

Geolocation/Geofencing Update

22-October-2015

Recent Occurences

- Geofencing API achieved FPWD status
 - Official draft at <http://www.w3.org/TR/geofencing/>
 - Living draft at <https://w3c.github.io/geofencing-api/>
- Call for comments on requiring authenticated origins for Geolocation API concluded
 - No consensus within group
 - Some progress: Chromium announced sunseting of support for non-auth. origins and geolocation
 - <http://lists.w3.org/Archives/Public/public-geolocation/2015Feb/0001.html>

Geofencing API

- Going through bug fixes
- Provided first cut at answers to TAG Privacy Questionnaire
 - <http://lists.w3.org/Archives/Public/public-geolocation/2015Oct/0003.html>
- Note also Geofencing API Privacy and Security section
 - <https://w3c.github.io/geofencing-api/#security-and-privacy-considerations>
 - Will be completed depending on consensus of group regarding questionnaire answers
 - May use TPAC F2F to finalize answers

Geofencing API – Privacy Considerations

- Geofencing API is built on Service Workers.
Why?
 - Well-suited for long-lived process such as geofencing
 - No defined use cases for requiring geofencing in the main thread
 - Would also be most likely built as extensions to existing geolocation API, which leads to ...
- Service workers require authenticated origins
 - In contrast to geolocation

Geofencing API – Privacy Considerations (cont.)

- But Service Workers require process persistence and some level of data persistence
 - Is it or is it not a concern? The privacy questionnaire does not definitively state that authenticated origins is sufficient mitigation.
- Can geofencing be abused in the context of a Pervasive Monitoring (RFC 7258) attack?
 - Leveraging an end user's location profile

Going Forward

- Need to complete privacy questionnaire and determine best way to incorporate in geofencing specification
- Guidance from PING on DeviceOrientation?
 - There are potential privacy abuses with this API
 - <http://lists.w3.org/Archives/Public/public-geolocation/2015Aug/0003.html>
 - API will be 'rebuilt' using Generic Sensor API being defined by DAP
 - DeviceOrientation is mature and ubiquitous; would face a lot of resistance trying to modify