Open *Source* | Open *Possibilities*

Multifactor Authentication based on User Contextual Data and the Mobile Web

September 11, 2014
W3C Web Crypto Workshop

QUALCOMM INNOVATION CENTER, INC.

Introduction

- The web ecosystem is trying to move beyond username/password for end user authentication
  - For example, FIDO Alliance recently released specifications targeted towards additional factors based on
    - Device integrated, end-user facing authenticators (Universal Authentication Framework, i.e. UAF)
    - External device-based second factor (Universal 2nd Factor, i.e. U2F)
- Coincidental with the development of HTML5, the W3C has also developed several device API's that have the potential to provide contextual data
  - Examples
    - Media capture and streams ("getUserMedia" or "gUM")
      - Local capture through microphone/camera of device environment
      - Audio fingerprinting, scene recognition, etc.
    - Geolocation
      - Geofencing or proximity to a given location
  - Contextual data can be used as to augment authentication factors

Considerations for Use of Device API's as an Auth Factor

- W3C device API's are powerful, but can a web service provider requiring authentication really leverage them?
  - Example: WebRTC API for browser-originated emergency call
    - Device location data is vital for PSAP (Public Safety Answering Pt.) in addressing call
    - Spoofable location is intolerable
    - Existing cellular telephony solutions already have means of securely obtaining and sending location data to PSAP
  - Web service provider also needs a secure means of reconfiguring an authenticator
    - Can require secure communication between service and trusted authenticator
- Obtaining contextual data can involve polling sensors frequently – this can be power consuming on handheld devices if implemented incorrectly
  - Examples
    - Geolocation: Geofencing requires long-lived geolocation processes
      - » A virtual geofence is defined by a centroid (usually a lat/lon pair) and radius
        - Geographic circle, but more complex polygonal geofences may be defined
    - gUM: audio fingerprinting from capture stream

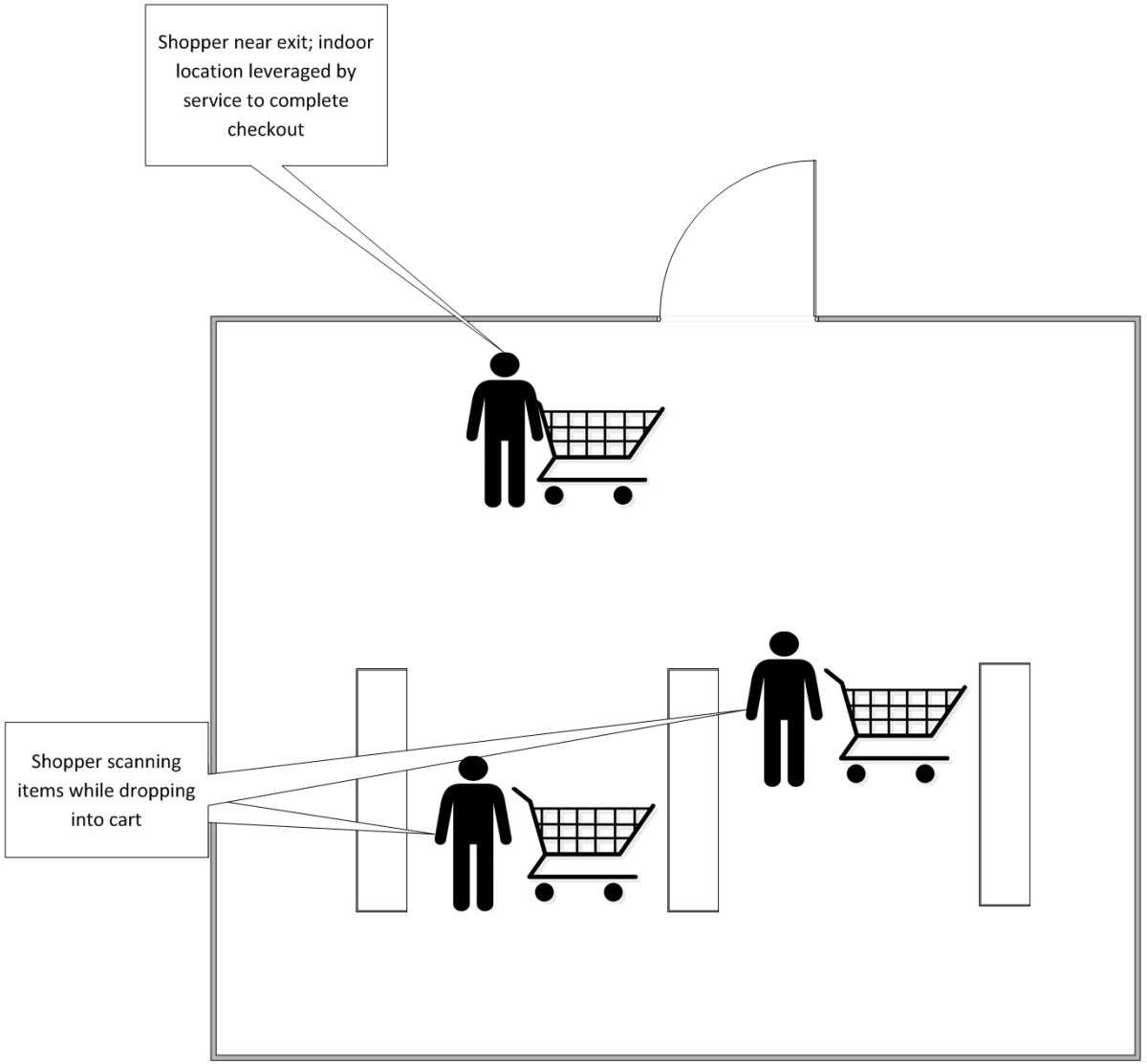
Example Use Case

Geofencing

Retail Mobile Web Payments Example (previously presented at W3C Web Payments Workshop)

- In-store shoppers who use bar code scanning on mobile device to scan in items as placing them into cart
- At checkout, device produces a final bar code to be scanned is displayed on mobile device and read
  - Customer automatically billed
- Can in-store location of shopper be leveraged instead?
  - Geofencing application

Mobile Web Payments – Going Forward



Dispatch and Delivery

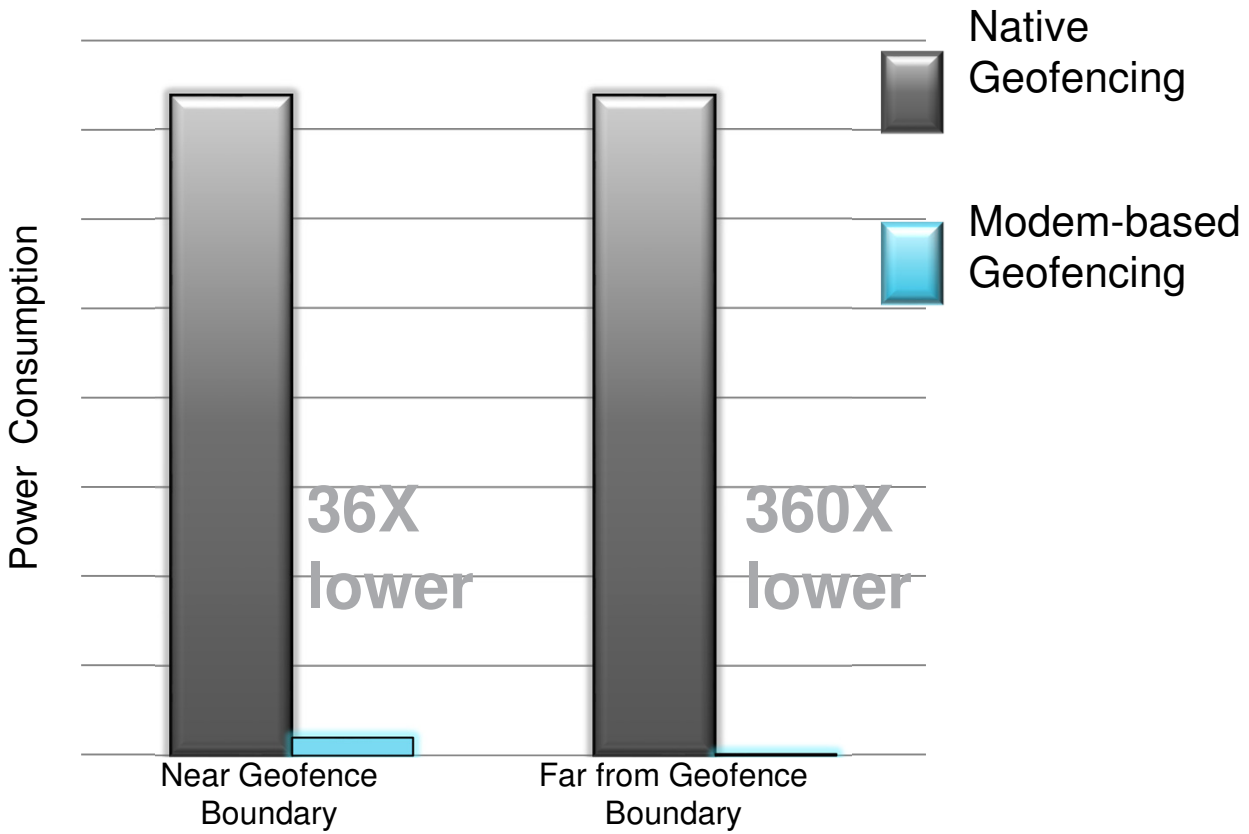
- Verification of successful delivery of some item or package
  - Ensure that delivery person's device is near target location
  - Dispatcher leverages this information for verification of delivery
- Delivery services often require timely delivery
  - Time of geofence breach occurrence could also be tracked
- Authentication can be based on a combination of geofence breach event data along with any other auth data sent by delivery person
  - Ideally would be communicated securely to dispatcher

Power Consumption Considerations for Continuous Auth

- Current Geolocation API does not have any kind of geofencing ability
  - Currently being defined by re-chartered W3C Geolocation Working Group
- Justification is that it is simple to develop a geofencing method in Javascript leveraging the existing API
- For mobile devices, particularly multi-core implementations, this is not only limiting but can be detrimental to performance
  - CPU/GPU/Modem partitioning
  - Running geofencing processes on modem is significantly less power consuming then at the app level (e.g. JS)

Geofencing on Modem Versus Apps Processor

Optimizes responsiveness with much lower power



Conclusions/Ways Forward

- Contextual data has a place alongside multifactor authentication for the web
- Such authenticators must be non-spoofable and trustworthy for web service providers
  - Service providers should also be able to reconfigure authenticators
    - Can trust run in both directions? Is trusted domain enough?
- Augmenting authentication with contextual data can have profound impacts for handheld devices
  - Power consumption can be an issue
- The suitability of current W3C device API's alongside multifactor auth is unclear
  - Can they take the place of secured HW-based authenticators?
  - Can they be efficiently implemented when compared to HW-optimized solutions?

Thank You

QuIC
QUALCOMM INNOVATION CENTER, INC.