FedCM

The Timing Attack Problem

Green = Added after the meeting based on feedback in the meeting & afterwards

Chrome Engineering

Aug 2022

8/31

Agenda

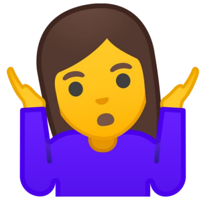
- TPAC
  - The timing attack problem, Multi-IDP choosers and Sequencing
- The IdP Sign-in Status API Proposal
- Sequencing Strategy Proposal
- Proof of Concepts
  - IdP Sign-in Status API
  - Multi-IDP

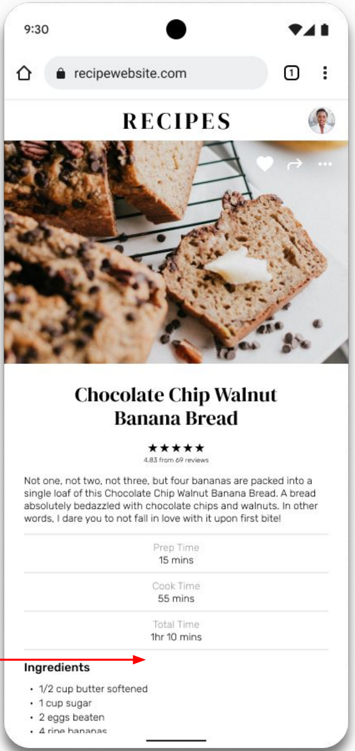
Recap: Silent Timing Attack Problem

The Problem: the Tracker gets a credentialed request (before the user consents) that allows them to SILENTLY track the specific user at the RP (through fingerprinting correlation).

```
fetch(`https://tracker.example/time.php?website=${window.location}`);

navigator.credentials.get({
  federated: {
    providers: [{
      "url": "https://tracker.example/”,
    }]
  }
});

GET time.php HTTP/1.1
Host: tracker.example
Website=rp.example

GET /rp.example/accounts.php HTTP/1.1
Host: tracker.example
Cookie: SID=212321

{ accounts: [] }
```

```
06/02/2022 10:32:31 PST ~~IP 201.299.99.00~~ SOME user       has visited rp.example
+
06/02/2022 10:32:32 PST ~~IP 201.299.99.00~~ User SID=212321 has visited SOMEWHERE
=
06/02/2022 10:32:32 PST ~~IP 201.299.99.00~~ User SID=212321 has visited rp.example
```




9:30
recipewebsite.com
RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★
4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Prep Time
15 mins

Cook Time
55 mins

Total Time
1hr 10 mins

Ingredients
- 1/2 cup butter softened
- 1 cup sugar
- 2 eggs beaten
- 4 ripe bananas

Proposal: stop silent tracking with the The IdP Sign-in Status API

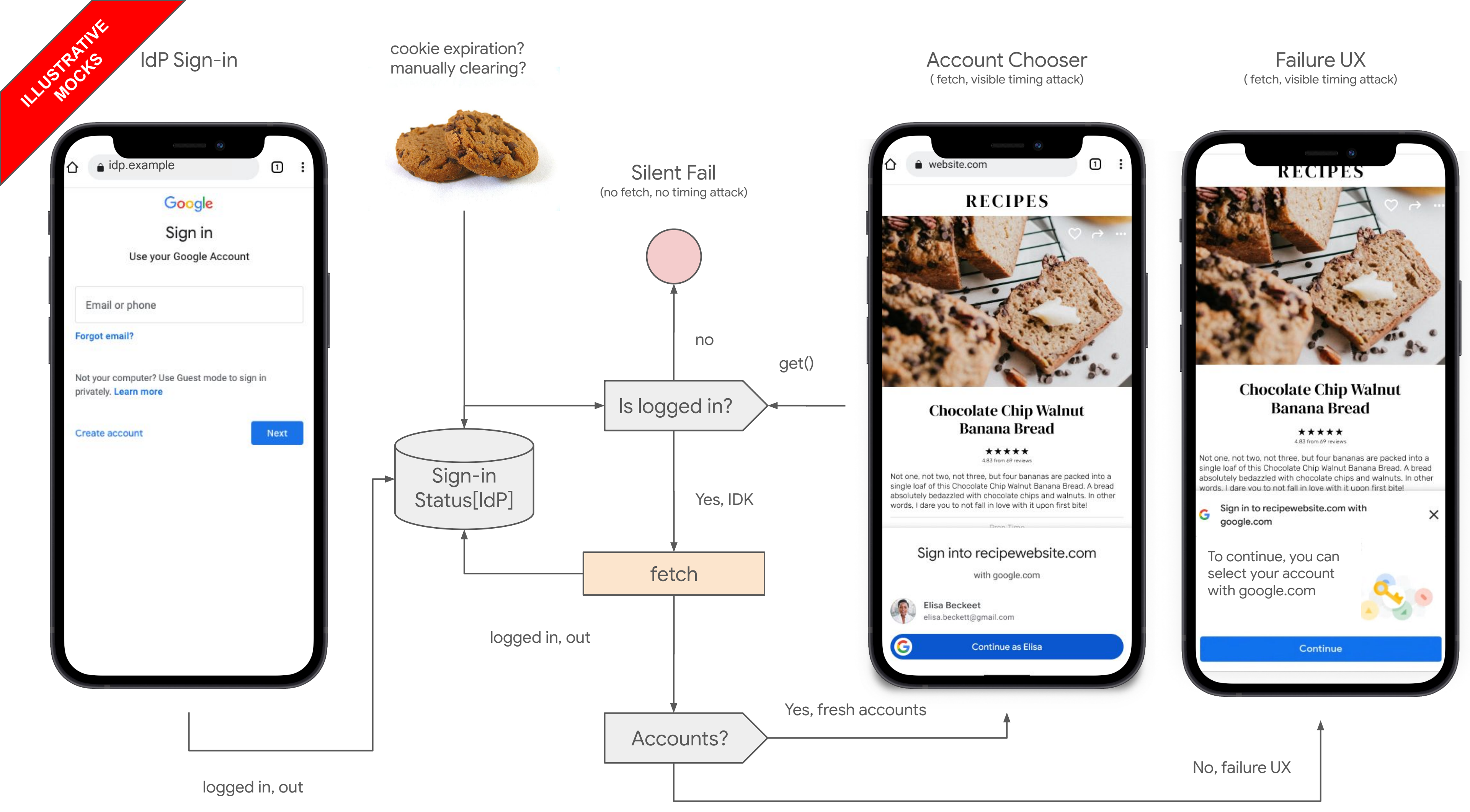
- Expose a Browser API to allow an IdP to store whether the user is “logged in”
- Since logging-in is specific to a browser instance/profile, this bit can reflect reality perfectly
- If the logged in bit is false for an IDP and an RP calls `navigator.credentials.get()` , we make no network request (we fail the JS call)
- Otherwise, we proceed as with the pull model (i.e. always fetch), **except** that we show UI even if an error/no accounts are returned (if the failure happens during the credentialed fetch)
  - This should not happen for “real” IDPs, since we know the user is logged in, except for network issues which should be rare
- This guarantees that we show fresh data
- Relatively easy improvement over pull API!
- For extra safety, we should only allow calling the IdP Sign-in Status API from toplevel frames and require some form of user activation (taking into account that this will likely be called on the page load *after* password form submission/user activation)
- Pros:
  - Not possible to track users silently (always observable by the user)
- Cons:
  - Does not prevent timing attack if tracker is able to call the IdP Sign-in Status API (including real IDPs).
    - Acceptable trade-off for Chrome
    - We heard from Ben before that it seemed like a reasonable trade-off
    - Question to Mozilla: is this an acceptable trade-off?

Proposal: The IdP Sign-in Status Semantics

1. Store one of 3-values for each IdP: undefined | true | false
   a. Sign-in-Status[IdP] = undefined | true | false (unknown, logged in or logged out)
2. On IdP login and logout
   a. IdP has a browser API that allows them to set their user's Sign-in Status
   b. Sign-in-Status[IdP] <= true | false
3. On navigator.credentials.get({identity: …})
   a. If Sign-in Status[IdP] is "undefined",
      i. Handles the case where the IdP's session happened before the API was available
      ii. Fetch accounts list
         1. Pays ONE "invisible" timing attack problem price once per IDP per browser profile
      iii. Sign-in Status[IDP] <= the result of the accounts list fetch
   b. If the Sign-in-Status[IdP] "false"
      i. Terminate the API, no timing attack
      ii. Do not show UI (or optionally show Two Tap)
   c. If the Sign-in-Status[IdP] is "true"
      i. Fetch account list
         1. Pays the "visible" timing attack problem
      ii. If network issues
         1. TBD
      iii. If no accounts are found
         1. Set the Sign-in Status to false
         2. Show a prompt that says "oops an error happened" (or optionally Two Tap + affordance to sign-in?)
      iv. If accounts are found
         1. Show prompt
4. Open Questions:
   a. How should we deal with changes from Browser UX (e.g. clearing cookies?)?
5. Question to Mozilla: is this what you had in mind? Would this work for you?

The IdP Sign-in Status API Proposal

1. Can only be called from toplevel frames and require some form of user activation
2. HTTP Headers API
   a. Likely useful (and where we should start), because a lot of IdPs centralize auth over 302s
   b. IdP-Sign-in-Status: action=login
   c. IdP-Sign-in-Status: action=logout
3. JS API (optional)
   a. Create a new interface, call it, say, IdentityProvider
   b. IdentityProvider.login()
   c. IdentityProvider.logout()
   d. Also move logoutRPs() to Identity Provider
4. Open questions
   a. Relationship with Login Status API
5. Question to Ben: we don't have to agree on the details API, but does this seem directionally correct to you? We considered some alternatives below, check if this seems right to you too?



Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

ILLUSTRATIVE MOCKS

Failure UX

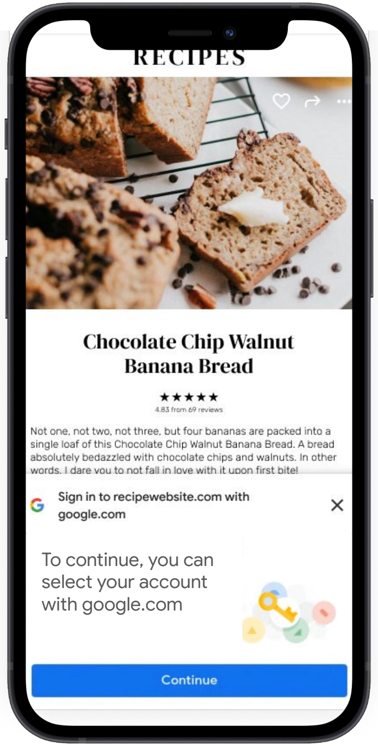
(opens a new tab or a modal?)

IdP Sign-in Url New Tab UX (or modal?)

(IdP does not learn who the RP is)
(configures it in the .well-known file)
(calls the IdP Sign-in Status API to close the tab and return)

Failure UX to Account chooser UX

(tab or modal? closed, returns)



RECIPES

Chocolate Chip Walnut Banana Bread

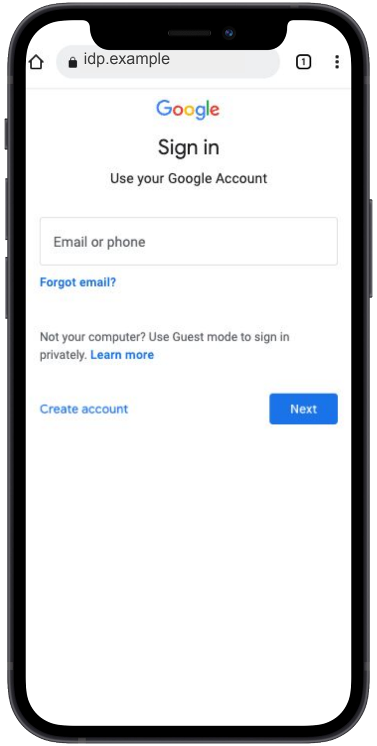
★★★★★
4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

To continue, you can select your account with google.com

Continue



idp.example

Google

Sign in

Use your Google Account

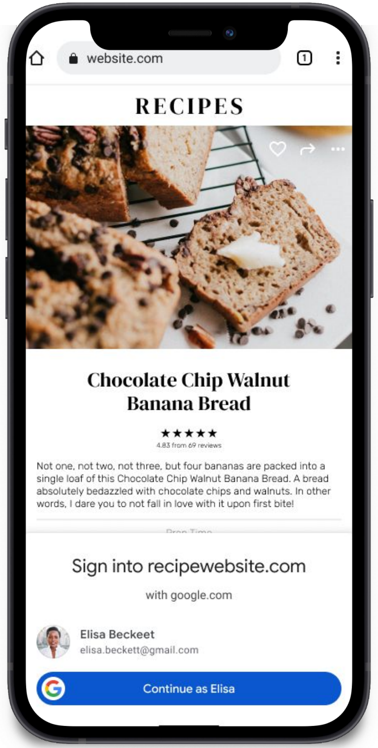
Email or phone

Forgot email?

Not your computer? Use Guest mode to sign in privately. Learn more

Create account

Next



website.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★
4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign into recipewebsite.com

with google.com

Elisa Beckeet
elisa.beckett@gmail.com

Continue as Elisa

Sequencing Strategies

1. We don't think we should launch the IdP Sign-in Status API without the confidence of production experience
2. We think that gathering production experience for the IdP Sign-in Status API will take us at least 2 quarters (but likely no more than 4 quarters): design > prioritize > implement > test > experiment > deploy.
    a. We are gathering more information on the estimates with IdPs
3. We are learning a lot by gathering production experience with IdPs
    a. Incentives (for IdPs and RPs)
    b. Deployment corner cases
4. It is hard to rely on FedCM until it is publicly available
    a. IdPs are asking us "In 2023, should we extend our products through Iframes/3PC or FedCM?".
    b. Browsers are asking us "So, should we solve this with the Storage Access API? First Party Sets?"
5. Browsers have generally favorable positions on FedCM, so
6. The riskiest part right now is to establish product market fit with IdPs
    a. Interoperability (is it available in chrome? will it be available in firefox/safari?)
    b. Incentives
    c. UX ROI
    d. Decrease costs
    e. Blind spots
    f. Stability: making backwards incompatible changes isn't fun for IdPs, but we think that might be preferable than "going quiet" for the next 4 quarters
7. Our Origin Trials has been running for 6 months (since M101). We I2EE once it until M107 (inclusive), expiring our "6 milestones no questions asked".
    a. That means we would likely have to interrupt our origin trials, which isn't ideal because for (3).
8. Question to Ben: do you agree that we would all benefit to have FedCM available before we ship the IdP Sign-in Status API?

Sequencing Proposal

1. Today, we propose we launch “FedCM Phase 1: the Baseline” **without** the IdP Sign-in Status API
    a. Publicly Framed to be insufficient for 3PCD
2. In a year, in conjunction with Browsers and IdPs, we launch “FedCM Phase 2: the IdP Sign-in Status API”
    a. In Q3/2022 Prototype the Sign-in Status API
    b. In Q4/2022 Dark Launch the Sign-in Status API
    c. In Q1/2023 Devtrials the IdP Sign-in Status API
    d. In Q2/2023 Origin Trials (I2E) the IdP Sign-in Status API
    e. In Q3/2023 Ship (I2S) the IdP Sign-in Status API
    f. In Q1/2024, in 12-ish months, work with IdPs to migrate them to the IdP Sign-in Status API
3. In two years, on 3PCD, we make a backwards incompatible change to enforce the “no silent tracking” by failing the API when the IdP Sign-in Status API wasn’t called
    a. Works without 3PC
    b. In Q3/2024, at 3PCD, make IdPs get the Interstitial UI or we fail the API without the IdP Sign-in Status API signal.
4. From that point on, we explore the Web Push API variations to propagate the user’s data
5. Question to Ben: WDYT?

ANNEX

Proof of Concept

1. On IdP login and logout
    a. Browser intercepts a IdP-Sign-In-Status header (and, for testing purposes, Google-Accounts-SignIn, Google-Accounts-SignOut)
    b. Writes to local storage, Sign-in-Status[IDP]
2. On navigator.credentials.get()
    a. Sign-in-Status[IDP] = undefined | false | true
    b. If Sign-in Status is “undefined”,
        i. Fetch accounts list (pay ONE invisible timing attack problem price once per IDP per browser profile)
        ii. Set the Sign-in Status
    c. If the Sign-in Status “false”
        i. Terminate the API (deliberately backwards incompatible)
        ii. Do not show UI
    d. If the Sign-in Status is “true”
        i. Fetch account list (paying the “visible” timing attack problem) and
        ii. If no accounts are found
            1. Set the Sign-in Status to false
            2. Show current prompt that says “fake user ‘user not found’”
        iii. If accounts are found
            1. Show prompt
3. Question to Ben: does this seem to be testing the riskiest parts? Anything else?

8/16

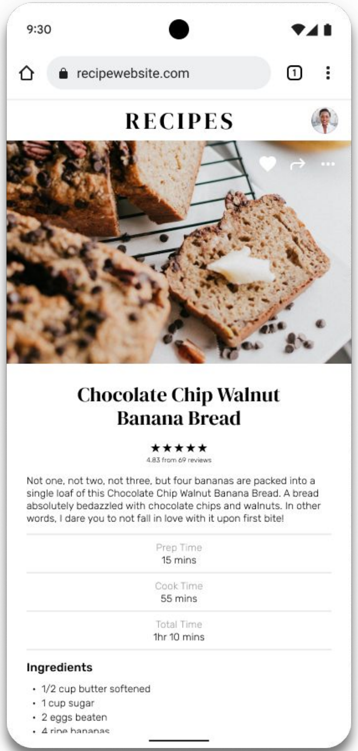
Recap: Timing Attack Problem

The Problem: the Tracker gets a request (before the user consents) that allows them to track the specific user (through cookies) at the RP (through fingerprinting correlation).

```
fetch(`https://tracker.example/time.php?website=${window.location}`);

navigator.credentials.get({
  federated: {
    providers: [{
      "url": "https://tracker.example/",
    }]
  }
});
```

```
GET time.php HTTP/1.1
Host: tracker.example
Website=rp.example

GET /rp.example/accounts.php HTTP/1.1
Host: tracker.example
Cookie: SID=212321

{ accounts: [] }
```

```
06/02/2022 10:32:31 PST ~~IP 201.299.99.00~~ SOME user       has visited rp.example
+
06/02/2022 10:32:32 PST ~~IP 201.299.99.00~~ User SID=212321 has visited SOMEWHERE
=
06/02/2022 10:32:32 PST ~~IP 201.299.99.00~~ User SID=212321 has visited rp.example
```



Chocolate Chip Walnut Banana Bread

★★★★★
4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Prep Time
15 mins

Cook Time
55 mins

Total Time
1hr 10 mins

Ingredients

- 1/2 cup butter softened
- 1 cup sugar
- 2 eggs beaten
- 4 ripe bananas

The push model

The Pull Model

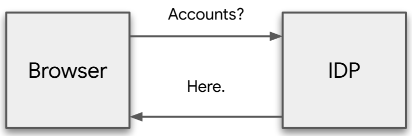
On demand, the browser makes an HTTP request to the IDP that returns the user's accounts.

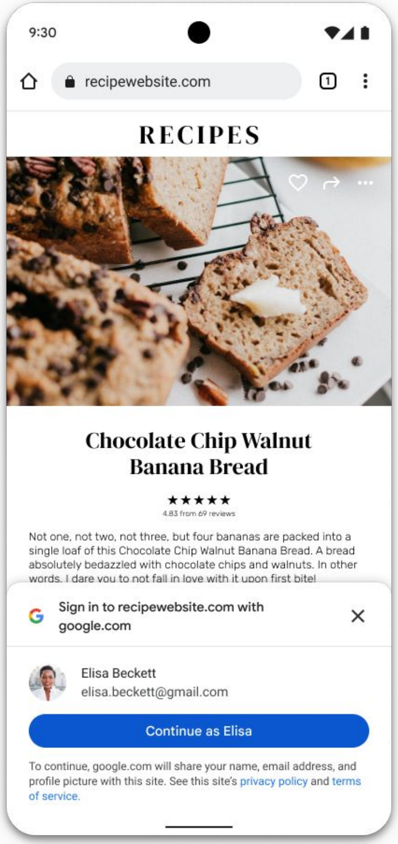
Pros

Simple to implement by IDP. Always in Sync.

Cons

Latency. Timing Attacks.




9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★

4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

Elisa Beckett
elisa.beckett@gmail.com

Continue as Elisa

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.

The Push Model

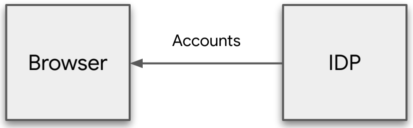
Ahead of time, the IDP saves in the browser the user's accounts.

Pros

Better UX. No attacks.

Cons

IDP announces all accounts rather than only the ones that would use federation. Can be out of sync (e.g. cookies can be cleared; out-of-band changes).



Hard requirement?

- **Must keep data fresh**, by some definition of freshness (hard legal requirement by IDPs)

Use case:

- User logs into IDP on their phone
- Later, user changes their display name on their desktop (e.g. for deadnaming reasons)
- User comes back to phone, goes to RP
- RP calls FedCM API
- Should show new name
  - At least if name was updated more than N hours ago
  - We have heard of some requirements that N <= 2

Evaluation Criteria

1. Privacy and tracking properties (e.g. timing attacks? leaking user behavior, e.g. when browser restarts? Leaks IP addresses, e.g. when they change cell networks?)
2. User experience (e.g. # of prompts)
3. Abuse vectors (e.g. unbounded resource consumption attacks)
4. Deployment Complexity (ergonomics) for IDPs (e.g. deployment complexity)
5. Browser implementation complexity

Ideal API

- Does not expose user’s IP address to IDP when user is not interacting with the IDP or an RP that uses this IDP
    - Could be solved by proxying, like Safari already does for paid users
- Does not expose browser usage patterns or user online status to IDP (e.g. ping IDP to update account list whenever browser starts)
- Does not rely on the user accepting additional permissions
- Of course, should actually solve the timing attack in all cases (ie. never make credentialed request to IDP at a time that can be correlated to an RP)
- Easy for IDPs to implement

This is hard!

Foundation: JS API

- Most of the options from the remaining slides require a JS API
- We propose the following, exposed to window and service workers
- Don’t want to bikeshed yet on exact names and which interface to put them
- Could also explore using the isLoggedIn API
- Could explore some version of navigator.credentials.store for this purpose

```
dictionary UserInfo {
  DOMString accountID;
  DOMString displayName;
  DOMString profilePictureURL;
}

// Must be called from IDP origin
void userLoggedIn(IdentityProvider idp, UserInfo user);
void userLoggedOut(IdentityProvider idp, DOMString accountID);

// match on accountID
void userDataChanged(IdentityProvider idp, UserInfo user);
```

User explicitly clears cookies

- By definition, if a user clears cookies they are logged out of the IDP(s)
- Browsers should observe cookie clearing action and implicitly call userLoggedOut when that happens
- Easy: Clear all cookies; clear all cookies for site
- What to do when user clears just one cookie?

- If cookies get cleared by content (HTTP/JS), we rely on the IDP to call userLoggedOut

Option: Web Push API

- IDP calls userLoggedIn when user logs in
- When user changes their details in the current device/browser/profile, IDP calls userDataChanged
- When user changes data on another device, IDP uses web push API to wake up service worker which calls userDataChanged
- Problems:
    - User has to accept permission to allow push API
    - API requires that the service worker shows a user-visible notification for each push
        - May be tolerable on the basis that changing name or profile picture is relatively rare?
    - Push services support acknowledgements, telling the sender whether the user is online
    - Possible battery life impacts of allowing IDPs to wake up users' phones
    - Allows service worker to ping IDP, exposing user's IP address

Option: Modified (“declarative”) web push API

- Idea: allow specially-formatted push messages that do not wake up service worker
- Instead, the payload contains the equivalent of the UserInfo dict
- Browser “intercepts” them and updates stored account list
- Do not need to show UI and does not expose user’s IP address
- Problems:
  - Does not solve acknowledgements problem that exposes online status
  - Still potential battery consumption concern
  - May still have to gate this on user granting permission because of the previous two points
  - Complicated for IDPs to implement

Option: Web Periodic Background Synchronization API

- Variation of push API proposal
- Instead of using web push, service worker syncs account list using periodic background sync API
- Does not require showing UI, other than installing PWA
- Problems:
    - Background sync API requires an installed PWA, making this likely infeasible for most IDPs
    - Exposes user’s IP address on every sync, unless browser does IP proxying

Option: Only stop **silent** tracking - seems most promising

- Use the `userLoggedIn`/`userLoggedOut` API only to store a “logged in” bit
- Since logging in is specific to a browser instance/profile, this bit can reflect reality perfectly
- If the logged in bit is false for an IDP and an RP calls `navigator.credentials.get()` , we make no network request
- Otherwise, we proceed as with the pull model (i.e. always fetch), **except** that we show UI even if an error/no accounts are returned (at least if the failure happens during the credentialed fetch)
    - This should not happen for “real” IDPs, since we know the user is logged in, except for network issues which should be rare
- This guarantees that we show fresh data
- Relatively easy improvement over pull API!
- For extra safety, we should only allow calling userLoggedIn from toplevel frames and require some form of user activation (taking into account that this will likely be called on the pageload *after* password form submission/user activation)
- We could allow IDPs to opt-in to this model using a field in the manifest, and fall back to two-tap otherwise
- Problem:
    - Does not prevent timing attack if tracker is able to call userLoggedIn (including real IDPs). **Question to Mozilla: Is this an acceptable trade-off?**
    - However, does prevent trackers from doing this silently

Infeasible option: Polling

- Idea: To keep user data stored in browser fresh, we poll the account list endpoint every N hours (2? 24?)
- Triggered either by `userLoggedIn` or first `navigator.credentials.get()` call
- To stop forever polling an IDP the user doesn't use, stop after M errors/empty account lists
- Problems
  - Exposes user's IP address on every periodic refresh, even when user is not using an RP or IDP's website (unless browser does IP proxying)
  - If only done while browser is active (likely), exposes user patterns
  - Consumes resources for something that may never be used

Option: Pull API + Caching - also provides (limited) protection against IDP tracking, unlike slide 12

- Cache results from `n.c.get()` for N hours, including errors
- Timing attack is only possible once per N hours
- Still need `userLoggedIn`/`userLoggedOut` API to invalidate cache when user logs in, to avoid API failing during cache lifetime
- Must limit userLoggedIn to toplevel frames to avoid RPs/IDPs colluding to call it in an iframe to guarantee a fetch in the next call
  - Or we could require providing the user info in the userLoggedIn call, avoiding the need to fetch
- Problems:
  - Timing attack still possible once per N hours

Discussion

Option we didn’t discuss - caching option + refresh button

- Start with the “cache for N hours” proposal, but provide a button to refresh the account data
- When user clicks the button, fetch account list and update UI
- An “escape hatch” when user wants newer information than we have
- Problems:
    - None

Future topics

- Is the “only stop silent tracking” proposal acceptable?
- Should the spec require both two-tap and the “stop silent tracking” solution, letting IDPs choose between them?
- If we frame the spec with two-tap as the default, should it allow browsers to only support one tap and only support IDPs that opt in to the “no silent tracking” option?
- For two-tap – should we support things like allowing users to sign in to the IDP

6/29

ILLUSTRATIVE MOCKS

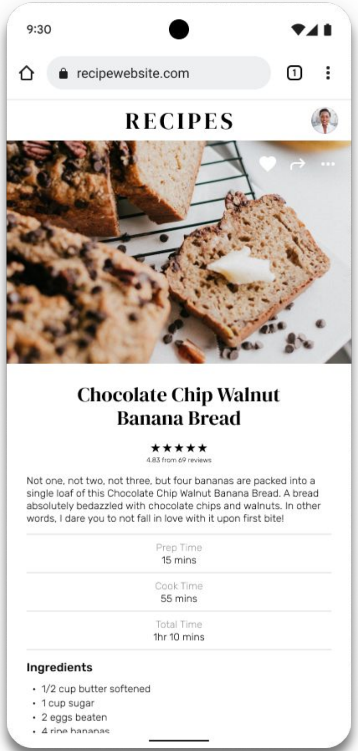
Timing Attack Problem

The Problem: a tracker gets a request (before the user consents) that allows them to track the specific user (through cookies) at the RP (through fingerprinting correlation).

```
fetch(`https://tracker.example/time.php?website=${window.location}`);

navigator.credentials.get({
  federated: {
    providers: [{
      "url": "https://tracker.example/",
    }]
  }
});
```

```
GET time.php HTTP/1.1
Host: tracker.example
Website=rp.example

GET /rp.example/accounts.php HTTP/1.1
Host: tracker.example
Cookie: SID=212321

{ accounts: [] }
```

```
06/02/2022 10:32:31 PST ~~IP 201.299.99.00~~ SOME user        has visited rp.example
+
06/02/2022 10:32:32 PST ~~IP 201.299.99.00~~ User SID=212321 has visited SOMEWHERE
=
06/02/2022 10:32:32 PST ~~IP 201.299.99.00~~ User SID=212321 has visited rp.example
```


9:30
recipewebsite.com
RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★
4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Prep Time
15 mins

Cook Time
55 mins

Total Time
1hr 10 mins

Ingredients
- 1/2 cup butter softened
- 1 cup sugar
- 2 eggs beaten
- 4 ripe bananas

ILLUSTRATIVE MOCKS

The push model

The Pull Model

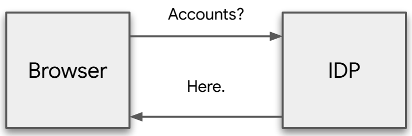
On demand, the browser makes an HTTP request to the IDP that returns the user's accounts.

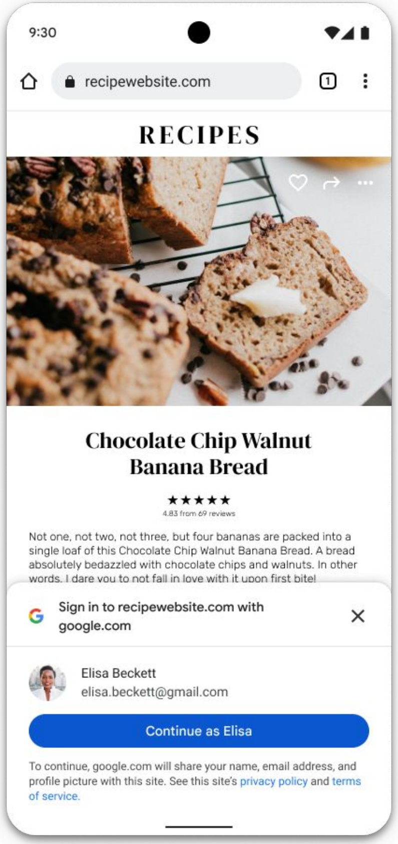
Pros

Simple to implement by IDP. Always in Sync.

Cons

Latency. Timing Attacks.





Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

Elisa Beckett
elisa.beckett@gmail.com

Continue as Elisa

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.

The Push Model

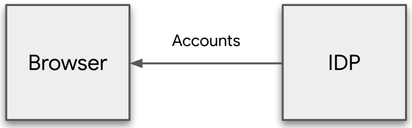
Ahead of time, the IDP saves in the browser the user's accounts.

Pros

Better UX. No attacks.

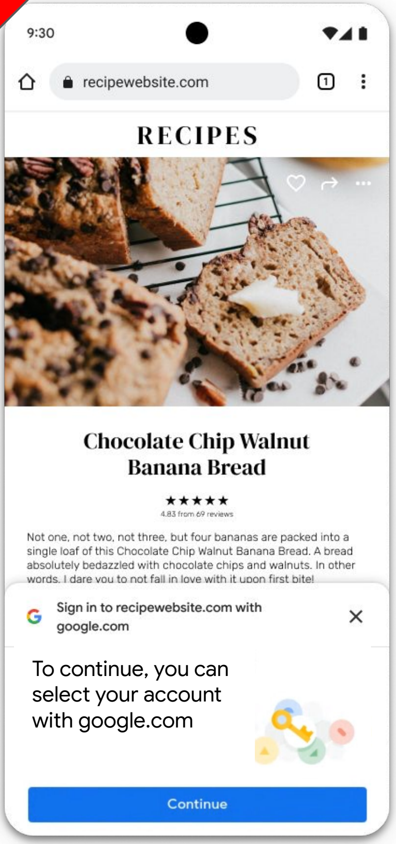
Cons

IDP announces all accounts rather than only the ones that would use federation. Can be out of sync (e.g. cookies can be cleared; out-of-band changes).



The Intuition

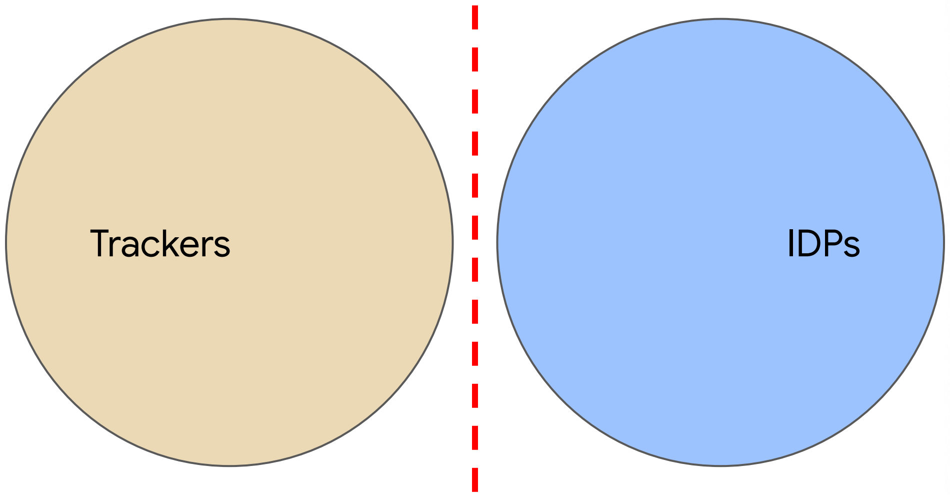
1. There are O(1000s) of IDPs with a T level of deployment competence
2. There are O(1) browsers with a T' level of deployment competence
3. T' >>> T
4. Time to deploy: T * O(1000) >>> T' * O(1)
5. If the push model costs A and the pull model costs B
6. Time to deploy: O(1000s) * T * A + O (1) * T' * A' >>> O(1000s) * T * B + O(1) * T'* B'
7. The intuition:
   a. if we can make O(1) * T'* B' work then we'd allow small IDPs to thrive O(1000s) * T * B
   b. What we heard from you in the past about lessons learned from Mozilla Personas: it is key to bring IDPs along for the ride.

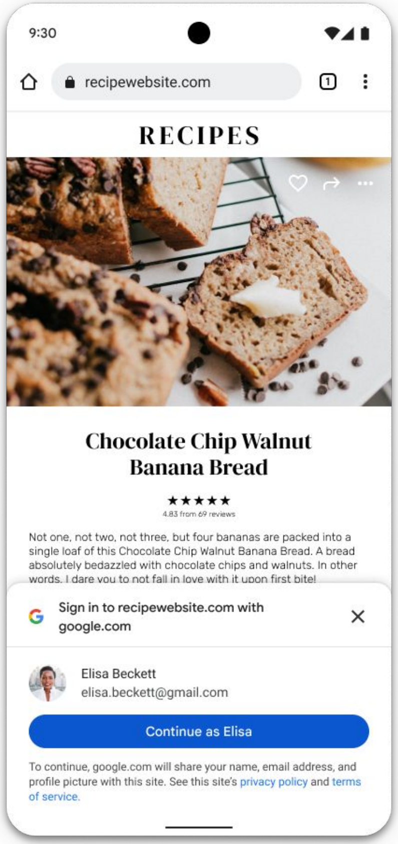


Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

To continue, you can select your account with google.com





Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

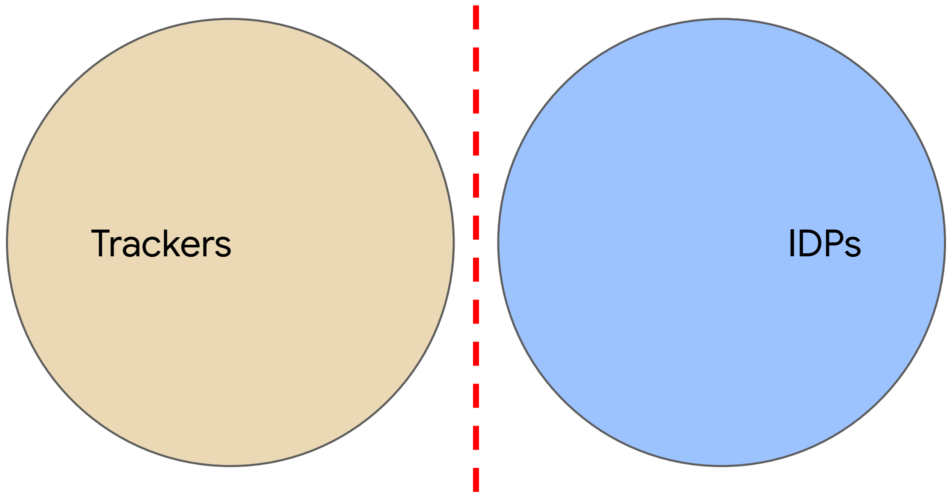
Sign in to recipewebsite.com with google.com

Elisa Beckett
elisa.beckett@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.

Show an **extra** sign-in specific but static UI for trackers and
Show a personalized UI for IDPs (it helps users)

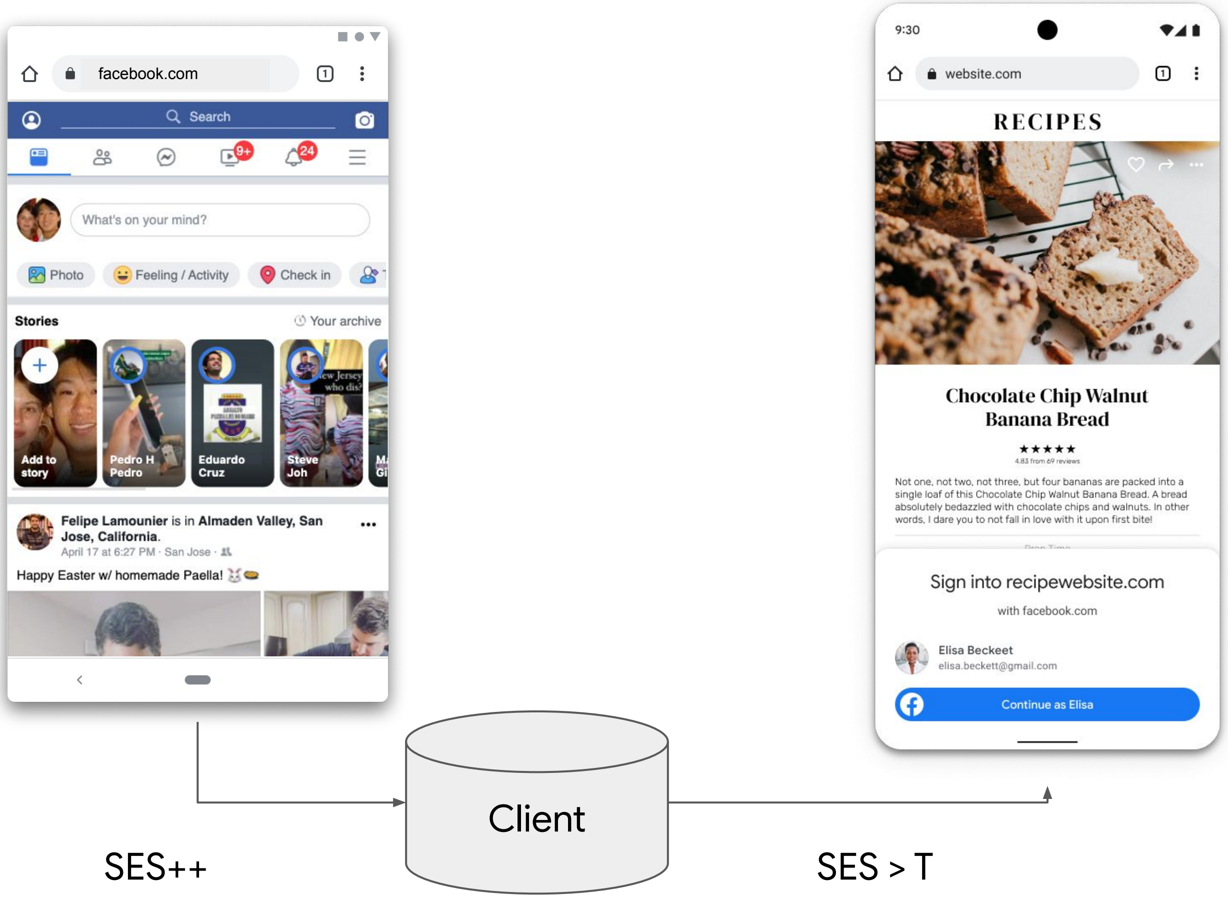
ILLUSTRATIVE MOCKS



How do you tell them apart?

ILLUSTRATIVE MOCKS

The IDP Site Engagement Score Assumption



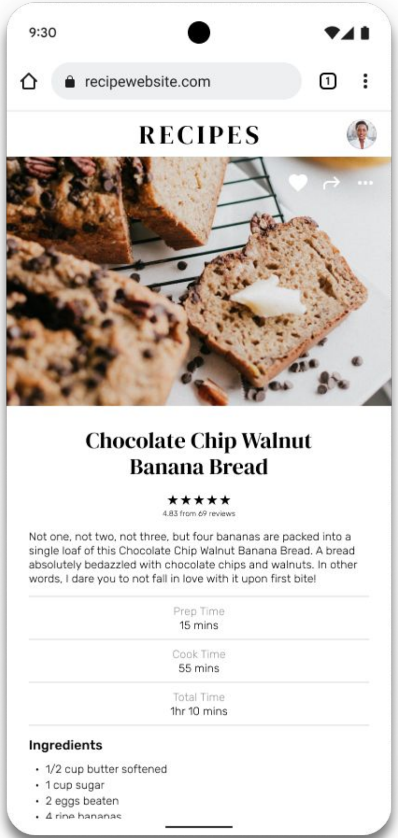
Felipe Lamounier is in Almaden Valley, San Jose, California.
April 17 at 6:27 PM · San Jose ·

Happy Easter w/ homemade Paella!

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

The IDP Service Assumption

ILLUSTRATIVE MOCKS



9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★
4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

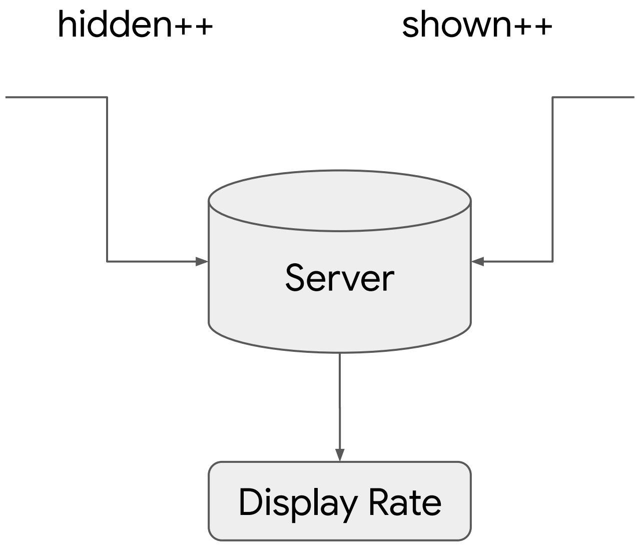
Prep Time
15 mins

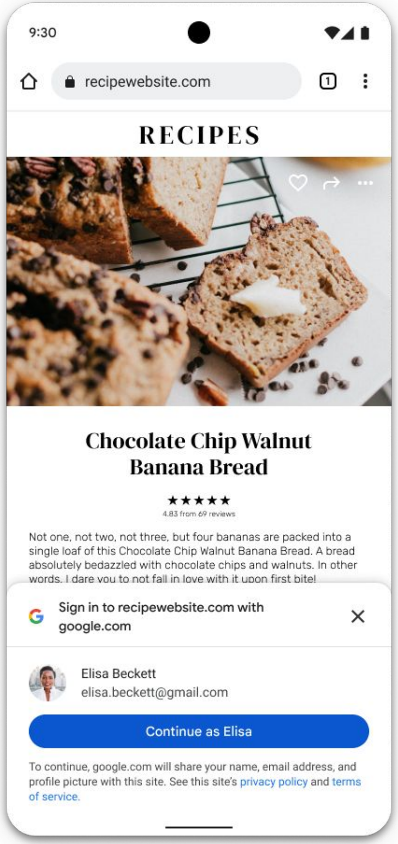
Cook Time
55 mins

Total Time
1hr 10 mins

Ingredients

- 1/2 cup butter softened
- 1 cup sugar
- 2 eggs beaten
- 4 ripe bananas





9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★
4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

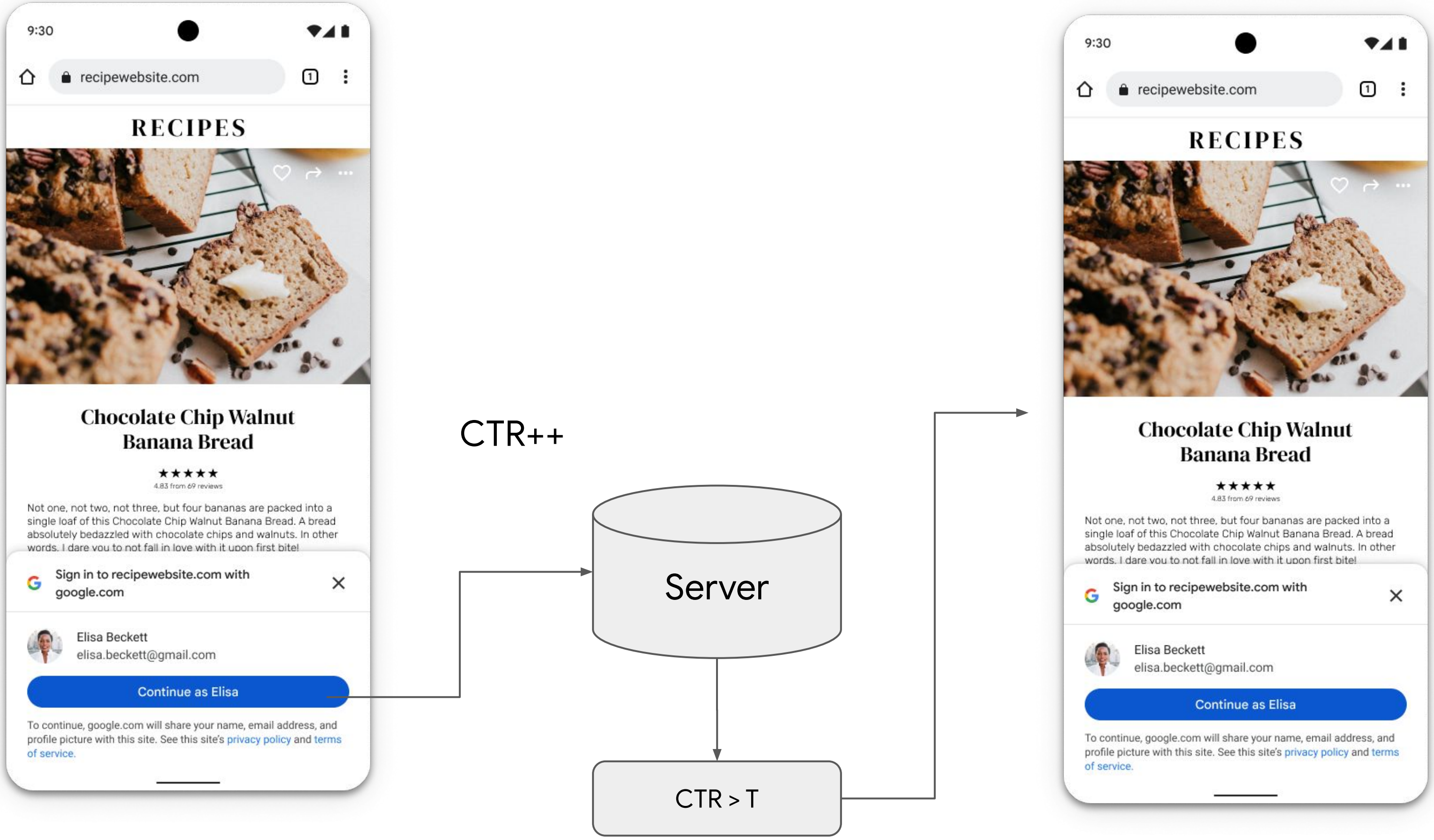
Elisa Beckett
elisa.beckett@gmail.com

Continue as Elisa

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.

ILLUSTRATIVE MOCKS

The IDP Performance Assumption



9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★
4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

Elisa Beckett
elisa.beckett@gmail.com

Continue as Elisa

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.

9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★
4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

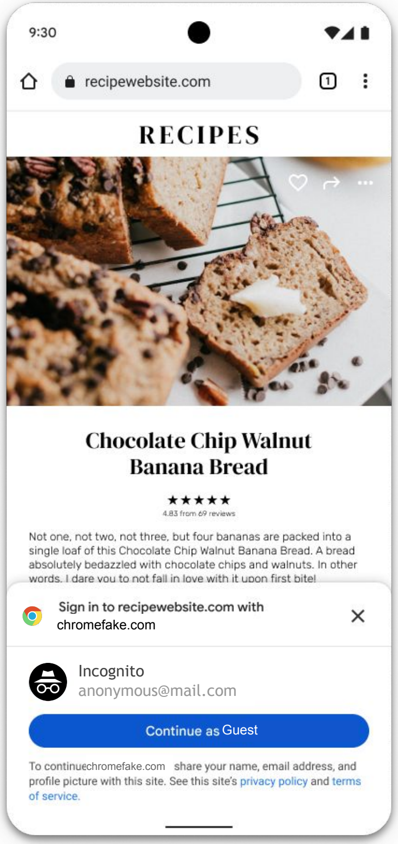
Elisa Beckett
elisa.beckett@gmail.com

Continue as Elisa

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.

ILLUSTRATIVE MOCKS

The IDP Identity Assumption



9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★

4.83 from 69 reviews

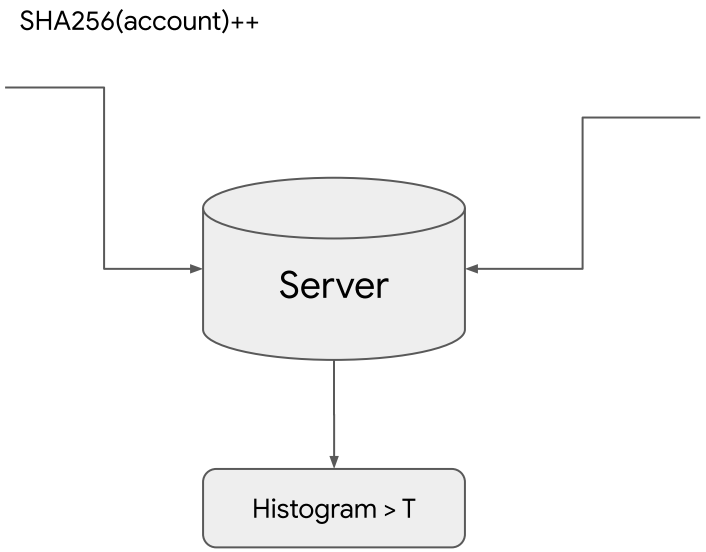
Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

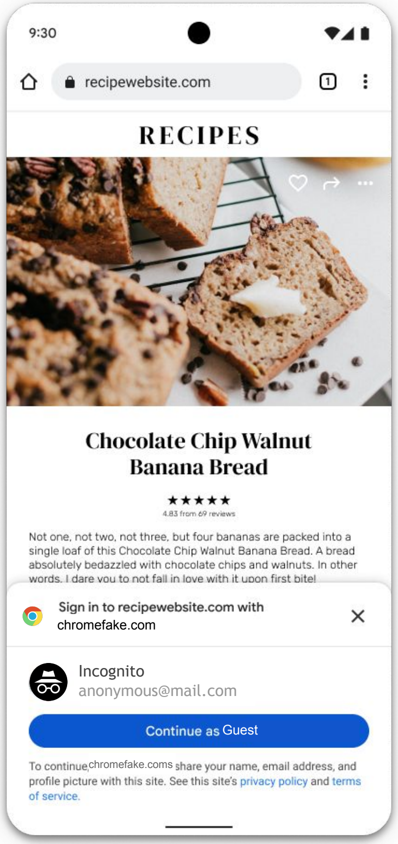
Sign in to recipewebsite.com with chromefake.com

Incognito
anonymous@mail.com

Continue as Guest

To continuechromefake.com share your name, email address, and profile picture with this site. See this site's privacy policy and terms of service.





9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★

4.83 from 69 reviews

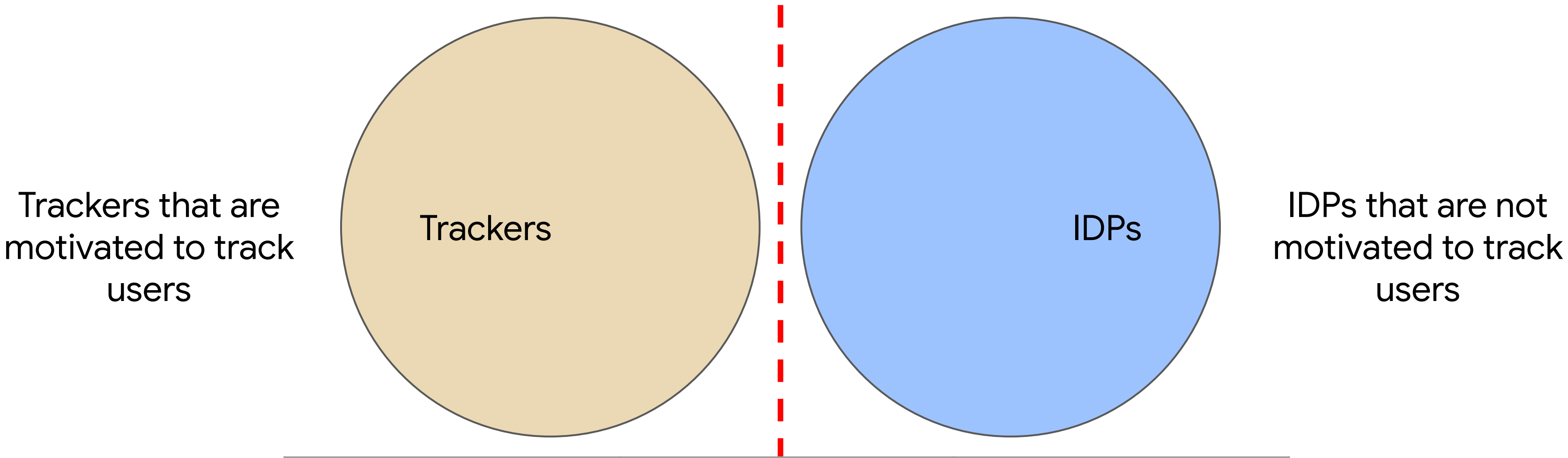
Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with chromefake.com

Incognito
anonymous@mail.com

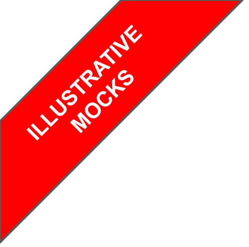
Continue as Guest

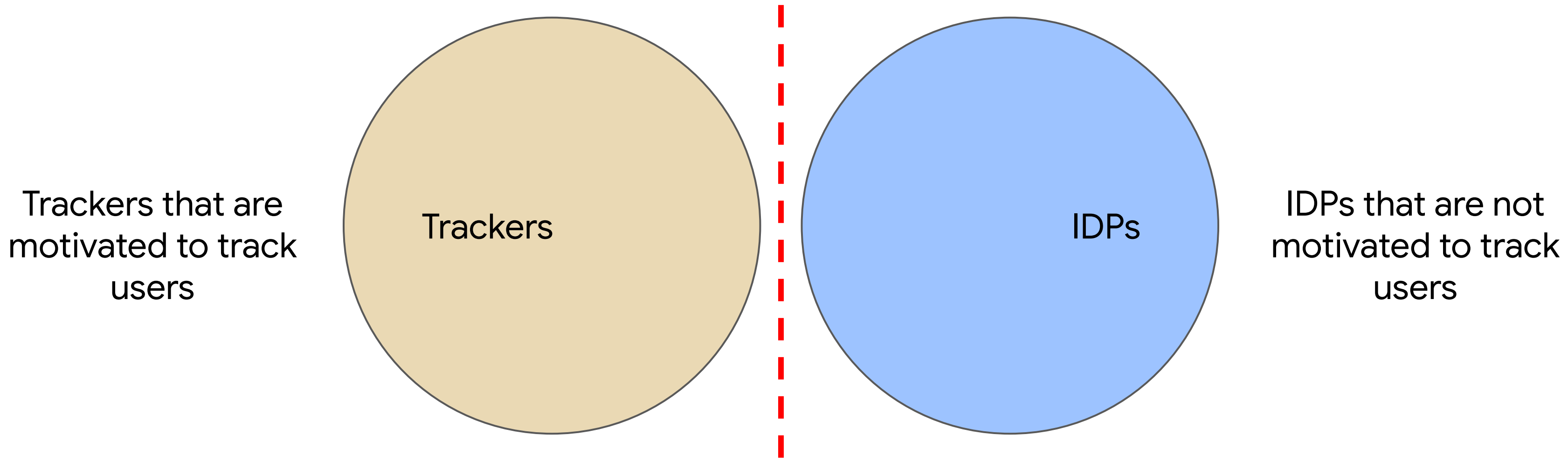
To continue chromefake.coms share your name, email address, and profile picture with this site. See this site's privacy policy and terms of service.


Trackers that are motivated to track users
Trackers
IDPs
IDPs that are not motivated to track users

	Trackers	IDPs
IDP Site Engagement Score	Low, never engaged	High, logged in
UI Display Rate	Low, intent to track users	High, IDP has the intent to provide the functionality for the user to log-in
Unique Accounts	Low, fake accounts	High, real accounts
Click Through Rate	Low, unrecognizable	High, user has an intent to log-in

ILLUSTRATIVE MOCKS

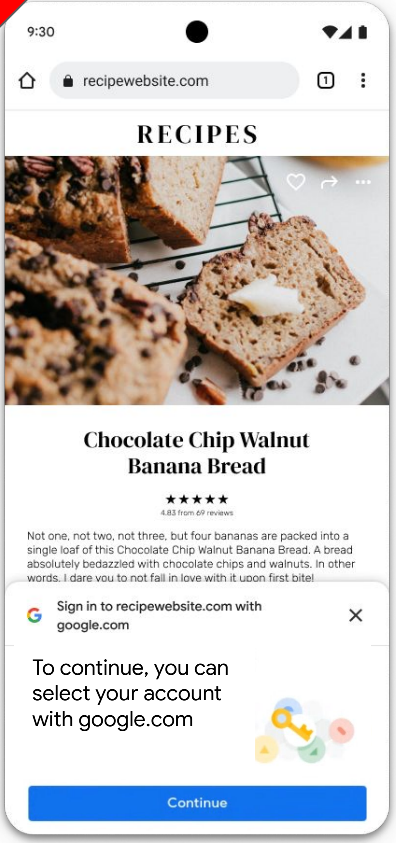



Trackers that are motivated to track users
Trackers
IDPs
IDPs that are not motivated to track users

Detection is going to be intrinsically statistical.
If IDPs want a more deterministic mechanism, we would provide them the push model.

This is one way that we think that the Login Status API could factor into.

ILLUSTRATIVE MOCKS


9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

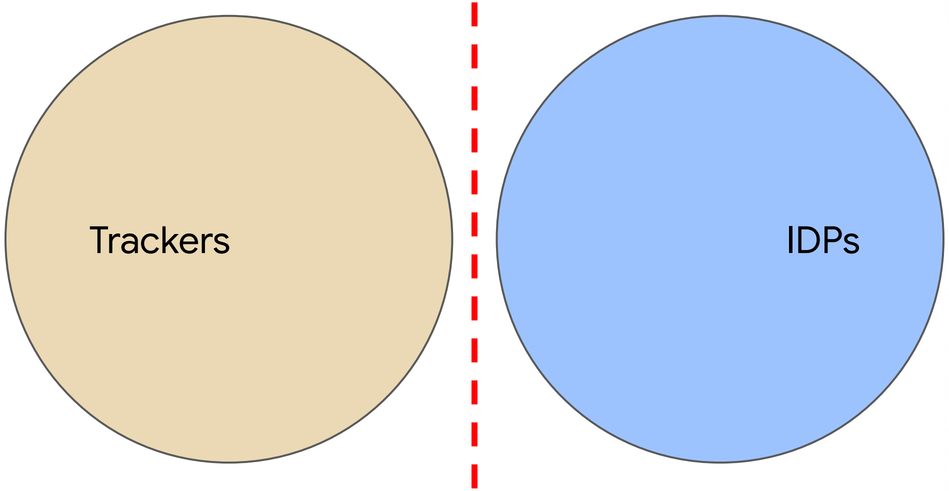
★★★★★
4.83 from 69 reviews

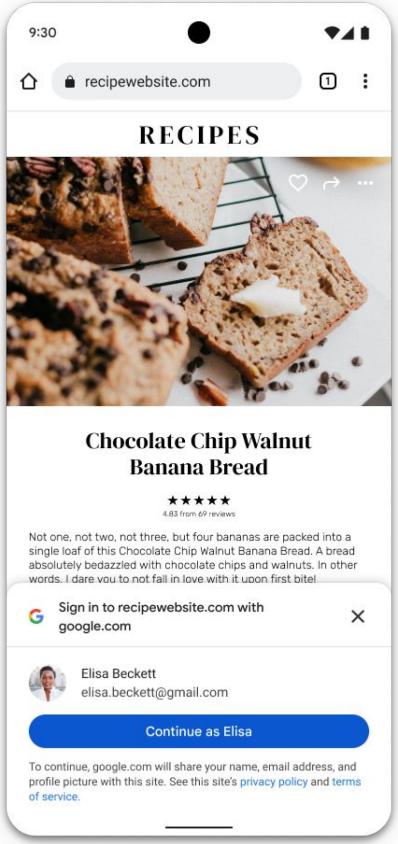
Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

To continue, you can select your account with google.com

Continue




9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★
4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

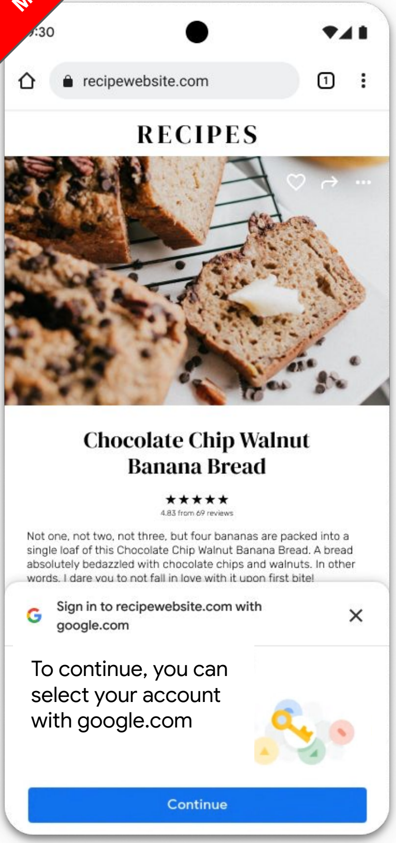
Elisa Beckett
elisa.beckett@gmail.com

Continue as Elisa

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.

Show an **extra** sign-in specific but static UI for trackers and
Show a personalized UI for IDPs

ILLUSTRATIVE MOCKS


recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

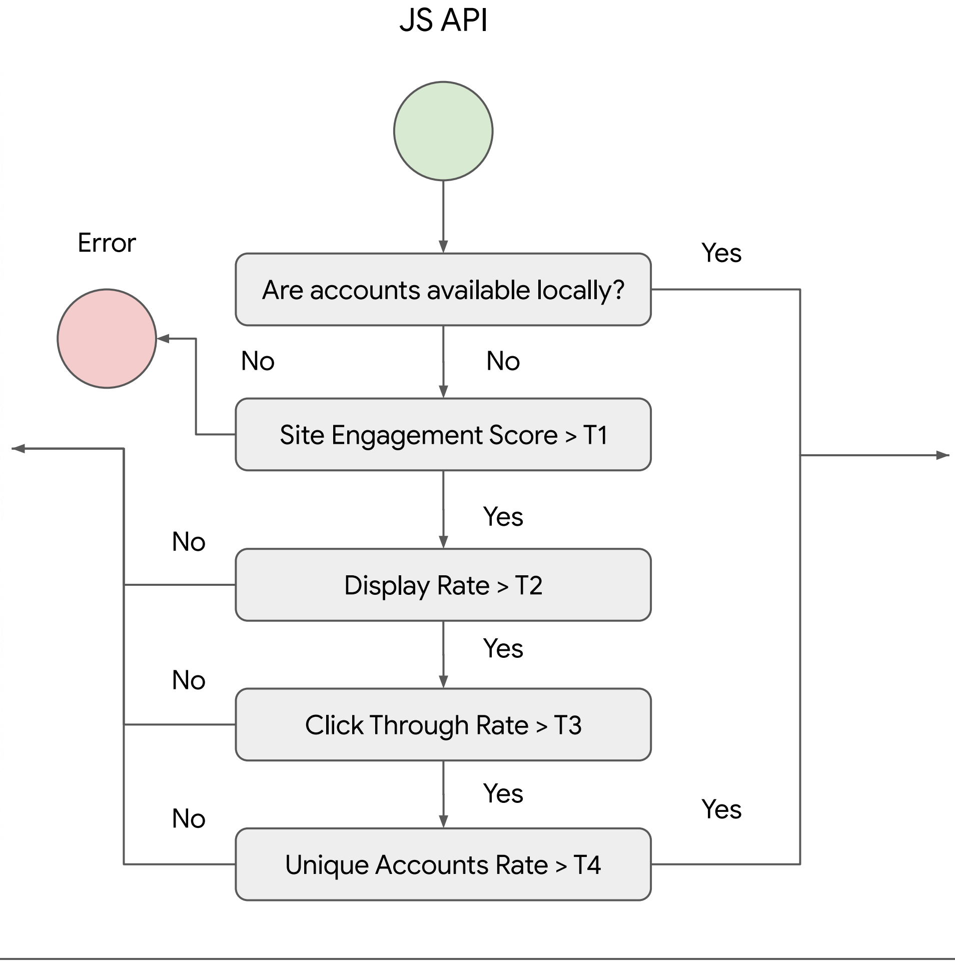
★★★★★
4.83 from 69 reviews

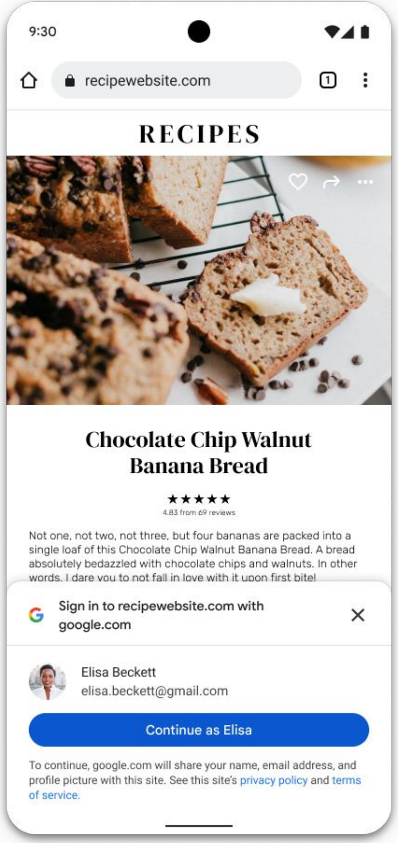
Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

To continue, you can select your account with google.com

Continue




9:30
recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★
4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

Elisa Beckett
elisa.beckett@gmail.com

Continue as Elisa

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.

ILLUSTRATIVE MOCKS

Discussion

1. Make sense?
2. Any other concerns?
3. If not
    a. help us form a favorable mozilla position? and
    b. Interested in co-editorship of FedCM?

NOTES

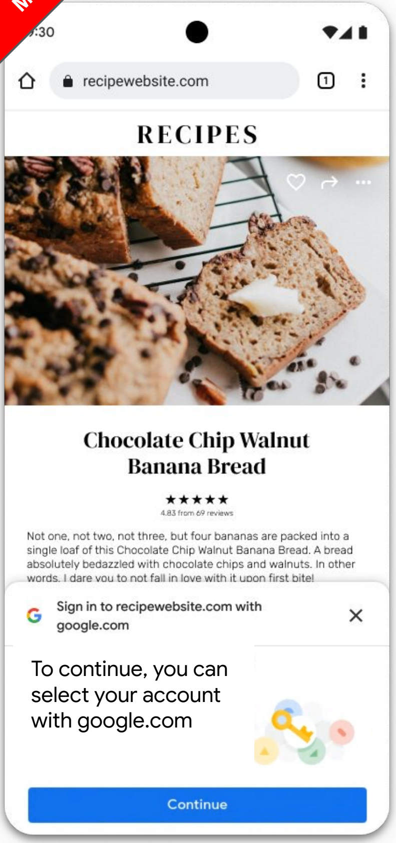
ILLUSTRATIVE MOCKS

What we heard from Mozilla

1. Heuristics:
    a. ben: lots of experience with the Storage Access API
    b. ben: here is a heuristic that was particularly useful for us in the past
        i. goto: neat, looking forward to hearing in more detail and incorporating.
        ii. npm: each browser can pick different heuristics and still interoperate.
2. Long term:
    a. ben: ack that the Push Model is more involved
    b. ben: Would it be possible to, long term, get rid of the heuristics?
        i. goto: highly desirable, but unclear if possible. we will work on the Push Model, but we expect:
            1. The Push Model to be more digestible for larger IDPs but harder on smaller IDPs
            2. How much cheaper can we make the Push Model for smaller IDPs?
            3. How much are smaller IDPs willing to use the Push Model?
            4. The heuristics buys us time to co-develop them with IDPs.
            5. Right now, it is unclear to us if it is possible to remove them, but we agree that the Push Model works better.
3. Is there anything else you'd like us to act on?
    a. ben: the timing attack problem was the biggest hurdle. will report back.
    b. goto: neat. we'd like to know if there is anything else you'd like us to act on.

ILLUSTRATIVE MOCKS

JS API



9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★

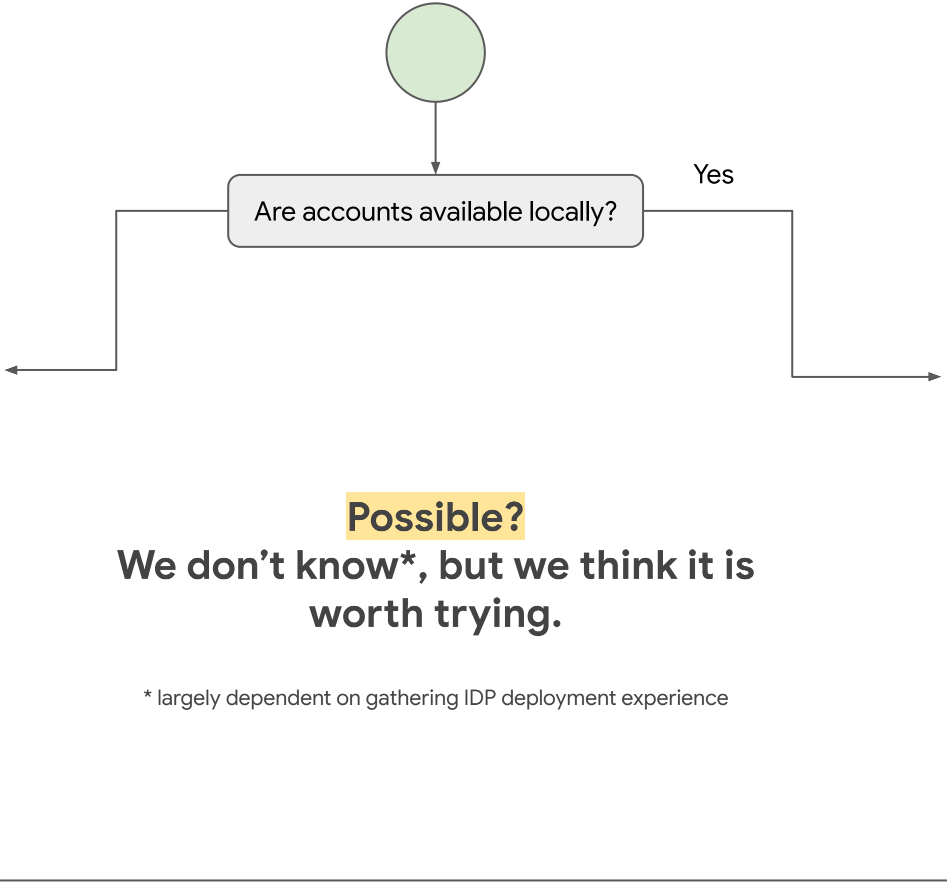
4.83 from 69 reviews

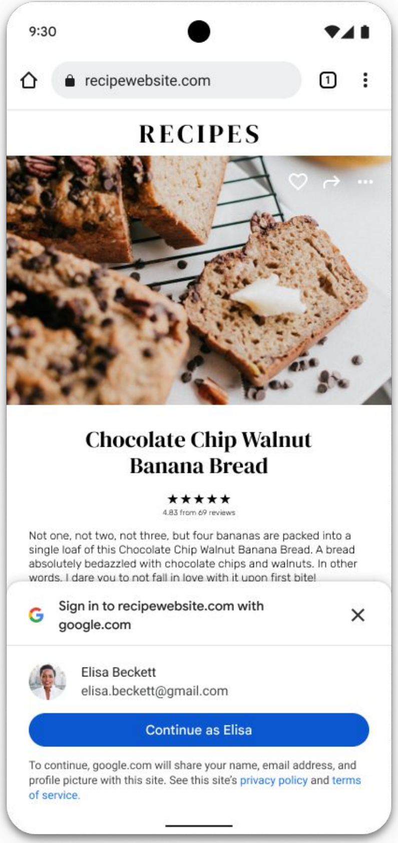
Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

To continue, you can select your account with google.com

Continue





9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★

4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

Elisa Beckett
elisa.beckett@gmail.com

Continue as Elisa

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.

Possible?

We don't know*, but we think it is worth trying.

* largely dependent on gathering IDP deployment experience

ILLUSTRATIVE MOCKS

The push model

The Pull Model

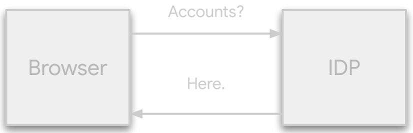
On demand, the browser makes an HTTP request to the IDP that returns the user's accounts.

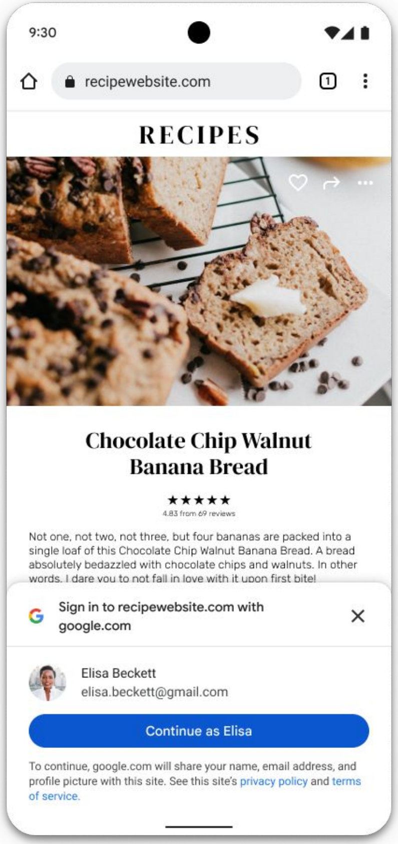
Pros

Simple to implement by IDP. Always in Sync.

Cons

Latency. Timing Attacks.




9:30

recipewebsite.com

RECIPES

Chocolate Chip Walnut Banana Bread

★★★★★

4.83 from 69 reviews

Not one, not two, not three, but four bananas are packed into a single loaf of this Chocolate Chip Walnut Banana Bread. A bread absolutely bedazzled with chocolate chips and walnuts. In other words, I dare you to not fall in love with it upon first bite!

Sign in to recipewebsite.com with google.com

Elisa Beckett
elisa.beckett@gmail.com

Continue as Elisa

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.

The Push Model

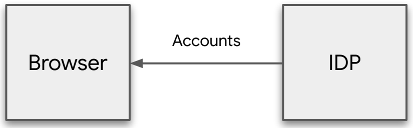
Ahead of time, the IDP saves in the browser the user's accounts.

Pros

Better UX. No attacks.

Cons

Harder to implement for IDPs. IDP announces all accounts rather than only the ones that would use federation. Can be out of sync (e.g. cookies can be cleared; out-of-band changes).



ANNEX

Alternatives Considered

1. Should it only affect IdPs or any website? Do we want a broader API that can be called by any website or just IdPs?
   a. Seemed smart to just be called by IdPs and to only affect FedCM
2. Where should it hang? navigator.credentials? navigator? IdentityCredential? A new interface?
   a. Ideally, it should be IdP specific and FedCM-specific
   b. navigator.recordLogin(); // not ideal because it is not IdP nor FedCM-specific?
   c. navigator.credentials.recordIdpLogin(); // not ideal because it is not FedCM-specific? Does it affect WebOTP? WebAuthn?
   d. navigator.credentials.store({identity: {new IdPSignedInStatus({status: “logged-in”, profile: { … }}}})); // IdPSignedInStatus seems awkward because it needs to be a credential?
   e. IdentiyCredential.idpLogin(); IdentityCredential.idpLogout(); // not ideal because it is not IdP specific?
   f. IdentityCredential.setIdPSignedInStatus(true || false); // christian: i think of IdentityCredential as an RP-thing, so it is kind of nice not to hang on it. Yi’s point: logoutRPs() is in IdentityCredential. Fair point.
   g. IdentityProvider.login(); IdentityProvider.logout(); // Rename the dict to something else, create a new interface, add a new static method
   h. IdentityProvider.logoutRPs();
   i. What else?
3. One method with a parameter?
4. Two methods without parameters? setUserLoggedInStatus(true | false), login({profileData: …}) and logout(), two separate methods seems favorable
   a. configURls? – default to “*”, later add per configURL bits

Rollout

- Proposal: Forwards incompatible option
    - I2S as previously planned
    - Implement userLoggedIn/etc API before or after
    - Once partner(s) are ready, require opt in to the userLoggedIn API to avoid fallback to two-tap
    - API is backwards compatible
    - UX is backwards incompatible :(– is that a problem ???
- Proposal #2: Forwards compatible option
    - Add to the I2S the userLoggedIn API
    - Goal: not make backwards incompatible changes
    - We'll only show the one tap UX if:
        - If we added a parameter in the .well-known file or the configURL to point to a specific header or cookie that represents whether the user is logged in, say, { userLoggedInHeader: "X-Chrome-Consistency-X", userLoggedInCookie: "SID" or maybe, userLoggedInIf: "__Secure-3PSID"}, then, we would at least be able to allow GSI to deploy on desktop at I2S
        - We could also support another newly introduced header, say, a standard one, say, X-User-Logged-In header.
        - And a JS API.
    - We could also support heuristics
        - SES
- Proposal #3: Expose API, store it, but ignore the value on the prompt
    - Expose Login Status API
    - Store it
    - Ignore during the prompt
- Proposal #4: The "undefined" state
    - Expose the Login Status API
    - Login bit: it can be "undefined", "true" or "false.
    - If "undefined", we fetch the accounts list and then set it to "true" or "false" depending on the results of the fetch
    - From that point on, the JS API is how you can change that bit
-

Sequencing

1. Two proposals:
   a. I2S without the IdP Sign-in Status API and Origin Trial the IdP Sign-in Status API later
      i. Christian: how long do we expect the gaia to take?
   b. Add the IdP Sign-in Status API to the Origin Trial and I2S them together
   c. I2S without the IdP Sign-in Status API BUT with the mitigations, and then OT the IdP Sign-in Status API to allow IdP to not have to rely on the mitigations

“The Bold Proposal”

1. Presupposes that the IdP Sign-in Status API is available (a JS and/or a HTTP HEADER API)
2. Presupposes that IdPs are using the IdP Sign-in Status API appropriately
3. If bit is “undefined”,
    a. Fetch accounts list
    b. Set the bit
    c. If the user is logged in
        i. Show the prompt
    d. Else
        i. Don’t know anything (pay the invisible timing attack problem price once)
4. If the bit “false”
    a. Terminate API (Do not show UI)
5. If the bit is “true”
    a. Fetch account list and
    b. If no accounts list is found
        i. Set the bit to false
        ii. Show a prompt that says “oops an error happened”
    c. If accounts are found
        i. Show prompt

“The Migration Mode Proposal”

1. Presupposes that the Login Status API is available (a JS and/or a HTTP HEADER API)
2. IDP’s can “temporarily” opt-into a “hackyModeWillGoAway: true” in the .well-known file.
3. If bit is “undefined” || hackyModeWillGoWay,
    a. Fetch accounts list
    b. Set the bit
    c. If the user is logged in
        i. Show the prompt
    d. Else
        i. Don’t know anything (pay the invisible timing attack problem price once)
4. If the bit “false”
    a. Terminate the API
    b. Do not show UI
5. If the bit is “true”
    a. Fetch account list and
    b. If no accounts list is found
        i. Set the bit to false
        ii. Show a prompt that says “oops an error happened”
    c. If accounts are found
        i. Show prompt

“The Heuristics Proposal”

1. This does NOT presuppose the IdP Sign-in Status API
2. We DO implement the heuristics
3. If bit is “undefined” || hackyHeuristicsSayThatThisIsAForRealIDP,
    a. Fetch accounts list
    b. Set the bit
    c. If the user is logged in
        i. Show the prompt
    d. Else
        i. Don’t know anything (pay the invisible timing attack problem price once)
4. If the bit “false”
    a. Terminate the API
    b. Do not show UI
5. If the bit is “true”
    a. Fetch account list and
    b. If no accounts list is found
        i. Set the bit to false
        ii. Show a prompt that says “oops an error happened”
    c. If accounts are found
        i. Show prompt