Transparency & Consent Framework

Cookie and Vendor List Format

Draft for Public Comment

v1.0a

Table of Contents

Introduction

In February 2017, the IAB Europe assembled parties representing various participants in the online advertising ecosystem, in particular parties from both the supply and demand side of the ecosystem, to work collectively on guidance and solutions to the requirements of the General Data Protection Regulation (the "GDPR"). That working group is currently known as the GDPR Implementation Working Group (or the "GIG")). One of the working groups within the GIG was tasked with developing guidance on consent as a legal basis of processing. An additional working group developed out of that working group to develop a standard solution for companies to use, where and if necessary, to request, obtain and disseminate consent to various parties in the online advertising ecosystem that may be relying on consent as a legal basis of processing and/or may have parties integrated with them that rely on consent.

This document is a draft for public comment. Please submit your general feedback to feedback@advertisingconsent.eu and any technical feedback to transparencyframework@iabtechlab.com by April 8, 2018.

About the Transparency & Consent Framework

The scope of the working group's initiative increased further into a broader initiative to develop an industry solution to

1. (allow website operators to control the vendors it wishes to allow to access to its users' browsers/devices and process their personal data and disclose these choices to other parties in the online advertising ecosystem;
2. seek user consent under the ePrivacy Directive (for setting cookies or similar tech and accessing info on a device) and/or the GDPR in line with applicable legal requirements and disseminate the consent status through the online advertising ecosystem;
3. have one place to go to
    (a) understand privacy-related disclosures about those vendors;
    (b) use those disclosures to make privacy-related disclosures to its users generally;
    (c) provide disclosures required to be provided by vendors that are Controllers to end users;
    (d) seek user consent in line with applicable legal requirements where vendors may require it under the ePrivacy Directive and/or GDPR for various purposes, and
    (e) disseminate the consent status through the online advertising ecosystem.

The various pieces of the framework are the following:

- A global vendor list
- The reference architecture (for cookie format and vendor list and related API's)
- Policy underlying
    - the disclosures to be made by vendors included on the global vendor list
    - the use of the global vendor list and the reference architecture

The transparency and consent framework and the various standards introduced by it, including the standard detailed below, are a work-in-progress and currently designed to be used for testing.

About the Transparency & Consent Standard

For purposes of this documentation, the following terms have the following definitions:

- "***CMP***" means a company that can read the vendors chosen by a website operator and the consent status of an end user (either service specific (through a first-party cookie) or global (through a third-party cookie). A CMP is not synonymous with a company that surfaces the user interface to a user (although it can be the same).
- "***Purposes***" mean the purposes for which a Controller enabled by a website operator is using personal data collected from (or received by a third party) about an end user.
- "***Daisybit***" means information compressed into a binary value and passed throughout the online advertising ecosystem through the OpenRTB specification.
- "***Vendor***" means a third party that a website operator is using in connection with surfacing content to its end users that either (1) accesses an end user's device or browser; and/or (2) collects or receives personal data about the website operator's end users. As such, a vendor need not be a Controller.

License

Consent cookie format and vendor list format v1.0

Version History:

v1.0a – Big-endian format specified as cookie bit format.
v1.0 - Initial version published December 19, 2017

In the proposed GDPR Global DaisyBit consent solution, what purpose does the cookie serve?

After the interaction between an end user and the Consent Manager Provider (CMP) UI, the consent info is stored as a third-party cookie in the user's browser. The data in the cookie answers the question: "Which vendors and purposes did the user give consent for?"

What information is stored in the cookie?

The data stored in the cookie is divided into 3 parts:

1. Metadata of the consent info, e.g. cookie version, when updated, vendor list version, what compression scheme applied on the data.
2. What purposes (globally-applicable, not per-vendor) the user has given consent for.
3. What vendors the user gave consent to.

What is not being supported in the cookie format?

- Multiple vendor lists support (or regional lists)
- Intelligent handling of deleted vendors
- Distinct recording of "Consent Revoked" (as opposed to "No Consent")
- "Not yet consented" state per vendor (it's either "Consent" or "No Consent", or entire-cookie-not-set).
- Legal audit trail stored in-cookie. However, the following details are stored for consent logging and audit record: LastUpdated, CmpId, CmpVersion, ConsentScreen, ConsentLanguage, and VendorListVersion.

What is contained in the vendor list JSON?

- A vendor list version
- A list of global purposes for vendors (maps PurposesAllowed bitfield to UI elements)

- A list of vendors with assigned VendorIds, purposes they need consent for, and a link to their GDPR policy page. VendorIds are incrementally-assigned and never reused; deleted vendors are just marked as deleted.

What is the location and format of the vendor (and standard purposes) lists?

The vendor list will be in JSON and hosted at https://vendorlist.consensu.org/vendorlist.json . Previous and the current version are available at https://vendorlist.consensu.org/v-*versionnum*/vendorlist.json , where *versionnum* is a decimal number from 0 to 4095 as indicated in the VendorListVersion cookie field.

Translations to non-English languages of the purposes will be hosted at https://vendorlist.consensu.org/purposes-*language*.json with older versions at https://vendorlist.consensu.org/v-*versionnum*/purposes-*language*.json where *language* is the two-letter lowercase ISO 639-1 language code.

The JSON format is:

```
{
  "vendorListVersion": 0, // will be incremented each change
  "lastUpdated": "2018-05-28T00:00:00Z",
  "purposes": [
    {
      "id": 1,
      "purpose": "Personalised Advertising",
       "description": "delivering tailored advertising based on your preferences or interests across services and devices."
    },
    … more purposes from id=2 up to no higher than id=24
  ],
  "vendors": [
    {
      "id": 1,
      "name": "Vendor Name",
      "purposeIds": [1, 4, 7], // list of data purposes
      "policyUrl": "https://vendorname.com/gdpr.html",
       "deletedDate": "2018-05-28T00:00:00Z", // if present, vendor should be considered deleted after this date/time
    },
```

```
    … more vendors, approximately 2000 for the initial list
  ]
}
```

What is the long-term plan for consent storage?

A third-party cookie isn't a long-term solution to auditable, permanent, user-keyed consent storage, and doesn't work today for browsers that block 3rd-party cookies or mobile apps. CMP's should work towards standardizing a more future-looking server-side consent retrieval mechanism as well,and can use this cookie as "consent caching" for that future implementation.

What is the format of the global vendor consent cookie?

Cookie Directive	**Value(s)**	**Notes**
Name	euconsent	Needs to be finalized
Host	.consensu.org	The DNS resolution for the name *cmp-name*.mgr.consensu.org will be delegated by the standardizing authority (IAB) to each CMP. CMP's will host their code, API's, and CDN at this domain or subdomains.
Path	/	
Max-Age	33696000	This represents thirteen 30-day months. Needs policy validation.
Value	Base64-encoding of the concatenated Cookie Value Fields bits described below	The binary bits should be padded at the end with zeroes to the nearest multiple of 8 bits, packed into a string of bytes, and have websafe-base64 encoding performed.

Vendor Consent Cookie Format

The following fields are stored in big-endian format:

Cookie Value Field Name	Number of bits	Value(s)	Notes
Version	6 bits	“1” for this version	Incremented when cookie format changes
Created	36 bits	Epoch deciseconds when cookie was first created	Deciseconds fits into 36 bits with enough precision to record a user’s consent action timing. Javascript: Math.round((new Date()).getTime()/100)
LastUpdated	36 bits	Epoch deciseconds when cookie was last updated	
CmpId	12 bits	Consent Manager Provider ID that last updated the cookie	A unique ID will be assigned to each Consent Manager Provider
CmpVersion	6 bits	Consent Manager Provider version	Each change to the CMP should receive a new version number, for logging proof of consent
ConsentScreen	6 bits	Screen number in the CMP where consent was given	The screen number is CMP and CmpVersion specific, and is for logging proof of consent
ConsentLanguage	12 bits	Two-letter ISO639-1 language code that CMP asked for consent in	Each letter should be encoded as 6 bits, a=0..z=25 . This will result in the base64-encoded bytes spelling out the language code (in uppercase).
VendorListVersion	12 bits	Version of vendor list used in most recent cookie update.	Vendor list versions will be released periodically. 12 bits allows for 78 years of weekly updates.
PurposesAllowed	24 bits	For each Purpose, one bit: 0=No Consent 1=Consent	Purposes are listed in the global Vendor List. Resultant consent value is the “AND” of the applicable bit(s) from this field and a vendor’s specific consent bit.

			Above fields are multiples of 6 bits to fit into base64-encoded bytes.
MaxVendorId	16 bits	The maximum VendorId for which consent values are given.	VendorIds are numbered 1 to *MaxVendorId*. Allows the cookie value to be interpreted without waiting for the vendor list fetch.
EncodingType	1 bit	0=BitField 1=Range	The consent encoding used. Either a **BitFieldSection** or **RangeSection** follows.
BitFieldSection			Encodes one consent bit per VendorId
BitField	*MaxVendorId* bits	For each Vendor, one bit: 0=No Consent 1=Consent	The consent value for each *VendorId* from 1 to *MaxVendorId*
RangeSection			Encodes a default consent value and any number of single or range override entries
DefaultConsent	1 bit	0=No Consent 1=Consent	Default consent for VendorIds not covered by a **RangeEntry**. VendorIds covered by a **RangeEntry** have a consent value the opposite of *DefaultConsent*.
NumEntries	12 bits	Number of entries to follow	*NumEntries* instances of **RangeEntry** follow.
RangeEntry			A single or range of VendorIds, whose consent value is the opposite of *DefaultConsent*
SingleOrRange	1 bit	0=Single VendorId 1=VendorId range	Whether a single VendorId or a start/end range of VendorIds is given
SingleVendorId	16 bits	A single VendorId	Exclusive with Start/EndVendorId.
StartVendorId	16 bits	The start of an	Exclusive with SingleVendorId. Must be

		inclusive range of VendorIds	paired with a EndVendorId
EndVendorId	16 bits	The end of an inclusive range of VendorIds	Exclusive with SingleVendorId. Must be paried with a StartVendorId.

Example cookie field values for the case:

- all VendorId consents given, except VendorId=9
- for VendorListVersion=8 which has 2011 VendorIds defined
- by Consent Manager Provider Id #7

Field	**Decimal Value**	**Meaning**	**Binary Value**
Version	1	Cookie Format Version #1	000001
Created	15100811449	2017-11-07T18:59:04.9Z	001110000100000101000100000000110010
LastUpdated	15100811449	2017-11-07T18:59:04.9Z	001110000100000101000100000000110010
CmpId	7	The ID assigned to the CMP	000000000111
CmpVersion	1	Consent Manager Provider version for logging	000001
ConsentScreen	3	Screen number in the CMP where consent was given	000011
ConsentLanguage	"en" (e=4, n=13)	Two-letter ISO639-1 language code that CMP asked for consent in	000100 001101
VendorListVersion	8	The vendor list version at	000000001000

		the time this cookie value was set	
MaxVendorId	2011	Number of VendorIds in that vendor list	0000011111011011
EncodingType	1	Range encoding (not bitfield)	1
DefaultConsent	1	Default is “Consent”	1
NumEntries	1	One “range or single” entry	000000000001
SingleOrRange[0]	0	A single VendorId (which is “No Consent”)	0
SingleVendorId[0]	9		00000000000001001
Base64-encoded cookie value			*TBD*

A “cookie workshop” page is available at http://gdpr-demo.labs.quantcast.com/user-examples/cookie-workshop.html to try out other values for the cookie to see its length and encoding.

How is publisher-specific consent stored?

Under this proposal, there are two types of publisher-specific consent: service-specific vendor consent, and a publisher’s purposes consent for its own data use (if needed).

Service-specific vendor consent can be implemented (optionally) by a CMP using the same Vendor Consent Cookie Format (as detailed above) with the Host, Path, and/or Cookie Name referencing a first-party cookie (or service-shared 3rd-party cookie), as customized by the publisher at initialization. The following table indicates which format is used by which cookie:

Cookie Format Type	**Cookie Location**	**Purpose**
Vendor Consent	3rd-party global location	global vendor consent

	(.consensu.org)	
Vendor Consent	1st-party publisher-configured name & location. Could instead be a 3rd-party shared location, for a service-specific cookie that spans publisher sites.	service-specific vendor consent (if configured, overrides global vendor consent)
Publisher Purposes Consent	1st-party publisher-configured name & location (again, could be a shared 3rd-party location as well)	publisher's own data usages consent (not looked at by vendors)

A publisher's purposes consent cookie store the publisher's own consent. The publisher can configure the CMP to request consent for any of the standard list of purposes (which utilizes the StandardPurposesAllowed bitfield and the standard list of purposes from the Vendor List) and/or custom purposes (initialized by the publisher). Differences from the Vendor Consent Cookie Format are:

- "PublisherPurposesVersion" augments "VendorListVersion" and is incremented each time the publisher modifies the requested purposes.
- "NumCustomPurposes" replaces "MaxVendorId" and is reduced to 6 bits.
- "BitField" encoding is always used and is indexed by CustomPurposeId's.
- Host, Path, and Cookie Name are initialized in the CMP by the publisher, and the cookie data is stored as a first-party cookie by the CMP javascript.

Publisher Purposes Consent Cookie Format

Cookie Value Field Name	**Number of bits**	**Value(s)**	**Notes**
Version	6 bits	"1" for this version	Incremented when cookie format changes
Created	36 bits	Epoch deciseconds when cookie was first created	Deciseconds fits into 36 bits with enough precision to record a user's consent action timing. Javascript: Math.round((new Date()).getTime()/100)

LastUpdated	36 bits	Epoch deciseconds when cookie was last updated	
CmpId	12 bits	Consent Manager Provider ID that last updated the cookie	A unique ID will be assigned to each Consent Manager Provider
VendorListVersion	12 bits	Version of vendor list used in most recent cookie update.	The vendor list provides the list of standard purposes, so the version of the vendor list used is recorded.
PublisherPurposesVersion	12 bits	Version of publisher purposes list used in most recent cookie update.	Publishers initialize the CMP with this version, and should increment it for any changes to the list of requested purposes so that consent can be re-obtained.
StandardPurposesAllowed	24 bits	For each standard purpose, one bit: 0=No Consent 1=Consent	Standard purposes are listed in the global Vendor List.
NumberCustomPurposes	6 bits	The number of custom purposes consent bits.	CustomPurposeIds are numbered 1 to *NumberCustomPurposes*. Custom purposes will be initialized in the CMP by the publisher.
			Above fields are multiples of 6 bits to fit into base64-encoded bytes.
CustomPurposesBitField	*NumberCustomPurposes* bits	For each custom purpose, one bit: 0=No Consent 1=Consent	The consent value for each *CustomPurposeId* from 1 to *NumberCustomPurposes*