Online Tracking and User Protection Mechanisms

A study on the technical implementation of user consent and Do Not Track (DNT)

DECEMBER 2017

About ENISA

The European Union Agency for Network and Information Security (ENISA) is a centre of network and information security expertise for the EU, its member states, the private sector and EU citizens. ENISA works with these groups to develop advice and recommendations on good practice in information security. It assists member states in implementing relevant EU legislation and works to improve the resilience of Europe's critical information infrastructure and networks. ENISA seeks to enhance existing expertise in member states by supporting the development of cross-border communities committed to improving network and information security throughout the EU. More information about ENISA and its work can be found at www.enisa.europa.eu.

Contact

For media enquires about this paper, please use press@enisa.europa.eu.

Table of Contents

Executive Summary

Online tracking techniques are increasingly used today on the internet and can he highly beneficial for service providers who can create detailed user profiles and accordingly advance their business, e.g. with the use of targeted behavioral advertising. Although users may also benefit from online tracking, for example through enhanced personalized services and functionalities, in most cases they are not aware of the full extent of tracking while they are surfing the internet, doing business on the web or communicating with their social networks. This can have serious adverse effects on their privacy.

On 10 January 2017, the European Commission adopted a proposal for a Regulation on privacy and electronic communications (EC Proposal), which will replace the current ePrivacy Directive 2002/58/EC and align privacy rules on electronic communications with the General Data Protection Regulation (GDPR). The EC proposal broadly recognizes the serious threats that online tracking and monitoring techniques pose to users of electronic communication services, for example by gaining access to information, storing hidden information and tracing users' activities. To this end, the EC Proposal stipulates that the use of processing and storage capabilities of users' terminal equipment and the collection of information from users' terminal equipment, including about its software and hardware, is conditioned on users' consent. It also provides for the possibility to express consent by using the appropriate technical settings of a software application providing access to the internet and mandates a new obligation to electronic communication software providers on privacy setting options.

Against this background and in the context of the ePrivacy legislation ongoing discussions, ENISA decided to provide a study on online tracking and relevant user protection mechanisms, paying particular attention to user consent, privacy settings and the implementation of the Do-Not-Track (DNT) standard.

The main recommendations of the report are drawn to the different stakeholder's groups (service providers, user agents, policy makers, regulators and others) concerning an enhanced implementation of user protection mechanisms against tracking.

A) Service providers

It is recommended that service providers:

- Provide clear information on their identity and how they use terminal storage and for what purpose in standardised machine-readable way, so that clients can act on it to protect privacy and be able to present it to the user in intelligible ways while not diminishing the user experience.
- Ask all their third parties to provide the purpose for which they will process users' data and then provide the users with that information when asking them to consent.
- Consider grouping third-parties by the purpose for which they are embedded on the service provider's website (e.g. advertising, social network, analytics, showing videos) and asking for consent for all the purposes for which a third party is embedded before this third party is 'called' on the web page.
- Restrict third-parties from loading into browsers by default and manage the data storage these parties use by communicating relevant information to user agents in a machine-readable, standardized, language independent and verifiable form.

B) User agents

It is recommended that user agents:

- Make default settings as privacy protective as possible.
- Implement features so that purpose-designed user consent signals can be recorded at a granular level, on at least a domain specific basis.
- Consider differentiating between domains the user has given consent versus domains he or she has not given consent (and therefore being able to apply different levels of protection to these domains).
- Consider options for reflecting an active choice from the user, which would constitute a valid consent mechanism.

C) Policy makers, regulators and other stakeholders

It is recommended that:

- Policy makers and regulators provide further guidance on the technical implementation of valid consent and privacy defaults, in co-operation with the industry of user agents and associations of service providers.
- The research community continues work in tracking prevention mechanisms, including the technical implementation and signalling of valid user consent.
- The European Commission and EU institutions in the field of privacy and security support relevant initiatives and research projects in this area (e.g. in the framework of H2020).
- The European Commission and EU institutions in the area of privacy and security promote the formulation of workshops and working groups with all relevant stakeholders in the field.
- EC standardisation bodies engage in the creation of relevant online data protection standards, in particular in the fields of signalling of user consent, icons symbols and functions, taking into account the underlying ePrivacy legal framework. Particularly, based on the new ePrivacy Regulation covering other forms of communications such as the over-the-top (OTT) technologies, standards will have to be expanded to define and standardize consent across a broad spectrum of media platforms.

ENISA will aim at further contributing in the discussions for online tracking prevention by supporting relevant projects and activities and co-operating with the main stakeholders in the field.

1. Introduction

1.1 Context

In 2002, the European Union launched the Directive 2002/58/EC [1] concerning the processing of personal data and the protection of privacy in the electronic communications sector (ePrivacy Directive), which complemented the Data Protection Directive 95/46/EC [2]. The ePrivacy Directive has been amended by Directive 2009/136 [3], designed to increase consumer protection and cover the protection of data and privacy on the web.

The ePrivacy Directive imposes the obligation to obtain the user's consent before storing any information or gaining access to any information already stored in the user's terminal equipment, unless the processing is necessary for carrying out a transmission of an electronic communication or for providing an information society service requested by the user (art. 5(3) ePrivacy Directive). Consent follows the definition of the Data Protection Directive, which as of 25 May 2018 will be repealed and replaced by the General Data Protection Regulation (EU) 679/2016 ('GDPR') [4] . Under GDPR consent is defined as '*any freely given, specific, informed and unambiguous indication of the data subject's wishes by which he or she, by a statement or a clear affirmative action, signifies agreement to the processing of personal data relating to him or her*'.

On 10 January 2017, the European Commission adopted a proposal for a Regulation on privacy and electronic communications[1] (EC Proposal). The proposed Regulation will replace the ePrivacy Directive and align privacy rules on electronic communications with the GDPR, taking also into account the technological advancements in the field , as well as the seek to improve on the requirement for consent of the ePrivacy Directive [5]. The EC proposal broadly recognizes the serious threats that online tracking and monitoring techniques pose to users of electronic communication services, for example by gaining access to information, storing hidden information and tracing users' activities.

To this end, the EC Proposal follows the opt-in regime of the ePrivacy Directive by stipulating that the use of processing and storage capabilities of users' terminal equipment and the collection of information from users' terminal equipment, including about its software and hardware, is conditioned on users' consent (art. 8(1) of the proposal)[2]. Recognizing the role of web browsers (and other similar applications) as mediators between users and information society service providers, the EC proposal also states that, where technically possible and feasible, consent may be expressed by using the appropriate technical settings of a software application providing access to the internet (art. 9(2) of the EC proposal). Moreover, Article 10 of the proposal poses the obligation to electronic communication software providers to offer the option to prevent third-party storage or access to information stored in the users' terminal equipment, as well as to inform users about the privacy setting options. It is worth mentioning that the EC Proposal is subject to change based on the remaining review procedure till its final adoption. At the time of writing of

[1] The ePrivacy Directive is currently under review in order to be modernised and aligned with GDPR. See latest information in https://ec.europa.eu/digital-single-market/en/news/summary-report-public-consultation-evaluation-and-review-eprivacy-directive

[2] As in ePrivacy Directive, in the EC proposal consent is not mandatory when the processing is necessary for carrying out the transmission of communication or to provide an information service requested by the user. In addition, consent is not mandatory for web audience measuring if such measuring is performed by the provider of an information society service requested by the user.

the current report, the European Parliament plenary of the 26th of October 2017, voted to approve an amended version of the e-Privacy Regulation[3,4].

Taking into consideration the aforementioned legal framework and relevant proposals, the technical implementation of consent mechanisms is a key element for preserving user privacy online. At the same time, browsers and/or other software linking a user to an online service should provide their products and services with privacy settings, allowing the user to prevent information from his or her terminal equipment being unexpectedly accessed or stored. In the last years, some extensions or add-ons to browsers have been developed to increase users' ability for monitoring and control, for example by being able to decide whether they want identifiers stored in their devices (e.g. regular cookies, flash cookies) or information collected about their device (e.g. fingerprinting). These tools enable users to control information flowing in and out of their devices. Given that often tracking mechanisms such as cookies are used to track individuals across web sites and deliver advertisement, the same functionalities also enable users to decide whether they accept to be tracked.

Moreover, in the last years there have been many active discussions about the Do-Not-Track (DNT) initiative [6] and many browsers have implemented Do-Not-Track options. Do-Not-Track initiatives aim at creating ways to enable users to signal whether they want to be tracked or not and set forth ways for companies to respond to such signals. Do-Not-Track relies on the obligation for the information society service (e.g. web site) to honor the request of users not to be tracked. In fact, DNT is the name of a request field in the HTTP protocol header and is used to express a choice not to be tracked when browsing from site to site on the web.

Against this background and in the context of the ePrivacy legislation discussions, ENISA decided to provide a study on online tracking and relevant user protection mechanisms, paying particular attention to user consent, privacy settings and the implementation of the DNT standard. The present report presents the results of this study.

1.2 Scope and objectives

Based on the aforementioned description, the objectives of the present report are as follows:

- To provide an overview of modern online tracking mechanisms.
- To provide an overview of existing online consent mechanisms (mainly on consent for cookies).
- To discuss the possibility to obtain consent via browser/application settings, focusing especially on conveyance/signaling of consent for tracking and the technical implementation of DNT.
- To address tracking prevention mechanisms via browser/application settings.

It should be noted that, although the focus of the report is mainly on browser applications, we also discuss other types of applications used for electronic communication (hereby all referred as user agents), and also Operating Systems (OS) which in some cases will be the relevant software for obtaining user consent. Clearly this report aims at providing an organizational and technical review of DNT and it is not meant to be used as legal guidance or targeted policy initiative that makes part of a formal legislative initiative in any way whatsoever.

[3] https://privacy-news.net/uploads/1d5b2400-b4fc-11e7-bd94-7ff24f8b617d.pdf
[4] http://www.europarl.europa.eu/thinktank/en/document.html?reference=EPRS_BRI%282017%29608661

The target audience of the report are developers of browsers and other applications that can be used for electronic communication, policy makers and regulators, as well as research and standardization bodies in the field of online privacy and data protection.

1.3 Methodology

The study was conducted by a dedicated ENISA expert group. It was mainly based on desk research, including literature reviews, regulatory documents, standards and technical specifications, as well as review of existing consent tools and technical implementation of privacy settings in software applications. The preliminary output of the study was shared with external stakeholders (especially regulators and researchers in the field). The feedback from these stakeholders and discussion with other partners was then used to finalize the results of the study.

1.4 Structure

The structure of the document is as follows:

- Chapter 2 provides an overview of the online tracking mechanisms. Specifically, the notion of tracking is introduced and defined and browser based tracking mechanisms are described, including third-party and first-party tracking. Moreover, the legal framework around online tracking based the EC proposal for an ePrivacy Regulation, as well as the General Data Protection Regulation (GDPR) is presented.
- Chapter 3 provides an overview of existing consent mechanisms and the privacy settings the user can configure in the user agent. Moreover, tracking prevention via user agent settings is discussed, backed up by some relevant examples (see also Annex B).
- Chapter 4 presents how the DNT initiative is used for signalling user consent for tracking implementation and provides a quick overview of the current user agent implementations (see also Annex A).
- Chapter 5 draws a number of final observations and conclusions.

The report complements past ENISA's work in the field of privacy and data protection, especially with regard to the technical implementation of GDPR and ePrivacy Directive[5].

[5] For more information, see: https://www.enisa.europa.eu/topics/data-protection

2. Online Tracking Mechanisms

While the discussion surrounding web tracking tends to focus on HTTP cookies, there is an extensive list of stateful and stateless technologies that can be used to collate web activities of online users. Moreover, many information service providers have their websites developed or hosted for them by other companies and do not have a thorough understanding of how their website visitors' data is handled. It has also become common practice for websites to embed "third-party" content, i.e. content that is hosted on the servers of other companies who may be tracking a website's visitors, often without the ultimately responsible company knowing about it (in some cases, third-party content may be dynamically introduced via embedded content of other third-parties). Similarly, it is common for websites to incorporate active content, such as JavaScript snippets or libraries, supplied by third-parties which may then introduce functionality to track users or collect personal data, and again, sometimes without the website company's knowledge.

In recent years, tracking technologies have been extensively studied and measured and researchers have found that third parties embedded in websites use numerous technologies, such as third-party cookies, HTML5 local storage, browser cache and device fingerprinting that allow the third party to recognize users across websites [7] and build browsing history profiles. Researchers found that more than 90% of Alexa top 500 websites [8] contain third party web tracking content, while some sites include as much as 34 distinct third party contents. Moreover, a sweep of web sites conducted under the Article 29 Working Party in 2014 [9] showed that 70% of the cookies recorded were third-party cookies. It was also shown that more than half of the third-party cookies were set by just 25 third-party domains.

2.1 The notion of tracking

2.1.1 Understanding tracking

Tracking may have several meanings, but for the purpose of this study one of the most valid ones is: '*following the trail or movements of (someone or something), typically in order to find them or note their course*[6]". There are two important observations with regard to this definition. First, in the online world, 'finding' or 'noting someone's course', does not necessarily mean that the individual needs to be directly identified (e.g. by name). On the contrary, it will usually be just enough to be able to distinguish him/her from any other individual, e.g. through a specific identifier that can be uniquely attributed to him/her. Secondly, as users are connecting to web and mobile services via certain devices (e.g. smartphones, laptops, wearables, IoT devices) and/or applications (e.g. browsers, mobile apps), tracking can be performed through these devices and/or applications, as long as they can be uniquely identified. In a converged online environment where users are utilising multiple devices (e.g. smartphone and laptop) for access to the same services, tracking efficiency is increased when all these devices (through their identifiers) can be linked to the same user.

When combined with data analytics, online tracking is a very powerful technique for collecting information about a user over time. Depending on the type and extend of tracking, this information can vary from detecting a user's interests upon visiting a specific web page (e.g. which pages he/she prefers) to a detailed analysis of the user's private life, including location data, personal interests and social relations, as well as sensitive data about his/her health, political beliefs or sexual preferences (e.g. through the combination of information derived from the user's visits to multiple web pages and/or online services). In that sense,

[6] http://www.oxforddictionaries.com/definition/english/track

tracking can be a powerful tool for profiling, which is defined in GDPR as: "*any form of automated processing of personal data consisting of the use of personal data to evaluate certain personal aspects relating to a natural person, in particular to analyse or predict aspects concerning that natural person's performance at work, economic situation, health, personal preferences, interests, reliability, behaviour, location or movements*".

Online tracking can be highly beneficial for service providers who can create detailed profiles of users and accordingly advance their business, e.g. with the use of targeted behavioral advertising[7]. In fact, some companies only rely on the income of targeted advertisement: they create websites offering free content to the users that are then exposed to advertisement (mostly targeted advertisement) during their consultation of the content. Some companies have developed tracking technologies allowing them to collect data from multiple sources and then reselling it to other companies, mainly for marketing purposes.

Although users may also benefit from online tracking, for example through enhanced personalized services and functionalities, in most cases they are not aware of the full extent of tracking while they are surfing the internet, doing business on the web or communicate with their social networks. This can have serious adverse effects in their privacy[8]. The "average user" should expect a clear reasoning from the service providers, in particular regarding the "purpose" of data collection. Moreover, tracking should be visible for the user to enable an informed decision making (and consent giving) [9,10].

2.1.2 Technical definition

The term **'**tracking' has been used in technical papers to describe the collection of data about someone's online activity. For example, this has been the underlying concept behind the development of the W3C's Do Not Track recommendation [6], primarily influenced by US based entities, which includes the following definition:

Tracking is the collection of data regarding a particular user's activity across multiple distinct contexts and the retention, use, or sharing of data derived from that activity outside the context in which it occurred. A context is a set of resources that are controlled by the same party or jointly controlled by a set of parties.

This definition does not fit exactly into the legal definitions in European law, however, although the concepts of online privacy and the protection of personal data basically cover the same ground. In particular, the W3C definition mainly focuses on the collection by third-parties ("outside the context in which it occurred"), whereas in Europe processing of personal data by any party is controlled by law, as well as access to the "private sphere" of the user's terminal equipment. It has always been recognised, both under the ePrivacy Directive and the EC proposal for an ePrivacy Regulation, that some access to or storage in user's terminal equipment is necessary in the normal course of online activity, and that user consent can be assumed if access is "strictly necessary to fulfil a purpose requested by the user".

---

[7] http://resources.infosecinstitute.com/means-and-methods-of-web-tracking-its-effects-on-privacy-and-ways-to-avoid-getting-tracked/#gref

[8] A clear demonstration of this is the case of company Target who was able to identify a pregnant teenage girl only from her conduction of online purchases, see: https://www.forbes.com/sites/kashmirhill/2012/02/16/how-target-figured-out-a-teen-girl-was-pregnant-before-her-father-did/#3b6a90b86668

[9] Refer to Asunción Esteve: The business of personal data: Google, Facebook, and privacy issues in the EU and the USA, International Data Privacy Law, Volume 7, Issue 1, 1 February 2017, Pages 36–47, https://doi.org/10.1093/idpl/ipw026

[10] Refer to Mary Madden : Most Would Like to Do More to Protect their Personal Information Online http://www.pewinternet.org/2014/11/12/most-would-like-to-do-more-to-protect-their-personal-information-online/

In practice, users assume some sort of persistence during a browsing session. Cookies are the method of choice to achieve a short-term persistence. Thus, unique user identifiers held in cookies are assumed to be strictly necessary if they are solely used to recognise the user agent for purely technical reasons, such as for load balancing for example. Typically those identifiers will have a very short lifespan, and are therefore discarded after a period no longer than a few hours [10].

Taking into account the aforementioned considerations, in the European context, a better technical definition of the term 'tracking' could be as follows:

Tracking is the collection or further processing of data held within or transmitted from a person's terminal equipment or derived from their online activity, beyond which is solely required to support the underlying communications channel, strictly necessary to fulfil a purpose that the person has specifically requested, or for other specific legally exempted purposes which have the required safeguards in place.

In the next sections we present in more detail different online tracking techniques.

2.2 Browser based tracking mechanisms

Broadly speaking, all online tracking techniques have from a conceptual perspective one common element: the attribution of a unique identifier to each connected device and/or application, which is used as the basis for the users' tracking. In some cases, the identifiers are being 'dropped' in the user's device by the tracker and followed up further on (e.g. with the use of cookies). In others, the identifiers are created from information that is sent/broadcasted by the user's device or extracted from the user's device upon specific requests of the tracker (e.g. device fingerprinting). Still it is important to note that in all cases the generation and tracking of the identifiers is facilitated by the standard way the web and mobile protocols operate for the delivery of services. In the following we provide an overview of main concepts and most common tracking techniques used today.

2.2.1 Third-party versus first-party tracking

When discussing the current state of tracking and tracking protection on the web, a key concept is whether the tracking is for or by a "third-party" or a "first-party". While the information society service providers must always ensure the privacy of visitors of their websites, entities from whom the user has not requested a service, but whose servers are accessed during the assembly of the website's content (and whose existence they are probably unaware of) perhaps present a bigger threat to privacy.

In the web context, a first-party [11] is the entity that controls the web domain a user has explicitly visited, sometime called the "top-level context" or the server addressed by the URL displayed in the browsers "location bar" when a site is visited. A third-party is any other entity controlling resources accessed while the page is being rendered or displayed, such as those for content embedded in the page, or external content accessed by script running in the page. Typical examples are widgets delivering advertising or the analytics required for targeted advertising, or "retargeting" where visitors to a website can be shown related online advertising when they visit other sites.

For example, when a user visits a news site, the browser may make additional requests to a social networking site; subsequently the latter site associates the news site visit with the user's profile data it holds.. Figure 1 shows an example of such tracking where LinkedIn.com is the third party.

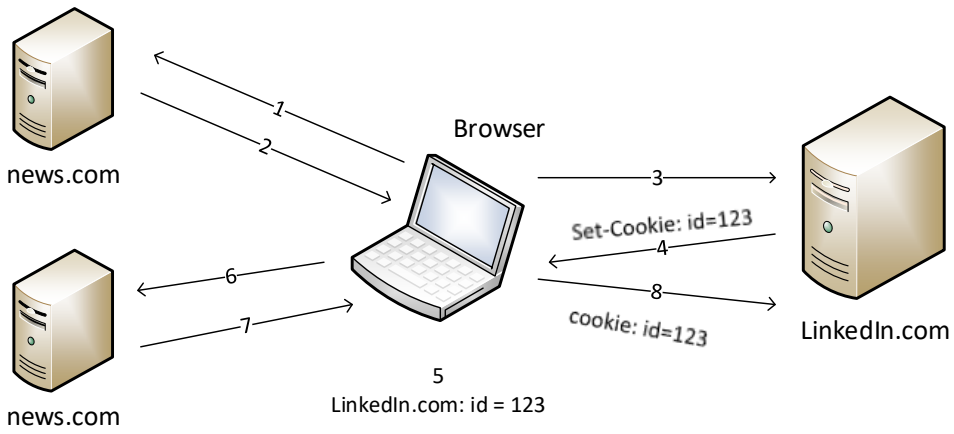


Figure 1: Third Party Tracking [12]

Web pages are often built using content from third-party elements, and some of these actively collect the web activity of users who visit the containing web page. To do this, visitors need to be recognized i.e. identified across multiple interactions with often many different websites. This is usually done via the use of "third-party" persistent cookies containing Unique User Identifiers (UUIDs), stored in the browser in a database keyed to the associated third-party domain. It is increasingly also done using "first-party" cookies keyed to the domain origin of the containing web page, but correlated with third parties cookies on other domains, via so called "cookie synching" techniques (see also 2.2.3.1 on cookies). Other techniques can also be used, such as "browser fingerprinting" (see also 2.2.3.2).

In the normal course of assembling content for a web page, possibly involving interactions with different servers, active script is only given access to the data associated with its own domain. This is enforced by browsers via the so-called Same Origin Policy, an important concept in the web application security model. An *origin* is defined as a combination of URI scheme, host name, and port number. This means that if a sub-resource is loaded from https://example.com/iframe.htm while the page http://mainsite.de/home.htm is being accessed, any script loaded by it is not allowed to access any data within the http://mainsite.de origin, and vice-versa, unless both parties purposely communicate between themselves.

Both servers (service providers) and user agents (browsers or users' devices) have a part to play in controlling tracking, but code in servers has the advantage that information is obtainable on whether, and how, tracking is employed, and for what purpose. Even so, they often have difficulty controlling what their embedded third-party resources are doing because the Same Origin Policy restrictions limits access to them. On the other hand, user agents may escape these restrictions, but usually do not have access to information about purpose necessary for fine grained control over tracking.

2.2.2 General purpose tracking prevention – Content blocking

Content blocking [13], [11] is a technique whereby browsers or browser extensions refuse to load some third-party content, depending on rules-based decision making. The decision can be based on a "blacklist" identifying content to be blocked or if non-consensual tracking or other unwanted behaviour has been dynamically detected.

While this can be highly effective, completely stopping any tracking content being loaded into browsers can be a very blunt instrument. Often it is necessary to block all the content from a domain so even benign non-tracking content that the user may want ends up being blocked, impairing their experience of the web. Without discoverable information about the purpose of embedded content, it is hard to recognise

tracking behaviour without many "false positives". Moreover, blacklists can be "gamed" by commercial entities to include benign non-tracking or privacy enabling content from their competitors, while domains that do track are removed if the companies responsible for them provide payment.

To counteract this, servers (of service providers) should collaborate with user clients (browsers) by declaring information on how and why they use storage (information held within users' terminal equipment) in a standardised machine-readable form. If servers for sub-resources do not supply this information themselves, then the server embedding them should make this available. Browsers can then use this information to implement tracking protection in a more straightforward manner subsequently improving the functionality of web sites.

2.2.3 Specific Tracking Techniques

2.2.3.1 Cookies

HTTP cookies have been part of the web almost since its beginning. They are the main way that individual HTTP transactions can be associated over time with an originating user agent, which is necessary because each transaction between a user agent and a server is essentially "stateless", i.e. each transaction is handled as a single unit without its surrounding context.

If cookies have been stored for a certain domain origin they will be included in the headers of every request for that domain sent to its server. The server can then associate different transactions from the same user agent with each other, using values contained in them to reconstitute the "session state" across a series of transactions.

Cookies that are stored when requesting a sub-resource, i.e. when an embedded element in a webpage has a different origin to the page the user has visited, are called "third-party" cookies, in contrast to "first-party" cookies which are stored in the visited website's own domain. Cookies in one domain are normally inaccessible from script in another, a consequence of the "Same Origin Policy".

- Privacy risk

While cookies are essential to store data needed to manage a user's "session" with a web server (of a service provider) over a few hours, they can also be used to uniquely identify the user's client (browser) over longer time periods of months, years, or decades. This may occur in order to recognize users that have explicitly logged-in to a service, or to build a record to support an online commerce application, but the purpose might be also to secretly track users by collecting their web activity building detailed profiles of them.

It is sometimes thought that, because of the Same Origin Policy, first-party cookies cannot be used to track people across the web because they are only accessible by the server of the first-party domain, but this is not true if the parties collaborate in some way.

It has become increasingly common for websites to host scripts which dynamically create embedded resources that address third party servers, via a URL (Universal Resource Locator) which includes data derived from a first-party cookie. The resulting unique identifier, originally derived from the first-party cookie, is then passed to the other server, which can then track the user on other sites which includes similar script. These unique identifiers can be correlated with each other, i.e. so they can be used to identify the same user, via a common profiling record for that user in an external database, by so-called "cookie synching". This uses some intermediary identifier, such as the source IP address of the user agent, or a short duration or session cookie in a third-party sub-resource, to link the first-party identifiers. This tracking method avoids the default third-party cookie blocking implemented by some browsers such as

Apple's Safari because the persistent identifier is held in a first-party cookie, which is never blocked. Even when Safari's recently improved its Intelligent Tracking Prevention [14] implementation, cookie synching using a third-party cookie still works, as long as the next site in the chain of linked sites is visited within 24 hours.

Some companies provide resources designed to be accessed in both a "first-party" and "third-party" context. This blurs the distinction between third-party and first-party storage because a UUID cookie can be placed when a resource is visited as a top-level page, and later the same data item can be accessed by the company when the resource is accessed in a third-party context on an entirely different page. To prevent this, companies can declare cookies that should be used only in a first-party context using the SameSite attribute [15]. One example is Facebook where their domain facebook.com is visited by millions of people who then receive UUID cookies which are sent back whenever other pages are visited that contain other Facebook content using the same domain facebook.com, e.g. their "Like" buttons or similar [15].

- Mitigation

Most user agents include functionalities that allow the user to examine cookies associated with a domain or a visited web page, showing expiry duration, their contents and what host domain they were associated with. This information however cannot explain what their intended use is, and the somewhat technical information that is available is often inappropriate for the user to decide whether to block or delete a cookie.

Such limitations in information reduce browsers' ability to protect users' privacy. Settings are usually available to block all cookies but this appears as too blunt a measure because most web sites require cookies to function, and as a result cookie blocking, especially for first-party domains, is hardly ever a user's preferred choice.

Some browsers offer more user control over cookies, for instance generally blocking their storage in a third-party context, but this sometimes inhibits useful functionality such as federated or cross-origin log-in. This has been improved to some extent by Safari's default "Allow [cookies] from Websites I Visit" setting[11] or Firefox's "Accept third-party cookies from visited" setting[12], which block cookie storage from embedded sub-resources, unless sites using the same domains have been explicitly visited by the user. Still, this can allow some popular sites to track users without their knowledge on other sites by getting these sites to include a sub-resource with the same domain as the popular first-party. It is also possible for third-parties to supply JavaScript to first-parties which intercepts internal links. When users navigate to these links the request is first sent to the third party which places a cookie before redirecting back to the internal link. Because this is a first-party request the browsers is "fooled" into accepting it, effectively bypassing any third-party cookie blocking.

Websites (of service providers) have an even harder job than user agents in controlling cookies used by sub-resources that may be embedded on pages. Although they are ultimately responsible for the placement of third-party hosted elements, they have no control over what further sub-resources these elements in turn load themselves.

[11] Safari for macOS Sierra: Manage cookies and website data using Safari, https://support.apple.com/kb/ph21411?locale=en_US

[12] Disable third-party cookies in Firefox to stop some types of tracking by advertisers, https://support.mozilla.org/en-US/kb/disable-third-party-cookies

The Same Origin Policy makes it hard for websites to manage any cookies used by any embedded sub-resource. It is possible to remove first-party cookies that may have been placed by external or third-party script inserted into the webpage, but third-party cookies cannot be accessed at all.

Usually the only control that websites have over third-party cookies is to block the content that inserts them, either by ensuring they are not inserted in the first place or by using Tag Manager (see also 3.3.2.3).

2.2.3.2 Fingerprinting

Fingerprinting [17] is the tracking of a user by deriving an identifier from information inherently, or independently, stored in or related to their terminal equipment or user agent (e.g. browser). This identifier can be re-calculated for every interaction with a webpage or third-party sub-resource enabling the user's browser to be recognized in subsequent visits. Common techniques are given in the table below.

Table 1: Common Fingerprinting techniques

FINGERPRINTING TECHNIQUES	DESCRIPTION
Canvas fingerprinting	Detecting minor difference in display hardware by reading back rendered text from a storage area mapped to the display. This technique is capable of only generating a "limited entropy" identifier, i.e. one of no more than about 12 bits in length, but it can nevertheless single-out the user when combined with other information such as the source IP address or the user agent header. The process of reading back the mapped image necessarily involves another interaction between the client and server, so introduces delay because of the extra "round-trip". This also means that a cookie must be used to associate the separate transactions, which is often unavailable to third-party sub-resources, limiting the utility of fingerprinting to them.
Font/plug-in detection	The various fonts and plug-ins supported by a browser can be detected and used to generate a unique signature, which the server can use to recognize the user in across separate interactions. This method suffers from the same extra "round-trip" delays and need for third-party cookies (if implemented by a sub-resource).
MediaStream	A unique stream identifier is generated by the Media Capture and Streams API, which can be reported back and used as a high-entropy identifier. This method suffers from the same extra "round-trip" delays and need for third-party cookies (if implemented by a sub-resource). Nevertheless, the resulting identifier is unique and capable of tracking the user over an extensive period.
WebRTC	The WebRTC (Web Real-Time Communication) API enables a collection of communications protocols to enable real-time communications such as required for video conferencing and screen sharing. It can be used to determine the locally administered IP address of the browser, i.e. the address behind any firewall or Network Translation (NAT) device. This can be used to generate a specific identifier for tracking purposes, even by a third-party sub-

	resource, usually without the user or the first-party controller even being aware of it. Security is also threatened because an external "bad actor" could use information about internal IP addresses to implement further attacks.
IP address/User agent header	The source IP address can be used to identify the user, especially if combined with other information such as the contents of the user agent header. Although many domestic routers implement IPv4 Network Address Translation, the addresses remain constant for long enough to support tracking over a period of days or weeks. When the replacement standard IPv6 becomes widespread there will be more bits to base a unique identifier on (128 versus 32 for IPv4), and Network Address Translation may be less required enabling IP addresses to be unique identifiers.
Client-Hints	Browsers can be instructed to send some identifying information in request packets to origin servers. This is a recently introduced API so has not yet been detected being used for fingerprinting, but the capability is there and should be noted in future.

- Privacy Risk

Although some of these techniques individually produce non-unique medium-entropy (short bit length) identifiers, they can be combined with others to create a more unique high-entropy identifier [18]. If embedded sub-resources implement these techniques to derive a unique, or almost unique, identifier they can then track users across all the sites the sub-resource is embedded on, without needing persistent cookies. This use of fingerprinting by sub-resources if often called "drive-by" identification or tracking, because it is done "invisibly" and the perpetrator often cannot be detected.

- Mitigation

It is very difficult for browsers to identify when some fingerprinting techniques are being used, and there is little that can be done at the individual browser level to mitigate them. Attempting to inhibit fingerprinting by restricting the information available to applications can often reduce necessary functionality of some web applications.

It is possible to reduce the threat of some forms of fingerprinting when cookies are blocked in a third-party context. When fingerprinting requires a further web request, for example when an identifier is calculated by downloaded script evaluating the fonts a browser has installed, which is then communicated back to the originating server, the two requests must be recognized as coming from the same client. This is often done by using a third-party cookie, which can have a very short duration, so having the browser block cookies in a third-party context can help. Nevertheless, servers can match multiple requests over short periods without using cookies by using source IP addresses, so this only reduces the threat to a minor extent.

The privacy risks associated with fingerprinting could be possibly be addressed at the standardisation stage, when new APIs are developed and discussed.

Content blocking is also an effective solution, only allowing domains that are known not to fingerprint to be loaded.

When the network layer protocol IPv6 becomes more widespread using IPv6 source addresses as unique identifiers will become more common. This threat can be mitigated by insisting that software providers and ISPs implement the privacy extensions for "stateless address autoconfiguration" defined in RFC4941 [19], and that ISPs are barred from supplying users with unique address prefixes and that the time-out period for addresses is less than 24 hours.

2.2.3.3 Web Storage

JavaScript can store name/value pair information directly into domain specific storage standard JavaScript web APIs. This can be either session related, “sessionStorage”, where it is not retained when the browser is reloaded, or permanently in “localStorage” [13]. Indexed transactional data can also be stored in browsers using the IndexedDB API [14]. This is not as efficient a tracking method as storing an identifier in a cookie because JavaScript must be used to access it, so reporting it back to a server takes at least another “round-trip”.

In many ways web storage can be perceived as a more privacy-friendly technique because it provides a way for personal data to be held locally, i.e. not on server hosted databases, and therefore more controllable by the user. The storage is only available to JavaScript and, unlike cookies, not automatically sent to servers in every request.

- Privacy Risk

The biggest risk from this form of tracking is its use to store copies of cookies so that they can be secretly regenerated even if the user has purposely deleted them. IndexedDB databases are especially problematic because there is no standardised ability in the IndexedDB API to enumerate database names, so it is not possible to determine when data is stored there by script supplied by third-parties. There is a new API[15] in development that can clear all site data for a particular origin that will be useful here.

- Mitigation

The use of this technique by third-party sub-resources is also often subject to the same restrictions as on third-party cookies, i.e. when sub-resources are blocked from storing cookies they also cannot access domains specific storage. It is possible for script to examine the contents of web storage with the sessionStorage and localStorage APIs, though not the IndexedDB API (so the latter may have to be detected in other ways).

2.2.3.4 Cache storage

The contents of pages and other information are cached within the browser (client-side cache) to minimise latency and redundant network activity.

- Privacy Risk

Although the duration of information stored is indeterminate, it can be used to hold an identifier for at least some period. The usual technique is to associate a web page with a unique identifier as the value of a “ETag” response header, which will then be returned in subsequent requests to the same page in the “If-None-Match” header. While this can single-out or efficiently track users, and does not need extra "round-trips", the identifiers are ephemeral as browsers periodically and randomly purge the client cache, so the technique is not as reliable as some other methods such as fingerprinting or persistent cookies.

[13] HTML5 Web Storage, https://www.w3schools.com/HTML/html5_webstorage.asp
[14] Indexed Database API 2.0, W3C Candidate Recommendation, 10 August 2017, https://www.w3.org/TR/IndexedDB
[15] Clear Site Data, W3C Working Draft, 20 July 2016, https://www.w3.org/TR/clear-site-data/

- Mitigation

It is not easy to detect or prevent cache based tracking, other than manually scanning for probable high-entropy values in ETag headers. Content blocking is probably the only alternative.

2.2.3.5 Flash cookies and local shared objects

Browser plugins, notably the video subsystem Flash Player, support a form of local storage called Local Shared Objects (LDO), which can store tracking identifiers in similar way to cookies; they are called "flash cookies". This is available to any resource embedding the object, and can be shared very easily.

- Privacy Risk

LDOs are accessible from any browsing context, so escape the checks and balances of the Same Origin Policy. They are also invisible to the containing browser, which is therefore not able to examine their contents and explain to the user what data is there or how long it persists. Even when the user requests that cookies and browser storage items are deleted, data in LDOs are retained. For this reason, flash cookies are often used to store copies of browser cookies or other online identifiers so that they can be secretly regenerated after a user has deleted them.

- Mitigation

The use of browser plugins is no longer encouraged, and there are now inbuilt capabilities in browsers to play video without them. Flash cookies are therefore becoming less common.

2.3 Legal framework

In this section we discuss the legal framework around online tracking based the EC proposal for an ePrivacy Regulation, as well as the General Data Protection Regulation (GDPR) (see also section 1.1.).

2.3.1 Proposal for ePrivacy Regulation

As in the case of the ePrivacy Directive, the proposed ePrivacy Regulation does not explicitly mention tracking but points to it, especially by regulating the protection of information stored and related to end-user's terminal equipment. In particular, the relevant provisions of the EC proposal are as follows:

- Terminal Equipment (article 8)

User consent is the one of the legal grounds (article 8(1)(b)) for enabling the use of storage and processing capabilities of the user's terminal equipment and the collection of information from the terminal equipment, which as described earlier is key to most modern tracking techniques today.

In addition to consent, other legal grounds can be invoked, in particular when processing is necessary for the sole purpose of carrying out the transmission of an electronic communication over an electronic communications network (article 8(1)(a)) or for providing an information society service requested by the end-user (article 8(1)(c)) or for web audience measuring (article 8(1)(d)). The latter legal ground is a novelty of the EC proposal and aims to create an exception from consent for first party audience measurement[16].

- Consent (article 9)

With regard to the notion of consent, the EC proposal refers to the GDPR definition of consent and mandates the possibility to obtain consent using the technical settings of user agents.

[16] See also relevant discussion in Article 29 Working Party Opinion 04/2012 on Cookie Consent Exemption, src: http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2012/wp194_en.pdf

- Privacy settings (article 10)

Finally, the EC proposal specifically addresses how software should be enhanced to offer options to consent after its installation and to block tracking, when technically possible, by default.

It should be noted that the EC proposal is subject to many discussions and is expected to evolve over the next few months, as for example the voted amended version of the e-Privacy Regulation by European Parliament plenary of the 26th of October 2017 [17]. As the text is still not finalized, we refrain of making a more detailed presentation and analysis of the aforementioned articles in the content of this report.

2.3.2 General Data Protection Regulation

The General Data Protection Regulation also does not tackle web and application tracking specifically; however, it provides some safeguards. While these safeguards may not be sufficient or specific enough to protect the confidentiality of electronic communications and enforce the privacy of electronic services users, they offer resort guarantee for the transition between the ePrivacy Directive and the upcoming ePrivacy Regulation.

For instance, the opposition to tracking can be implicitly covered by the article 21 GDPR which discusses the data subject's right to object. The first paragraph (1) of article 21 details this right as follows:

1.The data subject shall have the right to object, on grounds relating to his or her particular situation, at any time to processing of personal data concerning him or her which is based on point (e) or (f) of Article 6(1), including profiling based on those provisions. The controller shall no longer process the personal data unless the controller demonstrates compelling legitimate grounds for the processing which override the interests, rights and freedoms of the data subject or for the establishment, exercise or defense of legal claims.

In most, if not all, cases related to tracking, the interests, rights and freedom of the data-subject should prevail over the legitimate ground of the data controller, meaning that the controller will no longer process personal data once a person has objected. The second (2) and third paragraphs (3) of the Article 21 introduce specific provisions for direct marketing:

(2). Where personal data are processed for direct marketing purposes, the data subject shall have the right to object at any time to processing of personal data concerning him or her for such marketing, which includes profiling to the extent that it is related to such direct marketing.

(3). Where the data subject objects to processing for direct marketing purposes, the personal data shall no longer be processed for such purposes.

These provisions empower even more the user when his/her data are collected for marketing purpose; since the controller will never be able to demonstrate that his interest overcome interests, the right and freedoms of the data subject. Practically, this specific provision suggests that a data controller can never process personal data for direct marketing purpose once the data subject has opposed.

The fifth paragraph (5) of Article 21 addresses specifically information society services:

(5). In the context of the use of information society services, and notwithstanding Directive 2002/58/EC, the data subject may exercise his or her right to object by automated means using technical specifications.

[17] https://privacy-news.net/uploads/1d5b2400-b4fc-11e7-bd94-7ff24f8b617d.pdf

This paragraph (5) has been interpreted [20] as an implicit reference to DoNotTrack and to any similar mechanism which would define technical specifications allowing user to oppose to his or her data being processed. This article aims at empowering data subjects with a simple solution for exercising their right to object when using information society services.

The recital (32) GDPR also includes a reference to the use of technical settings in information society services, but this time to express consent.

The GDPR article 22 provision on profiling, also strengthens the users' right to object when the result of profiling (which as earlier discussed can be linked to tracking, see also 2.1) significantly affects them. In particular, article 22(1) states that *"The data subject shall have the right not to be subject to a decision based solely on automated processing, including profiling, which produces legal effects concerning him or her or similarly significantly affects him or her".*

3. Consent Mechanisms and Privacy Settings

After outlining the legal framework concerning the consent and the privacy settings, as defined mainly in the new ePrivacy Regulation and the General Data Protection Regulation (GDPR), this chapter tries to define consent based on other paradigms and define the usual practices for obtaining it. Specifically, different consent mechanisms (via browser settings, first party and third party consent tools) are described as well as how the user agent setting can be used for tracking prevention.

3.1 Definition of consent

The General Data Protection Regulation defines consent as *"any freely given, specific, informed and unambiguous indication of the data subject's wishes by which he or she, by a statement or by a clear affirmative action, signifies agreement to the processing of personal data relating to him or her;".* Furthermore, Article 7 specifies that "*Where processing is based on consent, the controller shall be able to demonstrate that the data subject has consented to processing of his or her personal data.*"

In the Opinion 15/2011 on the definition of consent, the Article 29 Working Party details the conditions required for a valid consent and details the meaning of "freely given", "specific" and "informed". A later opinion [10] provides details about consent obtained for cookies and could be read as providing guidance about consent for tracking in general.

- Specific

For consent to be specific, the data subject should *know "which data are processed and for which purposes."* Therefore, the data subject should know for which purpose his or her data are processed. The consent should also be granular in the sense that it cannot cover *"all the legitimate purposes".*

The difficulty of obtaining an informed consent on the web is that the entity which is in capacity of informing the user prior to obtaining his/her consent (the service provider) is often not the only entity tracking the user. In many cases, as already mentioned, third party trackers will collect and even share information about the user visiting the provider's website. The service provider may not be aware of all the third parties that will be tracking on its website, let alone the purposes for which they will process the data.

To address this challenge, either the service provider should ask all its third parties to provide the purpose for which they will process the data and then provide the user with that information when asking him or her to consent, or the third parties must directly inform the visitor of the purpose of the tracking, before asking him or her for consent.

- Informed

For consent to be informed, users must be aware of *"the recipients of possible transfers".* This means that consenting users must at the very least be aware of the parties (data controllers in GDPR) that will be processing their data. In the context of web and application tracking, this requires the user to be informed about the third parties that will be setting and reading cookies on their devices (i.e. processing their data). It is however not required to explicitly and individually name those parties; they can be just categorized.

In some instances, the categories of data controllers could be enough to deduce the purpose of the processing and thus satisfy the "specific" part of the consent.

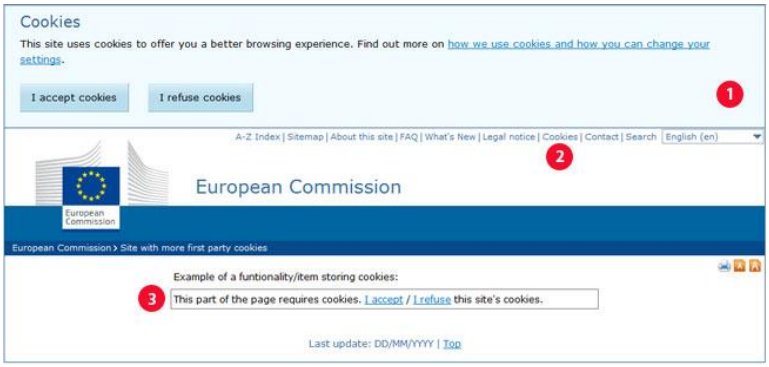
- Freely given

In its opinion 15/2011, Article 29 Working Party highlights the condition that *"data subject is able to exercise a real choice, and there is no risk of deception, intimidation, coercion or significant negative consequences if he/she does not consent. If the consequences of consenting undermine individuals' freedom of choice, consent would not be free."* In its guidance on obtaining consent for cookies [10], the Article 29 Working Party specifies how freely given consent can be obtained and details when restricting access can be a condition to the acceptance of cookies:

The emphasis on "specific website content" clarifies that websites should not make conditional "general access" to the site on acceptance of all cookies but can only limit certain content if the user does not consent to cookies (e.g.: for e-commerce websites, whose main purpose is to sell products, not accepting (non-functional) cookies should not prevent a user from buying products on this website).

3.2 Usual practices for obtaining user consent

The most widespread mechanism for user consent for cookies is probably the cookie header banner as described in the EU internet handbook [21]. In a proper implementation of such mechanism, the user is invited to make a choice to either accept or refuse cookies. Figure 2 shows such a web page with the cookie header banner on top of it.



1. The cookie header banner displayed on all pages of a site using cookies that require informed consent.
2. A link to the specific cookie notice page is also available.
3. This element of the page will only display its content once the user chooses to accept the site's cookies.

Figure 2: Cookie Consent Kit[18]

However, in many cases there is no way for the user to refuse cookies and still use the web page. Often, the only choice is to accept cookies by clicking "Continue". If the user does not accept cookies, he/she cannot use these websites[19].

Similarly, some websites inform the user about their use of cookies and just ask for a confirmation that the user understood the information (e.g. with a message such as "Got it"). Thus the user never explicitly gives his/her consent to accept cookies, but rather confirms that he/she read and understood the websites' cookie policy (see Figure 3).

[18] http://ec.europa.eu/ipg/basics/legal/cookies/index_en.htm
[19] See a relevant example in: http://www.jujuhq.com/2012/05/keeping-the-lid-on-your-cookie-jar/

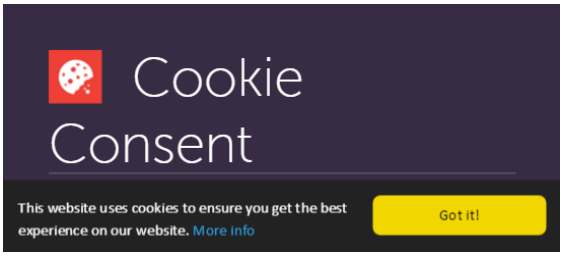


Figure 3: User' Confirmation[20]

These practices do not constitute valid consent in the sense of GDPR and as described in section 2.1. Having said that, it should be noted that opt-out controls are in some cases offered for the users to object to tracking. However, this is clearly not a way to consent, as the latter requires opt-in practices rather than opt-out.

Nevertheless, some server based consent tools have taken care to actually manage cookies, deleting those that require user consent if the user has not given it, or if the user has revoked his or her consent[21]. However, these tools only delete first party cookies, since the third party cookies cannot be directly managed.

Still, it is possible to block entire third-party domains from being loaded when the user has not given consent, using techniques such as dynamic CSP headers or tag management[22]. Even if nested browsing contexts do not actively manage their cookies in this way, the first-party site can block the JavaScript code using them when the user is deemed to be opted-out, either on first visit to the site or if consent revokes automatically after the consent duration expires. User consent can also be taken into account when a third-party video player is rendered[23].

When the user has been given the ability to give or withdraw their consent on specific websites they should be able to be informed of the current status. If Do-No-Track (DNT) is used as a consent signal (see Chapter 4 for more details on DNT), then the user agent can provide a User Interface feature to do this. For example, the browser extension Bouncer registers consent by using the DNT Consent API, i.e. consent is registered by sending the 'DNT: 0 header. When DNT is 0, consent must have been given, and the user should be informed about this[24]. Of course, not all user agents currently support the Consent API. In these situations, the sites can present the UI element themselves to remind the user what their consent status

[20] https://ourcodeworld.com/articles/read/131/top-5-best-cookie-policy-banner-javascript-plugins
[21] See relevant example in https://www.unilever.com/. The website has actively managed first-party cookies, deleting those that have not been declared as "strictly necessary to fulfil a purpose requested by the user". Moreover, see the cookie policy page (http://www.unilevercookiepolicy.com/en_GB/accept-policy.aspx) with opt-out button.
[22] See a relevant example in https://www.axe.com/de/home.html. On first visit, third-party JavaScripts that place cookies are blocked, and enabled only when the user opts-in. The consent panel has two buttons, accept or decline, giving the user an unforced choice, and in addition the consent expires after one year. If consent has not been given the site inserts the appropriate 'Content-Security-Policy header', so non-agreed third-parties are blocked by the user agent. The user can again revoke their consent at any time by pressing a button or buttons in the cookie policy page.
[23] See an example in https://www.axe.com/de/home.html. If consent has not been given the video is unable to play, and associated cookie placing domains are blocked. The user can be shown text explaining that cookies will be placed when they press an "accept and play" button.
[24] See in previous example: the shield icon (top right on the webpage) is used to inform the user. When consent is revoked or expires the DNT header reverts to DNT: 1 and the shield icon no longer has a C in it.

is[25]. The user may not initially recognise what these icons mean, and what they can do with them. One way to address this is to show a "reminder banner" the first time a user visits a site[26].

3.3 Consent mechanisms

3.3.1 Consent via browser settings

In its opinion 2/2010 on online behavioural advertising [22], the Article 29 Working Party specifically tackles validity of consent obtained by the way of user agent (browser) settings. Article 29 clarifies that first, *"data subjects cannot be deemed to have consented simply because they acquired/used a browser or other application which by default enables the collection and processing of their information."* The Working Party also requires that for such settings to provide a valid consent, they must have an actual effect on the tracking capacities of others parties, as *"it should not be possible to "bypass" the choice made by the user in setting the browser".* Finally, consent granted through the browser settings must be specific and therefore should adapt to future changes of purposes. That means that granularity should be better than "all or nothing".

As of today, the existing browsers offer only a limited set of options:

- Accept all cookies (default on IE, Chrome, Firefox)
- Accept only cookies that are set or accessed[27] by a first party (default on Safari)
- Accept only cookies set by a first party
- Accept no cookie.

The Article 29 Working Party suggested that for user-agent settings to be used to get a valid consent mechanism, the settings must reflect an active choice from the user.

This provision should be extended to cover all cookies that must be conditioned to consent. Indeed, as third party cookies are now blocked by default by some user-agents, advertisers and other tracking parties now use first party tracking cookies. Therefore, user-agents should be in a capacity to filter first party cookies based on their purposes. However, as discussed in Chapter 2, user-agent settings offer no such granularity and do not provide any tool to obtain a specific consent, and have no way of knowing the purpose of a given cookie.

3.3.2 First party consent tools

First parties (i.e. service providers, such as online shops, e-commerce websites, etc.) operate the website that is visited by the user. They are capable of contracting with a first set of third parties that will use cookies on their website. Often these third parties will be used to deliver a service: monetize ads slot, provide an analytic solution, share content on social networks, and embed multimedia content.

[25] See for example in https://www.gsksaglik.com, where an icon in the form of a "gate" symbol (bottom right) is presented. A "closed gate" indicates consent has not been given, or that it has been revoked, while an "open gate" that it has been given. The icon can let the user change the user's consent status at any time by showing a consent change panel when it is clicked. Note that the user agent is also able to show the user their consent status, due to the use of the DNT Consent API to register consent when it is available. When the site shows an "open gate", the shield icon shows a C, and vice versa.

[26] See for example in https://www.gsksaglik.com. The default is 'no consent' and this can be explained in the banner. In subsequent visits that banner does not appear and the user can change his or her consent status using the icon.

[27] Safari blocks storage by a third-party (embedded sub-resource) but not access. The cookie can get stored when the resource is visited as a first-party, but once a cookie is associated with a domain it always gets sent

3.3.2.1 Information

The service provider only knows the third parties he 'calls' on the website, but these third parties may in turn 'call' other third parties. The service provider may inform the users about the purpose for which third parties are being 'called', as long as the service provider is aware of the processing performed by all these third parties.

Moreover, in many cases third parties might be processing personal data for their own purposes, beyond the original purpose of the service provider. For instance, many service providers embed content from YouTube and Vimeo to show videos on their website. Therefore, from the service provider perspective the purpose of allowing these third parties to set cookies on their website is to show videos, but YouTube and Vimeo process the data collected through their cookies mostly for an advertising purpose (segment the audience of a website).

Taking into account the aforementioned considerations, there could be two layers of information when third parties are being 'called' in a service provider's website. The first layer would explain why third parties are 'called' and which parts of the service rely on these third parties (e.g. delivering video), and the second would explain how the third parties process the data for their own purposes (e.g. advertising).

Furthermore, as also mentioned earlier in the document, the same third party cookie could have different first party uses. For instance, an advertising cookie, set on an e-commerce website, will mostly be used to retarget a user, while script setting the same cookie in a news website will be used by the first party to monetize content.

3.3.2.2 Consent

Consent can only be granted once the user has been informed about the purposes of tracking. Therefore, explicit consent will require a positive action from the user who has been informed for all types of cookies, including those posed by third parties. Until this positive action occurs, the website must prevent third parties from 'dropping' cookies.

Only a few advertisers offer[28] (or planned to offer[29]) an option to let service providers inform them that they cannot set cookies. When they cannot inform their third parties that they cannot set cookies, service providers should in principle block all contents from their third parties setting cookies (which as shown in 2.1 is not always the case). The third parties could be grouped by the purpose for which they are embedded on the first party website (e.g. advertising, social network, analytics, showing videos) and consent has to be granted for all the purposes for which a third party is embedded before this third party can be called.

3.3.2.3 Tag Manager

A "tag manager" [30, 31] is also a technical implementation of the cookie consent that consists in blocking the execution of the script (also known as a "tag") of third parties. Practically, this requires encapsulating all scripts set by third parties within a conditional test: *if consent has been obtained, the third party script can be called, otherwise the script is not executed*.

---

[28] DoubleClick "Disable cookies on a per request basis" https://support.google.com/dfp_premium/answer/3202794?hl=en

[29] SmartAdServer noocookie parameter: http://www.geste.fr/sites/default/files/files/Gestion-des-cookies-Smart%20AdServer.pdf

[30] Google Tag Manager Overview, https://support.google.com/tagmanager/answer/6102821?hl=en

[31] Tag Commander, Take back control of your data, https://www.commandersact.com/en/products/tagcommander/

Depending on the framework used by the first party, this can be implemented using JavaScript, php or other languages used by the service provider. While effective in blocking cookies, fingerprinting, and most tracking tools, tag manager also blocks the third party content, meaning that some functionalities may not be available to a user who has not given his or her consent.

3.3.2.4 Content Security Policy

Servers sometimes need the assistance of user-agents to protect users from malware or illicit data collection. The ability to embed external resources into web pages as separate nested contexts i.e. by using the *iframe* tag, means script can be inserted that can dynamically embed further content, perhaps with the original designer of the site not being aware of it. In many situations, notably the insertion of programmatic advertising, the content that eventually is inserted may come from unknown entities, and while this may be intended, there is still a duty to ensure that illicit or dangerous content, e.g. malware, is not introduced into users' equipment.

W3C defines a standard for service providers to list the trusted parties that can be embedded in their pages [24], called the Content-Security-Policy API(CSP), and this is now widely supported by modern web browsers. CSP was introduced to prevent cross-site scripting (XSS), clickjacking and other code injection attacks resulting from execution of malicious content in the trusted web page context. CSP provides a standard method for website controllers to declare approved origins of content that user agents should be allowed to load on that website — covered types are JavaScript, CSS, HTML frames, web workers, fonts, images, embeddable objects such as Java applets, ActiveX, audio and video files, and other HTML5 features. Each of these content types have their own *source list* within the CSP header and if any of these are provided it becomes a whitelist, i.e. only content of that type that is identified on the list will be loaded and any other content of that type blocked by the user agent.

When the user agent detects a Content Security Policy (CSP) response header on a webpage, it will parse it to find the list of domains that are authorized to set content on the website.

Therefore, a CSP can be used to block content for which consent has not been granted [24]. For the first visit of a user on a webpage, the content security policy will list only domains controlled by the first party (service provider). If the user consents to the processing of his or her data by third parties, those third parties will be added to the CSP so they will not be blocked.

These actions would be even more effective if they were informed by the user agent being aware of the user consent status of specific domains. For example, a different CSP could apply when the user had given consent for tracking, e.g. the DNT request header value sent to the domain was "0".

There is also an associated standard under development called CSP Embedded Enforcement API [25], which lets a site insist that sub-resource browsing contexts honours a specified CSP. In this way user agents can be told to block iframes from being loaded when they have not been explicitly allowed by the site's designers or which refuse to respect the provided CSP.

3.3.3 Third party consent tools

Third parties know for which purposes they are processing data they collect by tracking users on the web. As such, they can provide accurate information about their data processing. However, users are hardly aware of the third parties that track them and hardly visit their website. Therefore, third parties can hardly ever directly deliver information about processing to the user and have to rely on first parties to deliver that information (assuming that the third party informed them).

A solution when consent can be obtained by actively pursuing the navigation on a website, is for third parties to adapt their script to not set cookies the first time they are called from a new first party [16] or when they receive the opt-out parameter in the URL. A user agent visiting a website for the first time would then not be tracked and would have a chance to oppose to cookies being set before pursuing his or her navigation.

However, this assumes that third parties have the capability to identify the first call from a first party (for instance through the referrer) and that the information provided to the user by the first party clearly mentions the third party opt-out tools or that the first party provides an opt-out solution.

While opt-out solutions could be technically feasible, they oblige the user to visit each third party for opting-out or to visit an opt-out portal to opt-out from many third parties at once. Moreover, this does not satisfy the opt-in regime of the ePrivacy legal framework. In particular, the valid consent described by the Article 29 Worky Party [22] opinion suggests that *"ad network providers should swiftly move away from opt-out mechanisms and create prior opt-in mechanisms".*

3.4 Tracking prevention via user agent settings

User agent settings can play an important role in preventing user tracking, by providing appropriate information to users, as well as presenting them with relevant choices and privacy prevention mechanisms. This section provides an overview on this topic, together with some specific examples.

3.4.1 Privacy settings

The user agent can provide information to the user about possible tracking prevention mechanisms and the relevant settings which the user may accordingly configure. The user friendliness of the interface is key in this respect. Unless the user is obliged to go through the settings (as for instance the permission request on Android phones), a complex user interface will either result in low adoption rate of privacy settings or users choosing the easy way out with express settings and leaving the default settings unchanged.

To this end, in the following we explore in more detail some aspects of user agent settings for tracking prevention, in particular granularity of settings, default value and possibility of user to update and revisit the settings. Moreover, Annex B provides some relevant examples.

The granularity offered to the user will vary with the type of tracking considered. A web tracking protection mechanism will often only protect against the collection of data about the user browsing habits. This protection will often be sufficient to meet the user's expectations but tracking mechanisms could offer a finer grained control, e.g. by blocking either the tracking of the user location or the tracking of the browsing habits.

Broadly speaking, tracking protection has several dimensions:

- the granularity of the items the user wants to block/consent, can be either:
  - a list of domains
  - a specific type of tracking company
  - a specific type of technology.
- where the user wants them to be blocked:
  - on all websites
  - only on chosen websites
  - on a chosen group of websites.
- for consent to be specific, the users should be able to choose for which purpose he or she consents (or opposes) to tracking:

- all purposes (e.g. audience measure, advertising and other purposes declared)
- only a specific purpose (e.g. advertising).

- the type of data:
  - all types of data for which permissions are necessary
  - all types of data that are not strictly necessary for the application
  - all types of data that are not necessary for the current utilization of the application

  only a specific type of data (e.g. microphone, contacts, pictures, location on a smartphone).

All these dimensions should be taken into consideration while discussing privacy settings for user agents.

3.4.2 Default value

Article 29 in its opinion on Behavioral advertising states that the default settings of user agents should not permit third parties to track the user. In practice, the choice offered to the user varies based on:

- The information offered about the existing control:

Some settings will quickly be summarized and the user will be asked to accept them or to click on more settings. Often, in order to not be overly complicated, the information provided to the user omits some details that could have influenced the user decision.

- How quickly can the settings be changed

The question is if the the user should go through multiple screens and long list of settings before being able to change the settings.

- When is the user informed about the default settings

Information provided at first run or during the installation may lack the context to help the user make a good decision. On the other hand, offering in context notification may sometime be adequate, for instance prompting the user for sharing data about words typed on the keyboard while they are typing a message.

3.4.3 Changing mind

To compensate for the lack of control offered during the first choice of settings, a compensative measure could be to facilitate the control a posteriori. If the user can easily revisit choices, he or she may adapt them based on his or her experience. The granularity of the choice offered then will impact users' comprehension and willingness to revisit their choice.

4. Signalling consent for tracking with DNT

Following the description of consent mechanisms and the use of browser settings to prevent tracking, in this Chapter we focus in particular *on the deliberate communication of user consent between browsers and servers*, or otherwise *the signalling of user consent*. After analysing how consent is communicated between servers and clients and how client-servers collaborate to this end, we explicitly focus on the DNT header, which is a 'preference expression' setting in a browser, designed to allow the *signalling of user consent for tracking* to websites. This mechanism, if correctly and broadly implemented by servers and clients, can greatly support users' protection against online tracking.

4.1 Communicating consent between servers and clients

A user's consent for tracking must be stored somewhere so it can be acted upon in future transactions. This necessitates the use of some persistent storage in the user agent, either to store a low-entropy indication of whether consent has been given or not, or to store a unique identifier whose existence signals the fact that consent has been given. The identifier can then link to a record in a server-side database giving more information on what has been agreed to.

When user agents request content from servers they communicate information about themselves in the HTTP request headers. The server, or script executing in the context created by the server, can use the standard Cookie header to convey this information, which can also contain an attribute specifying how long the cookie should last, i.e. when the consent should automatically be revoked. For example:

1. A "low-entropy" cookie stores a value indicating consent has been given, e.g.

```
Cookie: Consent=yes;path=/;expires=Sun, 16/07/2017 23:30:00 GMT";
```

2. A UUID cookie is used to let the server look up a user specific data record, where the fact that consent has been given can be stored. This use of storage to maintain a consent record can be explained to the user at the time they are asked for it. The UUID will be automatically deleted when the cookie expires, and therefore also the link to the specific data record.

```
Cookie: Consent= 1a2b3c4d;path=/;expires=Sun, 16/07/2017 23:30:00 GMT";
```

While using a cookie for this is relatively straightforward it does not in itself solve the problem of signaling user consent to embedded third-parties. While this can be achieved in the case of nested browsing contexts (e.g. *iframes*) using readily available APIs such as `postMessage` [32], this is of no use for non-active content such as images. Even in the case of active contexts there is no currently standardised way to format the messages and so this would require separate protocol agreements between the parties.

Using cookies to indicate consent also restricts the use of expiration caching because cookies often contain unique values making as many caches exist as there are users, i.e. the `Vary` header is not useful for inhibiting redundant requests for pages which may be held in intermediate caches. While it is possible to tell user agents and intermediary devices not to bother requesting new content unless certain headers have changed by using the `Vary` header, this is usually pointless in the case of the `Cookie` header because every user agent probably has a unique set of them.

[32] https://developer.mozilla.org/en-US/docs/Web/API/Window/postMessage

These problems do not arise with the DNT header. This is an HTTP request header that can contain the value “1” (for no tracking consent) or “0” (for tracking consent), sent in every request when the DNT “general preference” is enabled in the user-agent. Servers for any content, i.e. not only nested browsing contexts but images or any other web resource can receive and act on the header. A JavaScript API has been defined so consent can be given for specific domains, which means that the DNT value of “0” (DNT: 0) is sent to those domains, either when they are accessed by embedded content on particular sites (site-specific consent), or on any transaction on the web (web-wide consent). The value can only exist in 2 states so itself cannot be used to track the user. This also means the header can appear in a "`Vary`" cache header so expiration caching can be implemented very efficiently.

Using a dedicated header also has the advantage that user-agents can recognise it themselves, and act upon it. A cookie is simply a pair of strings, a “name” and a “value”, and there is no standardised naming convention to allow user agents to recognise what purpose a particular name (or value for that matter) is intended for.

On the other hand, DNT: 1 has the universally recognised meaning of Do-Not-Track with DNT: 0 having its reciprocal meaning i.e. that consent has been given for tracking.

It follows that user-agents can not only adjust their tracking protection behaviour dependent on the existence of user consent, they can also implement User Interface mechanisms to revoke or give consent themselves. If a set of domains are recorded as having been given explicit consent then this fact can be conveyed to the user, perhaps long after they have forgotten they gave it, using a suitable UI. They can then amend their choice, e.g. revoke their consent, at any time they choose, without having to revisit the site, and be reminded that they have given consent by their user agent periodically.

This means that not only must the necessary information about the identity of the data controller and the purposes for tracking be communicated from the service provider to the user-agent, this information must be machine-readable because the user agent must be able to automatically parse it, so it can be presented to the user in a standardised and meaningful manner.

The Do-Not-Track specification [6], discussed within the Tracking Protection Working Group [26], covers how information passed from the server to the user agent by defining a Tracking Status Resource (TSR), a JSON resource that is located at a specified relative path for any domain. This represents an extensible JavaScript object that can hold the necessary information. In [26] a possible schema for how transparency information could be included in a TSR is given. All current discussions regarding implementation issues of DNT are discussed in the GitHub page[33].

4.2 Client-server collaboration

Enabling both clients and servers to recognise when a user has given their consent or not, and for what purpose, radically improves the capability for different entities in the web ecosystem to offer users more control. It creates the possibility for user agents and servers to collaborate, creating more effective ways to protect privacy.

Servers should make information available to user-agents that helps them communicate to the user the purpose for storage e.g. what services cookies or web storage items are designed to support. Every data controller responsible for the content should identify themselves, especially if servers are being accessed

[33] GitHub page, https://github.com/w3c/dnt

as embedded third-parties and, if they have obtained user consent and this has not been registered in the DNT header, communicate the fact to the user agent.

User agents can make this information available to the user in easily located User Interface (UI) controls. It will always be available, and the user will be able to know where to find it, what it means and be able to make choices dependent on it.

Similarly, if the user agent itself can obtain the user consent for tracking by specific domains, the fact that it has been registered should be communicated to the appropriate servers. Once this ability becomes widespread there will be less need for servers to obtain consent themselves, which will be less of a burden for websites. In addition, user agents will be able to differentiate between domains the user has given consent to versus domains they have not given consent, and therefore be able to apply different levels of protection to them, for example by blocking domains which do not declare their support for DNT (by supporting a properly formatted Tracking Status Resource at the "well-known" location).

User agents increasingly offer their users options to help protect them from being tracked, especially by embedded third-parties. These options primarily address how cookies are processed, but also deal with other potential repositories of personal data such as HTML web storage or the user agent cache. If user agents have access to machine-readable information on the identity of third-parties, e.g. by each party declaring a DNT TSR, and the purposes for all parties for storage access, then they will be able to provide users with fine control over their web experience, without having to block needed functionality.

4.3 Do-Not-Track header

As already mentioned, the DNT header differs from others in that it can be used to signal to other domains, i.e. it is designed to be cross-origin. Once a user has set their general preference not to be tracked, the signal will be included in every HTTP request to first-party websites, in addition to all the servers handling their embedded third-party elements. In addition, the JavaScript Consent API (also known as the User Granted Exception API), can register a user's consent to the first party and some or all the embedded third-parties on a particular website, or to a particular third-party resource anywhere on the web. This allows the user agent to communicate user consent, when it is given, to embedded third-parties of a website using standard user-agent signals. Not all user-agents currently support the Consent API so this functionality has up to now been developed in a proprietary way (generally known as out-of-band consent) for non-DNT compliant user-agents.

DNT is not only about user control (i.e. respecting a user's control or tracking preference), but also transparency (i.e. presenting information about tracking behavior in a standardized way). If sites, or their embedded third-parties, do not communicate privacy practices that can be clearly understood and verified, users lose trust in them. Browsers and browser extensions designed to enable tracking protection will also be able to use machine-readable tracking declarations to inform decisions on content blocking and other privacy protective measures.

DNT defines methods for declaring such things as the identity of the site's owner (data controller under GDPR), its tracking policy (listing why tracking occurs and the purposes for which the collected personal data is used), the compliance regime it operates under, the other host domains it controls, how consent can be given or revoked, and how its tracking behaviour has been modified in the light of a specific tracking preference. This is done in standardised machine-readable elements available to both first party and third-party servers in the Tracking Status Resource (TSR), which returns a JSON object when content is requested from the DNT well-known location (./.well-known/dnt/). Because this information is delivered

transparently and available to anyone, regulators can check its veracity and ensure that companies are held accountable for it.

Browser extensions and browsers, designed to give users control over non-consensual tracking or stop intrusive advertising, can examine the DNT status to recognise when a user has given consent, detect if a domain respects DNT by checking for the existence of the TSR, or examine the TSR to determine if embedded third-parties share the same data controller. This will allow extension developers to make intelligent decisions about sites, so that trustworthy sites and third-parties will not have to have their content or functionality arbitrarily blocked.

If a site does use tracking, it must determine for every incoming HTTP request, whether a user has given his or her consent to being tracked or not given his or her consent. When user has not given consent, "tracking data" must not be collected, stored, or retained unless certain exemptions or alleviations apply. "Tracking data" means anything that can be used to single-out the user, i.e. unique identifiers in cookies, IP addresses, derived unique identifiers such as user-agent fingerprints, etc.

The default formal standard for sites that comply with Do Not Track is available from the W3C. The formal standard for the technical aspects of Do Not Track is Tracking Preference Expression [6]. Other projects[34] and documents such as the EFF's Do Not Track Policy [28] also exist, and it is expected that regulators in Europe and elsewhere will issue detailed guidance on DNT compliance in their jurisdictions.

The DNT setting is externally verifiable and provided it has properly been selected in a web request, the website's server is required to respect it; the general terms and conditions of the web site can include suitable provisions thereto. The main risk to the integrity of the signal lies within the browser or user agent, where an attacker may be able to register user consent, without being noticed, when in fact no consent has been given by the legitimate user. The current version of the DNT API allows embedded sub-resources, i.e. nested browsing contexts, to initiate the DNT API, i.e. to prompt their domain origin, and perhaps subsidiary embedded origins, to be sent the DNT:0 (consent) signal. This could become a greater risk when browsers implement tracking protection procedures for particular domain origins contingent on the status of the DNT signal, and it is likely to then turn them to a target more easily. This underlines the importance of regulatory oversight in a way that action can be taken against potential perpetrators. Also browsers should ensure that users monitor the status concerning their consent in relation to tracking.

4.3.1 Technical DNT implementation by service providers (websites)

Implementation can be broken down to 5 main functional areas.

1. The site must contain a reference to a Tracking Status Resource (TSR) that can give user agents information about the site's tracking purpose, if any.
2. Unless the site does not ever track users, each incoming request should be checked to see if it contains a `DNT` header and whether it denotes a Do-Not-Track signal (`DNT:1`) or that consent for tracking is provided (`DNT:0`). In Europe, if no `DNT` header exists then the site must assume that the user has not consented to tracking. In other jurisdictions the default may be an implied consent. It is also possible that the user has consented to tracking using another mechanism for giving explicit consent, not described in the DNT protocol, referred to here and in other DNT documents as ‘out-of-band consent’.
3. Action taken depending on the deemed tracking preference.

[34] See for example: Cookie Clearing house, https://cch.law.stanford.edu/statusnext-steps.1.html

- If the request signifies user consent for tracking, tracking data can be used without restriction, as long as it is compatible with the information given to the user when consent was obtained.
- If the request is a Do-Not-Track signal, which means that user has not provided consent for tracking, and no exemption or alleviation applies, then the tracking data must not be retained. This normally means that UID cookies should not be placed (and, at least in Europe, should be deleted if they are already there), JavaScript used for fingerprinting should not be loaded, location data with less than "City level" granularity and any user identifying data in logs (such as IP addresses) deleted or otherwise permanently de-identified.

4. A site's tracking behavior in the context of a particular request should be indicated in the Tracking Status Value (TSV). The TSV can be communicated in various ways.
   - It can be recorded in the `Tracking` property of the TSR. The TSR is a JSON encoded resource located at a particular location on the site's domain. If the site's resources use the host name `example.com` then the TSR would be at `https://example.com/.well-known/dnt/`. The TSR can be dynamic with different JSON returned depending on request headers. For example, a TSR with the `Tracking` property set to C or T can be returned if the DNT request header value is "0", while a `Tracking` property of "N" would be returned when DNT was "1" [35].
   - If the value can change for individual requests, and the TSR has not been designed to be dynamic, the `Tk` response header can be used. Either:
     - The `Tracking` property can be set to `?` (`Dynamic`) and the TSV delivered in the value of the `Tk` response header, or:
     - A sub-tree of Tracking Status Resources can be maintained and the specific one that applies in the current request indicated using the status-id mechanism.
5. If the site uses the user's consent to enable tracking then this can best be achieved by informing the user of the purpose and retention limits etc., then, when they have given their consent by checking a box or clicking a button, using the DNT Consent API to register the fact in the browser. The Consent API can be used at any time, even if the user has not enabled a general DNT preference, i.e. even when DNT is unset. As the script API is not currently supported by some browsers an out-of-band technique, probably using cookies, can be used instead. In this case a Tracking Status Value of C must be communicated in the Tk response header or in the `Tracking` property of a dynamic or request specific TSR.

4.3.2 Extensions to get a valid consent

As we have explained, service providers often have less than complete control over what third-parties exist on their websites and how and why these parties access the terminal storage and personal data of their site's users.

Being able to restrict what third-parties are loaded into browsers and manage the storage they use, is a useful control mechanism for service providers. Not only can they minimize the legal and compliance risks associated with unlawful access by third-parties to storage and personal data, they can also minimize the unwanted exfiltration of data from their sites and create a better managed and more transparent environment.

This could be achieved by communicating transparency information to user agents in a machine-readable, standardized, language independent and verifiable form. This could declare what other domains are

[35] The tracking status value is case sensitive, and is defined: "C" — tracking with consent, "T" — tracking, "N" — not tracking.

expected to be referenced or embedded by a site, and what their relationship to them is. In addition, the purpose for using browser storage such as cookies could be declared, so it can be explained to the user in standardized language independent terms, to allow users to make informed decisions on whether cookies etc. should be kept or removed. A suggested schema for how this information can be presented in a machine-readable way is contained in Annex B, consisting of:

- The list of first and third parties that are collecting data
- For which purposes these parties are processing these data

A similar schema has been proposed [29], [30] to the W3C's Tracking Protection Working Group[36], and may be considered for inclusion in a future revision of the recommendation. Also under consideration is a proposal to include a reference to the purposes that a user has given its agreement for within the DNT header after the first "0" character, and an additional property for the API parameter dictionary for specifying it[37].

For example, one suggestion has been the “OtherParty” array which contains a list of third-party domains, the third party content that the first party has knowingly integrated into its website (or web application). As some third parties’ *iframes* go on to load their own third parties, the first party may not be aware of all third parties that could end-up being accessed by the user-agent. However, the “OtherParty” property would list the third parties that the website owner has contracted with and knowingly integrated.

The user-agent could prefetch specific transparency information for each domain in the "OtherParty" list, and could then inform the user about the third parties that are collecting data on the website and the purposes for which they collect it. The user-agent could then make the information available to the user, and ask for their consent when required. Also, because the user agent is able to detect when the user has not given their consent, it can then block the content of, or inhibit tracking by, domains that the first-party has not included in the list.

Furthermore, the user may have previously selected the purposes for which he or she would agree to have his or her data processed, the user agent could automatically take the decision to grant or reject the exception request. That would assume that a common taxonomy is shared by the user agent and the third parties processing data to define the purposes.

Another technique could be to use the Content-Security-Policy response header (CSP) [23] to limit the set of domains that can be loaded by the user-agent (see also section 3.3.2.4 on CSP). Servers could deliver different CSP headers depending on whether DNT is set or not in the request, or not.

In the future, an extension to the CSP API could allow user-agents themselves, perhaps informed by the transparency information described in Annex B or the through monitoring of the DNT Consent API, to enable a less restrictive set after a user gives its informed consent to some or all the embedded third-parties.

4.4 DNT implementations

Soon after the DNT standard proposal in 2009, several large internet companies announced to implement the HTTP header field. Among those, we differentiate between browser makers (such as Mozilla) and content providers (such as Twitter). Some companies are part of both groups (such as Microsoft). This

[36] W3C's Tracking Protection Working Group, https://www.w3.org/2011/tracking-protection/
[37] Issue 60 currently under consideration, https://github.com/w3c/dnt/issues/60

section touches upon the specific ways those big players reacted ("implemented") DNT (or related technologies), focusing on their business/methodological perspective rather than the actual technical implementation.

4.4.1 Browser implementations of DNT

Today, all major browsers support the DNT http header field, e.g. Mozilla Firefox, Google Chrome, Microsoft Internet explorer and Edge-browser, Opera and Apple Safari. All these browsers adopted DNT and allow users to activate the setting. However, it is interesting to explore how these browsers set the default configuration of DNT. In particular, the development of DNT-default settings over time gives an indication on how DNT is currently perceived by the industry: many browsers activated DNT by default at the time of first implementation to better protect their users' privacy. A few years later, many of these browsers changed this default setting (for example, Microsoft removed the DNT - ON default setting in their new browsers in 2015[38]. In more detail:

- **Mozilla Firefox**: DNT is disabled by default[39] in regular browsing, however it is activated in "private browsing". This implies that Mozilla expects the user either to welcome tracking during the day-to-day web browsing or willingly choose to browse "in private mode". Private mode limits the UX in many other aspects besides turning off DNT. Only if the user dives deeply into the browser settings, he/she might enable DNT also during regular browsing.
- **Google Chrome**: DNT is disabled by default. Incognito (private) mode browsing is possible, but the DNT settings are the same as in "regular" browsing mode[40].
- **Microsoft Edge and Internet Explorer**: DNT is disabled by default. In a dedicated blog post [41], Microsoft argues that due to legal ambiguities, the DNT setting (in particular the default setting) may not be considered a "deliberate choice by the user. In the absence of user choice, there is no tracking preference expressed."
- **Opera** disables DNT by default as the other browser do. DNT can be enabled in the settings. In addition, Opera offers "site preferences" which allow the user to specify certain settings (including DNT) to be applied *per site*[42] . This way, users can enable DNT for "general browsing" and disable it for certain sites that they visit regularly and trust and thus allow to track.
- **Apple Safari** (which also disables DNT by default) does not explicitly mention "Do Not Track" as an option for end users, but offers the respective setting and explains the general and expected behaviour, i.e. possible ignorance of web sites[43].

4.4.2 Web sites / service provider's implementation

Even though implementing an appropriate behavior of web servers is easy to accomplish, many web service providers do not implement any treatment of DNT header fields in http-requests. This is also a situation that changed over time: in the early days of DNT several companies implemented DNT and respected the "do not track" request by some users. Over time, many of them rolled back the respective policy and ceased supporting DNT as a web standard[44].

---

[38] https://techcrunch.com/2015/04/03/microsoft-disables-do-not-track-as-the-default-setting-in-internet-explorer/
[39] https://support.mozilla.org/en-US/kb/settings-privacy-browsing-history-do-not-track
[40] https://support.google.com/chrome/answer/114836?hl=en
[41] https://blogs.microsoft.com/on-the-issues/2015/04/03/an-update-on-microsofts-approach-to-do-not-track/
[42] http://help.opera.com/Windows/12.10/en/notrack.html
[43] https://support.apple.com/kb/PH21447?locale=en_US
[44] See http://donottrack.us/implementations for a list of companies that once committed to DNT. Many of them changed (revoked) this commitment in following years. For example, while Twitter in 2012 announced to implement

Two factors influence the consequences of this trend to move tracking settings from the browser to the web service provider:

1. Standardization: sites behave differently and may change the policy at will. While personalization in general is enhancing user experience and customer satisfaction and may allow detailed user specific configuration, it also bears some serious (privacy) challenges: service(company)-hosted personalization relies on the companies' privacy policy and other rules that may change over time and according solely to the company decisions; the user needs to read those policies carefully and keep all the different flavors of tracking policies in mind. There is no "standard" and no neutral third party regulating this inherent conflict/issue.

2. Non-registered visitors: only registered users can "personalize" the service experience. Anonymous users who did not register or log-in to the service cannot set any "personalization". Thus seemingly "anonymous" users who have no profile with personalization or privacy settings have no chance to refuse tracking.

Concluding the view on DNT implementation, we find that DNT adoption was higher shortly after its initial introduction rather than today.

While all browsers are likely to offer purpose designed settings to obtain, register and communicate consent the forthcoming e-Privacy Regulation is also likely to mandate it. Taking into account the increasing number of people concerned about online privacy, any browser, or browser extension, that enables this functionality by, for example properly implementing the DNT API and other elements described in the Tracking Protection Expression [6] is likely to increase the chances of success. A DNT browser similar to a neutral search engine could be offered as a proxy service which intercepts traffic from a subscribing user's agent and implements appropriate procedures in the proxy's server to obtain, register and communicate consent.

It should be noted, however, that the future adoption of the DNT protocol may change depending on the final text of the proposed ePrivacy Regulation. For example, amendments to the draft ePrivacy Regulation recently agreed by the European Parliament require browser providers to implement browser settings for signaling consent. In particular, in the EP agreed draft it is stated (amendments for article 10) that " *the settings shall lead to a signal based on technical specifications which is sent to the other parties to inform them about the user's intentions with regard to consent or objection*", "*This signal shall be legally valid and be binding on, and enforceable against, any other party*" and that it "*may be expressed or withdrawn at any time both from within the terminal equipment and by using procedures provided by the specific information society service*" [45], i.e. the same indication can be recognized and acted on within both user agents and servers.

and support DNT (https://blog.twitter.com/official/en_us/a/2012/new-tailored-suggestions-for-you-to-follow-on-twitter.html), in 2015 the privacy policy changed and refers to user service settings for twitter customers (https://twitter.com/personalization and https://twitter.com/en/privacy): "*We [twitter] respond to these [personalization] settings rather than the Do Not Track browser option, which we no longer support*". Twitter retained a form of DNT indication allowing website publishers to indicate DNT by defining an HTML Meta tag (https://dev.twitter.com/web/overview/privacy#what-privacy-options-do-website-publishers-have).

[45] http://www.europarl.europa.eu/sides/getDoc.do?type=REPORT&reference=A8-2017-0324&language=EN

5. Conclusions and Recommendations

Tracking is being extensively used nowadays to identify and collect information about online users. Numerous technologies, such as third-party cookies, HTML5 local storage, browser cache and device fingerprinting are utilised to allow third parties to recognize users across websites and build browsing history profiles. From the perspective of service providers, tracking provides desired functionality including personalization, site analysis and targeted advertising. From the user perspective, however, it can introduce serious privacy risks, which the user cannot easily understand or manage. Therefore, users should be able to explicitly provide their consent once they have been informed about the purposes of tracking, which mainly will require a positive action from the users' side.

As shown in the analysis of this study, general privacy settings in user agents cannot be unambiguous indication of user consent. Users often cannot disable cookies or other access to data stored across all sites because the functionality of the services they require would be seriously restricted. Even if settings can be easily applied to specific service providers, there may not be a secure or privacy protective way for user agents to indicate this.

On the contrary, consent should be capable of being signalled using purpose-designed mechanisms, unambiguously stored in a standardised way, recognisable to both server and clients. Clients can use this commonly understood indication of consent to provide more intelligent means to protect privacy and personal data, without diminishing users' experience of the services they require.

Based on the analysis in previous Chapters, the following recommendations can be drawn to the different stakeholder's groups (service providers, user agents, policy makers, regulators and others) for an enhanced implementation of user protection mechanisms against tracking.

D) Service providers

It is recommended that service providers:

- Provide clear information on their identity and how they use terminal storage and for what purpose in standardised machine-readable way, so that clients can act on it to protect privacy and be able to present it to the user in intelligible ways while not diminishing the user experience.
- Ask all their third parties to provide the purpose for which they will process users' data and then provide the users with that information when asking them to consent.
- Consider grouping third-parties by the purpose for which they are embedded on the service provider's website (e.g. advertising, social network, analytics, showing videos) and asking for consent for all the purposes for which a third party is embedded before this third party is 'called' on the web page.
- Restrict third-parties from loading into browsers by default and manage the data storage these parties use by communicating relevant information to user agents in a machine-readable, standardized, language independent and verifiable form.

E) User agents

It is recommended that user agents:

- Make default settings as privacy protective as possible.

- Implement features so that purpose-designed user consent signals can be recorded at a granular level, on at least a domain specific basis.
- Consider differentiating between domains the user has given consent versus domains he or she has not given consent (and therefore being able to apply different levels of protection to these domains).
- Consider options for reflecting an active choice from the user, which would constitute a valid consent mechanism.

F) Policy makers, regulators and other stakeholders

It is recommended that:

- Policy makers and regulators provide further guidance on the technical implementation of valid consent and privacy defaults, in co-operation with the industry of user agents and associations of service providers.
- The research community continues work in tracking prevention mechanisms, including the technical implementation and signalling of valid user consent.
- The European Commission and EU institutions in the field of privacy and security support relevant initiatives and research projects in this area (e.g. in the framework of H2020).
- The European Commission and EU institutions in the area of privacy and security promote the formulation of workshops and working groups with all relevant stakeholders in the field.
- EC standardisation bodies engage in the creation of relevant online data protection standards, in particular in the fields of signalling of user consent, icons symbols and functions, taking into account the underlying ePrivacy legal framework. Following the forthcoming ePrivacy Regulation covering other forms of communications such as the over-the-top (OTT) technologies, defining and standardizing consent-giving across a broad spectrum of media platforms will be a matter to address in the context of standardisation.

ENISA will aim at further contributing in the discussions for online tracking prevention by supporting relevant projects and activities and co-operating with the main stakeholders in the field.

6. References

[1] "Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector (Directive on privacy and electronic communications)," [Online]. Available: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32002L0058&from=EN.

[2] "Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data," [Online]. Available: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:31995L0046&from=EN.

[3] "Directive 2009/136/EC of the European Parliament and of the Council of 25 November 2009 amending also Directive 2002/58/EC concerning the processing of personal data and the protection of privacy in the electronic communications sector," [Online]. Available: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32009L0136&from=EN.

[4] "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, repealing Directive 95/46/EC (GDPR)," [Online]. Available: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from=EN.

[5] E. Commission, "Evaluation and review of Directive 2002/58 on privacy and the electronic communication sector," [Online]. Available: https://ec.europa.eu/digital-single-market/en/news/evaluation-and-review-directive-200258-privacy-and-electronic-communication-sector.

[6] W. C. R. 1. O. 2017, "Tracking Preference Expression (DNT)," October 2017. [Online]. Available: https://www.w3.org/TR/tracking-dnt/.

[7] S. C. Q. M. L. T. a. C. J. H. A. Soltani, "Flash cookies and privacy," *AAAI Spring Symposium Series, Intelligent information privacy management,* pp. 158-163, 2010.

[8] T. K. a. D. W. F. Roesner, "Detecting and defending against third party tracking on the web," in *In NSDI'12 Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, San Jose, CA, April 25-27, 2012.

[9] "COOKIE SWEEP COMBINED ANALYSIS – REPORT," [Online]. Available: http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2015/wp229_en.pdf.

[10] A. 2. D. P. W. Party, “Working Document 02/2013 providing guidance on obtaining consent for cookies,” [Online]. Available: http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2013/wp208_en.pdf.

[11] “Web Tracking: Mechanisms, Implications and Defenses. IEEE “SAC 056 - SSAC Advisory on Impacts of Content Blocking via the Domain Name System, ICANN,” [Online]. Available: http://www.icann.org/en/groups/ssac/documents/sac-056-en.pdf.

[12] N. B. T. R. Dolière Francis, “Control What You Include! Server-Side Protection Against Third Party Web Tracking,” in *International Symposium on Engineering Secure Software and Systems, ESSoS*, 2017.

[13] I. Society, “Internet Society Perspectives on Domain Name System (DNS) Filtering,” [Online]. Available: http://www.internetsociety.org/sites/default/files/pdf/dns-filtering_20110915.pdf.

[14] “Intelligent Tracking Prevention,” WebKit.org, June 2017. [Online]. Available: https://webkit.org/blog/7675/intelligent-tracking-prevention/.

[15] M. G. M. West, “Same-site Cookies, Internet-Draft 6265,” April 2016. [Online]. Available: https://tools.ietf.org/html/draft-west-first-party-cookies-07.

[16] T. r. p. f. t. B. P. Commission, “Facebook Tracking Through Social Plug-ins,” June 2015. [Online]. Available: https://securehomes.esat.kuleuven.be/~gacar/fb_tracking/.

[17] R. U. a. others, “A classification of web browser fingerprinting techniques,” in *7th International Conference on New Technologies, Mobility and Security (NTMS)*, 2015.

[18] C. E. S. E. M. J. A. N. C. D. G. Acar, “The Web never forgets: Persistent tracking mechanisms in the wild,” in *In Proceedings of CCS 2014*, November 2014.

[19] “Privacy Extensions for Stateless Address Autoconfiguration in IPv6, RFC 4941,” [Online]. Available: https://tools.ietf.org/html/rfc4941.

[20] “DNT:1 extension for audience measurement (EU ePR),” [Online]. Available: https://github.com/w3c/dnt/issues/65.

[21] “The EU Internet Handbook, Cookies,” [Online]. Available: http://ec.europa.eu/ipg/basics/legal/cookies/index_en.htm.

[22] A. 2. D. P. W. Party, “Opinion 2/2010 on online behavioural advertising,” [Online]. Available: http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2010/wp171_en.pdf.

[23] W. W. Draft, “Content Security Policy Level 3,” September 2016. [Online]. Available: https://www.w3.org/TR/CSP3/.

[24] V. Toubiana, “Implementing cookie consent with “Content Security Policy”,” October 2014. [Online]. Available: https://unsearcher.org/enforcing-cookie-consent-with-content-security-policy.

[25] W3C, “Content Security Policy: Embedded Enforcement,” September 2017. [Online]. Available: https://w3c.github.io/webappsec-csp/embedded/.

[26] W3C, “Tracking Protection Working Group,” [Online]. Available: https://www.w3.org/2011/tracking-protection/.

[27] “"Guide for Sites" in the TPWG Wiki,” [Online]. Available: https://trackingprotection.github.io/Implementation/DNTGuide/.

[28] “Understanding EFF’s Do Not Track Policy,” [Online]. Available: https://www.eff.org/pages/understanding-effs-do-not-track-policy-universal-opt-out-tracking.

[29] B. S. R. v. E. L. U. Mike O'Neill, “Proposal to the TPWG, DNT 1.1,” October 2017. [Online]. Available: https://w3c.github.io/dnt/DNTBugs.html#status-representation-additions.

[30] B. S. Mike O'Neill, “Additional TSR properties supporting European Privacy & Data Protection Law,” Unofficial Draft 13 October 2017, [Online]. Available: https://w3c.github.io/dnt/drafts/Transparency.html.

[31] A. 2. D. P. W. Party, “Working Document 02/2013 providing guidance on obtaining consent for cookies,” [Online]. Available: http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2012/wp194_en.pdf.

[32] A. 2. D. P. W. Party, “Opinion 04/2012 on Cookie Consent Exemption,” [Online]. Available: http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2012/wp194_en.pdf.

[33] A. 2. D. P. W. Party, “Opinion 15/2011 on the definition of consent,” [Online]. Available: http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2011/wp187_en.pdf.

[34] B. Systems, “Discovered in the wild: a new method bypassing Safari's third-party cookie blocking,” January 2015. [Online]. Available: https://baycloud.com/thirdparty-redirect.

Annex A: A suggested schema for machine-readable information about storage use or the purposes for processing personal data

A suggested schema for how this information can be presented in a machine-readable way is contained in this Annex, consisting of:

- The list of first and third parties that are collecting data
- For which purposes these parties are processing these data

Similar schema has been proposed[46] to the W3C's Tracking Protection Working Group[47], and may be considered for inclusion in a future revision of the recommendation. Also under consideration is a proposal to include a reference to the purposes that a user has given its agreement for within the DNT header after the first "0" character, and an additional property for the API parameter dictionary for specifying it[48].

Information can be presented to the user-agent using the following JSON (JavaScript Object Notation) schema. The data should be machine-readable so it can be readily parsed by browsers or browser extensions. The information can be presented to the user in a standardised and language independent way, and data protection procedures can be followed should the user select them, or if they are enabled by default.

This could, for example, be provided as properties in the Do-Not-Track Status Object, or Tracking Status Resource returned by a GET request to the URI https://www.exampleco.com/.well-known/dnt/.

ITEMNAME	TYPE	DESCRIPTION
controllerIdentity	*ControllerIdentity* object	An object indicating the identity of the data controller, contact details, privacy policy, about page etc.
storageUses	Array of StorageUse objects	An array of objects indicating the types of terminal storage used, their purpose, duration etc.
otherParties	Array of otherParty objects	An array of objects indicating the domains of sub-resources that may be present on a site, and the controllers responsible for them. This lets website managers specify the domains they expect to be present, and supply the required identity information for them, if they do not yet present it themselves i.e. in the domains own Status Object.
legitimateInterests	String	A human readable string explaining the legitimate interests, if any, pursued by this controller

[46] Proposal to the TPWG 1st September 2016, https://w3c.github.io/dnt/DNTBugs.html#status-representation-additions
[47] W3C's Tracking Protection Working Group, https://www.w3.org/2011/tracking-protection/
[48] Issue 60 currently under consideration, https://github.com/w3c/dnt/issues/60

ControllerIdentity

ITEMNAME	TYPE	DESCRIPTION
name	String	The name of the controller or service provider
contacts	String	URI of a page showing contact details for the controller or service provider, their representative if applicable, their data protection officer etc.
about	String	URI of a human-readable web resource describing the controller or service provider
privacyPolicy	String	URI of a human-readable web resource describing the privacy policy
cookiesPolicy	String	URI of a human-readable web resource describing the cookies policy, if applicable
termsOfService	String	URI of a human-readable web resource describing the Terms of Service, if applicable

StorageUse

ITEMNAME	TYPE	DESCRIPTION
type	String	The type of user agent storage, one of: "c": an http cookie, "l": an item in localStorage, "s": an item in sessionStorage, "e": an item in the cache e.g. ETag "f": fingerprinting, i.e. any procedure used to derive an identifier from existing data "o": Any other storage in the user agent or device. This must be described in the **description** property.
description	String	A description of the storage use, optional but required if **type** or **purpose** is "o"
name	String	The name of the storage item, if applicable
purpose	Array	An array of Strings indicating all the purposes that the storage item is used for, which can any of: "a": advertising or commerce. Used for targeted or behavioural advertising, other commercial purposes or the analytics gathered to support them "f": functional. Solely used to persist state between HTTP transactions. Any data retained is generic in nature and will not be used to profile the user. "s": Analytics. Solely used to gather aggregated statistics on website usage. Any data retained is generic in nature and will not be used to profile the user. "l": login. Solely used for persisting an identifier for an authenticated, logged-in user. "o": any other purpose, which must be described in the **description** property.
duration	String	A string indicating the number of seconds before this item will be automatically deleted, or the string "session" if the item is automatically deleted at the end of a user session.

ITEMNAME	TYPE	DESCRIPTION
exemption	Array	If consent not needed, then one or more bases for the exemption. One of: "n": strictly necessary to fulfil a requested service, "t": solely used for carrying out the transmission of a communication "a": used for 1st party analytics (to be updated in light of changes to draft EPR)
sharedWith	Array	An array of *ControllerIdentity* objects indicating the other controllers with which this storage item is shared, if any.

OtherParty:

ITEMNAME	TYPE	DESCRIPTION
name	String	The domain name component of the sub-resource origin
type	String	The type of relationship this domain has to the first-party, one of: "s": same-party. This domain is managed by the first-party, "p": the controller of this domain is a contracted data processor for the first-party, i.e. it does not use any personal data collected for its own purposes. "n": this controller does not use terminal storage or collect personal data "o": Any other party.
controller	*ControllerIdentity* object	A *ControllerIdentity* object indicating the controller responsible for this domain, if known and is not the first-party, if it uses terminal storage or processes personal data, and if its identity is not specified by the domains own Status Object.
description	String	A description of the storage use, optional but required if **purpose** is "o"
purpose	Array	An array of Strings indicating all the purposes that the first-party controller is aware this domain uses terminal storage or processes personal data for, which can be any of: "a": advertising or commerce. Used for targeted or behavioural advertising, other commercial purposes or the analytics gathered to support it. "f": functional. Solely used to persist state between HTTP transactions. Any data retained is generic in nature and will not be used to profile the user. The **type** property can only be "s" or "p". "s": Analytics. Solely used to gather aggregated statistics on website usage. Any data retained is generic in nature and will not be used to profile the user. The **type** property can only be "s", "n" or "p". "l": login. Solely used for persisting an identifier for an authenticated, logged-in user. "o": any other purpose, which must be described in the **description** property.
exemption	Array	If consent is not needed, then one or more bases for the exemption. One of: "n": strictly necessary to fulfil a requested service, "t": solely used for carrying out the transmission of a communication "a": used for 1st party analytics (to be updated in light of changes to draft EPR)
sharedWith	Array	An array of *ControllerIdentity* objects indicating the other controllers with which this storage item is shared, if any.

Examples:

```
{
controllerIdentity:
        {
          "name": "ExampleCo Ltd.",
          "contacts": "https://www.exampleco.com/contact_us.htm",
          "about": "https://www.exampleco.com/about.htm",
          "cookiesPolicy": "https://www.exampleco.com/privacy.htm#cookies",
          "privacyPolicy": "https://www.exampleco.com/privacy.htm",
},
"storageUse":
         [
            {
                "type": "c",
                "name": "_ga",
                "purpose": "a",
                "duration": "63072000", // 2 years
                "sharedWith":
                {
                   "name": "Google Inc.",
                   "contacts": "https://www.google.com/contact/",
                   "about": "https://www.google.com/about/",
                   "cookiesPolicy": "https://www.google.com/policies/privacy/?hl=en#infocollect",
                   "privacyPolicy": "https://www.google.com/policies/privacy/?hl=en",
                },
               },
               {
                  "type": "c",
                  "name": "SESSION", // deleted when browser is unloaded.
                  "purpose": "f",
                 "duration": "session",
                },
                 {
                  "type": "c",
```

```
                "name": "piwik_uid",
                "purpose": "s",
                "exemption": "a",
                "duration": "86400", // 24 hours
            }
        ],
        "otherParties":
        [
            {
                "name": "cdn.jquery.com",
                "type": "n"
            },
            {
                "name": "service.exampleco.com",
                "type": "s"
            },
            {
                "name": "www.google-analytics.com",
                "type": "o",
                "controller":
                {
                    "name": "Google Inc.",
                    "contacts": "https://www.google.com/contact/",
                    "about": "https://www.google.com/about/",
                    "cookiesPolicy": "https://www.google.com/policies/privacy/?hl=en#infocollect",
                    "privacyPolicy": "https://www.google.com/policies/privacy/?hl=en",
                },
                "purpose": "a"
            },
        ]
}
```

Annex B: Privacy setting examples

In this Annex, an overview of different ways offered to the user to control his or her permission settings is given. Specifically, the interface offered for application permission settings, for browser settings and for an online setting aggregator is described.

B.1.1 Android app permission

Android application permission interface and default values have changed with the most recent version of the mobile operating system.

- Default

Since the sixth version of the operating system, most permissions are disabled by default. Applications that are pre-installed on Android may have been granted permissions that the users can later revoke, whereas applications installed by the user will not be granted permissions by default: the user will be informed about the requested permission upon installation and the application will be requesting the permission the first time it needs it (see Figure 4). When an installed application requests a permission, the user will be prompted in context and can decide to grant or reject the permission request. In some cases, the permission is necessary for the application to function properly but in some instances only specific features or advertising components will rely on the permission. In those cases, the access to the storage or computing capabilities of the device is clearly not necessary.

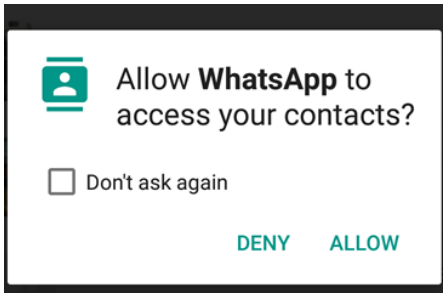


Figure 4: Prompting for a permission on Android

- Granularity

Permissions are organized by categories/families; the user can control the permission at the family level (see

Figure 5 Android App permission controls grouped by families

Figure 6 Detailed Android App permission information

) but can see them at a more granular level (see Figure 6). However, some data collections are not covered by the permissions system. For instance, being able to access to the Advertising ID does not require a permission, similarly accessing to the Wi-Fi MAC address of the terminal does not require to have user's permission (as shown on Figure 6). This setting controls the permission for the applications and covers both the application service provider and all the third parties it may include. With the current version of Android, an application developer cannot ask a permission for a specific purpose. If the permission is granted, the data could be shared with many third parties.

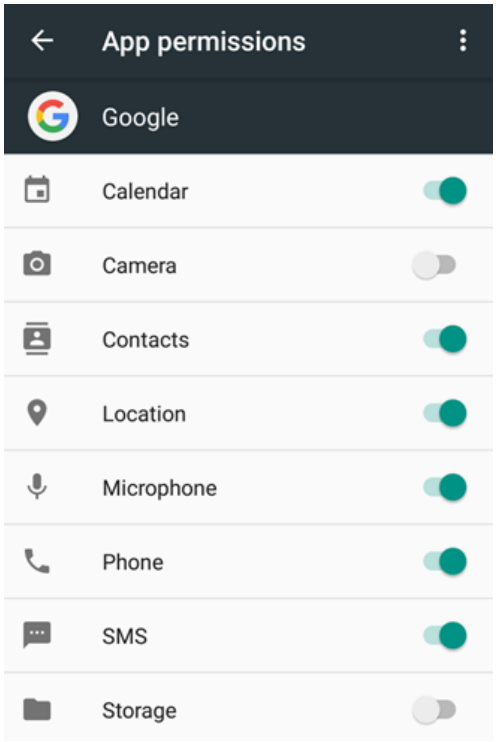


Figure 5 Android App permission controls grouped by families

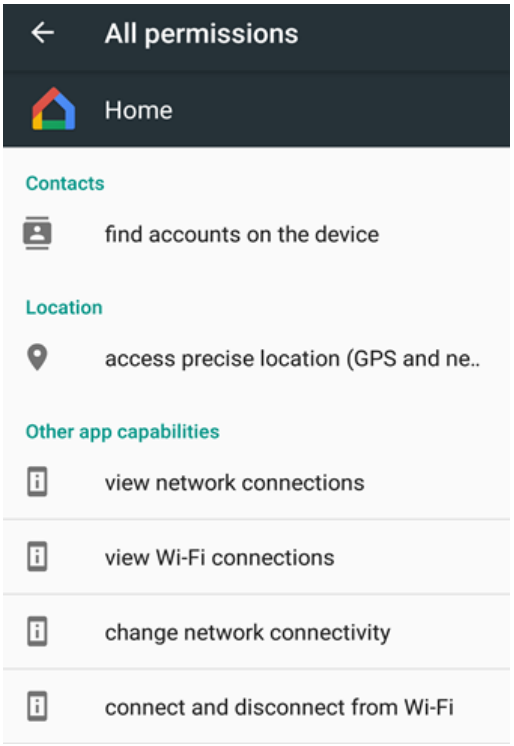


Figure 6 Detailed Android App permission information

- Revisiting the settings

Users can change the settings at any time and remove or grant permissions requested by the application. This is the only solution for user to not grant permissions to pre-installed applications as those applications will not request permissions by default.

B.1.2 Browser permission

Although they are not yet used to ask for consent to tracking, browsers permissions are already implemented and utilized to ask users if messages can be pushed or to ask access to their locations.

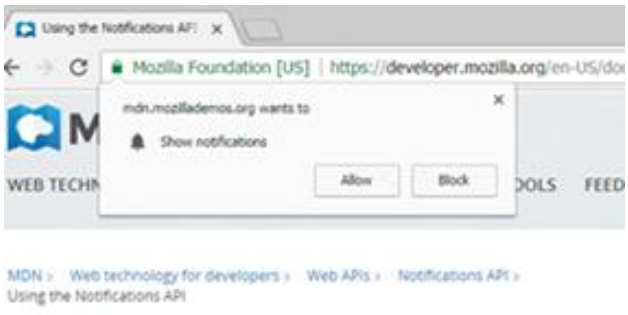


Figure 7: Chrome prompting for a "Notification" permission

- Default setting

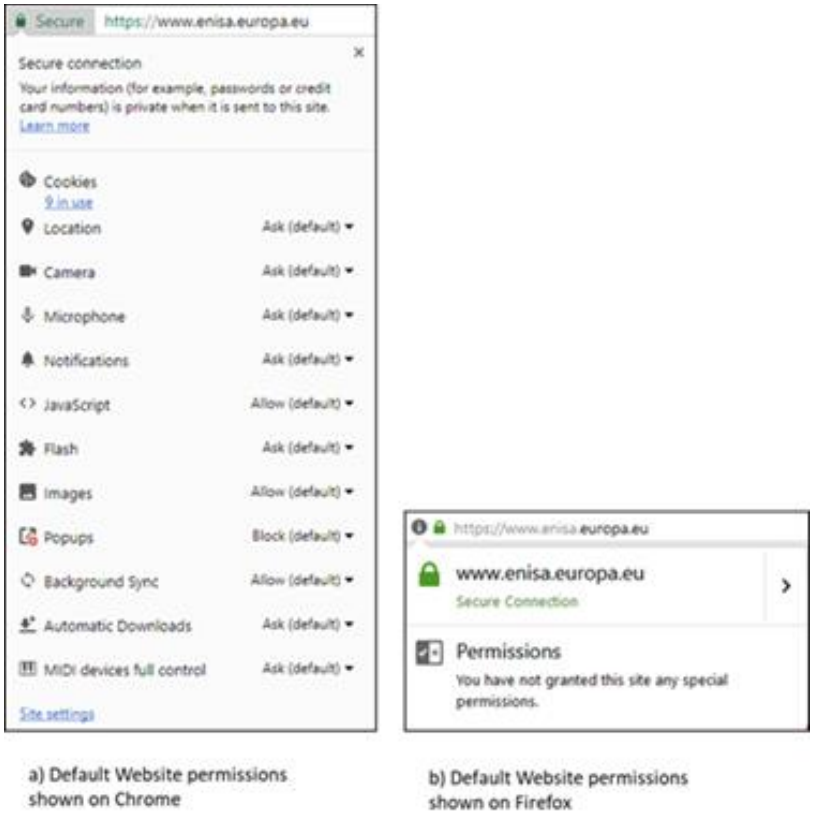
By default, permissions are not granted to first party and their third parties. Permissions have to be requested and the user will be shown a popup and asked for a choice when a website asks for the permission.

- Granularity

The setting is granted on a first party basis. The choice can be configured according to two parameters: the permission type and the first party. There is no way to the party requesting the permission to specify a purpose for the location tracking.

- Revisiting the setting

Settings are updated when the user grants or rejects a permission on a visited website. The user can change his or her decision by resisting the page. The permission model varies a bit with the different browsers.



a) Default Website permissions shown on Chrome

b) Default Website permissions shown on Firefox

Figure 8: Permissions shown on the browser

B.1.3 Online Setting aggregator

The advertising industry representatives have created a regional "opt-out" aggregator that centralizes opt-outs provided by advertisers.

- Default setting

By default, third parties will assume that there have been granted a consent to track the user on any website. Most tracking entities will impose this contractually on the first party to obtain consent before embedding the "tracking script" and therefore will assume that they can start to track as soon as they have been called. Therefore, by default third parties will track the browsers that are "calling" them.

- Granularity

Since this is organized by the advertising industry, the opt-outs only impact tracking for an advertising purpose. However, the granularity of the opt-out is not the same for all advertisers: some provided an opt-out to behavioural advertising while others provide an opt-out to tracking.

The opt-out can either be configured for all advertisers or only for a few of them.

- Revisiting the setting

The settings remain available on the advertising industry representative website. Service providers often provide a link to it in their cookie policy.

ENISA

European Union Agency for Network
and Information Security
Science and Technology Park of Crete (ITE)
Vassilika Vouton, 700 13, Heraklion, Greece

Athens Office

1 Vasilissis Sofias
Marousi 151 24, Attiki, Greece

PO Box 1309, 710 01 Heraklion, Greece
Tel: +30 28 14 40 9710
info@enisa.europa.eu
www.enisa.europa.eu