Phish in Sheep’s Clothing: Exploring the Authentication Pitfalls of Browser Fingerprinting

Xu Lin, Panagiotis Ilia, Saumya Solanki, Jason Polakis
University of Illinois at Chicago, USA
xlin48@uic.edu

August, 2022
Risk-based authentication and two-factor authentication (2FA)

- 2FA creates friction for users
- Certain websites only trigger 2FA for suspicious login attempts
Advanced risk-based authentication that uses browser fingerprinting

1. Visit login page

2. Page with fingerprinting script

3. Send login, password, fingerprints

4. Grant access or trigger 2FA

- Fingerprints don't match
- Fingerprints match
Threat Model

The attacker tricks the user into visiting a malicious website and entering their credentials.
Overview of our attack workflow

Figure 1: Overview of our attack workflow that misuses browser fingerprints for bypassing ancillary security checks.

Fingerprint Extractor 0.1
Generate fingerprinting JavaScript

Fingerprint Spoofers 1.0
Detect fingerprinting and report fake values
Phase 1: Attacker visits target websites and "extracts" their fingerprinting code

1. Enable FP-extractor extension
2. Visit target-website
3. "Extract" fingerprinting code
4. Page with fingerprinting script

Automatically replicate the fingerprinting process of target websites
Phase 2: attacker obtains user’s credentials and fingerprints

1. Deploy phishing site
   - https://www.phish-website.com

2. Visit phishing website
   - John Doe
     - ***********

3. Page generates fingerprints of user's device

Collect fingerprints

Generate fingerprints identically to the ones expected by target websites
Phase 3: attacker spoofs fingerprints and bypasses 2FA mechanism

1. Enable FP-Spoofer extension
2. Visit target website
3. Page with fingerprinting script
4. Spoof fingerprints, send login, password
5. Successful Login
   OK
6. Grant access or trigger 2FA

Fingerprints Match !!!
How FP-Extractor Extension Works

1. Inject code that hooks fingerprinting properties & methods.

```javascript
Object.defineProperty(MediaDevices.prototype, 'enumerateDevices', {
  value: () => {
    fpTrace.push('enumerateDevices');
    return originalPromise;
  }
})
```

2. Code runs at “document_start”.
3. Keep track of accesses with their arguments.
   - Dynamic FP attributes (e.g., WebGL) can vary across websites.

```javascript
if (fpTrace.includes('enumerateDevices')) {
  fpCode += `navigator.mediaDevices.enumerateDevices().then...`
}
```


Target website

Phishing website
How FP-Spoof Extension Works

- Take victim’s fingerprints as input
- Hook fingerprinting APIs
- Override/delete/add values to match the victim’s values

For complex FPs
- No need to manipulate intermediate values
- Only spoof the final values, e.g., toDataURL for Canvas, offsetWidth and offsetHeight for Fonts

```javascript
Object.defineProperty(HTMLSpanElement.prototype, "offsetWidth", {
  get: function(){
    if (isSupportedFont) {
      return customWidth;
    } else {
      return fallbackFontWidth;
    }
  }
});
```
Fingerprint Spoofing Demo

Attacker spoofs their device’s fingerprints to mimic those of the victim’s device.
Experimental Evaluation

- Crawled Alexa top 20K
  - Logged FP APIs being used
  - Top sites employ more advanced fingerprinting techniques on login pages vs home pages

- Select 300 popular sites that implement FP and support 2FA for manual analysis
  - 14 use fingerprints for remembering user’s device
    - More prevalent among high-value financial services!
    - Risk-based authentication + FPs = emerging trend

<table>
<thead>
<tr>
<th>Website</th>
<th>Top 10K Home</th>
<th>Top 10K Login</th>
<th>Top 10K-20K Home</th>
<th>Top 10K-20K Login</th>
</tr>
</thead>
<tbody>
<tr>
<td>Navigator</td>
<td>5,510</td>
<td>5,403</td>
<td>5,589</td>
<td>5,371</td>
</tr>
<tr>
<td>Window</td>
<td>5,261</td>
<td>5,104</td>
<td>5,272</td>
<td>4,968</td>
</tr>
<tr>
<td>Screen</td>
<td>5,209</td>
<td>4,682</td>
<td>5,231</td>
<td>4,473</td>
</tr>
<tr>
<td>Timezone</td>
<td>5,035</td>
<td>4,617</td>
<td>4,934</td>
<td>4,282</td>
</tr>
<tr>
<td>Canvas</td>
<td>1,224</td>
<td>1,254</td>
<td>1,077</td>
<td>879</td>
</tr>
<tr>
<td>Canvas Fonts</td>
<td>179</td>
<td>380</td>
<td>142</td>
<td>237</td>
</tr>
<tr>
<td>WebRTC</td>
<td>221</td>
<td>313</td>
<td>192</td>
<td>210</td>
</tr>
<tr>
<td>AudioContext</td>
<td>290</td>
<td>351</td>
<td>223</td>
<td>234</td>
</tr>
</tbody>
</table>
## Risk-based authentication mechanisms in popular web services

<table>
<thead>
<tr>
<th>Website</th>
<th>BasicFP</th>
<th>Fingerprinting Technique</th>
<th>Fonts</th>
<th>Audio</th>
<th>IP Address Restrictions</th>
<th>Vulnerable</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Canvas/WebGL</td>
<td></td>
<td></td>
<td>IP Check</td>
<td>Bypass</td>
</tr>
<tr>
<td>Bank-A</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>Bank-B</td>
<td>×</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>CreditCard</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>→</td>
</tr>
<tr>
<td>Trading-A</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>Trading-B</td>
<td>×</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>→</td>
</tr>
<tr>
<td>Tax-A</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Tax-B</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>Tax-C</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>Tax-D</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>eCommerce-A</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>eCommerce-B</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>RideSharing</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>→</td>
</tr>
<tr>
<td>Food&amp;Beverage-A</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Food&amp;Beverage-B</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

- We completely bypass 2FA in 9/14 websites that use FPs for authentication!
- Attack only prevented by IP address checks.
- We inject X-Forwarded-For header (used by proxies) with the user’s IP to bypass IP-checks (→).
Phishing website datasets

Three Phishing website datasets

Local server

Phish-A

Local server

Phish-B

Actual websites

APWG


Use VisibleV8 to log native functions and property accesses
Phishing and Fingerprinting

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Time Period</th>
<th>Sites</th>
<th>JS</th>
<th>FP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phish-A</td>
<td>31/05/2018 – 19/06/2019</td>
<td>71,343</td>
<td>39,618</td>
<td>29,312</td>
</tr>
<tr>
<td>Phish-B</td>
<td>31/10/2018 – 05/05/2020</td>
<td>82,431</td>
<td>40,777</td>
<td>36,733</td>
</tr>
<tr>
<td>APWG</td>
<td>05/05/2020 – 12/04/2021</td>
<td>173,269</td>
<td>93,568</td>
<td>85,491</td>
</tr>
</tbody>
</table>

- The majority collect fingerprints, with 73.98%, 90.08% and 91.36% across the 3 datasets respectively.
- An increase in the number of websites collecting browser fingerprints over time.
### Phishing sites that obtain all necessary fingerprints for bypassing 2FA

<table>
<thead>
<tr>
<th>Target</th>
<th>Sites</th>
<th>Bypass</th>
<th>Sites</th>
<th>Bypass</th>
<th>Sites</th>
<th>Bypass</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bank-A</td>
<td>83</td>
<td>1</td>
<td>685</td>
<td>14</td>
<td>330</td>
<td>74</td>
</tr>
<tr>
<td>Bank-B</td>
<td>1549</td>
<td>-</td>
<td>2,683</td>
<td>-</td>
<td>327</td>
<td>-</td>
</tr>
<tr>
<td>CreditCard</td>
<td>89</td>
<td>61</td>
<td>0</td>
<td>0</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td>Trading-A</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>RideSharing</td>
<td>7</td>
<td>0</td>
<td>363</td>
<td>1</td>
<td>1378</td>
<td>5</td>
</tr>
</tbody>
</table>

APWG dataset
- more recent
- visited actual websites
Are phishers adapting their targets?

- The sharp decline in phishing sites targeting Bank-B could be due to the IP address requirement.
Summary

- Developed fully automated system for replicating target sites’ fingerprinting process, and replaying users’ stolen fingerprints.
- Presented an empirical analysis of the use of browser fingerprinting for augmenting web authentication in the wild.
- Demonstrated attacks that completely bypass 2FA in high-value services.
- Presented a large-scale study on the use of browser fingerprinting techniques by phishing sites.
- Disclosed our findings to affected vendors.