The Dangers of Human Touch: Fingerprinting Browser Extensions through User Actions

Konstantinos Solomos, Panagiotis Ilia, Soroush Karami, Nick Nikiforakis, Jason Polakis
ksolom6@uic.edu
An Emerging Privacy Problem

- Fingerprinting browser extensions
  - Arbitrary websites detect extensions and track the user
  - No permissions
  - Reveal personal-sensitive information

- Side channel inference techniques
  - Web Accessible Resources (Sjosten et al. CODASPY ‘17)
  - Behavioral fingerprints (Starov & Nikiforakis IEEE S&P ’17, Karami et al. NDSS ’20)
  - Style Modifications (Laperdrix et al. USEC ‘21)
An Emerging Privacy Problem

• Fingerprinting browser extensions
  • Arbitrary websites detect extensions and track the user
  • No permissions
  • Reveal personal-sensitive information

• Side channel inference techniques
  • Web Accessible Resources (Sjosten et al. CODASPY ‘17)
  • Behavioral fingerprints (Starov & Nikiforakis IEEE S&P ‘17, Karami et al. NDSS ’20)
  • Style Modifications (Laperdrix et al. USEC ‘21)
Extensions: Complex & Dynamic Behavior

• Specialized features triggered by **user interactions**
  • Text Selection
  • User Input
  • Right Click
  • Context Menu
  • Hotkeys
  ...

➤ How do user interactions affect the fingerprintability of extensions?
Threat model
Methodology
Methodology
Preparatory Phase

• Parse *manifest.json* extract permissions & structure
  • `ContextMenu` → Right-click action
  • `Browser_Action` → Extension icon

• Static analysis to identify user-driven capabilities
  • `addEventListener` *(click, scroll, keypress, …)*
  • Categorize and group by the target action
    • `Mouseup, Mousedown, Mousemove, Mouseover`
    • `Click, Doubleclick, Scroll, Select`
    • `Keypress, Keyup, Keydown`
    ....
User Interaction Templates

• **Browser actions**
  
  • Extension’s browser icon, Popup page, Configuration page
User Interaction Templates

• **Browser actions**
  • Extension’s browser icon, Popup page, Configuration page

• **Mouse actions**
  • Doubleclick, Select, Highlight
  • Mousemove, Mousedown, Mouseup, Mouseover
User Interaction Templates

• **Browser actions**
  - Extension’s browser icon, Popup page, Configuration page

• **Mouse actions**
  - Doubleclick, Select, Highlight
  - Click, Mousedown, Mouseup, Blur, Focus

• **Keyboard actions**
  - Single keystroke, Repetitive keystroke
  - Combined Hotkeys
Methodology
Extension Fingerprinting via User Actions

• Honey Page
  - Adopted by Carnus [Karami et al.]
  - Forms, clickable elements, dynamic elements, dropdown lists
  - Textual content of 8 popular languages

• Exercise extension according to their structure & permissions
  \[\text{Actions}_{\text{exti}} : \{\text{extension-icon, right-click, mouse, keyboard, \ldots}\}\]
  \[\text{Actions}_{\text{extj}} : \{\text{right-click, popup page, mouse, keyboard, \ldots}\}\]
Extension Fingerprinting via User Actions

• Generate **fingerprint** after each action
  
  • Trigger each action **independently**
  • Collect the behavioral fingerprint

  • Outer HTML modifications (**DOM**)
  • Intra-communication (**broadcasted messages**)
  • Inter-communication (**resources loaded**)

\[
Fingerprint_{exti} : \{\text{right-click [background-color:blue]}, \text{key}_M [msg:abc]\}
\]
Experimental Evaluation

• 3 Datasets [2018-2021]
  • 41K extensions
  • Fingerprinted: 5,531 (13%)

• Overview
  • 89% of extensions triggered by extension icon
  • Mouse events: highlight term and right-click (75%)
  • Keyboard interactions: single keystroke and 2 key combination (83%)

➢ Effectively replicate user interactions and trigger extensions
Experimental Evaluation
Attack: Page Simulated Events

• Generate artificially crafted interaction events
  • JavaScript API Dispatch Event
  • Replicate all mouse and keyboard events
    • Click, Scroll, Select, Mouse Move, ...
  • Bypass real user interactions

• Browsers origin verification mechanism
  • event.isTrusted {True, False}
  • Rarely used by developers
Attack Evaluation

• Leverage artificial events to trigger extensions
  • Select term \{mousemove, ...,mouseover,highlight, ..., doubleclick\}
  • Enable form \{mousemove, ..., mouseup, click, ..., click\}

• Vulnerable extensions: 1,513 (67 %)
  • 88% of mouse events
  • 65% of keyboard events

• Triggering 20 extensions < 0.5 seconds
Conclusion

• Novel extension fingerprinting vector that employs user interactions to fingerprint extensions

• Evaluated user-triggered extension fingerprinting and detected 1,820 hidden extensions

• Demonstrated the lack of security checks by triggering extensions through artificial actions

• Proposed a countermeasure for automatic incorporation of safeguards in the extension’s code
Thank you!

Feel free to reach out with any questions:
ksolom6@uic.edu