(Cross-)Browser Fingerprinting via OS and Hardware Level Features

Yinzhi Cao  
Lehigh University  
yinzhi.cao@lehigh.edu

Song Li  
Lehigh University  
sol315@lehigh.edu

Erik Wijmans  
Washington University in St. Louis  
erikwijmans@wustl.edu
Roadmap

• Background
• Design
  • Existing features
  • New features
  • Improvement of existing features
• Evaluation
• Conclusion
Background

- **Background**
- Design
  - Existing features
  - New features
  - Improvement of existing features
- Evaluation
- Conclusion
User Tracking
Identifier

Stateful

• Definition: Identify users by previously stored information
  • Cookie
  • Supper cookie

Stateless

• Definition: Identify users by features without stored information
  • AmlUnique\(^1\), Panoticlick\(^2\)
    • User agent string
    • List of plugins

[2]: https://panopticlick.eff.org/
Cross-browser fingerprinting

• Survey Result: 70% of the surveyed users use two or more browsers regularly
• Problem: Single-browser fingerprinting can’t identify user when changing browser
• Key Insight: Adopting hardware and OS level features
  • i.e., cross-browser invariant
Design

- Background
- **Design**
  - Existing features
  - New features
  - Improvement of existing features
- Evaluation
- Conclusion
## A list of our features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Single-browser</th>
<th>Cross-browser</th>
</tr>
</thead>
<tbody>
<tr>
<td>WebGL based GPU rendering result</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
<tr>
<td>Supported language</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
<tr>
<td>Number of CPU virtual cores</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Screen ratio</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>List of fonts (JS based)</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
<tr>
<td>Audio context</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
</tbody>
</table>
## New features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Single-browser</th>
<th>Cross-browser</th>
</tr>
</thead>
<tbody>
<tr>
<td>WebGL based GPU rendering</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
<tr>
<td>Supported language</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
<tr>
<td>Number of CPU virtual cores</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Screen ratio</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>List of fonts (JS based)</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
<tr>
<td>Audio context</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
</tbody>
</table>
WebGL rendering

- Most of WebGL animations are rendered by GPU
- Different GPUs render pictures in different ways

Result of GPU 1

Result of GPU 2

Subtraction of result 1 and result 2

Subtraction of result 1 and result 2 (x200)
Example: texture mapping
WebGL Rendering

- Texture
- Light
- Camera
- Model
- Transparency
- Complex Lights
- Anti-aliasing
- Special textures
Virtual CPU cores

• Ways to get
  • New API (navigator.hardwareConcurrency)
  • Side channel detection (Run different number of JavaScript workers and measure the time)[3]

[3]: https://github.com/ofitn-oswg/core-estimator
Writing scripts (Supported language)

- Source: 36 popular different writing scripts (Wikipedia popular writing scripts)
- Result: Shown as languages (supported) or boxes (unsupported)
## Improvement of existing features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Single-browser</th>
<th>Cross-browser</th>
</tr>
</thead>
<tbody>
<tr>
<td>WebGL based GPU rendering</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
<tr>
<td>Supported language</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
<tr>
<td>Number of CPU virtual cores</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Screen ratio</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>List of fonts (JS based)</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
<tr>
<td>Audio context</td>
<td>Yes</td>
<td>Yes (Need Modification)</td>
</tr>
</tbody>
</table>
Screen resolution and depth

- Existing Solution: screen.width, screen.height
  - Including AmlUnique etc.
  - Not stable for even single browser
- Problem: affected by screen zoom levels
- Method to solve this problem
  - Use the detected resolution times zoom level
  - Screen ratio
Audiocontext

Fingerprinting audio card

- Method: Input some audio resources and capture the results\cite{4}
- Problem: Influenced by both browsers and audio card

Our improvement: AudioContext

- sampleRate (44100)
- maxChannelCount (2)
- numberOfInputs (1)
- numberOfOutputs (0)
- channelCount (2)
- channelCountMode (explicit)
- channelInterpretation (speaker)

List of fonts

- Method: JavaScript based\(^5\)
  - Try to use different fonts
  - If not installed, back to default font
  - Measure the width of characters

\[\text{Default font} \quad \alpha\beta\chi\delta\varphi\gamma\kappa\lambda\mu\nu\pi\theta\rho\sigma\tau\]
Width 2 inches

\[\text{Abadi MT Condensed light} \quad \text{abcdefgijklmnopqrstuvwxyz}\]
Width 1.5 inches

- Improvement: Select different subsets from 4,422 fonts (based on different OS)
  - E.g., Segoe WP and FreeMono

\(^5\): [https://github.com/Valve/fingerprintjs2](https://github.com/Valve/fingerprintjs2)
Evaluation

- Background
- Design
  - Existing features
  - New features
  - Improvement of existing features
- Evaluation
- Conclusion
Evaluation

• 3,615 fingerprints
  • Amazon Mechanical Turk
  • Microworkers
• All over the world
## Normalized entropies

<table>
<thead>
<tr>
<th></th>
<th>Ours</th>
<th>AmlUnique</th>
<th>Panoticlick</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Agent</td>
<td>0.612</td>
<td>0.570</td>
<td>0.531</td>
</tr>
<tr>
<td>List of Plugins</td>
<td>0.526</td>
<td>0.578</td>
<td>0.817</td>
</tr>
<tr>
<td>List of Fonts(Flash)</td>
<td>0.219</td>
<td>0.446</td>
<td>0.738</td>
</tr>
<tr>
<td>Screen Resolution</td>
<td>0.285</td>
<td>0.277</td>
<td>0.256</td>
</tr>
<tr>
<td>Timezone</td>
<td>0.340</td>
<td>0.201</td>
<td>0.161</td>
</tr>
<tr>
<td>Cookie Enabled</td>
<td>0.001</td>
<td>0.042</td>
<td>0.019</td>
</tr>
</tbody>
</table>
## Overall results

<table>
<thead>
<tr>
<th></th>
<th>Single-browser</th>
<th>Cross-browser</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>AmIUnique</strong></td>
<td>90.84%</td>
<td>10.82</td>
<td></td>
</tr>
<tr>
<td><strong>Known features</strong></td>
<td></td>
<td>68.98%</td>
<td>6.88</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td>99.24%</td>
<td>10.95</td>
<td>83.24%</td>
</tr>
</tbody>
</table>

**Uniqueness**: The percentage of fingerprints which are unique in all fingerprints

**Stability**: The percentage of fingerprints which are same in different browsers
## New features results

<table>
<thead>
<tr>
<th>Feature</th>
<th>Single-browser</th>
<th>Cross-browser</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entropy</td>
<td>1.40</td>
<td>0.98</td>
</tr>
<tr>
<td>Screen Ratio</td>
<td></td>
<td></td>
</tr>
<tr>
<td>List of Fonts (JavaScript)</td>
<td>10.40</td>
<td>6.58</td>
</tr>
<tr>
<td>Audio Context</td>
<td>1.87</td>
<td>1.02</td>
</tr>
<tr>
<td>CPU Virtual cores</td>
<td>1.92</td>
<td>0.59</td>
</tr>
<tr>
<td>Writing Scripts</td>
<td>2.87</td>
<td>0.51</td>
</tr>
<tr>
<td>GPU Texture test</td>
<td>3.5</td>
<td>2.26</td>
</tr>
<tr>
<td>GPU Light test</td>
<td>3.52</td>
<td>2.27</td>
</tr>
<tr>
<td>All cross-browser features</td>
<td>10.92</td>
<td>7.10</td>
</tr>
</tbody>
</table>
## Cross-browser Fingerprinting Uniqueness and Stability

<table>
<thead>
<tr>
<th>Browser</th>
<th>Chrome</th>
<th>Firefox</th>
<th>Edge</th>
<th>IE</th>
<th>Opera</th>
<th>Safari</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chrome</td>
<td>99.2% (100%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Firefox</td>
<td>89.1% (90.6%)</td>
<td>98.6% (100%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge</td>
<td>87.5% (92.6%)</td>
<td>97.9% (95.9%)</td>
<td>100% (100%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IE</td>
<td>85.1% (93.1%)</td>
<td>91.8% (90.7%)</td>
<td>100% (95.7%)</td>
<td>100% (100%)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Opera</td>
<td>90.9% (90.0%)</td>
<td>100% (89.7%)</td>
<td>100% (100%)</td>
<td>100% (60.0%)</td>
<td>100% (100%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Safari</td>
<td>100% (89.7%)</td>
<td>100% (84.8%)</td>
<td>N/A</td>
<td>N/A</td>
<td>100% (100%)</td>
<td>100% (100%)</td>
<td></td>
</tr>
<tr>
<td>Others</td>
<td>100% (22.2%)</td>
<td>100% (33.3%)</td>
<td></td>
<td></td>
<td>100% (50%)</td>
<td></td>
<td>100% (100%)</td>
</tr>
</tbody>
</table>

### Uniqueness (Stability)
Observations

- Our fingerprintable features are highly reliable
  - The removal of one single feature has little impact on the fingerprinting results
- DataURL is implemented differently across browsers.
Conclusion

• Background
• Design
  • Known features
  • New features
  • Improvement of known features
• Evaluation
  • Conclusion
Conclusion

Single browser
- Improved the uniqueness of existing work
  - 90.84% (AmIUnique) -> 99.24% (ours)

Cross browser
- A reliable and usable approach to fingerprint machine
- 83.24% uniqueness and 91.44% stability
Thank you!

- Website: http://www.uniquemachine.org/
- Questions?