Poster: FedCM for Research and Education

Erwin Kupris
Munich University of Applied Sciences
Munich, Germany
erwin.kupris@hm.edu

Tobias Hilbig
Munich University of Applied Sciences
Munich, Germany
tobias.hilbig@hm.edu

Thomas Schreck
Munich University of Applied Sciences
Munich, Germany
thomas.schreck@hm.edu

***Abstract*—The misuse of web technologies for tracking users on the Internet poses a threat to user privacy. Such technologies include third-party cookies and bounce tracking servers. Browser vendors and other stakeholders agreed to phase out some of these technologies in the near future. This impacts not only trackers and advertisers, but also legitimate usages such as authentication flows in identity federations. The industry aims to solve these issues via an emerging API called "Federated Credential Management" (FedCM), transforming the login process into a browser-mediated flow. Our research focuses on how to improve the user experience of FedCM within multilateral federations, which are frequently used in the Research and Education (R&E) sector. Specifically, we suggest ways to filter the large number of Identity Providers (IdPs) commonly found in the R&E context and display them, while automating the IdP discovery process. We provided our suggestions to the working group, also considering user privacy aspects. Incorporating these changes into the FedCM API accordingly could pave the way for a privacy-preserving and user-friendly sign-in experience in R&E federations.**

***Index Terms*—FedCM, federation, security, privacy, academia**

I. Introduction

The ongoing exploitation of certain web technologies for tracking poses a threat to user privacy and data protection. To mitigate these infringements, browser vendors have agreed to phase out some of these technologies. The most prominent example is third-party cookies, which are expected to be deprecated later this year. Bounce tracking, i.e., a specific chain of redirects invisible to the user, serves similar purposes. However, legitimate usages for these technologies do exist, which will be affected by this discontinuation.

Federated authentication mechanisms fall into this category. SAML2 and OAuth are the most commonly used protocols for this purpose. Both employ redirect-based login flows that are nearly indistinguishable from malicious usage. Moreover, federated applications often use services such as SeamlessAccess for discovering a user's Identity Provider (IdP) [1]. To offer a comfortable user experience, third-party cookies are required.

To prevent the breakage of these technologies, the Federated Credential Management API (FedCM) was proposed. It enables a secure, privacy-preserving, and dynamic authentication process mediated by the browser. FedCM is being developed by the W3C and has the status of a draft community group report. A working group with the goal of standardizing this API was recently established. The latest draft of the FedCM API is already implemented in the Chrome browser and experimental features can be tested in the Canary version.

In FedCM, the browser acts as a mediator between Relying Parties (RPs) and IdPs [2]. When a user visits an RP, it can call the FedCM API by providing one or more IdP URLs. The browser then issues requests to the IdP to obtain information about user accounts that have an active session with that IdP. Afterwards, the browser asks the user to select an account and consent to the federated authentication in a mediated dialog. Upon selection, the browser issues a final request to the IdP that includes the session cookie of the chosen account. The IdP returns an opaque identity assertion to the browser, which is relayed to the RP, thereby concluding the FedCM flow.

The current version of the FedCM API covers use-cases of social, bilateral federations, e.g., "Login with Google" or "Login with Facebook". However, multilateral federations, common in Research and Education (R&E), typically have different requirements. Although FedCM does not yet meet these requirements, it has the potential to not only fulfill them but also address other long-standing issues.

In this work, we propose ideas for extending FedCM to function effectively in the R&E sector. They can also be applied to other federated architectures such as Open Banking.

II. Problem Statement

R&E federations differ from bilateral federations by having the following specific characteristics: Instead of a single or a few public IdPs, users from R&E institutions can usually choose from thousands of IdPs when authenticating to an RP. Due to this extensive number of IdPs, "Where Are You From" (WAYF) services are used to determine a user's home IdP. Moreover, session lifetimes are commonly shorter, e.g., about an hour long. R&E IdPs usually do not offer a dedicated login page. Instead, when accessing a compatible service, users are redirected to the IdP for authentication. Finally, it is possible for RPs to not be explicitly registered at each IdP. These differences prevent the latest version of the FedCM API from being used in R&E federations.

Our contributions in this poster focus on two aspects of FedCM: (1) FedCM should only present compatible IdPs to the user, i.e., IdPs accepted by the RP and affiliated to the user. (2) FedCM should provide an enhanced user experience when the API is called using multiple, multilaterally federated IdPs. This would enable an automated, user-friendly WAYF process for R&E federations. In addition, we discuss privacy aspects that become relevant when extending FedCM to better support multilateral federations.

III. Filtering IdPs

In the early stages of FedCM, an RP could initiate the API using only a single IdP per call. However, RPs often provide federated authentication with multiple providers. This requires RPs to display individual buttons for each supported IdP, initiating FedCM for the chosen IdP. This is known as the so-called *NASCAR* problem, because users are presented with an overwhelming number of logos and buttons.

To avoid this issue in FedCM, two experimental features have already been added to the API: (1) A list of IdPs might be provided in the API call instead of a single one [3]. (2) IdPs can register themselves in the user's browser [4]. In the so-called *any* mode, RPs can then call FedCM for all registered IdPs without explicit knowledge about them. While still being actively worked on, both of these features enable RPs to initiate FedCM using a larger number of potential IdPs.

However, there are many scenarios in which only IdPs suitable for federated authentication with a specific RP should be considered in the FedCM procedure. For example, an RP might only be federated with IdPs that are part of the global R&E inter-federation eduGAIN. In this case, other IdPs should not be considered during the FedCM procedure to avoid unnecessary requests by the browser. This section provides suggestions for realizing this functionality.

A. IdP-list approach

RPs in R&E federations can usually fetch lists of compatible IdPs from a federation operator or a separate metadata discovery service. With this approach, an RP can already initiate FedCM using only IdPs it is federated with, omitting any unsuitable ones. While this solution is straightforward, it presents another challenge: A list of all IdPs with which an RP is federated might be extensive, e.g., there are more than 5,000 possible IdPs in eduGAIN. It is not practical for the browser to issue requests to all of these IdPs. Therefore, the following approaches might be considered to filter such extensive lists provided by the RP.

FedCM introduced the Login Status API to prevent unnecessary requests to IdPs. This API enables IdPs to set a status, either *logged-in* or *logged-out*, for their domain within the browser. Before initiating any requests to an IdP, the browser checks that the status for that IdP is set to *logged-in*. If the RP provides multiple IdPs, this verification process occurs concurrently for each. Consequently, only IdPs that have set this status, i.e., IdPs with which the user has previously interacted, are considered by the browser.

This solution works well for social federations because session lifetimes are typically very long. In other scenarios, however, sessions are usually shorter, especially in the academic or banking sectors. Such IdPs can utilize this API by consistently setting the status to *logged-in*, preventing them from being filtered out by the browser if the user is not logged into them. While this solution might be perceived as misusing the API, it would work with the current FedCM implementation in Chrome without necessitating any changes.

Instead of utilizing the login status API to infer with which IdPs the user has already interacted, FedCM's IdP registration can be used to filter IdP lists sent by an RP. In the proposed *any* mode, every IdP that has been previously registered in the browser is considered for the regular FedCM flow. Because the RP does not call the API with IdPs that it is compatible with, this mode needs to be adjusted to not consider unsuitable IdPs that might have registered. Such an adjustment can combine the IdP registration with lists of IdPs sent by RPs during the API call, resulting in a *some* or *certain* mode. If an IdP has been previously registered in the browser and is included in the list of compatible IdPs provided by the RP, it can reasonably be assumed that it is suitable for federated authentication. Therefore, such an IdP should be considered by FedCM, regardless of its login status.

B. affiliationHint approach

Instead of the RP sending a whole list of IdPs, we suggest it signals its federation affiliations. This could be realized via an additional attribute, e.g., *affiliationHints*, that the RP provides to FedCM. Similarly, the IdP would mark its federation affiliations during IdP registration. When the API is called, the browser compares IdPs registered with identical *affiliationHints* and exclusively considers exact matches. Since R&E federations are often structured hierarchically, this attribute can contain multiple entities, as shown in the following example.

The Munich University of Applied Sciences (HM) is a member of the German federation DFN-AAI and the eduGAIN inter-federation. HM's IdP registered itself in the browser with *affiliationHints = [”hm.edu”, ”dfn.de”, ”edugain.org”]*. The user accesses an RP operated by the Sapienza University, which is a member of the Italian GARR and eduGAIN. The RP sends *affiliationHints = [”uniroma1.it”, ”garr.it”, ”edugain.org”]* within the API call. The browser then determines that RP and IdP share an affiliation, i.e., eduGAIN. Afterwards, the regular FedCM flow continues with HM's IdP.

If an entity is part of more federations, this list can be simply extended. In scenarios where such a clear hierarchy does not exist, other hints might be used, e.g., *”EU-bank”* or *”US-bank”*. The exact structure of these hints must be clearly defined to ensure alignment between RPs and IdPs. Furthermore, both RPs and IdPs can restrict their *affiliationsHints* as they desire. For example, by excluding eduGAIN from its hints, an RP can ensure that only users within its own national federation are presented with an option to access it via FedCM.

C. OpenID Federation Approach

Instead of *affiliationHints*, the RP can call FedCM by including OpenID Federation trust chains [5]. Each trust chain represents a signed path from the RP to one of its trust anchors. For filtering IdPs, the *entityIds* within a trust chain can be parsed and used similarly to the *affiliationHint* approach. The RP's trust chains can subsequently be used to verify the trust relationship between the RP and IdP. This approach is similar to our previous study, which proposes a way to automate the IdP discovery process in multilateral federations [6].

IV. WAYF OPERATION MODE

As previously discussed, users within R&E federations can authenticate at federated RPs via thousands of potential IdPs. Consequently, RPs in such federations commonly integrate WAYF services to determine the IdP with which the user wants to authenticate. This process is cumbersome from a user experience perspective, because it frequently involves selecting one's home organization from an extensive list. Additionally, the user experience of WAYF services is further impaired by the deprecation of third-party cookies. The emerging FedCM API has the opportunity to automate this long-standing issue within R&E federations.

When the FedCM API is called by the RP, it currently offers the user a selection of logged-in accounts. We propose FedCM to incorporate an "organization chooser" in addition to the current account chooser dialog. This dialog leverages some of FedCM's experimental features, i.e., IdP registration and "button mode". Instead of showing only logged-in accounts, the selection dialog should include organizations that have registered themselves in the browser and are not filtered out by the methods proposed in Section III. If the user selects an organization, a subsequent login should be facilitated at the IdP by opening a pop-up window at the login URL the IdP previously registered with. However, R&E IdPs might not allow users to authenticate at the IdP directly. Instead, the federated login procedure can only be initiated via a redirect from an RP, including the necessary parameters such as the RP's entity identifier. Therefore, R&E IdPs will need to develop alternative solutions, e.g., integrating a separate RP at the login URL.

If FedCM were to be adopted in R&E federations in the future, changes to all involved IdPs and RPs would be necessary. During the transition period, it would be beneficial for FedCM to support basic WAYF functionality. Normally, FedCM requests an opaque token from the IdP for the selected account and returns it to the RP. Instead of returning a token, we suggest that FedCM should offer an option to return the IdP selected by the user. This can be realized via an additional parameter called *wayf* that is set by the RP in the API call. Upon selection, the browser skips the retrieval of the token from the assertion endpoint. After the selected IdP is returned, the existing, possibly redirect-based federated login flow is executed. Realizing this functionality would require minimal changes to the affected RPs, IdPs, and the FedCM API.

V. PRIVACY CONSIDERATIONS

After a list of logged-in accounts has been queried, the FedCM flow continues by fetching client metadata about the RP from the IdP. Apart from receiving the RP's metadata, this request also ensures that a trust relationship between IdP and RP exists. In the public IdP use case, RPs are always registered at the IdP, making such a request possible. However, in multilateral federations, this is often not the case. Instead, metadata is either centrally managed, for example in SAML2, or resolved dynamically in OpenID Federation.

In SAML2 based federations, the IdP typically maintains and regularly updates extensive XML files containing the metadata of all RPs within a federation. Therefore, the IdP can simply locate the RP's metadata in these files and return it. For federations based on the OpenID Federation protocol, this process presents a potential privacy challenge. It is vital for FedCM to never disclose the user's affiliation, i.e., their IdPs, to an RP before they give explicit consent. The regular metadata resolution in OpenID Federation would violate this rule if the IdP started this process with a request to the RP [5]. A malicious RP can correlate such a request with the user's running browser session and infer the user's affiliation. In our previous work, we presented an alternative approach to verifying the trust relationship between IdP and RP [6]. This method requires the RP to initiate the FedCM API by including its OpenID Federation trust chains, as stated in Section III-C. In addition, this solution does not disclose the RP a user visits to the IdP before the user consents.

VI. CONCLUSION AND FUTURE WORK

The Federated Credential Management API (FedCM) is an emerging standard that aims to improve privacy, security, and the overall user experience of authenticating to federated webservices. In the R&E sector, specific requirements exist that FedCM, as of today, does not fully cover. Our poster presents ideas on how this API can be extended to better support the R&E sector and to offer a user friendly sign-in experience. We have already proposed these suggestions to the working group [7].

In future work, we plan to build a proof of concept of FedCM at our university's IdP. This includes implementing a FedCM plugin for the Shibboleth IdP software and integrating the necessary API endpoints. Furthermore, we plan to analyze its security through a threat model analysis. Finally, we envision a usability study representative of higher education institutions, including students, staff, and faculty members.

ACKNOWLEDGMENTS

We are grateful for having had the opportunity to discuss our initial idea for this work at TDI / OSW 2024, particularly with Giuseppe De Marco, Tim Cappalli, and Samuel Goto.

REFERENCES

[1] Coalition for Seamless Access, "SeamlessAccess," 2024. [Online]. Available: https://seamlessaccess.org/

[2] N. P. Moreno, "Federated Credential Management API," W3C, Draft Community Group Report, Mar 2024. [Online]. Available: https://fedidcg.github.io/FedCM/

[3] W3C FedID CG, "Allow multiple idps to be used," GitHub Issue, 2022. [Online]. Available: https://github.com/fedidcg/FedCM/issues/319

[4] T. Looker et al., "Allow IDP registration," GitHub Issue, 2023. [Online]. Available: https://github.com/fedidcg/FedCM/issues/240

[5] R. Hedberg, M. B. Jones, A. A. Solberg, J. Bradley, G. De Marco, and V. Dzhuvinov, "OpenID Federation 1.0 - draft 34," The OpenID Foundation, 2024.

[6] E. Kupris, T. Hilbig, D. P. Sugar, and T. Schreck, "A-WAYF: Automated Where Are You From in Multilateral Federations," in *2nd International Workshop on Trends in Digital Identity (TDI 2024)*, 2024.

[7] E. Kupris and T. Hilbig, "FedCM for Research and Education," GitHub issue, 2024. [Online]. Available: https://github.com/fedidcg/FedCM/issues/563

FedCM for Research and Education

Erwin Kupris Tobias Hilbig Thomas Schreck

HM Munich University of Applied Sciences

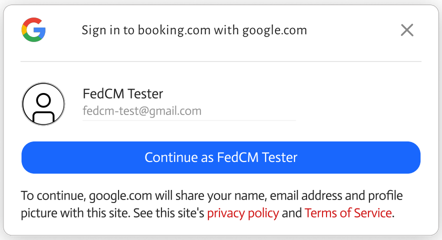
Problem Statement

Current web technologies are misused for **tracking users** on the Internet, including **third-party cookies**, **link decoration**, and **bounce tracking**. Major browsers will phase out these technologies in the near future to enhance user privacy. However, this also impacts legitimate usages, such as authentication flows in identity federations.

FedCM [1] is an emerging technology that aims to solve this issue by providing a **privacy-preserving** and **user-friendly sign-in experience**. Current implementations meet the requirements of bilateral federations, such as social logins. However, Research and Education (R&E) federations are multilateral, which leads to several conflicts with FedCM. Therefore, this research aims to **advance FedCM's compatibility with R&E federations** by proposing a number of improvements.

What's FedCM?

The **Federated Credential Management API** [1] by the W3C Federated Identity Community Group is currently in draft status. It is already available in Chromium-based browsers, Social-Login IdPs (Google) and some RPs. The browser mediates the login process between Relying Party (RP) and Identity Provider (IdP) in a privacy-preserving and user-friendly manner. Experimental features include: Support for multiple IdPs in one API call, IdP registration in the browser, facilitating sign-in at the IdP.



Sign in to booking.com with google.com

FedCM Tester
fedcm-test@gmail.com

To continue, google.com will share your name, email address and profile picture with this site. See this site's privacy policy and Terms of Service.

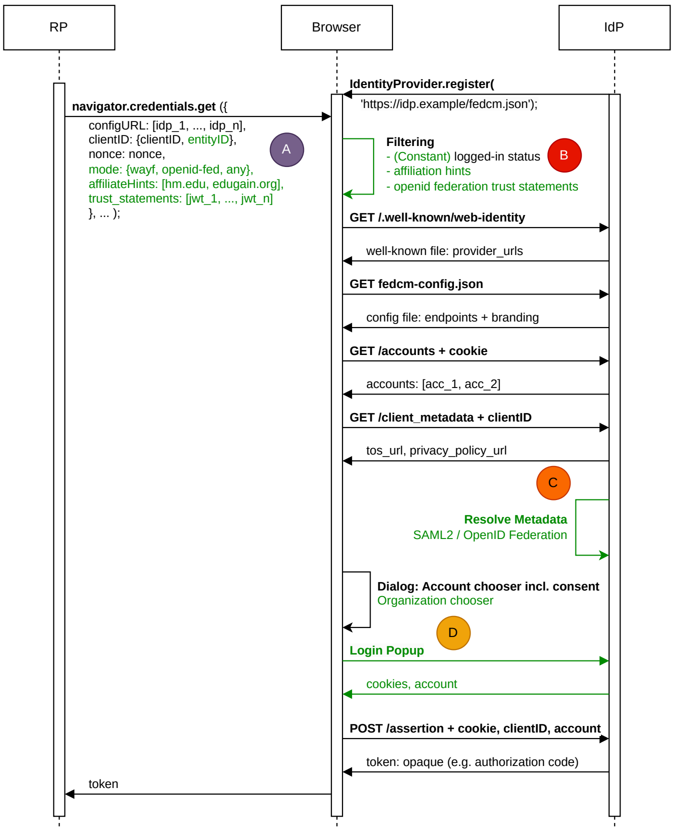
Figure 1. Mockup of FedCM's standard account chooser dialog in Chrome while accessing the RP booking.com.

R&E Federations vs. Social Logins

There can be thousands of IdPs that users can choose to authenticate with at an RP [2]. Where Are You From (WAYF) services are used to find a user's home IdP, e.g., SeamlessAccess [3]. In R&E federations:

- Session lifetimes at IdPs are usually short
- IdPs usually do not offer direct login pages like Google or Facebook
- RPs do not have to be explicitly registered at each IdP

FedCM Sequence Diagram



navigator.credentials.get ({
configURL: [idp_1, ..., idp_n],
clientID: {clientID, entityID},
nonce: nonce,
mode: {wayf, openid-fed, any},
affiliateHints: [hm.edu, edugain.org],
trust_statements: [jwt_1, ..., jwt_n]
}, ...);

IdentityProvider.register(
'https://idp.example/fedcm.json');

Filtering
- (Constant) logged-in status
- affiliation hints
- openid federation trust statements

GET /.well-known/web-identity

well-known file: provider_urls

GET fedcm-config.json

config file: endpoints + branding

GET /accounts + cookie

accounts: [acc_1, acc_2]

GET /client_metadata + clientID

tos_url, privacy_policy_url

Resolve Metadata
SAML2 / OpenID Federation

Dialog: Account chooser incl. consent
Organization chooser

Login Popup

cookies, account

POST /assertion + cookie, clientID, account

token: opaque (e.g. authorization code)

token

Figure 2. Sequence diagram of the standard FedCM flow (black) and our proposed additions (green).

Filtering IdPs

FedCM supports the Login-Status API [1] for querying status information related to accounts. IdPs without logged-in accounts are currently not shown to the user. While FedCM for R&E should also filter unsuitable IdPs, shorter session lifetimes must be considered as well. We therefore propose three possible solutions:

IdP-list and constant logged-in status:

- An RP calls the FedCM API using a (large) list of IdPs it is federated with
- IdPs have a constant logged-in status. This works already, but might be considered misusing the API
- Our suggestion: Treat previously registered IdPs differently and do not require them to be logged-in
- Possible issue: Full list might be too extensive (>5k entries)

affiliationHints:

- IdPs use their affiliation for registration, e.g.:
  `affiliationHints = ["uniroma1.it", "garr.it", "edugain.org"]`
- RPs send affiliation in the API call, e.g.:
  `affiliationHints = ["hm.edu", "dfn.de", "edugain.org"]`
- Result: If the browser does not find a match, it filters out the IdP

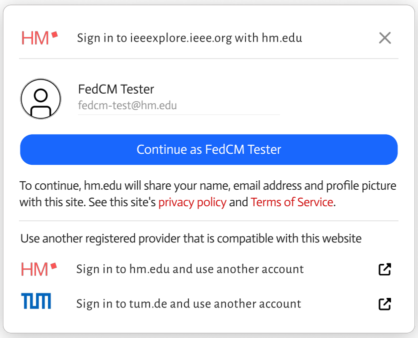
OpenID Federation Trust Chains:

- RP calls the API using its OpenID Federation [4] trust chain
- entityIDs within the statements of the chain represent affiliations
- Filtering works similarly to affiliationHints
- Side effect: Browser can use the trust chain to verify the trust relationship between the RP and IdP

Organization Chooser and Automated WAYF

WAYFs often have poor UX and are also impacted by the deprecation of third-party cookies. With FedCM for R&E, there is an opportunity to automate this cumbersome process:

- Add an organization chooser to the account chooser dialog
- Shows organizations / IdPs that previously registered in the browser, even without logged-in accounts
- Open a popup window at the IdP, if the user selects an organization
- FedCM then queries the account endpoint again and directly fetches the token
- Side effect: Short session lifetimes would no longer be an issue
- Problem: R&E IdPs often do not offer direct logins → Integration of an RP service at this URL needed



Sign in to ieeexplore.ieee.org with hm.edu

FedCM Tester
fedcm-test@hm.edu

To continue, hm.edu will share your name, email address and profile picture with this site. See this site's privacy policy and Terms of Service.

Use another registered provider that is compatible with this website

Sign in to hm.edu and use another account

Sign in to tum.de and use another account

Figure 3. Mockup of the envisioned FedCM organization chooser dialog.

Privacy Considerations

By design, FedCM fetches the metadata of the RP from the IdP and verifies their trust relationship. In R&E federations, RPs are not explicitly registered with each IdP. Therefore, IdPs in FedCM for R&E need to discover the RP's metadata first:

- SAML2 [5]: IdPs can query centrally managed metadata files, e.g., from eduGAIN
- OpenID Federation: IdPs resolve metadata dynamically, starting with a request to the RP
  → Potential privacy issue because this leaks the user's IdP to the RP before consent!
- Mitigation using OpenID Federation trust chains presented in detail in previous work [6]
- Alternative: IdPs could resolve trust and metadata differently, e.g., directly via the trust anchor

Future Work

- Already proposed suggestions to the community group [7]
- Build a PoC at our university's IdP by implementing a Shibboleth plugin
- Analyze the resulting security through a threat model analysis
- Analyze resulting usability through a study representative of higher education institutions

References

[1] N. P. Moreno, "Federated Credential Management API," W3C, Draft Community Group Report, Mar 2024. [Online]. Available: https://fedidcg.github.io/FedCM/

[2] eduGAIN, "eduGAIN entities database," 2024. [Online]. Available: https://technical.edugain.org/entities

[3] Coalition for Seamless Access, "SeamlessAccess," 2024. [Online]. Available: https://seamlessaccess.org/

[4] R. Hedberg, M. B. Jones, A. A. Solberg, J. Bradley, G. De Marco, and V. Dzhuvinov, "OpenID Federation 1.0 - draft 34," The OpenID Foundation, 2024.

[5] N. Ragouzis, J. Hughes, R. Philpott, E. Maler, P. Madsen, and T. Scavo, "Security assertion markup language (saml) v2.0 technical overview," OASIS, Tech. Rep., 2008.

[6] E. Kupris, T. Hilbig, D. P. Sugar, and T. Schreck, "A-WAYF: Automated Where Are You From in Multilateral Federations," in *2nd International Workshop on Trends in Digital Identity (TDI 2024)*, 2024.

[7] E. Kupris and T. Hilbig, "FedCM for Research and Education," GitHub issue, 2024. [Online]. Available: https://github.com/fedidcg/FedCM/issues/563

seclab.cs.hm.edu 9th IEEE European Symposium on Security and Privacy, Vienna, July 8-12, 2024 erwin.kupris@hm.edu