

Attribute Sync Service

Introduction

The Data Sync Service provides implementers with the option of storing items on the Adhese server, which can be used for inventory, campaign insights, and targeting. The storage can be linked to any kind of identifier as per configuration.

Server Side Memory

The memory service allows an implementer to save certain data points linked to a unique key. The key can be either a user ID, which is only allowed with the user's explicit consent, or any other non-personal key (e.g., a DOOH device ID).

The event is a string that will be saved and reported exactly as sent in. A human-readable label can be added in the Adhese inventory configuration.

The third parameter of the memory service is optional and indicates the time-to-live of the saved data point. If no value is passed, the default of 14 days will be applied.

```
https://ads-[customer].adhese.com/usersync/handlers/[service_handler]/user_sync?id=[event1],[event2]&u=[userid]&ttl=[expiration_in_minutes]
```

Example

```
https://ads-[customer].adhese.com/usersync/handlers/ec_view/user_sync?id=pricewatch&u=123abc&ttl=20160
```

Example with multiple events

```
https://ads-[customer].adhese.com/usersync/handlers/ec_view/user_sync?id=pricewatch,laptop,macbook&u=123abc&ttl=20160
```

Removing Data for a Key or User

All data stored in Redis will expire after a given time. The 'right to be forgotten' is one of the most important reasons, and it is essential to provide a way to remove stored data at the request of the rights holder (the end-user).

To remove data, simply call the user sync service with the user ID or key you want. The service will remove all data stored for that key and then remove the key itself.

Example

```
https://ads-[customer].adhese.com/usersync/handlers/ec_view/user_sync?iu=123abc
```

User sync for Open Market OpenRTB

A very specific application of the Data Sync Service can be used to support the OpenRTB User Sync process.

The User Sync process is designed to match P users with SSP users. We recommend syncing users once every day at the earliest opportunity.

The User Sync service is triggered by Adhese directly on the client, enabling the SSP to establish a connection and transfer the resulting user ID to Adhese.

According to the setup, a simple or complex workflow is available. A complex workflow leads to a 1st party user sync cookie, while a simple workflow leads to a 3rd party user sync cookie.

User Consent: This application of the sync service depends on a cookie that contains a unique ID. Users need to be informed about this and give consent prior to calling the service. Adhese has the contractual right to audit the use of this service to ensure it is legally implemented.

Simple workflow

1. The user navigates to newz.zz
2. Browser calls SSP endpoint `https://ssp.com/sync?partner=adhese`
3. The SSP redirects to the user sync endpoint
4. Browser calls Adhese user sync endpoint `https://user-sync.adhese.com/handlers/ssp/user_sync?u=IH3D8DXL-20-KV8Y`
5. We issue a cookie `Set-Cookie: ssp_uid=IH3D8DXL-20-KV8Y; Domain=.adhese.com; Expires=Fri, 25-Dec-2024 09:57:40 GMT; Path=/`
6. Browser requests an ad on `https://ads-adhese.com/ad/sl_foo_-bar`
7. We read ssp_uid cookie to send it over to the SSP's API

Complex workflow for 1st domain implementations

If a publisher serves on its own domains, the process is more complicated. There are no calls made to *.adhese.com.

1. The user navigates to publisher-domain.com
2. Browser registers request domain with Adhese user sync discovery `https://user-sync.adhese.com/handlers/ssp/user_sync_discovery?domain=adhese.publisher-domain.com&protocol=https`

protocol query parameter is optional, it defaults to https

1. We read the domain parameter
2. We issue a cookie: `Set-Cookie: user_sync_domain=adhese.publisher-domain.com; Path=/; Expires=Fri, 25-Dec-2024 10:16:23 GMT`
3. The browser calls the SSP endpoint `https://ssp.com/sync?p=publisher`
4. SSP redirects to user sync discovery
5. The browser calls Adhese user sync discovery `https://user-sync.adhese.com/handlers/ssp/user_sync_discovery?u=IH3D8DXL-20-KV8Y`
6. We read the user_sync_domain cookie
7. We redirect the call to the user sync endpoint running on the correct domain
8. The browser calls Adhese user sync endpoint `https://adhese.publisher-domain.com/handlers/ssp/user_sync?u=IH3D8DXL-20-KV8Y`
9. We issue a cookie: `Set-Cookie: ssp_uid=IH3D8DXL-20-KV8Y; Expires=Fri, 25-Dec-2024 09:57:40 GMT; Path=/`
10. Browser requests an ad on `https://adhese.publisher-domain.com/ad/sl_foo_-bar`
11. We read the ssp_uid cookie to send it over SSP's API

Revision #10

Created 18 October 2024 07:45:26 by Casper Steuperaert

Updated 25 March 2025 08:55:32 by Casper Steuperaert