Global Privacy Platform

Streamlining technical privacy and data protection signaling standards into a singular schema and set of tools which can adapt to regulatory and commercial market demands across channels.

March 2021

Draft in Request for Public Comment until April 8, 2021

This document has been developed by the Global Privacy Working Group.

IAB Tech Lab's Global Privacy Working Group will streamline technical privacy standards into a singular schema and set of tools which can adapt to regulatory and commercial market demands across channels. This group is open to members with technical expertise interested in contributing to development of industry privacy and data protection standards. This working group looks to the IAB Privacy Lab, local IABs and other relevant voices for policy requirements. This group will build on the foundations and experience of IAB Tech Lab's GDPR and CCPA Technical Working Groups. It will support the existing work and markets of those groups while expanding to cover new markets and channels.

Global Privacy Working Group Roster

At the date of publishing, Global Privacy Working Group Roster is made up of 295 individuals representing 127 organizations. Full roster details can be viewed here.

About IAB Tech Lab

Established in 2014, the IAB Technology Laboratory (Tech Lab) is a non-profit consortium that engages a member community globally to develop foundational technology and standards that enable growth and trust in the digital media ecosystem. Comprised of digital publishers, ad technology firms, agencies, marketers, and other member companies, IAB Tech Lab focuses on solutions for brand safety and ad fraud; identity, data, and consumer privacy; ad experiences and measurement; and programmatic effectiveness. Its work includes the OpenRTB real-time bidding protocol, ads.txt anti-fraud specification, Open Measurement SDK for viewability and verification, VAST video specification, and Datalabel.org service. Board members/companies are listed at https://iabtechlab.com/about-the-iab-tech-lab/tech-lab-leadership/. For more information, please visit https://iabtechlab.com.

IAB Tech Lab Contact

Alex Cone
Sr. Director, Product Management

Global Privacy Platform

Overview

About this RFC

IAB Tech Lab's Global Privacy Working Group is making considerable progress toward a Global Privacy Platform (GPP) architecture which will enable data use transparency and control for users, resulting signals to the digital advertising supply chain supporting adherence to regional regulations and norms. This aim is clearly aligned with the public mission of the Global Privacy Working Group:

> Streamline technical privacy and data protection standards into a singular schema and set of tools which can adapt to regulatory and commercial market demands across channels.

This is an ***early*** RFC document. It is being made open for a period of time for public comment on Global Privacy Working Group's progress to date. **Why "*early*"? It does not represent every detail and nuance required to support the full ambitions of a ready to deploy GPP.** This is not a final product, but a good faith effort toward the mission and overall vision. Tech Lab's Global Privacy Working Group is publicly declaring its intent to create something close to what is detailed in this document. See How to Submit Comments below.

Goals for Early RFC

- Formalize IAB Tech Lab's intent to bring a GPP technical standard to market
- Solicit feedback on the high-level architectural designs developed by the Global Privacy Working Group to date
- Use this feedback to catapult Global Privacy Working Group forward in the directions most important to our industry and the broader public
- Demonstrate tangible progress toward supporting new markets like Canada and those covered by IAB's Cross Jurisdiction Privacy Project (CJPP) seeking to provide users with data use transparency and control
- Clearly link GPP to Tech Lab's simultaneous work on Addressability and Accountability in an industry experiencing major disruptions rooted in data protection and privacy concerns

GPP Value

Users

Internet users will see incremental predictability in transparency and control over time as privacy and data protection norms converge. Many more users, in countries previously uncovered by powerful digital advertising transparency and control tools, can see into and have a say over data uses for digital advertising.

Publishers, Advertisers and Ad Technology Products

Businesses will see reduced cost of maintaining privacy and data protection controls for users across the regions they work in. GPP adopters can adapt to regional changes and even gradual convergence in privacy and data protection experiences without technology switching costs.

Guiding Principles

- Reduce fragmented privacy signaling technologies that are costly for implementers
- Accommodate regional and country differences in privacy and data protection norms even if over time there is convergence
- Regions and countries should have support for their jurisdictions without needing to get other regions and countries to agree to adaptations
- Inclusive of existing in-use consent formats, recognizing that there have been a number of disparate efforts that need to be reflected.
- The cost associated with adopting new privacy and data protection signaling technologies can be significant therefore we need to both minimizes costs and provide value with new features
- Flexibility is critical to support the needs of users, publishers, advertisers and the advertising technology products they use.

Multi-Jurisdictional Design

There is a proliferation of data protection laws particular to countries and regions and we cannot ignore the issue of jurisdictional overlap of data protection laws. **These RFC designs acknowledge "jurisdictional overlap."** Jurisdictional overlap is a consequence of the extra-territorial nature of various data protection laws, including the GDPR which applies in the European Union but can also be applied in other jurisdictions under a few circumstances. In other words, data protection laws can be applied outside of the jurisdiction in which the law is adopted. For example, Country X's privacy law may apply to a user based in Europe who visits a country X digital property. This can result in a situation where an interaction falls within the scope of two or more jurisdictions, which could lead to participants, particularly digital properties and CMPs, needing to get consent compliant under multiple jurisdictions. Similarly, downstream vendors would need to consider which jurisdiction was relevant to their action. This can be problematic where there is a significant difference in requirements between different jurisdictions—for instance, some purposes may be specific to a jurisdiction, or there may be different legal requirements. Such situations could be supported by combining user preferences across different jurisdictions within a single signal.

To do this, policy inputs to date indicate that the technical architecture should clearly indicate which jurisdiction(s) are supported by a given GPP signal. The proposed approach, detailed below in the "Proposed Architecture", is to have sections (where sections equate to country or regional jurisdiction) within the GPP string, where each section is dedicated to one jurisdiction with legal bases or permissions specific to that jurisdiction. CMPs only generate a GPP string section for the jurisdiction(s) that the digital property requests and transmit that information to downstream vendors, along with an indication of the jurisdiction(s) that the digital property will apply. Participants thus make their own determination over how to proceed with the information provided to them via the digital property's CMP.

In practice the string could support a number of sections. For example, if a user visits the same digital property, but from a new location in one of the supported GPP jurisdictions, that digital property can choose to apply the local jurisdiction's relevant policy framework and generate a new section in the GPP string specific to that jurisdiction. Vendors will have an indication that the digital property is now applying another jurisdiction to the transaction and will also know the initial choice made in the original jurisdiction as well as the 'new' choice.

Note: *IAB Canada represents a new market seeking support by the technical designs in this document. IAB Canada recently released its own flavor of Transparency & Consent Framework policy for public comment.*

Proposed Architecture

"TC String" as a Starting Point

This proposal builds on the Transparency & Consent Framework v2.0 (TCF) concept of a "TC String" composed of flexible and discrete "Segments", expanding these to support multiple existing and new privacy formats. This is our preferred path given the broad adoption of TCF v2.0 across digital properties, CMPs and ad technology products.

This architecture seeks to minimize disruptions to currently adopted privacy signaling, such as the TCF v2.0 and USPrivacy, while at the same time giving potential GPP adopters reasons to make the update to their production implementations.

Basic Signal Makeup

GPP Sections encode:

- Disclosures to and control by the user, with a given granularity dependent on regional norms (by-purpose, by-vendor, by-legal-basis, or "omnibus" all or nothing)
- Metadata about the context of those choices (timestamps, versions, CMP info, regions, publisher info, regional or jurisdictional applicability, integrity signatures)
- Possible additional legal, publisher, or framework restrictions

Section specifications will clearly define which of the above data are represented, and in what form. Whenever possible, the various *technical* enumerations, that have been developed for TCF v2.0 can be used directly or adapted:

- Integer identifiers for version & screens, CMP ID's, vendors ID's, purposes: integers
- CMP ID's from TCFv2
- Where applicable to the jurisdiction or region, the vendor ID's and data processing purposes as enumerated in the TCFv2 GVL can be reused in whole or adapted. Additionally, the version

See “Discrete Sections” below for more detail.

Header

We are introducing a concept of a Header section on top of the TC String-inspired architecture. The purpose of the Header is to identify which regions’ transparency and control signals are included in a string payload and be a table of contents where to find each region in the string payload (broken into discrete sections). It is basically an ordered list of discrete sections that equate to different regions and counties and their jurisdictions. It lets readers understand what is present in the string and in what order. (See Discrete Sections below)

The header contains only a GPP version, the region ID(s) and index of the place of the associated region section in the string. The header delegates regional policy versions and technical encoding versions to each substring section so that each may develop independently of each other and the header design. (See Discrete Sections below)

Region IDs

This is an example of how Region IDs are enumerated. It is designed in a way to avoid needing to redesign anything about TCF upon which this proposal is based.

Section ID	Description
1	EU TCF v1 section (see note below)
2	EU TCF2 section (see note directly below)
3	GPP Header section (see note directly below)
4	GPP signal integrity section
5	Canadian TCF section
6	USPrivacy *Unencoded Format* section
7	USPrivacy *Encoded Format* segment. (see example below)

...	...

***Note:** In order to make it simple to distinguish a GPP string from the existing IAB Europe TCF v2.0 TC String the first space in the header should be the version. This would allow current implementations to more easily understand and adapt to a GPP string. If the reader of a string finds "C" as the first character this indicates the string is IAB Europe's TCF v2.0 ("2" in bits corresponds to letter "C" in base64). If the reader of a string finds a "D" as the first character this indicates the string is GPP ("3" in bits corresponds to letter "D" in base64).*

Header Encoding

The Header consists of the following encoded fields:

Position	Type	Description
Type	Int(6)	Fixed to 3 as "GPP Header field"
Version	Int(6)	Version of the GPP spec (currently 1)
Sections	Range	List of Section IDs that are contained in the GPP string. Each ID represents a discrete Section that will be contacted to the Header Section. The IDs must be represented in the order the related Sections appear in the string. This is required to make real time string processing less resource intensive.

Discrete Sections

IAB Europe's TCF v2.0 introduced the concept of Segments to privacy and data protection signaling. This design allows for flexibility needed by GPP. With certain tweaks described here, the TCF v2.0 TC String-inspired concept of discrete segments can be used to support multiple regions from one architecture while maintaining the ability to modify these discrete sections as needed.

Each string segment is scoped to the same body that updates the spec. This allows for regional policies sovereignty to make changes that might include more delimited information. For example, what if in TCF v3.0 "out of band" vendors were eliminated and resulted in the removal of DisclosedVendors and AllowedVendors?[1] That should not require a version bump to the GPP string specification.

[1] This is a hypothetical example and not meant to be read as something that is actually happening.

Delimiters

In order to be backward compatible with IAB TCF and USPrivacy string format the delimiter used to separate segments is “~” (tilde).

Note: *URL-safe characters are important to meet the integration needs of those not reading privacy signals server side or via the client-side APIs. URL-safe characters are:*

- *A-Z, a-z, 0-9*
- *- (minus)*
- *. (dot)*
- *_ (underscore)*
- *~ (tilde)*

“.” and “-” and “_” are in use which leaves “~” as the only possible delimiter unless we re-use “.”.

Section Encoding

Each region’s discrete section is encoded according to that region’s needs. This means today’s implementations that read and adapt to TCF v2.0 signals or US Privacy signals need not change their logic for a given discrete section of the string, as long as the implementation is aware of where the discrete section is.

For new sections, the following guidelines are envisioned. Guidelines like these help developers quickly adopt new regions and be able to parse new sections without the need to reinvent new data types. The format follows the encoding logic introduced by the TCF due to its prevalence in market and because this RFC builds so heavily upon TCF v2.0 technical design.

Possible data types may include but are not limited to:

Type	Encoding	Description
Boolean	1 bit	0=true, 1=false
Integer (fixed length of x)	x bit	A fixed amount of bit representing an integer. Usual lengths are 3, 6 or 12 bit. Example: int(6) “000101” represents the number 5
String (fixed length of x) (including country codes)	x*6 bit	A fixed amount of bit representing a string. The character’s ASCII integer ID is subtracted by 65 and encoded into an int(6). Example: int(6) “101010” represents integer 47 + 65 = char “h”
Datetime	36 bit	A datetime is encoded as a 36 bit integer

		representing the 1/10th seconds since January 01 1970 00:00:00 UTC. Example JavaScript representation: Math.round((new Date()).getTime()/100)
Bitfield (fixed length of x)	x bit	A fixed amount of bit. Usually, each bit represents a boolean for an ID within a group where the first bit corresponds to true/false for ID 1, the second bit corresponds to true/false for ID 2 and so on.

Range	variable	A range field always consists of the following fields:
		1. int(12) - representing the number of items to follow
		2. (per item) Boolean - representing whether the item is a single ID (0/false) or a group of IDs (1/true)
		3. (per item) int(12) - representing a) the single ID or b) the start ID in case of a group
		4. (per item + only if group) int(12) - representing the end ID of the group
		
		Example:
		int(12) = 2 // 2 items
		Bool = 0 // item 1 is type single ID
		int(12) = 3 // ID of item 1
		Bool = 1 // item 2 is type group
		int(12) = 5 // item 2 start ID
		int(12) = 8 // item 2 end ID
		
		Range = [3,5,6,7,8]
		Bits = 000000000010 0 000000000011 1 000000000101 000000001000
		
		Note: items may not be in sorted order.

When defining a new Section, regional policy writers should consider the above format in order to describe their segment.

Example Implementation

Example Field name	Example Type	Example Description
Version	int(6)	Version of Specification XYZ
LastUpdated	datetime	Datetime of last update
OptOutPurposes	Bitfield(6)	Purposes for which the user opted out,

		each bit representing the purpose ID
...	...	...

Note: *It is recommended to use Field names in CamelCase and without any special chars or space. This allows to use the same field names within other APIs (e.g., GPP JS API or GPP Mobile API)*

USPrivacy *Encoded Format* Example

In order to support backward compatibility, the Section for USPrivacy is not encoded in the format below, but in the original human readable “1NYY” format (see “Region IDs” above). In order to align USPrivacy to the same encoding format, we propose a Section definition for “USPrivacy *Encoded Format*” below.

Field name	**Type**	**Description**
Version	int(6)	The version of this string specification used to encode the string
Notice	Int(2)	Has notice been provided as required by 1798.115(d) of the CCPA and the opportunity to opt out of the sale of their data pursuant to 1798.120 and 1798.135 of the CCPA. Possible values: 0 = not applicable 1 = yes 2 = no
OptOutSale	Int(2)	Has user opted-out of the sale of his or her personal information pursuant to 1798.120 and 1798. If CCPA applies, only 1 (yes) or 2 (no) can be used. Possible values: 0 = not applicable 1 = yes 2 = no
LSPACovered	Int(2)	Publisher is a signatory to the IAB Limited Service Provider Agreement(LSPA) and the publisher declares that the transaction is covered as a “Covered Opt Out Transaction” or a “Non Opt Out Transaction” as those terms are defined in the Agreement. Possible values: 0 = not applicable 1 = yes 2 = no

Sub-Sections / Segments

If a section uses sub-sections in order to separate information or to be more flexible, it can use the delimiter "." (dot) to separate the sub-sections from each other. TCF v2 uses this method.

Creating a GPP String

In order to create the GPP string:

1. For each section:
    a. When section is using the recommended base64-websafe encoding:
    Create a bit representation of the Section's header (if it exists) and each sub-section and convert them to base64-websafe without padding (i.e., removing "=" at the end) and concatenate the header and all sub-sections using "." (dot).
    b. When section is using a different encoding:
    Ensure that the data is websafe and does not include the "~" (tilde) character.
2. Create a bit representation of the GPP header section. Include all IDs for discrete sections in a sorted order. Convert it to base64-websafe without padding.
3. Concatenate the GPP-header as first item to the encoded versions of the discrete sections (step 2) using "~" (tilde) as delimiter.

Dealing with Length

TCF v2.0 implementers are finding many examples of strings too long for certain applications. An example case is when a digital property uses TCF v2.0 publisher restrictions to set all flexible vendors to consent as a legal basis and the context is Accelerated Mobile Pages (AMP). The signal length may become more than the context allows. A design that expects additional discrete Sections to the TC String-inspired concept is likely to run into the same challenges. This is especially if certain regions' policies and thus Section encodings are very similar. This is already the case with IAB Canada's derivative of IAB Europe's TCF policy. While we do not currently have an answer for this, the Global Privacy Working Group seeks public comments during RFC to overcome these challenges.

GPP APIs

JavaScript

Our goal is to have an API similar to `__tcfapi` but more general given the nature of GPP. This could look something like:

```
__gpp(version, command, callback, parameter)
```

…with the following supported commands:

- `ping`

- `addEventListener`
- `removeEventListener`
- `hasSection`
- `getSection` (output parsed section as an object)
- `getSectionField` (requires parameter to be present. Output a named field from a section)
- [sectionname].[command] (e.g., "`tcf.getTCData`" or "`usp.getUSPData`")

> ***Note:*** *this is not an exhaustive definition of a JS API. It is provided to give the public an idea of where this proposed architecture is headed.*

Non-web (Mobile, CTV, etc.…)

We seek the input of experts in non-web environments to propose how we can copy the relevant pieces of a JS API into non-web logic and interfaces.

Signal Integrity

The Global Privacy Working Group is committed to introducing signal integrity technology for GPP. Right now, the group is evaluating two proposals for signal integrity.

- Cryptographic Security Foundations for Programmatic Ads Ecosystem
- JWT Consent Token Proposal

The direction chosen may have an effect on this RFC's overall architectural proposal.

Vendor Registry

We recognize there are existing vendor lists (see note with non-exhaustive list below). In the near-term it is recommended that close cousins to TCF (see IAB Canada's TCF policy in public comment) build upon the current Global Vendor List. Notably, the proposed GPP string architecture is supported by the Global Vendor List (GVL) and CMP list. A benefit of this near-term approach is the utilization of the current vendor ID, where there is overlap in TCF derived policy registrants. TCF-like or derived policy is described as a direct or near direct adoption of TCF's purposes, special purposes, features and special features) jurisdictions.

For non TCF-like approaches, Tech Lab's Transparency Center, which is already hosting the Limited Service Provider Agreement (LSPA) signatory list, could be a central API that pulls from local privacy and data protection signaling policy registries. During public comment, the Global Privacy Working Group will discuss this path.

> Known vendor lists:

- TCF v2.0 GVL
- Google Additional Consent (companies on Google's ATP list not part of TCF v2.0)
- NAI opt out
- DAA opt out
- Limited Service Provider Agreement Signatory List (includes publishers and advertisers)

Unified Libraries and SDKs

It is our intention to support official open-source projects to aid integration to this spec. Existing official open-source projects like iabtcf-es could be extended and consolidated to work with GPP.

Vision for Updates from TCF v2.0 and/or USPrivacy to GPP

We reiterate the intention of this proposed architecture is to support privacy and data protection signaling world-wide. That means the proposed architecture supports TCF v2.0 and USPrivacy in addition to new markets. The future finalization of any GPP technical specification does not equate to an end of life for region-specific technical specifications outside of GPP. However, it is the Global Privacy Working Group's aim to design GPP in such a way that transition would be straightforward and as low cost as possible. It is also the working group's aim for GPP to provide value that leads market participants to adopt it. This way, after a period of time, we are left with only a single technical standard to maintain and support.

How to Submit Comments

Comments on this RFC may be submitted to globalprivacy@iabtechlab.com.