Simple Policy Negotiation for Location Disclosure

Nick Doty
School of Information
UC Berkeley
npdoty@ischool.berkeley.edu

Erik Wilde
School of Information
UC Berkeley
dret@berkeley.edu

ABSTRACT

Relying on non-enforceable normative language to persuade Web sites to make their privacy practices clear has proven unsuccessful, and where privacy policies are present, they are notoriously unclear and unread. Various machine-readable techniques have been proposed to address this problem, but many have suffered from practical difficulties. We propose a simple standard for transmitting policy information just-in-time and enabling simple negotiation between the site and the user agent. In particular, we detail how this could improve privacy of the *W3C Geolocation API*, but also suggest the possibility of extension to other application areas in need of privacy and policy negotiations.

1. PROBLEMS

1.1 Current Situation

Web sites that use the W3C Geolocation API rarely, if ever, follow the privacy practices required by the specification for disclosing their data usage practices to the user [4]. More broadly, privacy policies and the "notice and consent" model on the Web are widely considered unsatisfactory,[1] prompting uncertainty in the marketplace and the threat of varied and burdensome government intervention.[2] As the Web begins to include more advanced APIs with access to more sensitive information (location, address book, camera, etc.), the existing method of out-of-the-way privacy policies will only become less acceptable.

1.2 Machine-readable Approaches

In response to these concerns, various machine-readable approaches to describing privacy policies have been proposed over many years. In 1997, the W3C launched the *Platform for Privacy Preferences (P3P)* [2] Project and P3P was promoted to a W3C recommendation in 2002. However, the standard for describing the full contents of privacy policies in machine-readable metadata never reached widespread use, plagued by the complexity of its vocabulary, resistance from companies who did not want a straightforward description of their privacy practices and a lack of browser implementations [12]. The Mozilla-led *Privacy Icon Project* [11] provides a similar (albeit simpler) approach based on laundry label-style iconography of a Web site's policy practices, but some have argued that it will suffer the same fate as P3P due to the similarity of high-level practices in the marketplace.[3]

Alternatively, the IETF-driven *GeoPriv* [3] attempts to reverse this approach by putting privacy policies in the hands of users instead of services: a user transmits her own privacy preferences about how data is used with the data itself, while Web sites are bound by their market or legal obligations to respect those preferences. After extensive debate, the GeoPriv proposal to the W3C Geolocation Working Group was voted down. Opponents thought the proposal too complex for Web developers to realistically implement — and in our research many uses of the Geolocation API have been simplistic — and that automatically sending a standard set of preferences was too inflexible for the variety of use cases — how should a site purely for location-sharing handle location data with the attached policy that it not be shared with anyone?

Objections to the GeoPriv and other proposals also included concerns that the browser should not ever make promises on behalf of a Web site. For reasons we will not elaborate on here,[4] we never found this objection compelling. Given Mozilla's interest in Privacy Icons it also appears that at least some browser makers would accept policy hooks that are ultimately enforced by the market or by regulation rather than by technology itself.

We are heartened by the renewed attention in this area and to the breadth of proposals put forward within DAP and previous workshops. We believe combining aspects of the user selectivity control proposed by DAP for selecting contacts and the expression of policy in Policy Rulesets can allow for true negotiation of policy: privacy or otherwise.

2. NEGOTIATION FRAMEWORK

We propose a standard for real-time negotiation of policies between Web sites and users. First, requesting Web sites specify the acceptable range of policy conditions that apply to their use case; second, users choose from the options available (or instruct their user agents to choose automatically on their behalf); finally different amounts of data at varying levels of precision is returned by the user agent with the chosen policy requirements attached. Below we show

[1] `http://nyti.ms/auyhoZ`
[2] See, for example, the EU Article 29 regulations and draft Congressional legislation in the US.

[3] `http://www.w3.org/2010/api-privacy-ws/report.html`
[4] See the CDT's response to arguments against the binding rules approach and UC Berkeley's argument for privacy hooks, both at the July W3C workshop on privacy.

W3C Workshop on Privacy and Data Usage Control, October 2010, MIT.

one possible implementation of this model for Geolocation and privacy, but the model can be extended to other APIs and other policy objectives beyond privacy.

For Geolocation, fields are chosen to support privacy in the dimensions of: *minimization*, *user control*, *notice*, *consent*, *secondary use*, *distribution*, and *retention* [4]. The particular fields and values are drawn from GeoPriv, Privacy Rulesets, and other proposals discussed within the Geolocation Working Group and at the July W3C privacy workshop [15], but those details may be debated or changed without fundamentally altering the proposal.

- `precision`: Specific to geolocation, the geographic precision of the data varies for different use cases. The values of `exact`, `street`, `city`, and `country` may be allowed.
- `sharing`: Values of `internal`, `affiliates`, `unrelated-companies`, and `public` are allowed, with their meanings as defined in the Privacy Rulesets proposal [1].
- `retention`: Also drawn from the Privacy Rulesets proposal, values of `no`, `short`, and `long` are allowed.
- `usage`: A short, human-readable string to explain the site's usage of location information. This is simply a disclosure, not a negotiated field. (Given its lack of constraints and machine-accessible structure, we recognize that there may be some debate over whether to include this field at all.)
- `policyUrl`: Another disclosure (non-negotiated) field: the URL to the full privacy policy describing in greater detail the use of location information. Web sites are encouraged to use links to direct the user to the relevant part of a longer policy page.

For each negotiated field, the requesting Web site provides an array of acceptable options for its use case. For example, a location-sharing site with the express purpose of publishing exact locations to the world will only provide `['exact']` and `['public']` for the `precision` and `sharing` fields. Conversely, a mapping site could accept several levels of precision for centering the map in the user's area (`precision: ['exact', 'street', 'city']`) and different levels of retention for remembering (or not) the user's position for the next visit (`retention: ['no', 'short']`). (For a full sample function call for the mapping site, see Figure 1, which might be rendered as in Figure 2.)

```
navigator.geolocation.getCurrentPosition({
  precision: ['exact', 'street', 'city'],
  retention: ['no', 'short'],
  sharing: ['internal'],
  usage: 'center the map near you',
  policyUrl: 'http://example.com/privacy#location'
}, successCallback, errorCallback);
```

Figure 1: Sample position call for a mapping Web site using negotiated ranges

In order to improve on the current situation, policy fields (in this case, at least `precision`, `retention`, `sharing` and `usage`) must be mandatory — otherwise, as it is now, most

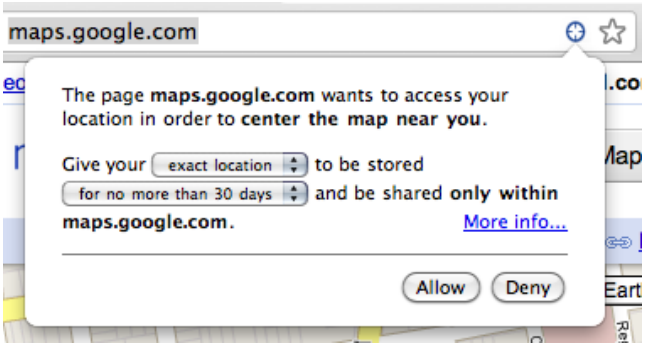

Figure 2: Sample user interface for negotiation.

Web sites will only have a disincentive to provide this policy information up-front if competitor sites don't disclose their practices. For other scenarios it's an interesting open question whether all negotiated fields need to be mandatory, or what semantics are implied by an omitted field; for further discussion of the meta-model, see Section 3.

Although a site might specify only a single value for each field, sites are encouraged to accept the widest possible range that makes sense, enabling their site for users with different privacy preferences. This is much the way that web developers are recommended to enable graceful degradation (or, if you prefer, progressive enhancement) for clients that may not all have the same technical capabilities. For sites that do not provide ranges ("take-it-or-leave-it"), the proposal degrades to a situation where the site notifies the user in a visible way of their existing, non-negotiable privacy policy — still an improvement over the current situation.

Response objects include an additional `policy` property on the existing `position` object with the user's choice for each negotiated field; see Figure 3 for an example response. The response will be one of the options that the developer specifies in the function call, so no unexpected or indecipherable response is possible in the normally functioning case.

```
{
    coords: {lat: 37, lon: -122},
    policy: {
        precision: 'exact',
        retention: 'no',
        sharing: 'internal'
    }
}
```

Figure 3: Sample position response object with a user's policy choices.

While an API call often will specify sets of values (or ranges of values) as shown in Figure 1, the result of a successfully negotiated policy will always only contain single values, representing the policy agreed to by the user (agent) and the actual data that is bound to this policy.

2.1 Advantages

This is certainly not the first proposal for negotiation of privacy (several additions were proposed for negotiations on top of P3P [8, 9], for example) nor the first (or last, we sus-

pect) proposal for handling privacy in Web APIs with access to sensitive information, like Geolocation. Nevertheless, we believe that this mechanism for negotiation has the following advantages over alternatives:

1. *Just-in-time*: As opposed to site-wide descriptions of practices (P3P and Privacy Icons), policy negotiation happens around particular pieces of sensitive information, at the moment when they are being shared. Users, therefore, should be more directly aware of the implications involved.

2. *User-controlled, but with site-specified options*: Users have the ultimate say on the use of their information (as in GeoPriv), but sites can specify which range of options make sense for their use case. Sites, therefore, never have a reason to ignore attached preferences and no information is sent in cases where an acceptable policy has not been negotiated.

3. *Non-repudiation*: Recipients cannot argue after the fact that they did not know the user's expectations for retention or use of information; this enables market and regulatory forces to punish bad actors. Similarly, users cannot claim after the fact that the site was deceptive or that they had not been informed.

4. *Ease-of-use*: Since many Web developers who use the Geolocation API are neither sophisticated users of JavaScript nor experienced computer programmers, this approach requires only simple parameters (arrays of strings, with straightforward documentation of the enumerated options) and no parsing of XML or RDF in responses (as would be required with P3P, GeoPriv, XACML, or various Semantic Web approaches).

Requiring sites to specify each field can act as a forcing function for Web developers to consider these policy issues and simultaneously saves them the time spent designing a policy disclosure of their own (which, as we have seen, few are willing to do). For backwards compatibility, sites that use `navigator.geolocation` without specifying policy can prompt a warning in the browser before continuing, encouraging Web sites to enable policy disclosure and negotiation without breaking existing functionality on the Web.

3. EXTENSIBILITY

While the approach we have presented thus far is specifically designed to support privacy policy negotiation for location information, it may be applied across a wider variety of use cases. It could either be extended to cover *privacy policies* regardless of the nature of the information, or it could be even extended to cover *policy negotiation* in general.

The W3C's *Device APIs and Policy (DAP) Working Group* is considering a variety of APIs that will provide access to privacy-sensitive information and services on the local platform, such as the contacts database, the personal calendar, the to-do list, the camera, microphone, messaging functionality, even the local file system. Clearly, all these scenarios must have well-defined ways of controlling access and dealing with privacy issues, and we believe that the framework we describe in Section 2 can be generalized and extended to cover the greater challenges that arise for a whole landscape of sensitive APIs.

We do not believe it is necessary that the framework be designed and implemented such that it dynamically adapts to new use cases, but we do believe that a unified model across a variety of APIs and services would make it easier for developers to understand the policy framework, use it, and reuse code that works with it. In order to reuse the framework across scenarios, some parameters may need to be generalized, most notably the information that is exchanged. The best way to proceed might be to identify a set of data types that are supported, such as ordered and unordered lists of values, numeric ranges, date or time ranges, etc. These data types would be best determined by looking at the requirements of policies across a variety of scenarios, perhaps all of the APIs currently under consideration by the DAP Working Group.

For platforms supporting a variety of APIs, it should be possible to reuse considerable amounts of code for dealing with (privacy) policies if the underlying framework is consistent. Furthermore, the W3C *Web Notifications Working Group* that just started[5] will work on a framework on how to use platform-specific notification mechanisms from within applications, and basing user interactions on such a mechanism (even though it would be up to the platform on how to expose policy negotiations to the user) might further improve the user experience and ease of use of the proposed framework.

In addition to contributing to harmonization across APIs, a standardized configuration file format could allow sharing user configurations (which sites they trust with what data under what conditions) with friends; trusted sources (be it governments, browser makers, privacy advocates or colleagues) could publish recommended configurations. Configuration can also be shared with other devices. This is particularly powerful in scenarios including interfaceless devices[6] that could make decisions based on a user's previously configured preferences. For example, for navigation devices (for cars and bicycles) it might be much easier to set the privacy preferences externally, so that the devices do not need to expose any UI elements. They could still respect a user's privacy settings, so that the navigation device is allowed to use full GPS resolution internally, but is only allowed to expose city-level resolution to external services, still good enough to receive traffic alerts.

3.1 Other Sample Use Cases

Though the Geolocation API and location privacy are a good candidate for the sort of negotiated policy approach described in Section 2, we believe the same framework could be extended to other use cases. We list a few possibilities below, for the sake of illustration.

- *Contacts:* Working Drafts of the Contacts API [13] already provide for selective user control over which fields in the address book will be made available to the requester, similar to the data minimization proposed here for geolocation precision. But given the

[5]A very early draft [5] of the API has been made available but will likely change substantially before getting close to finalization.

[6]A scenario increasingly likely to become reality with the *Web of Things* [6] and similar approaches working towards creating a more intricate connection between the information-oriented world of the Web and the physical world.

sensitivity of revealing personal contact information, users may want to specify further privacy restrictions: limitations on retention and sharing to be sure, but perhaps also limitations on whether the information can be used by the recipient to contact the subjects. The exact set of policy negotiations in this case probably will be more complex than in the case of geolocation, but this is almost unavoidable given the more complex structure of a user's contacts database.

- *Web Storage:* A good example of policy not directly about privacy, the Web Storage API [7] currently contains suggestions about disk space limitations,[7] proposing a "mostly arbitrary limit of five megabytes per origin". Since different sites may have legitimately different storage requirements based on their use case, and users of mobile devices may wish to control their disk usage, negotiation may be useful.[8] Sites may provide a recommendation or request some range of disk space accompanied by a usage explanation; users can ultimately decide the amount from a slider UI or some pre-set configuration.

- *Media licensing:* Sites that ask users to upload pictures, video, or audio of their own creation may benefit from policy negotiation to suggest licenses that fit their use case (Wikipedia requiring certain accessible licenses and photo-sharing sites allowing users to assert full copyright) and accept embedded or attached licenses from the user agent along with the media file. This kind of policy might be interesting for applications using media uploads (through regular forms or through file system access [10]) or using direct access to media capture capabilities of a device [14].

These are not intended to be concrete proposals, particularly given our lack of expertise with the particular APIs involved, but should give some idea of how the proposed negotiation framework could be extended to other APIs and other policy objectives.

4. CONCLUSIONS

Given that the current disclosures of privacy policies on the Web are consistently out-of-the-way and unread and the challenges faced by adoption of machine-readable approaches towards a Policy Aware Web, we believe a simple model for negotiating policy that can be built into new Web APIs can dramatically increase support of privacy on the Web. Further, though, we hope that a similar model for negotiation could apply to other policy issues, be it copyright or disk usage. Important questions remain both about the meta-model (what types of issues can be negotiated in this way) and about which policy issues apply to any particular API or scenario: questions we believe can be answered in the DAP and API-specific Working Groups.

Our goal at the workshop will be to gather the group's feedback on this proposal (and ones like it) both to see how a concrete negotiation proposal might be brought to the Geolocation Working Group and to investigate whether a more general framework like this one could be designed within the DAP Working Group.

[7] `http://www.w3.org/TR/webstorage/#disk-space`

[8] Imagine a Web-based mapping service that would like to store maps for offline usage on a user's computer. These images will likely take up much more space than five megabytes, but given the value of having this information available even in offline mode users might be willing to prove substantial storage space to this kind of application.

5. REFERENCES

[1] Alissa Cooper, John B. Morris, and Erica Newland. Privacy Rulesets: A User-Empowering Approach to Privacy on the Web. In W3C Workshop on Privacy for Advanced Web APIs [15].

[2] Lorrie Faith Cranor, Marc Langheinrich, Massimo Marchiori, Martin Presler-Marshall, and Joseph M. Reagle. The Platform for Privacy Preferences 1.0 (P3P1.0) Specification. World Wide Web Consortium, Recommendation REC-P3P-20020416, April 2002.

[3] Jorge R. Cuellar, John B. Morris, Deirdre K. Mulligan, Jon Peterson, and James M. Polk. Geopriv Requirements. Internet RFC 3693, February 2004.

[4] Nick Doty, Deirdre K. Mulligan, and Erik Wilde. Privacy Issues of the W3C Geolocation API. Technical Report 2010-038, School of Information, UC Berkeley, Berkeley, California, February 2010.

[5] John Gregg. Web Notifications. World Wide Web Consortium, Editor's Draft, June 2010.

[6] Dominique Guinard and Vlad Trifa. Towards the Web of Things: Web Mashups for Embedded Devices. In *Second Workshop on Mashups, Enterprise Mashups and Lightweight Composition on the Web*, Madrid, Spain, April 2009.

[7] Ian Hickson. Web Storage. World Wide Web Consortium, Working Draft WD-webstorage-20091222, December 2009.

[8] M. Maaser, S. Ortmann, and P. Langendörfer. NEPP: Negotiation Enhancements for Privacy Policies. In *W3C Workshop on Languages for Privacy Policy Negotiation and Semantics-Driven Enforcement*, 2006.

[9] S. Preibusch. Privacy Negotiations with P3P. In *W3C Workshop on Languages for Privacy Policy Negotiation and Semantics-Driven Enforcement*, 2006.

[10] Arun Ranganathan. File API. World Wide Web Consortium, Working Draft WD-FileAPI-20091117, November 2009.

[11] Aza Raskin and Arun Ranganathan. Privacy: A Pictographic Approach. In W3C Workshop on Privacy for Advanced Web APIs [15].

[12] Ari Schwartz. Looking Back at P3P: Lessons for the Future, November 2009.

[13] Richard Tibbett. The Contacts API. World Wide Web Consortium, Working Draft WD-contacts-api-20100817, August 2010.

[14] Dzung D. Tran, Ilkka Oksanen, and Ingmar Kliche. The Capture API. World Wide Web Consortium, Working Draft WD-capture-api-20100401, April 2010.

[15] *W3C Workshop on Privacy for Advanced Web APIs*, London, UK, July 2010.