Privacy of Geolocation Implementations

Marcos Cáceres, Opera Software ASA,
marcosc@opera.com

License and disclaimer

Although I work for Opera Software, the position presented here is my own.

Abstract

This paper critically evaluates the privacy aspects of Web Browsers that implement the W3C's Geolocation Specification. This paper concludes by making a number of recommendations that may be applicable to browser vendors and standards-setting bodies, such as the W3C.

Introduction

There is a growing mountain of evidence [1][2][3][4][5][6][8] that some government and commercial institutions have exploited software, hardware, and laws to undermine individuals' fundamental human right to privacy (see Article 12 of [9] and Article 8 of [10] for declarations of right to privacy). In the worst cases, misuse of personal information has led to serious privacy violations: individuals unjustly find themselves deprived of access to goods, services, or even employment because an institution has violated their privacy – without any avenue for recourse [1]. Where privacy has been erroneously violated, as in the case with mistaken identity or human error, it can take months or even years for an individual to have the damage undone [1][2][3]. In more severe cases, individuals have found themselves incarcerated for exercising their human right of freedom of expression [8]. This as a direct result of commercial entities providing personally identifiable information to law enforcement agencies of countries that have little or no respect for human rights [8]. In some states, such as the United States, individuals have virtually no legal right to control, modify, or access data about themselves [1][2]; nor can they see who can access that data, or who that data is being sold to (and for what purpose) [1]. Bottom line is: because of the Internet our privacy and reputation is more at threat than at any other time in human history [2].

In his book, *Database Nation: The Death of Privacy in the 21st Century*, Garfinkel [1] warns:

> "Few engineers set out to build systems designed to crush privacy and autonomy, and few business or consumers would willingly use or purchase these systems if they understood the consequences. What happens more often is that the privacy implications of a new technology go unnoticed. Or if the privacy implications are considered, they are misunderstood. Or if they are understood correctly, errors are made in implementation. In practice, just a few mistakes can turn a system designed to protect personal information into one that destroys our secrets."

The roll-out of "advanced Web APIs", which is really an euphemism for APIs that access device capabilities and other personally identifiable data, will add more data that needs to be carefully protected by all those involved: the public, businesses, browser makers, standards setting bodies, operating system vendors, hardware manufacturers, and law makers. A balance needs to be struct between adhering with local and international laws, protecting the rights of individuals, and being able to conduct business across the globe [8].

Critical framework

In this paper, I critically evaluate the privacy aspects of a Web browsers that have implement the W3C's Geolocation Specification. By critically evaluating these browsers, I aim to ascertain a set of practical recommendations that engineers and standards bodies may use to protect the privacy of end-users.

In the book, *The Future of Reputation: Gossip, Rumor, and Privacy on the Internet*, Solove [2] argues that privacy can be evaluated in terms of *confidentiality*, *accessibility*, and *control*. I define these terms with the following questions:

Accessibility:
Are options and information pertaining to privacy accessible to the end-user?

Control:
Given the accessibility to privacy options, how much control does the system afford the end-user over privacy settings?

Confidentiality:
Does the system afford to the end-user anonymity or alternative means of protecting their privacy?

The Web Browsers evaluated in this paper are:

- Mobile Safari, on the Iphone OS 3.0.
- Chrome 6.0.422.0 dev, MacOS X.
- Firefox 3.6, on MacOS X.
- Opera 10.54 (developer preview), on Windows 7 – *it must be emphasized this version of Opera is a developer preview release and not a final shipping product. At the time of writing, no shipping version of the Opera browser supports geolocation.*

For each browser, I examine how confidentiality, accessibility, and control afford privacy to end-users.

Click to Confirm in Mobile Safari

Figure 1. A website viewed in Safari on iPhone 3.0 requesting to use the end-user's current location. But what will the data be used for? And where is the appplication getting the data from?

As can be seen in Figure 1, when a web page attempts to access geolocation services on Mobile Safari, the browser presents the end-user with a dialog that states "[URL] Would Like To Use Your Currrent Location" with two options: "Don't Allow" and "OK".

This "click to confirm" model suffers from a number of privacy issues: For one, the confirmation dialog does not give any indication to the end-user how their location is being derived: Is the location-provider the GPS? or is it the WIFI, or the cellular network, or a Web service? or a combination of those? and under what privacy policy does the location-provider provide that information? The iPhone provides no accessible means of viewing or changing the geolocation provider; hence an end-user has no control over the geolocation provider or even of knowing if their data is being encrypted on request.

Another privacy issue of Mobile Safari is that the confirmation prompts are *modal*: the user cannot fully view or interact with the underlying application to make an assessment of what the application might do with the positioning data, without first rejecting geolocation access to the website. Also, it is generally accepted that this kind of modal confirmation dialog lead to 'click fatigue': whereby users simply become accustomed to clicking "OK" to every prompt without grasping the consequences of their actions, and without having any real control over what personally identifiable data gets used, what it will be used for, or how long that data will be kept, or even if it will be made available (sold) to third parties. The privacy policies that govern geolocation services are buried three-levels deep in the "Settings" menu of the iPhone, under the "Legal" option, which contains about 50 pages of legalese and no searchable index!

Similar confirmation dialogs are found in the iPhone's native applications (e.g., the *Camera* and *Maps* applications). If a user changes their mind about allowing location services, there is usually no way for them to revoke geolocation access without either quitting the application, uninstalling the application, or finding some other convoluted way to revoke access to geolocation services (e.g., having to globally disable location services on the device through the "Settings" menu). What is worst is that once a user grants an application access to geolocation services three times, the system grants access to location services forever – or until the device is "reset", meaning resetting back to factory default settings. Applications that get granted access then do not generally provide an end-user with a means to revoke that access. This is also true on Mobile Safari: even after clearing the cache, history, and cookies Mobile Safari still grants websites access to geolocation without prompting the user.

In summary, Apple's Mobile Safari browser (and iPhone 3.0 in general) provides end-users with limited access to privacy controls. It also provides no means of seeing which Website have access to geolocation, nor once granted can that access be easily revoked by an end-user. The OS, however, provides means of achieving confidentiality by allowing the end-user to globally disable location services, WIFI, and cell-tower communication (via "Airplane Mode").

Information Bars and policies

On the desktop, Firefox, Opera, and Chrome move away from the "click to confirm" model by instead using an "information bar": when a Web site attempts to access geolocation services, a bar slides down from the top of the browser that allows the end-user to either grant or deny access to geolocation services. Examples of an information bar can be seen in Figure 2 and Figure 4. The information bar is *non-modal* in that blocks the website from accessing geolocation services, but does not block the user from interacting with the website. By having a chance to explore the website before enabling geolocation services, an end-user may get a better idea of what their location may be used for (and perhaps even have chance to read the privacy policy of the website). In terms of accessibility and control over privacy, this is certainly a step forward from Mobile Safari's modal confirmation dialog.

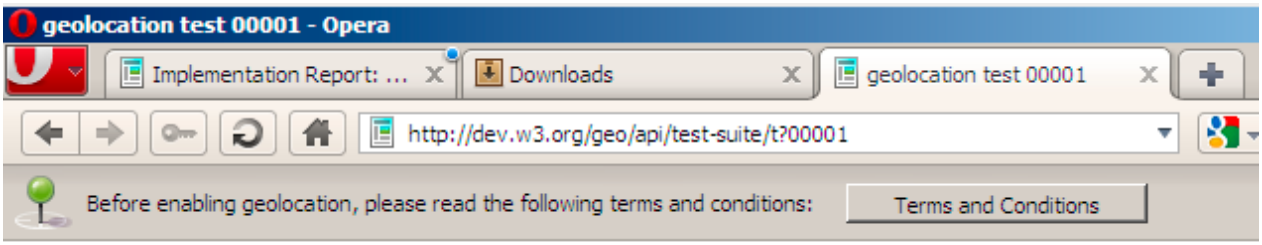


Figure 2. Opera's "Terms and Conditions" for using geolocation services.

The three desktop browsers do not behave the same way the first time a website requests access to geolocation services. Opera, on the one hand, first asks the end-user to read and confirm the "Terms and Conditions" under which geolocation services are provided by Opera and the location provider (in this case, Google). The presented "Terms and Condition", which can be seen in Figure 3, are not written in legalese; but rather are written in plain English to help end-users understand the privacy implications of enabling geolocation. However, I would argue that because of the "Terms and Conditions" window has a End-User-License-Agreement (EULA) look-and-feel, it is unlikely many users will read it (perhaps a consequence of 'click fatigue'). Another issue is that once the user clicks "Accept", there seems to be no way of re-accessing the "Terms and Conditions" window. Opera should provide a means for end-users to access the "Terms and Conditions" after they have clicked "Accept".

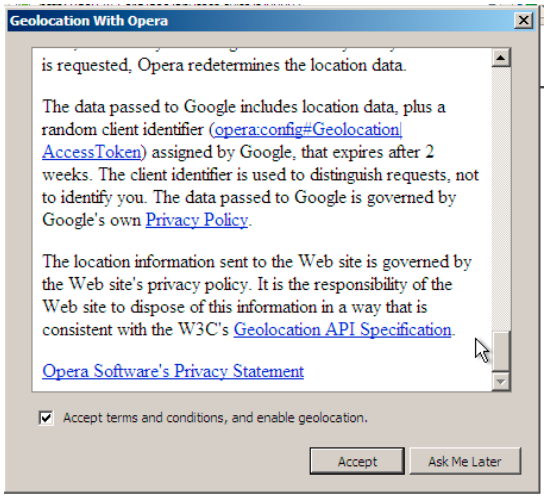


Figure 3. Opera's "Terms and Conditions" for using geo-location services.

Firefox and Chrome, on the other hand, don't present any such "Terms and Conditions", but instead provide a "Learn More…" link (seen in Figure 4). Clicking on the "Learn More…" link leads to an information page about location services that identify the location-provider, which is again Google, and links to various privacy policies. Unlike Opera's "Terms and Conditions" window, the "Learn More…" link is part every request to access geolocation – a good thing from a privacy perspective.

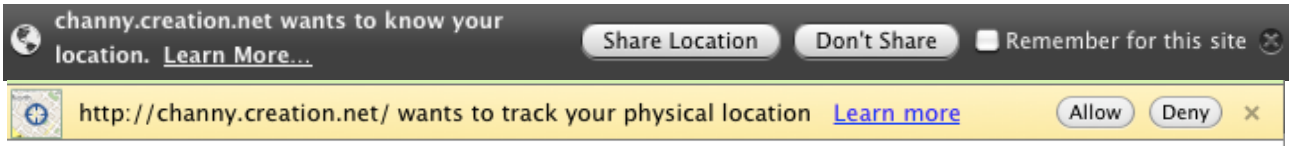
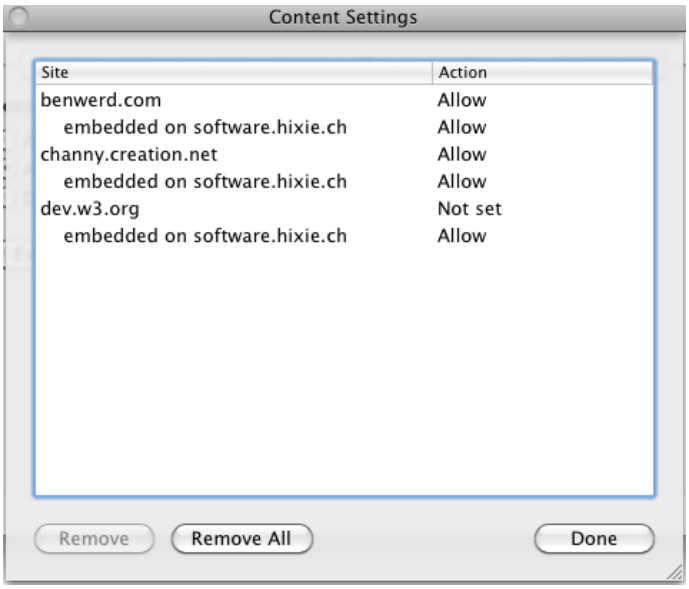


Figure 4. The 'Learn More...' links of Firefox (above) and Chrome (below) encourages users to go and find out more about location services. However, there is little evidence that people do go to find out more or understand the privacy implications of enabling geolocation.

Despite the similarities between Firefox and Chrome, there are two shortcomings to the way Firefox allows the user to control a website's access to geolocation services. The first shortcoming being that, in order to revoke a websites access to geolocation services, the end-user needs to go through a non-obvious multi-step process: according to Mozilla's geolocation information page, the end-user must "1. navigate to the site to which you've given permission, 2. go to the Tools menu, then select Page Info. 3. Select the Permissions tab. 4. Change the setting for Share Location". In other words, there is no accesible centralized user interface where an end-user can view the sites that have been granted access to geolocation services. Because the end-user must first return to the website that was originally granted access, they risk exposing their location. I'll note that Opera also suffers from this problem: there is no central place to view which websites have acces to geolocation services. And access must be revoked on a website-by-website basis.

Chrome, however, does provide access to a central place to view which websites have acces to geolocation services, with the added advantage of seeing if geolocation service requests were made by websites or embedded components (iframes) within a website.



The second shortcoming being that, upon returning to a website, there is no way for the end-user to know if, or when, the website is accessing their geolocation information. In other words, there could potentially be nothing on a Web page to indicate that geolocation services are being used by the webpage. Indicators is subject of discussion in the next section.

Figure 5. With two clicks, Chrome allows the end-user to see and remove all websites that have acces to geolocation.

Lastly, a final privacy issue with Opera is that it fails to identify which website (origin) is requesting geolocation access (see Figure 6): it simply says “The Web site is requesting your location.” The problem being that websites can contain other websites via iframes. As a consequence, an attacker can embed a HTML iframe and trick the user into thinking that the site they are looking at is the one requesting their location. For this reason, the W3C’s Geolocation Specification [7] mandates in the strongest terms that “The user interface must include the URI of the document origin”. Opera is aware of this problem will fix it in their next release.

Figure 6. Opera does not display the address of the website that is requesting access to geolocation services. This violates the W3C's Geolocation Specification's privacy conformance requirements [7

Activity Indicators and Controls

As discussed in the previous section, Firefox lacked any means of indicating via the user interface that geolocation was actively being used by a website. In order to let the user know if/when a website is accessing their geolocation services, Opera and Chrome display an indicator in the address bar, which can be seen in Figure 7. It must be noted that the icons differ stylistically between the two browsers, though they are used consistently throughout each browser. They also differ in the information they provide when clicked.

Figure 7. Indicators of chrome and Opera showing active geo-location.

In both browsers, the geolocation indicator serves as a means to access the geolocation controls for a website. By clicking on the icon, the end-user can disable geolocation services, as well as revoke the website’s access to geolocation services in the future.

Location Providers

The location provider is the means by which the browser derives an end-user’s geographical position. In the three desktop browsers that I have discussed in this paper, the provider has been a third-party: Google. From the end-user’s perpective, being able to access information about the provider via some user interface allows the end-user to know who the provider is, and how the location requests to the provider are being made. That is, is the provider someone the end-user can trust and is the communication channel with the provider secured?

Firstly, Google Chrome does not provide any user interface for accessing or controlling the provider. Although Firefox provides a means to access and control the default location provider, it the makes the process particularly unapproachable to end-user. Firstly, it requires the end-user to access the ‘about:config’ browser configuration page, which jokingly warms the user that "This may void your warranty". And then requires the end-user to manually change the values of key-value pairs seen in Figure 8:

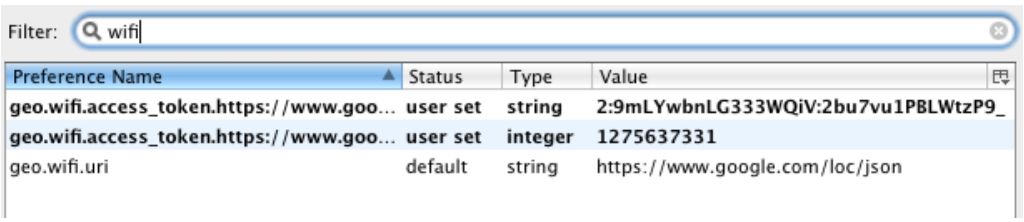


Figure 8. Firefox's somewhat unfriendly way of accessing and controlling the geolocation provider.

Although also requiring the end-user to know about a browser the configuration page ("opera:config"), Opera provides a more accessible means of controlling the providers, when compared to Firefox, which can be seen in Figure 9.

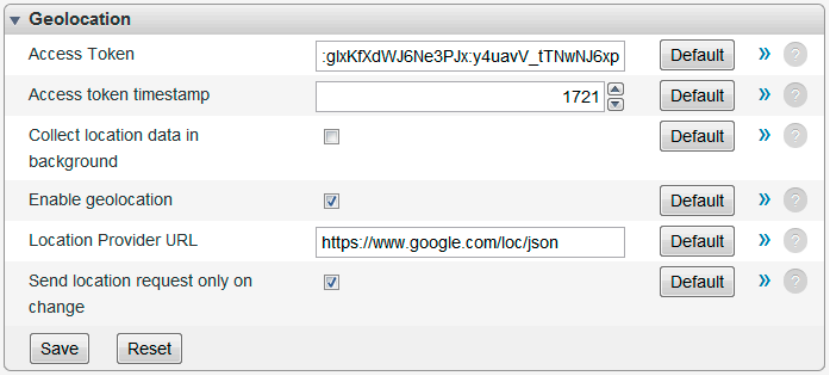


Figure 9. Opera's somewhat more friendly way of accessing and controlling the location provider.

Note that there is a disconnect between the location provider's settings and the privacy policy for the provider.

Final remarks and recommendations

As there is no international law that fully protect individuals, or individuals may be subject to laws that violate their rights, due diligence must be taken by engineers to protect end-users on multiple levels. Those protections must be immutable in the API, and, if standardized, part of the core set of conformance requirements of a specification.

Having said that, and based on all I've read on this subject, I make the following recommendations:

1. Browsers must avoid confirmation dialogs that don't allow users to make informed decisions, such as those found on Apple's iPhone 3.0.
2. Digital signatures must be only used to verify data integrity and verify continuity of authorship, and should not exclusively be used as a means of enabling APIs.
3. Where device capabilities are required by applications, those capabilities must be declared by the author up-front. Where capabilities are declared and potentially affect a user's privacy, the system must provide a user interface to view which capabilities will be used by an application.
4. As exemplified by [7], and where necessary, standards bodies must mandate that conforming user agents provide user interfaces that give end-users access and control over their privacy.
5. Where it makes sense, it must be possible to change providers – particularly third-party providers. And where the provider is a third party, the provider's privacy policy must be accessible and intrinsically linked in the user interface.
6. Communication between the provider and the client must be encrypted, particularly in the case of third-party providers.
7. Browser vendors must work together to reach *de facto* or *de jure* standardization of iconic activity indicators (as has effectively been achieved with the padlock indicator for secure communication).
8. Policy and law makers should work with the public to come up with more accessible privacy policies (i.e., like the creative commons icons, but for privacy policies).
9. When it comes to storing privacy choices that an end-user has made, such as which provider to use, those choices should be treated as sensitive data and appropriately secured via, for example, cryptographic means. End user's should be informed of any attempt by a third party to temper with their choices.

References

[1] Garfinkel, S., *Database Nation: The Death of Privacy in the 21st Century*. *2000*.

[2] Solove, D. J. *The Future of Reputation: Gossip, Rumor, and Privacy on the Internet*. 2007 .

[3] Zittrain, J., *The Future of the Internet–And How to Stop It*. 2009.

[4]
O'Hara, K. and Shadbolt, N. *The Spy in the Coffee Machine: the End of Privacy as We Know It*. 2008.
[5]
Lessig, Lawrence. *Code and other laws of Cyberspace*. 1999.
[6]
Schneier, B. *Secrets & Lies: Digital Security in a Networked world*. 2000.
[7]
Popescu, Andrei (Ed.). *Geolocation API Specification*, W3C Working Draft, 07 July, 2009.
[8]
Goldsmith, J. and Wu, T. *Who Controls the Internet: Illusions of a Borderless World*.. 2006.
[9]
Universal Declaration of Human Rights. ohchr.org.
[10]
The European Convention on Human Rights and its Five Protocols. 4 November, 1950.