



Federated Identity & The Federated Credentials Management API

Kristen Chapman
kristen.chapman@salesforce.com

What Is Identity Federation?

OAuth, OpenID, SAML, Federated Identity Management, authorization vs. authentication, credentials, SSO, security domains, and so on and so on...

There are a lot of terms associated with identity federation - and it's often pretty confusing.

At its most basic, though, identity federation is simply about linking an individual's digital identity across different sites or services.

(example: using a Twitter account to log in to a media site)



Why Talk About It Here?

1

Advertising
is arguably one of
the economic
backbones of the
web

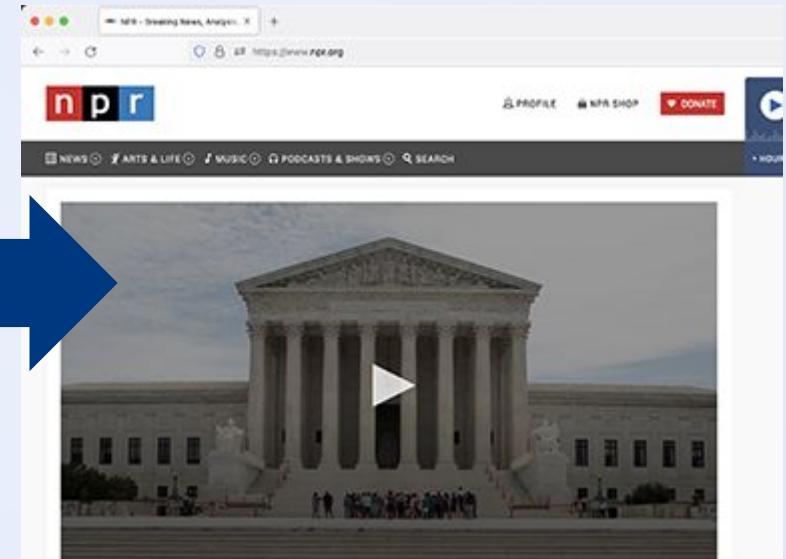
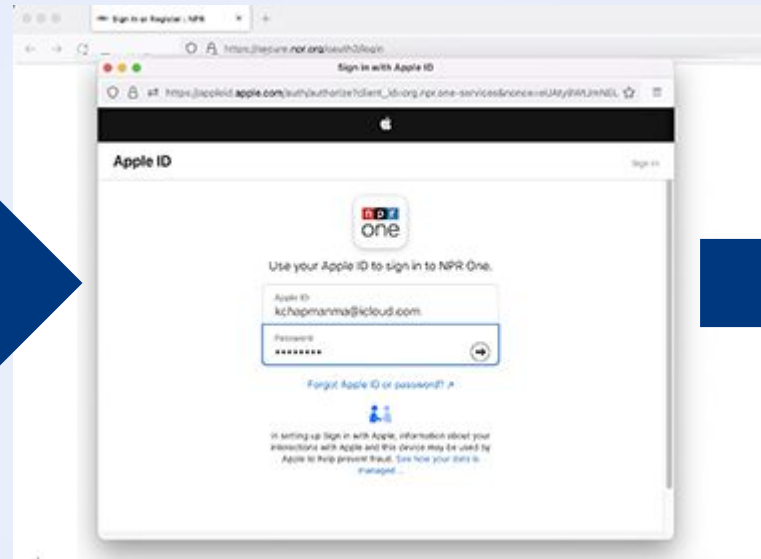
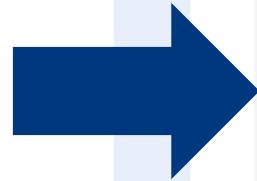
2

**Federated
Identity**
could be
considered one of
the usability
backbones of the
internet

3

The Overlap
is they both rely on
the same web
primitives:
third-party cookies,
link decoration, etc .

One Example Of Federated Identity



Clicks to login to a site

User can choose to login directly to the site or with an Identity Provider

Chooses an Identity Provider

If they're not logged in, presented with the login form for the IDP

Back to the Relying Party

With the Identity Provider sending back authentication / authorization data

Behind The Scenes



Navigation tracking, URL parameters...

Go to application

Redirect to IDP

Redirects

3P cookies

IFrames, pop-ups...

If not logged in, return login form

Check if logged in

Submit form with login credentials

Send back auth data

Verify credentials

Pass auth data to Relying Party

Validate data

Access tokens & ID tokens

Return application page

Some Of The Challenges Involved

- Many of the proposed privacy changes for tracking will also disrupt federated identity.
- It has its own privacy concerns since it can also be used to collect data about users:
 - Identity Providers can see which sites a user visits
 - Relying Parties can learn information about the user from the Identity Provider
 - Often uses global identifiers like email addresses
- It's a massive problem. Identity federation has been around a long time - and different protocols (SAML, OAuth, OIDC, etc.), implementations and use cases have evolved online.
- It's used by very diverse institutions: B2B, B2C, B2E, financial, federal, health services, universities for both students and researchers, etc.
- The solution can't require all of these different parties to do too much work, or it won't be feasible in a timely manner.



Federated Credentials Management API

Formerly Known As WebID

- This is Google's proposal for how to support Federated Identity without third-party cookies.
- It's currently only focused on the third-party cookie dependency, since that's the most pressing change.
- They are also trying to require the least amount of work from Relying Parties (the Sites/Services).
- It's actively being discussed in the Federated Identity CG.

The screenshot shows a web browser window displaying the GitHub repository page for 'FedID CG Federated Credentials Management'. The browser's address bar shows the URL 'https://github.com/fedidcg/FedCM'. The page content includes a 'README.md' file with the following text:

FedID CG Federated Credentials Management

This is the repository for the W3C's FedID CG Federated Credentials Management API.

Explainer: [explainer/README.md](#)

Work-in-progress specification: <https://fedidcg.github.io/FedCM/>

Introduction

As the web has evolved there have been ongoing privacy-oriented changes ([example](#)) and underlying privacy [principles](#). With those changes some underlying assumptions of the web are changing. One of those changes is the deprecation of third-party cookies. While overall good for the web, third-party cookie deprecation leaves holes in how some existing systems on the web were designed and deployed.

Federated Credentials Management API aims to fill the specific hole left by the removal of third-party cookies on federated login. Historically this has relied on third-party cookies or navigational redirects in order to function as they were the primitives provided by the web.

The [explainer](#) and [spec](#) provide a potential API and the rational behind how that API was designed.

Contributing

Much of the FedCM specification has evolved due to the experimentation detailed in the [explainer](#). The explainer documents give a good overview of the *why* of the FedCM API. Please read over the documents to understand how the current API has evolved.

There are several ways to contribute to the Federated Credential Management API.

Google Considered Three Alternatives

Permission-Oriented

The user-agent tries to warn users about potential tracking risks - and prompts the user for their permission to continue.

Mediation-Oriented

The user-agent acts as a mediator between the Relying Party and the Identity Provider. The user-agent can then control what data is exchanged.

Delegation-Oriented

The user-agent still acts as the mediator - but in this scenario they take over more work from the Identity Provider to stop the IdP from observing which sites the user is visiting.

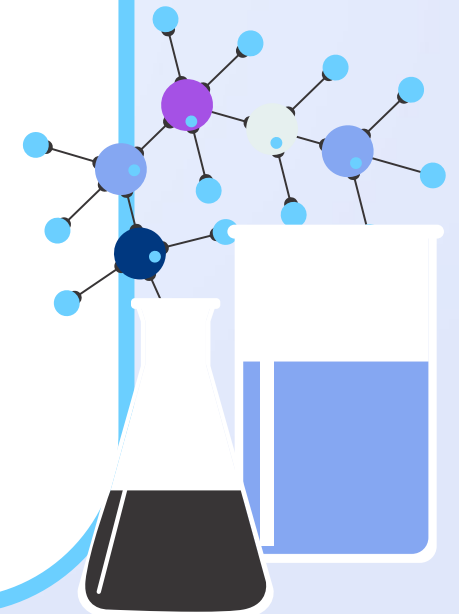
Mediation-Oriented Is The Focus For Now

 User Agent  Relying Party



The Use Cases Google Considered

- Sign-Up
- Sign-In
 - Prompted Sign-In
 - Auto Sign-In
- Sign-out
 - Relying Party
 - Identity Provider
- Session Management
- Revocation / Account Cancellation
 - With the Relying Party
 - With the Identity Provider
- Authorization
 - The user also has access to the resources



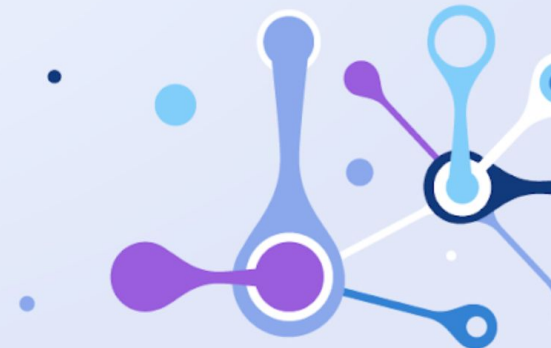


The Current Status

- The FedID CG is gathering and documenting different use cases, and working with Google on the proposal.
- Outstanding questions/concerns:
 - What should the role of the user-agent be here? Many of the browser vendors are also identity providers, which is problematic.
 - Google and Microsoft are active in the FedID CG - but we really need more involvement from the other browsers. Will they support the FedCM proposal or are expecting other APIs to be used?
 - What's the relationship going to be like between the privacy APIs in general in terms of federated identity?

For More Information

- Identity vs. Browser Changes video from the 2021 OAuth Security Workshop: PDF of the slides
- Authentication vs. Federation vs. SSO overview article
- Common Federated Identity Protocols: OpenID Connect vs. OAuth vs. SAML 2.0 on HackEDU
- Federated Identity articles on ScienceDirect
- FedCM Proposal: <https://github.com/fedidcg/FedCM>
- The FedCM HOWTO
- FedCM Draft Report: <https://fedidcg.github.io/FedCM/>
- FedCM at BlinkOn 15: video and slides
- FedCM at TPAC 2021: video and minutes





Thank You!

If you'd like to join the FedID CG, you can do so at:
<https://www.w3.org/community/fed-id/>

Another Example Of Federated Identity



Employee

The screenshot shows a web browser window titled "My Apps Dashboard X". The page features a search bar at the top with the placeholder text "Search your apps". Below the search bar, there is a "My Apps" section with a "Sort" button. The applications are organized into two rows. The first row contains four blue boxes representing company apps: "Company App 1 Ex: Email", "Company App 2 Calendar", "Company App 3 Tech Support", and "Company App 4 HR". The second row contains four colored boxes representing partner apps: "Partner A App 1 Analytics" (yellow), "Partner A App 2 CMS" (yellow), "Partner B App 1 CRM" (red), and "Partner C App 1 Video Calls" (purple).

Category	App Name	Example / Description
Company	Company App 1	Ex: Email
Company	Company App 2	Calendar
Company	Company App 3	Tech Support
Company	Company App 4	HR
Partner A	Partner A App 1	Analytics
Partner A	Partner A App 2	CMS
Partner B	Partner B App 1	CRM
Partner C	Partner C App 1	Video Calls