The Web’s Sixth Sense:
A Study of Scripts Accessing Smartphone Sensors

Anupam Das
North Carolina State University

Joint work with Günes Acar, Nikita Borisov and Amogh Pradeep

https://sensor-js.xyz
Web Browsing is Increasingly Mobile

Desktop vs Mobile vs Tablet Market Share Worldwide
Jan 2012 - Sept 2018
New Mobile Web APIs

- Touch Events
- Vibration
- WebXR (VR/AR support)
- Sensors
  - Orientation
  - Motion
  - Ambient Light
  - Proximity

```javascript
window.addEventListener("devicemotion", motionHandler);

function motionHandler(evt){
    // Access Accelerometer Data
    ax = e.accelerationIncludingGravity.x;
    ay = e.accelerationIncludingGravity.y;
    az = e.accelerationIncludingGravity.z;
    // Access Gyroscope Data
    rR = e.rotationRate;
    if (rR !== null){
        gx = rR.alpha;
        gy = rR.beta;
        gz = rR.gamma;
    }
}
```
Sensor APIs

- **Orientation**
  - Orientation about the X, Y, Z axis (°)

- **Motion**
  - Accelerometer (m/s²)
  - Accelerometer w/o gravity (m/s²)
  - Gyroscope (°/s)

- **Ambient light**
  - Light sensor (lux)

- **Proximity**
  - Proximity sensor (cm)
No Permissions for Sensor APIs

● Available to any web page **without** permission check

● Try it! https://sensor-js.xyz/demo
API Exposure Risks

• **Keylogging**
  • PIN recovery\(^1\), keystroke recovery from nearby keyboard\(^2\)

• **Surreptitious recording**\(^3\)
  • Accelerometer and Gyroscope are low-fi microphones!

• **Surreptitious geolocation**
  • Motion changes (e.g., subway)\(^4\)
  • Ambient light changes

• **Fingerprinting**
  • Stateless tracking\(^5\)

• **Biometrics**
  • e.g., gait

---

In This Talk

We look at -

• which websites and scripts use sensors?
• ...for what purposes?
• what can be done to mitigate the risks?
Data Collection and Analysis

Crawler: OpenWPM-mobile
- Based on OpenWPM framework
- Develop a Mobile version
  - Emulate mobile environment: user agent, screen size, extensions, fonts, etc.
  - Capture addEventListener calls
  - Generate sensor APIs events and return realistic sensor data stream
# Sensor Access

<table>
<thead>
<tr>
<th>Sensor</th>
<th># sites</th>
<th># script domains</th>
</tr>
</thead>
<tbody>
<tr>
<td>Motion</td>
<td>2653</td>
<td>384</td>
</tr>
<tr>
<td>Orientation</td>
<td>2036</td>
<td>420</td>
</tr>
<tr>
<td>Proximity</td>
<td>186</td>
<td>50</td>
</tr>
<tr>
<td>Light</td>
<td>181</td>
<td>35</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>3695</strong></td>
<td><strong>603</strong></td>
</tr>
</tbody>
</table>

**Including...**
- cnn.com
- taobao.com
- tmall.com
- cnet.com
- alibaba.com
- foxnews.com
- zillow.com
- wells Fargo.com
- reuters.com
- bloomberg.com
- groupon.com
- hotels.com
### Who is using sensors?

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Top 3 domains</th>
<th># sites</th>
<th>Top rank</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Motion</strong></td>
<td>serving-sys.com</td>
<td>815</td>
<td>67</td>
</tr>
<tr>
<td></td>
<td>adsco.re</td>
<td>648</td>
<td>570</td>
</tr>
<tr>
<td></td>
<td>doubleverify.com</td>
<td>517</td>
<td>187</td>
</tr>
<tr>
<td><strong>Orientation</strong></td>
<td>adsco.re</td>
<td>648</td>
<td>570</td>
</tr>
<tr>
<td></td>
<td>alicdn.com</td>
<td>417</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>yieldmo.com</td>
<td>83</td>
<td>100</td>
</tr>
</tbody>
</table>
Exfiltration detection

• Trigger sensor events with easy-to-recognize values:
  
  \[ 42.1234 \text{ (fixed)} + 0.00005468 \text{ (random)} = 42.12345468 \]

• Look for raw and base64 encoded values in the request URLs and payload

<table>
<thead>
<tr>
<th>Domain</th>
<th>Sensors</th>
<th>Encoding</th>
<th># sites</th>
<th>Top site</th>
</tr>
</thead>
<tbody>
<tr>
<td>b2c.com</td>
<td>AOPL</td>
<td>b64</td>
<td>53</td>
<td>498</td>
</tr>
<tr>
<td>perimeterx.com</td>
<td>A</td>
<td>b64</td>
<td>45</td>
<td>247</td>
</tr>
<tr>
<td>wayfair.com</td>
<td>A</td>
<td>b64</td>
<td>7</td>
<td>1136</td>
</tr>
<tr>
<td>moatads.com</td>
<td>O</td>
<td>raw</td>
<td>5</td>
<td>3616</td>
</tr>
</tbody>
</table>

Clustering to understand use cases

• Low-level features: JS API
  • `get_window.navigator.userAgent`
  • `set_window.document.cookie`
  • `call_HTMLCanvasElement.toDataURL`
  • `addEventListener_deviceMotion`, ...

• High-level features: fingerprinting
  • Canvas, Battery, AudioContext, ...

• ~400 features per script

• Use DBScan for clustering
• Refinement techniques to reduce “noisy” cluster
• Use Moss to look at source code similarity
• Manual analysis of 3–5 scripts in each cluster
Use Cases

- Tracking
  - Fingerprinting, audience recognition, session replay
- Fraud detection
  - Bot detection
- Feature detection
- Gesture control
- Parallax tilt scrolling
- Responsive design
- RNG
Fingerprinting

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Canvas FP</th>
<th>Canvas Font FP</th>
<th>Audio FP</th>
<th>WebRTC FP</th>
<th>Battery FP</th>
<th>Any FP</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Motion</td>
<td>56.7</td>
<td>0.2</td>
<td>19.8</td>
<td>6.8</td>
<td>5.6</td>
<td>62.7</td>
<td>501</td>
</tr>
<tr>
<td>Orientation</td>
<td>36.2</td>
<td>3.4</td>
<td>5.7</td>
<td>6.2</td>
<td>4.5</td>
<td>41.7</td>
<td>650</td>
</tr>
<tr>
<td>Proximity</td>
<td>2.1</td>
<td>0.0</td>
<td>47.9</td>
<td>0.0</td>
<td>49.0</td>
<td>51.0</td>
<td>96</td>
</tr>
<tr>
<td>Light</td>
<td>19.5</td>
<td>1.2</td>
<td>56.1</td>
<td>15.9</td>
<td>57.3</td>
<td>76.8</td>
<td>82</td>
</tr>
</tbody>
</table>

Percentage of sensor-using scripts that also perform fingerprinting.
What can be done?

- Ad blockers, tracking protection mode?
  - blocklists miss the long tail (blocking rate: 1.8%-8.6%)
  - some sites serve scripts as first-party to avoid blocklists
- Feature Policy API
  - enables publishers to control what APIs are accessible
- Block sensor access from insecure and cross-origin iframes (W3C)
  - browsers don’t always follow recommendations
What can be done? (cont’d)

• Default to low resolution readings: ask user for high-precision readings if needed
• Visual indication when sensors are accessed
• Private browsing/incognito mode: lower resolution or disable by default
• (Future work...)
• Apple has turned off access to accelerometer and gyroscope by default in Safari since iOS 12.2

• As of May 9, 2018 Firefox (version 60) disabled proximity and light sensor APIs

https://www.macrumors.com/2019/02/04/ios-12-2-safari-motion-orientation-access-toggle/
Thanks for listening!

Paper, code, and data: sensor-js.xyz

Collaborators
- Günes Acar, Princeton Univ.
- Nikita Borisov, UIUC
- Amogh Pradeep, NEU