Promoting Privacy by Developing Real-Time Web Standards with “Both Screens” in Mind

Alissa Cooper
John Morris
Center for Democracy & Technology

September 19, 2010

I. Introduction

Many applications that were once only available as installable desktop software, including email and word processing applications, have already been available for a number of years as web-based applications. As these web-based versions were being developed, use of web browsers and web technologies was more confined to the desktop environment than it is today. With massive growth in the adoption of cellular data connections and smartphones [9], the application development environment is far more heterogenous than it once was [8]. Designers of web-based applications can no longer expect their users to be sitting in front of a sizable laptop or desktop PC screen. Instead, the development of new web-based applications, including real-time video and voice applications, demands a design vision that accounts for both desktop- and mobile-based usage. Developing for “both screens” has important implications for a number of design and standardization issues, including privacy.

II. Privacy differences between desktop and mobile

Transparency and user control are two core components of systems designed to afford privacy protection. The primary ways that these components manifest themselves in the browser context is through user interface features and user preferences. For example, when a browser UI displays a “lock” icon in the URL bar to signal the use of SSL, or when it changes the appearance of the chrome to indicate that a private browsing mode is active, it is providing visual cues that inform the user about the privacy implications of his or her activity. When the browser provides an interface that allows cookies to be removed or blocked, or when it allows the user to clear the browsing history, it is providing preference mechanisms that give users some control over information about their browsing. Transparency and user control are but two of a larger set of components required to ensure the privacy-protective deployment of web technologies, which include restrictions on data use and sharing, accountability mechanisms, and a number of other components [11]. But much of how users perceive the privacy experience afforded by an application is determined by the transparency and control features that it offers.

The capabilities of a small mobile device to support transparency and control are vastly different than those of a full-sized laptop or desktop computer. The smaller screens and smaller range of input devices (fingers and tiny keys instead of mice and large keyboards) mean that user interface features are harder to see and user controls are harder to manipulate. The more challenging mobile form factor has translated into reduced transparency and control in the mobile setting [5]; for example, trying to delete individual cookies or read the information from an extended validation certificate is either difficult or impossible on many mobile devices.

At the same time, for many users mobile devices are both more personal and more uniformly integrated than desktops or laptops. They store address books, call logs, and photos snapped impulsively. They are equipped with a suite of sensors – cameras, microphones, thermometers, multiple network interfaces – that, when combined, can provide intimate details about the owner's whereabouts and behavior [1]. They are often furnished out-of-the-box with calendars and address books that share integrated files and data management systems. And their small size means that they travel with people at all times, from professional to social to personal settings. While full-size laptops and desktops may perform some subset of these functions, mobile devices provide the entire package. As a consequence, the need to protect user privacy interests is in some ways greater in the mobile context than it is in the desktop context. Mobile devices may provide more avenues for potential abuse and the risks of data loss or compromise may be more immediate, yet transparency and user control are more constrained.

III. Standardization to bridge the gap

These discrepancies present obvious challenges for the standardization of new technologies to bring real-time interactive applications to the web. Designers cannot assume that their applications will be consumed in a desktop-only context, nor can they justify a desktop-only mentality in a world of increasing mobile growth. Furthermore, developing web-based applications with some notion of uniformity between the desktop and mobile experiences will serve to increase the salience of the web in a mobile environment where widgets and downloadable applications have gained significant popularity. Bringing real-time applications to the web presents an opportunity to showcase the web's "develop once, run anywhere" virtue in a mobile space where that virtue has heretofore been underappreciated.

Ongoing web standardization efforts can provide lessons and guidance in this endeavor. In some respects, the story of the W3C's Device APIs and Policy (DAP) working group [12] has thus far been one of been one of seeking to reconcile a desktop-centric view with a mobile-centric view. The DAP WG is standardizing a suite of web APIs that give web applications access to various device capabilities (address book, camera, calendar, etc.). The development of the group's Policy Framework [3], which provides a security model meant to address a wide variety of deployment scenarios, has been an effort to extend and generalize frameworks primarily developed in the mobile/widget context to the web at large. The Access Control Use Cases and Requirements [2] has evolved to now make clear distinctions between different kinds of authorization scenarios that apply in different desktop and mobile contexts. And the group's work that is perhaps most directly relevant to real-time applications – Media Capture, which provides access to camera and microphone capabilities – has been split entirely into two separate specifications, one of which uses an HTML5 form element and is thereby geared toward traditional web applications [6], and the other of which provides scripted access suitable for widgets or the web [10]. Exploring why these particular paths are being pursued and the extent to which reconciliation or separation of desktop and mobile contexts is taking place is vital to real-time web application standardization efforts.

Deployment experience with existing standards for advanced web applications is also relevant. Because user interface considerations are often viewed as out of scope for technology standardization efforts, the standardization of privacy features often comes in the form of normative guidance or requirements The W3C Geolocation API [7], for example, provides normative guidance to browser developers about providing transparency and user control with respect to location information. Deployment of transparency and control features has been somewhat uneven, however, and the feature differences between desktop and mobile browser clients are notable. For example, the amount of information that users receive about sites requesting their location varies from browser to browser and between mobile and desktop browsers [4].

For future standardization efforts, it is important to understand the extent to which normative privacy guidance truly achieve "standardization" in the sense of common user experiences across different deployment contexts. Standardization efforts in the real-time applications area should seek not only to standardize privacy requirements, but also to ensure that those requirements actually achieve some uniformity of privacy experiences.

IV. Conclusion

The web is ripe for real-time applications. But the way that people consume the web is changing rapidly, and the growth of mobile devices demands that new standardization efforts account for a greater variety of usage contexts. Starting with a unified vision for standardization efforts will not only help to improve the privacy features of real-time web applications across the board, but it will also help to showcase the virtues of the web on mobile platforms.

References

[1] Shane Ahern, Dean Eckles, Nathan Good, Simon King, Mor Naaman, and Rahul Nair. "Over-Exposed? Privacy Patterns and Considerations in Online and Mobile Photo Sharing," *Proceedings of the SIGCHI conference on human factors in computing systems*, San Jose, CA (2007).

[2] Laura Arribas, Frederick Hirsch, and Dominique Hazaël-Massieux, Eds., Device API Access Control Use Cases and Requirements (W3C Editor's Draft), http://dev.w3.org/2009/dap/policy-reqs/ (September 10, 2010).

[3] Laura Arribas, Paddy Byers, Frederick Hirsch and David Rogers, Eds., Policy Framework for Device APIs (W3C Editor's Draft), http://dev.w3.org/2009/dap/policy/Framework.html (June 29, 2010).

[4] Marcos Cáceres, "Privacy of Geolocation Implementations," *W3C Workshop on Privacy for Advanced Web APIs*, http://www.w3.org/2010/api-privacy-ws/papers/privacy-ws-21.pdf (July 2010).

[5] Jason Hong and James Landay, “An architecture for privacy-sensitive ubiquitous computing,” *Proceedings of the 2nd international conference on mobile systems, applications, and services*, Boston, MA (2004).

[6] Ilkka Oksanen and Dominique Hazaël-Massieux, Eds., HTML Media Capture (W3C Editor’s Draft), http://dev.w3.org/2009/dap/camera/ (August 4, 2010).

[7] Andrei Popescu, Ed., Geolocation API Specification (W3C Candidate Recommendation, http://www.w3.org/TR/geolocation-API/ (September 7, 2010).

[8] Kristen Purcell, Roger Entner, and Nichole Hendersen, *The Rise of Apps Culture*, Pew Internet and American Life Project, http://pewinternet.org/Reports/2010/The-Rise-of-Apps-Culture/Overview.aspx (September 14, 2010).

[9] Aaron Smith, *Mobile Access 2010*, Pew Internet and American Life Project, http://pewinternet.org/Reports/2010/Mobile-Access-2010/Summary-of-Findings.aspx (July 7, 2010).

[10] Dzung Tran, Ilkka Oksanen, Ingmar Kliche, Eds., The Media Capture API (W3C Editor’s Draft), http://dev.w3.org/2009/dap/camera/Overview-API.html (September 3, 2010).

[11] U.S. Department of Homeland Security, *Privacy Policy Guidance Memorandum, The Fair Information Practice Principles: Framework for Privacy Policy at the Department of Homeland Security,* http://www.dhs.gov/xlibrary/assets/privacy/privacy_policyguide_2008-01.pdf (December 2008).

[12] W3C Device APIs and Policy Working Group, http://www.w3.org/2009/dap/ (2010).