Exploring and Mitigating Privacy Threats of HTML5 Geolocation API

Hyungsub Kim, Sangho Lee, Jong Kim
Pohang University of Science and Technology (POSTECH)

Annual Computer Security Applications Conference (ACSAC) 2014

Presenter: Hyungsub Kim

2014-12-11
Location-based service (LBS)

- Use real-time geo-data to provide information
  - Discovering the nearest coffee chains
- Are used in a variety of contexts
  - Travel information, proximity-based marketing, fraud prevention

http://www.starbucks.com/store-locator/search
Web (HTML5 Geolocation API)

- Realize LBSs via the Web
- Obtain latitude, longitude, altitude, heading, speed
- Need to obtain permissions from users
- Methods: ① `getCurrentPosition()` ② `watchPosition()`

```
function showPosition(position) {
    var latlon = position.coords.latitude +"," + position.coords.longitude;
    var img_url = "http://maps.googleapis.com/maps/api/staticmap?center=" + latlon + "&zoom=13&size=500x400&maptype=hybrid&sensor=true";
}
```

<Obtain a Google map image based on the current geolocation>
Web (HTML5 Geolocation API)

- Realize LBSs via the Web
- Obtain latitude, longitude, altitude, heading, speed
- Need to obtain permissions from users
- Methods: ① `getCurrentPosition()` ② `watchPosition()`

Track user’s location

```javascript
navigator.geolocation.watchPosition(showPosition);

function showPosition(position) {
  var latlon = position.coords.latitude +"," + position.coords.longitude;
  var img_url = "http://maps.googleapis.com/maps/api/staticmap?center=" + latlon +"&zoom=13&size =500x400&maptype=hybrid&sensor=true";
}
```

<Obtain a Google map image based on the current geolocation>
Permission model

When a user grants permissions to a web site

All web pages of the web site can access the user’s location.

Geolocation exceptions

<table>
<thead>
<tr>
<th>Hostname pattern</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://www.w3schools.com:80">http://www.w3schools.com:80</a></td>
<td>Allow</td>
</tr>
</tbody>
</table>

<A list of Geolocation permissions on Chrome browser>

Employ a per-domain permission model
Motivation & goal

**Motivation**
- HTML5 Geolocation API can violate a user’s location privacy.
  - Due to its coarse-grained permission and location models

**Goal**
- Explore the current status of the Geolocation API
  - Discovering overprivileged web sites
  - Finding vulnerable web browsers
- Mitigate the privacy threats of the Geolocation API
  - Supporting fine-grained permission and location models
  - Inspecting the location sensitivity of each web page
Outline

- Privacy threats
  - Real-world privacy problems
  - Proposed scheme
  - Evaluation
(1) No per-method permission

www.w3schools.com wants to use your device’s location.  
Learn more

Deny | Allow

www.w3schools.com wants to use your device’s location.  
Learn more

Deny | Allow

< Permission dialogs >

1. Problem: Cannot distinguish the two different methods

The website can silently track the geolocation changes of a user.
(2) No re-confirm process

2. Problem: Web sites preserve permissions even after their contents are changed.
3. Problem: All webpages can access the exact location of a user.
Outline

- Privacy threats
- **Real-world privacy problems**
- Proposed scheme
- Evaluation
(1) How browsers implement API

- Investigate top 60 Android web browsers
- Detect 39 web browsers that support the Geolocation API

4. Problem: Do not ask for user permissions

<table>
<thead>
<tr>
<th>Geolocation permission</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Permanent &amp; temporary</td>
<td>18</td>
</tr>
<tr>
<td>Permanent only</td>
<td>7</td>
</tr>
<tr>
<td>No permission check</td>
<td>14</td>
</tr>
<tr>
<td>Not available</td>
<td>21</td>
</tr>
</tbody>
</table>

<How the 60 Android web browsers support the Geolocation API>

<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>#Downloads</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baidu Browser</td>
<td>4.1.0.3</td>
<td>10,000,000+</td>
</tr>
<tr>
<td>Maxthon Browser for Android</td>
<td>4.3.0.2000</td>
<td>5,000,000+</td>
</tr>
<tr>
<td>Angel Browser</td>
<td>12.30z</td>
<td>500,000+</td>
</tr>
<tr>
<td>Maxthon Web Browser for Tablet</td>
<td>4.0.4.1000</td>
<td>500,000+</td>
</tr>
<tr>
<td>Exsoul Web Browser</td>
<td>3.3.3</td>
<td>100,000+</td>
</tr>
<tr>
<td>Full Screen Browser</td>
<td>2.3</td>
<td>100,000+</td>
</tr>
<tr>
<td>Harley Browser</td>
<td>1.3.4</td>
<td>100,000+</td>
</tr>
<tr>
<td>Maxthon Browser for Pioneer</td>
<td>2.7.3.1000</td>
<td>100,000+</td>
</tr>
<tr>
<td>Safe Browser - The Web Filter</td>
<td>1.2.5</td>
<td>100,000+</td>
</tr>
<tr>
<td>Baidu Browser for Tablet</td>
<td>1.3.0.2</td>
<td>100,000+</td>
</tr>
<tr>
<td>Habit Browser</td>
<td>1.1.25</td>
<td>100,000+</td>
</tr>
<tr>
<td>Browser Omega</td>
<td>2.6.1</td>
<td>50,000+</td>
</tr>
<tr>
<td>Jelly Web Browser</td>
<td>1.1.4</td>
<td>10,000+</td>
</tr>
<tr>
<td>Zomi Mobile Browser</td>
<td>2.6.6</td>
<td>10,000+</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>16,770,000+</strong></td>
<td></td>
</tr>
</tbody>
</table>

<Vulnerable browsers doing not ask for permissions>

More than 16 million users have installed the 14 vulnerable browsers!
(2) How web sites utilize API

Why does the website want to know my pinpoint location?

It is unnecessary for me to know pinpoint location!

City: New York, NY 10028, USA
Pinpoint: 99-109 E 81st St New York, NY 10028, USA
(2) How web sites utilize API

- Collect 1196 web pages that use the Geolocation API
  - Inspect web sites listed on Alexa.com
  - Use the Google search engine with keywords (e.g., “near me” and “around me”)
  - Use an HTML code search engine (globalogiq.com)
- Analyze categories and location sensitivity of the web pages

5. Problem: About half of web pages do not need to use exact geolocation
Outline

- Privacy threats
- Real-world privacy problems
- Proposed scheme
- Evaluation
Design goals

- To support a fine-grained permission model
  - Per-method permissions to separate positioning and tracking
  - Per-webpage permissions

- To support a fine-grained location model
  - Verify the changes of web pages

- To recommend appropriate privacy settings
  - Inspect the location sensitivity of each web page
(1) Verify changes of web pages

Fuzzy hash value: c98e8c29770f9

< The first visit >

< stored hash value >

< The second visit >

< new hash value >

4) Compare the new hash value with the stored one

Fuzzy hash value: a1169830770b7837ef8

< new hash value >

Fuzzy hash value: 3a2290

< new hash value >

1) Extract the resources of the page
2) Compute a hash value of them
3) Store the value in DB

Browser ask for user permissions again!

How to remove dynamic contents?
(1) Verify changes of web pages

1) Access the same page at the same time

2) Inspect changed contents over the visits

3) Compute a hash value while excluding the dynamic content

Proposed scheme (3/7)
New permission dialog

Per-page: touch.groupon.com/nearby
Per-domain: touch.groupon.com

Per-page permissions

*know*: geolocation.getCurrentPosition();
*track*: geolocation.watchPosition();

Per-method permissions

Recommend appropriate privacy setting

Inspect location sensitivity of the webpage

A web page "touch.groupon.com/subscriptions" wants to *know* your location.

This webpage requires *city* level of granularity

Inspect web page

⚠️ Remember this choice
Allow
Don’t allow
(2) Inspect location sensitivity

1) Extract static contents from the two rendered web pages

2) Visit a web page multiple times with five different addresses

3) Compare the static contents with the five rendered web pages

4) Recommend a privacy setting to a user
Demonstration videos

Demonstration videos

Outline

- Privacy threats
- Real-world privacy problems
- Proposed scheme
- Evaluation
Accuracy

- Randomly choose 200 web pages
- Correctly estimate the location sensitivity
  - Comparing the results of our scheme with manual inspection
  - Accuracy: 93.5%
Choose Top 20 web pages using the Geolocation API from *Alexa.com*

Measure time to inspect location sensitivity
  - About 1.8 times longer than the page loading time

Is negligible because it happened just one time.
  - Only when the browser visits the web page for the first time.

*<Inspection time normalized to page loading time>*

Average: 1.8 times
Conclusion

- Consider the privacy problems of the Geolocation API
  - Point out coarse-grained permission and location models
  - Detect vulnerable web browsers and overprivileged web sites
- Propose a scheme to enhance the Geolocation API
  - Support fine-grained permission and location models
  - Recommend appropriate privacy settings according to automatic location sensitivity inspection
Thank you

Are there any questions?