Protect My Choices v2.0 (Beta): How to read the AdChoices Signal

Background

When consumers install the Protect My Choices (v2.0) extension – which is currently available for the desktop versions of Google Chrome and Mozilla Firefox and the desktop and mobile versions of Apple Safari – any preferences expressed through the WebChoices tool will be stored as a string in the extension, according to the AdChoices Signal specification document. The value of this string can be read by any participant with the code on page.

Accessing User Preferences | JavaScript 'Reader'

Web participants can access these user preferences from the extension, as required and when available, in real time. Vendors who serve ads or collect data about users, such as ad servers, exchanges, DSPs, DMPs, CDPs, etc., will need to listen for the "**ExtensionLoaded**" event name on the top application level. This sample JS code confirms that the extension was loaded:

```
window.addEventListener('message', (event) => {
    if (event.data.type === "ExtensionLoaded") {
        // Do something when the extension is loaded
    }
});
```

If the extension is loaded, the following JavaScript code retrieves the user preference string:

```
window.postMessage({type: "GetAdPreferences"}, "*");
```

When combined, the implementation code should look something like this:

```
window.addEventListener('message', (event) => {
  if (event.data.type === "ExtensionLoaded") {
    window.postMessage({type: "GetAdPreferences"}, "*");
  } else if (event.data.type === "AdPreferences") {
    const adPrefs = event.data.data; // Store preferences
    }
});
```

Vendors should send the preference string to themselves so that it can be passed on to other parties. If vendors need to know the users' preferences before ad decisioning, they should delay ad decisioning until after retrieving the string.[1]

[1]Those who integrate into the AdChoices tools can receive the string along with an identifier via API for any backend (asynchronous) preference or suppression operations.

Participating vendors may also access user choices via a header string created by the Protect My Choices 2.0 extension. When users have the extension installed, the extension will insert the "X-Adchoices" (Chrome) or “Cookie2” (Safari) string into a header to be attached to every request on any website. The header value can then be decoded to interpret the AdChoices Signal as demonstrated in the examples below.[2]

Example Strings

Example 1: No Global Preference, Three Participant Preferences

Base64 (base64url[3]):
BYVHiWSADABAAIQAwAAA

Binary:
000001011000010101000111100010010110010010000000000011000000000000100000000000010000100010000000000011000000000000000

Annotated Binary:

Bits	Field	Value/Description
000001	*(Header Section)* Version	Version 1 of the specification is used.
01100001010100011110001001011001	Timestamp	1632756313 seconds since epoch, or Monday, September 27, 2021 3:25:13 PM UTC.
0010	Global IBA Choice Status	2; the user has not expressed a preference.

[2] We have sample code available upon request for reading and decoding the AdChoices Signal in a browser header.
[3] The spec requires a base64url string. Padding characters (‘=’) must not be used. See RFC 4648 sec 5.

000000000011	*(Per-Participant Records Section)* Number of Per-Participant Records	3 per-participant records will follow.
000000000001	*(First Participant Choice Status Record)* Participant ID	Participant ID 1 (as defined in the participants reference file).
0000	Choice Status	0; the user has explicitly chosen to limit data collection and use for IBA.
000000000010	*(Second Participant Choice Status Record)* Participant ID	Participant ID 2.
0001	Choice Status	1; the user has explicitly chosen to allow data collection & use for IBA.
000000000011	*(Third Participant Choice Status Record)* Participant ID	Participant ID 3.
0000	Choice Status	0; the user has explicitly chosen to limit data collection and use for IBA.
000000000000	*(Per-Category Records Section)* Number of Per-Category Records	0 per-category records will follow.

Example 2: No Global Preference, One Category Preference

Base64 (base64url):
BYVHiWSAAABAZEA

Binary:
000001011000010101000111100010010110010010000000000000000000000001000000011001
0001

Annotated Binary:

Bits	Field	Value/Description
000001	*(Header Section)* Version	Version 1 of the specification is used.
01100001010100011110001001011001	Timestamp	1632756313 seconds since epoch, or Monday, September 27, 2021 3:25:13 PM UTC.
0010	Global IBA Choice Status	2; the user has not expressed a preference.
000000000000	*(Per-Participant Records Section)* Number of Per-Participant Records	0 per-participant records will follow.
000000000001	*(Per-Category Records Section)* Number of Per-Category Records	1 per-category records will follow.
000000011001	*(First Category Record)* Category ID	Category ID 25: Travel (as defined in the category reference file).
0001	Category Preference	1; the user has expressed an explicit preference to allow/include this category.

Example 3: Global Preference Only (for IBA)

Base64 (base64url):
BYVHiWQAAAAA

Binary:
000001011000010101000111100010010110010000000000000000000000000000

Annotated Binary:

Bits	Field	Value/Description
000001	*(Header Section)* Version	Version 1 of the specification is used.
01100001010100011110001001011001	Timestamp	1632756313 seconds since epoch, or Monday, September 27, 2021 3:25:13 PM UTC.
0000	Global IBA Choice Status	0; the user has expressly chosen to limit IBA.
000000000000	*(Per-Participant Records Section)* Number of Per-Participant Records	0 per-participant records will follow.
000000000000	*(Per-Category Records Section)* Number of Per-Category Records	0 per-category records will follow.

In this example, there are no per-participant or per-category records, so no such records appear, however the number of per-participant or per-category records fields are always required for correct parsing.

Example 4: No Global Preference, Four Participant Preferences

Base64 (base64url):
BYVHiWSAEDsB54AzQUeAAAA

Binary:
000001011000010101000111100010010110010010000000000100000011101100000011110011110000000000110011010000101010001111000000000000000000

Annotated Binary:

Bits	Field	Value/Description
000001	*(Header Section)* Version	Version 1 of the specification is used.
011000010101000111100010010110 01	Timestamp	1632756313 seconds since epoch, or Monday, September 27, 2021 3:25:13 PM UTC.
0010	Global IBA Choice Status	2; the user has not expressed a preference.
000000000100	*(Per-Participant Records Section)* Number of Per-Participant Records	4 per-participant records will follow.
000011101100	*(First Participant Choice Status Record)* Participant ID	Participant ID “00236” (33Across).
0000	Choice Status	0; the user has explicitly chosen to limit data collection and use for IBA.
011110011110	*(Second Participant Choice Status Record)* Participant ID	Participant ID “01950” (Adform).

0000	Choice Status	0; the user has explicitly chosen to limit data collection and use for IBA.
000011001101	*(Third Participant Choice Status Record)* Participant ID	Participant ID “00205” (Amazon).
0000	Choice Status	0; the user has explicitly chosen to limit data collection and use for IBA.
010100011110	*(Fourth Participant Choice Status Record)* Participant ID	Participant ID “01310” (Beeswax).
0000	Choice Status	0; the user has explicitly chosen to limit data collection and use for IBA.
000000000000	*(Per-Category Records Section)* Number of Per-Category Records	0 per-category records will follow.

Example 5: No Global Preference, Three Category Preferences

Base64 (base64url):
BYVHiWSAAADABEAUQExA

Binary:
000001011000010101000111100010010110010010000000000000000000000011000000000001000100000000001010001000000010011 0001

Annotated Binary:

Bits	Field	Value/Description
000001	*(Header Section)* Version	Version 1 of the specification is used.
01100001010100011110001001011001	Timestamp	1632756313 seconds since epoch, or Monday, September 27, 2021 3:25:13 PM UTC.
0010	Global IBA Choice Status	2; the user has not expressed a preference.
000000000000	*(Per-Participant Records Section)* Number of Per-Participant Records	0 per-participant records will follow.
000000000011	*(Per-Category Records Section)* Number of Per-Category Records	3 per-category records will follow.
000000000001	*(First Category Record)* Category ID	Category ID 1: Automotive (as defined in the category reference file).
0001	Category Preference	1; the user has expressed an explicit preference to allow/include this category.

000000000101	*(Second Category Record)* Category ID	Category ID 5: Dating
0001	Category Preference	1; the user has expressed an explicit preference to allow/include this category.
000000010011	*(Third Category Record)* Category ID	Category ID 19: Pets
0001	Category Preference	1; the user has expressed an explicit preference to allow/include this category.

Important note: In all of the aforementioned examples, the binary strings are constructed first, according to the AdChoices string specification. The base64url value is derived by converting the binary to base64url. Converting the base64url values back to binary can result in values that don't 100% match the original binary strings. This is because of padding (extra zeros) added to the end of the string due to the conversion process. This extra padding should be ignored when parsing the binary strings. (Given that the strings include fields for "number of per-participant records" and "number of per-category records," it will be evident when the padding begins.)

For example, a binary string according to the spec may look like this:
00000101100001010100011110001001011001001000000000001100000000000100000000000001000010000000000011000000000000000

But when the same string is decoded from its base64url value, it looks like this:
00000101100001010100011110001001011001001000000000001100000000000100000000000001000010000000000011000000000000000**000000**

The strings are identical except for the added zeros at the end. If not accounted for, this may cause some automated tests to fail.

Last updated: April 16, 2025
v.2.0.5