Multifactor Authentication based on User Contextual Data and the Mobile Web

Jon Azen
Qualcomm Technologies Inc.
5775 Morehouse Drive
San Diego, CA 92121
USA
jazen@qti.qualcomm.com

Ian Brettell
Qualcomm Technologies Inc.
5775 Morehouse Drive
San Diego, CA 92121
USA
ianb@qti.qualcomm.com

Laurence Lundblade
Qualcomm Technologies Inc.
5775 Morehouse Drive
San Diego, CA 92121 USA
llundbla@qti.qualcomm.com

Giridhar D. Mandyam
Qualcomm Innovation Center
5775 Morehouse Drive
San Diego, CA 92121
USA
mandyam@quicinc.com

Mike Milikich
Qualcomm Innovation Center
9600 N. Mopac, Ste 900
Austin, TX 78759 USA
milikich@quicinc.com

***Abstract*—The W3C has recently embarked on several efforts related to security and authentication in the web, including the Web Cryptography API and the efforts of the Web Payments Community Group. However, integration of hardware-based multifactor authentication places requirements on browser design particularly in mobile devices. This position paper discusses those unique challenges, and provides suggestions for standardization directions for the W3C.**

Keywords-authentication, authorization, continuous authentication

I. INTRODUCTION

Web authentication is common and widely accepted, but multifactor authentication on the web is still relatively new. Simple username/password authentication methods (combined with transport layer security) often are acceptable to relying parties, despite the risk involved.

Nevertheless, there is an interest in leveraging hardware based authentication by service providers, browser vendors and hardware manufacturers. For instance, the Fast Identity Online (FIDO) Alliance [1] has leveraged the expertise of various companies to deliver public specifications for multifactor authentication wherein hardware authentication can be leveraged by applications without exposing user information to hostile parties. Some examples of hardware-enabled authentication are fingerprint recognition, supplementary device presence (e.g. smart card), and manual PIN code entry.

However, the W3C has introduced several device-level API's that have found widespread adoption in browsers that could be leveraged as part of a multifactor authentication scheme. For instance, API's such as the W3C Geolocation API [2] or Media Capture and Streams specification [3] can be leveraged for device contextual information as part of authentication (e.g. utilizing a virtual geofence or face/scene recognition).

The W3C device API's have provided powerful tools to developers for a wide variety of application, but it is not clear that relying parties (RP's) can trust information accessed by a web application running in a device-based browser even if that web application is from the RP itself for many reasons. For one, a compromised browser can spoof device data. Secondly, browser-executable script can be accessed by hostile parties. Thirdly, standard transport layer security methods available to browsers rarely involve mutual authentication of RP's along with end user devices.

In addition, leveraging user contextual data involves constant monitoring of the user context itself. This could mean polling of sensor data on the device, which can result in shorter battery life. As a result, "continuous authentication" methods that leverage user contextual data should be designed considering hardware impacts. This is difficult to do with many W3C specifications, as they are meant to target multiple hardware platforms.

This paper outlines some of the motivations behind continuous authentication as part of a multifactor authentication method, and discusses potential solutions.

II. EXAMPLE USE CASES FOR CONTINUOUS AUTHENTICATION: GEOFENCING FOCUS

As mentioned previously, contextual authentication can be leveraged as part of any multifactor authentication method. As an example, an end user's location in relation to a virtual geofence may be considered as part of authenticating the user or a specific user action. Two examples of leveraging the breach of a virtual geofence in the context of multifactor authentication are provided in this section.

A. *Mobile Assisted Shopping*[1]

In-store mobile-assisted shopping provides an example of leveraging geofencing as part of the checkout process. One of the earliest examples of this was The Metro Group in Germany [6]. Metro in-store shoppers leveraged NFC in their devices to scan items as they take them from shelves, and when they are

[1] This example was provided in a previous paper by the authors [5].

ready to check out they are presented with a bar code that they have scanned on their way out of the store.

While the bar code on the device is one form of authentication, it could be considered redundant if the user's device is determined to be breaching a geofence boundary near the checkout area (e.g. near the exit of the store). This is depicted in Figure 1. Leveraging a user's location relative to a geofence boundary as an authentication mechanism in this case would have multiple benefits. For instance, the end user can be spared the inconvenience of having to present the device at a checkout station when leaving the store. Moreover, retail establishments can be spared need to install a barcode scanner at a particular location, which requires an investment in additional in-store infrastructure [7].

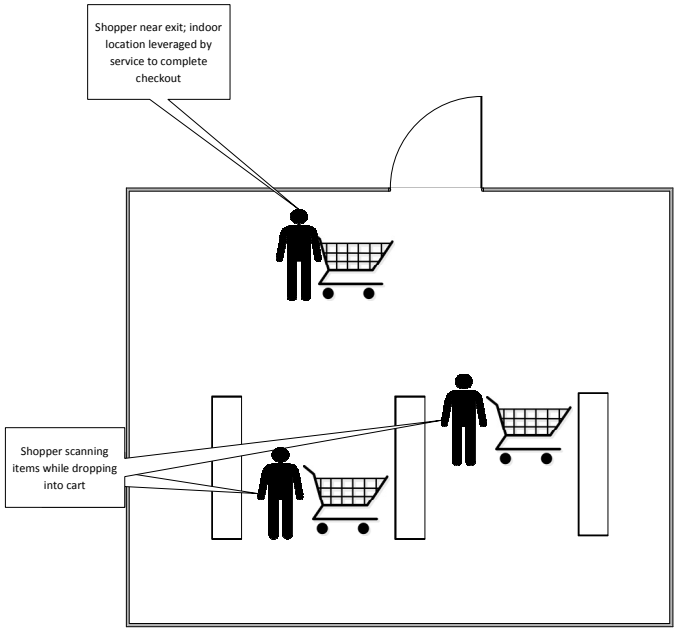

Figure 1: Shopper Checkout Using Location

B. *Displatch and Delivery*

In many dispatch and delivery use cases, verification of successful receipt of a physical package involves interaction between a delivery person and the intended recipient. However, the relying party (usually a dispatch service of some sort) can leverage detection of a virtual geofence breach as part of the authentication process (which in this case is really verification that the delivery person is physically present at the location of the intended recipient) [8].

Moreover, additional contextual information regarding the package delivery can be leveraged as part of authentication such as time of delivery (i.e. time of geofence breach event detection). Dispatchers who serve as relying parties (RP's) can therefore leverage this information in verification of the package delivery provided that this information can be obtained and delivered in a trusted manner.

III. Implementation Considerations for Continous Authentication: Battery Life Focuse

The current W3C Geolocation API does not provide for any hardware-based geofencing capability. As a result, any web applications that are to determine geofence breach events must do so via Javascript code that leverages the W3C Geolocation API. Many mobile devices however are based on a partitioning where applications such as the browser run on a general purpose microprocessor, while location-specific functionality such as the GPS processing run on the modem (which also implements connectivity functionality for wireless access). Implementing geofencing on the microprocessor has profound consequences as far as power consumption is concerned. This is because of several reasons, much of which has to do with the constant polling by the microprocessor to the modem to monitor the current location.

The difference in power consumption between geofencing implemented at the application layer (e.g. leveraging the current W3C Geolocation API) and geofencing implemented on the modem is depicted in Figure 2. This data was measured on actual Qualcomm hardware. Note in particular that modem-based geofencing adjusts location polling frequency based on detected distance from the geofence boundary, therein resulting in power consumption savings. Although this is possible to replicate this within Javascript, different implementations of the W3C API still may rely on fixed polling frequencies thus not resulting in any meaningful power savings.

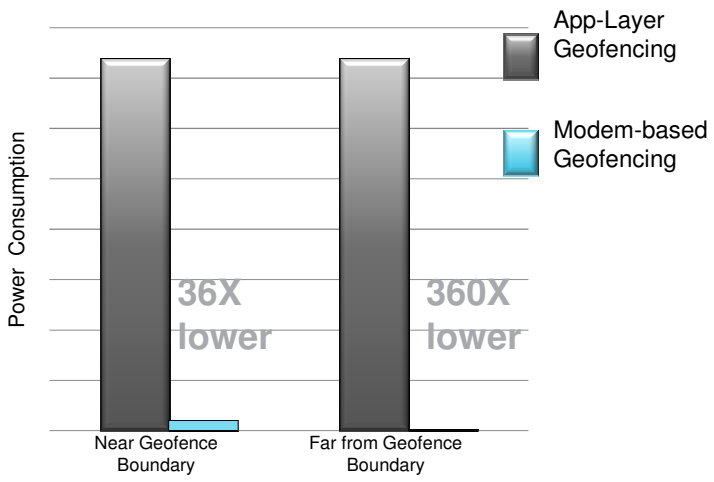

Figure 2: Geofencing and Power Consumption

IV. Conclusions

User contextual data based on device information (such as those obtained by sensors or geolocation technology) can be useful as part of any multifactor authentication method. In addition, the W3C has defined several device API's that can provide such information to web applications. However, exposure of such data to web applications makes it difficult for relying parties to leverage such data as part of any multifactor authentication method. Moreover, continuous authentication applications such as those leveraging geofencing may result in

increased power consumption when implemented at the web application layer. Although the W3C Geolocation Working Group has been recently re-charted to explore the addition of lower-power geofencing to the API, in the specific case of multifactor authentication there will still be a need for hardware-based authentication leveraging device information, so as to account for any security concerns with browser-based applications. It is desirable for user agents to be able to therefore implement W3C device API's for web applications that do not require trusted authenticators, while also allowing for web applications to leverage hardware authenticators that use identical device functionality but may implement it in a more secure and efficient manner.

REFERENCES

[1] https://fidoalliance.org/.

[2] The Worldwide Web Consortium. *Geolocation API Specification*. http://www.w3.org/TR/2010/CR-geolocation-API-20100907/. September 2010.

[3] The Worldwide Web Consortium. *Media Capture and Streams*. http://www.w3.org/TR/mediacapture-streams/. September 2013.

[4] The Worldwide Web Consortium. *Proximity Events*. http://www.w3.org/TR/2013/CR-proximity-20131001/. October 2013.

[5] Mandyam, Giridhar and Milikich, Mike. "Mobile Web Payments: Challenges and Ways Forward." *W3C Web Payments Workshop*. March 2014.

[6] Sakr, Shafir. "Will shoppers be enticed by new ways of paying**?**". BBC Online News Service. http://www.bbc.co.uk/news/business-12310810. January 31, 2011.

[7] Zhang, Feng et al. "Location-Based Authentication and Authorization using Smart Phones". *IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. June 2012. pp. 1285-1292.

[8] "Geofencing Breaks Through Efficiency Barriers." Truckinginfo.com. http://www.truckinginfo.com/channel/fleet-management/article/story/2011/05/geofencing-breaks-through-efficiency-barriers.aspx. May 2011.