes responsibilities. Browsers become active mediators in both federated sign in (FedCM) and wallet-based credential presentation (DC API). Identity providers and wallets must integrate with these APIs via standardized endpoints and protocols. Relying parties depend more directly on the web platform.

This has several implications:

- Security: Browser mediated flows can reduce phishing and click jacking risks and allow better integration with operating system level authenticators.
- Privacy: By controlling UI and partitioning state, browsers can limit tracking, but they also gain more visibility into identity flows, which must be constrained.
- Interoperability: Standard APIs can reduce fragmentation but also create new lock in vectors if only a subset of browsers or platforms adopt them, or if certain wallet providers dominate.

For public administrations, this means that choices made by browser and operating system vendors directly affect the feasibility and reliability of national eID and EUDI wallet solutions.

5.2. Security and privacy gaps and research directions

Despite significant progress, several gaps remain:

IdP centric versus wallet centric models: FedCM improves IdP centric SSO but does not eliminate IdP tracking. Wallet centric models using DC API and VC 2.0 can shift control towards the user but require more complex infrastructure and governance.

Cryptographic privacy enhancements: Research like BISON shows that per-RP pseudonyms can be derived in a way that hides the RP's identity from the IdP [19]. Similar ideas could be standardized in future extensions to OIDC and FedCM.

Secure cross device flows: Hybrid WebAuthn flows and DC API based cross device presentations significantly reduce phishing and relay risks, but real world deployments need careful user experience design and threat modelling, especially where citizens use multiple devices and wallets.

Transparency and governance: EUDI ARF Topic F emphasizes that wallets must remain sovereign. Browsers and operating systems should not silently override wallet choices or leak information about credential use [6]. Ensuring this in practice will require both technical means (for example APIs with clear consent and policy hooks) and regulatory oversight.

Post quantum aspects are currently not considered and require further research.[2]

5.3. Implications for public administrations

For public administrations, the analyzed web standards imply a staged evolution.

Short term:

- Improve existing OIDC or SAML deployments with WebAuthn or FIDO2 as strong authentication factors, as ID Austria already does.
- Consider FedCM for major national IdPs to improve user experience and resilience against tracking prevention changes.

Medium term (EUDI rollout):

- Align national wallet implementations and ID schemes with VC 2.0, OID4VCI or OID4VP, and the Digital Credentials API, in line with EUDI ARF.

[2] A-SIT is working in that direction in the projects PREPARED and POSEIDON.

- Pilot remote flows (same device and cross device) for high value use cases, including cross border access and attribute presentation.

Long term:

- Explore privacy enhancing extensions (for example BISON like pseudonymization, minimal disclosure credentials, advanced cryptography) and governance mechanisms ensuring wallet neutrality and platform accountability.
- Plan for coexistence of multiple models (IdP centric SSO, wallet centric identity, sector specific schemes) and define interoperability profiles.
- Plan for the post-quantum transition.

Public administrations also need to invest in testing and conformance: ensuring that web applications, wallets, and browsers correctly implement these standards and that changes in browser behavior do not silently degrade eGovernment services.

6. Conclusion

Browser integrated identity APIs are transforming how identity is managed on the web. FedCM and WebAuthn/FIDO2 are sufficiently mature to be used today to improve the security, usability and privacy of national eID systems, including eGovernment services. FedCM offers a browser mediated alternative to classic redirect based SSO, while WebAuthn underpins phishing resistant authentication and can satisfy high assurance level requirements when combined with certified authenticators.

The Digital Credentials API, together with Verifiable Credentials 2.0 and OpenID for Verifiable Credentials (OID4VCI or OID4VP), is emerging as the key web standards stack for wallet-based identity under eIDAS 2. It provides a general-purpose interface between EUDI wallets and web relying parties, enabling remote presentation of identity and attribute credentials in both same device and cross device scenarios.

However, the wallet-based ecosystem is still evolving. Implementations are emerging in Chrome, Safari, and Android, and the EUDI Architecture and Reference Framework continues to refine its guidance. Public administrations should therefore treat DC API based solutions as strategically important but not the sole foundation in the immediate term. EU funding of development for open source platforms and browsers could be a strategy building block for stability and European sovereignty.

By aligning national eID schemes such as ID Austria with identity-related web standards while recognizing the constraints and open issues, Member States can position themselves to benefit from the European Digital Identity Framework and enable secure, privacy respecting digital interactions for citizens and businesses.

References

[*] OpenAI. ChatGPT 5 [Large language model] was used to aid research and to edit parts of the text.
[1] W3C, "Federated Credential Management API", W3C First Public Working Draft, 2023.
[2] MDN Web Docs, "Federated Credential Management (FedCM) API", 2025.
[3] W3C, "Digital Credentials", W3C Working Draft, 2025.
[4] W3C FedID Working Group, "Digital Credentials API – GitHub repository".
[5] European Union, eIDAS 2: "Regulation (EU) 2024/1183 of the European Parliament and of the Council of 11 April 2024 amending Regulation (EU) No 910/2014 as regards establishing the European Digital Identity Framework", Official Journal L 2024/1183, 30 April 2024.
[6] European Digital Identity Cooperation Group, "Topic F – Digital Credentials API (EUDI ARF Discussion Paper)", 2025.
[7] W3C Verifiable Credentials Working Group, "Verifiable Credentials Data Model v2.0", W3C Recommendation, 2024.
[8] OpenID Foundation, "OpenID for Verifiable Credential Issuance 1.0", Final Specification, 2025.
[9] OpenID Foundation, "OpenID for Verifiable Presentations 1.0", 2025.
[10] W3C, "Web Authentication: An API for accessing Public Key Credentials Level 1", W3C Recommendation, 2019.
[11] FIDO Alliance, "Client to Authenticator Protocol (CTAP)", including CTAP 2.2 documentation.
[12] Chrome Developers, "Introducing the Digital Credentials API origin trial", 2024.
[13] Chrome Developers, "Digital Credentials API: secure and private identity on the web", 2025.
[14] Apple Developer Documentation, "Requesting a mobile document on the web", 2025.
[15] Deep summary of EUDI Digital Credentials API architecture, "Digital Credentials API Architecture – eudi.dev", 2025.
[16] ID Austria, "Anbindung mit OpenID Connect", ida.gv.at, accessed 2025.
[17] ID Austria or Austrian Federal Chancellery, "Generelle Informationen zur ID Austria", including QES information, 2025.
[18] A Trust, "FIDO Tokens – Als zweiten Faktor zur Auslösung der ID Austria", 2025.
[19] J. Heher, S. More, L. Heimberger, "BISON: Blind Identification with Stateless scOped pseudoNyms", arXiv:2406.01518, 2024.
[20] J. Heher et al., "BISON - GitHub repository", 2024.
[21] Chrome Platform Status, "Digital Credentials API (presentation support)", feature 5166035265650688, 2025.
[22] W3C FedID Working Group, "FedCM – GitHub repository", 2025.
[23] Corbado, "Digital Credentials API (2025): Chrome & Safari Support Live", 2025.
[24] Android Central, "Google makes it easier to share digital credentials on your Android phone", describing Android Credential Manager's support for OpenID4VP or OIDC4VCI, 2025.
[25] Example open source integration, "ID Austria demo – Next.js", GitHub repository rudgal/id-austria-demo.