

Master Thesis Defense

Felix Kybranz
kybranz@campus.tu-berlin.de
Matriculation number: 380341

Supervised by:
Prof. Dr. Jean-Pierre Seifert
Prof. Dr. Florian Tschorsch
M.Sc Julian Fietkau



Uncovering Obfuscated Fingerprinting Techniques. A Large Scale Security and Privacy Analysis of Device Identification.



Agenda

- Thesis Goal
- Web-Device Fingerprinting
- Design & Implementation
- Evaluation
- Conclusion
- Future work



Thesis Goal

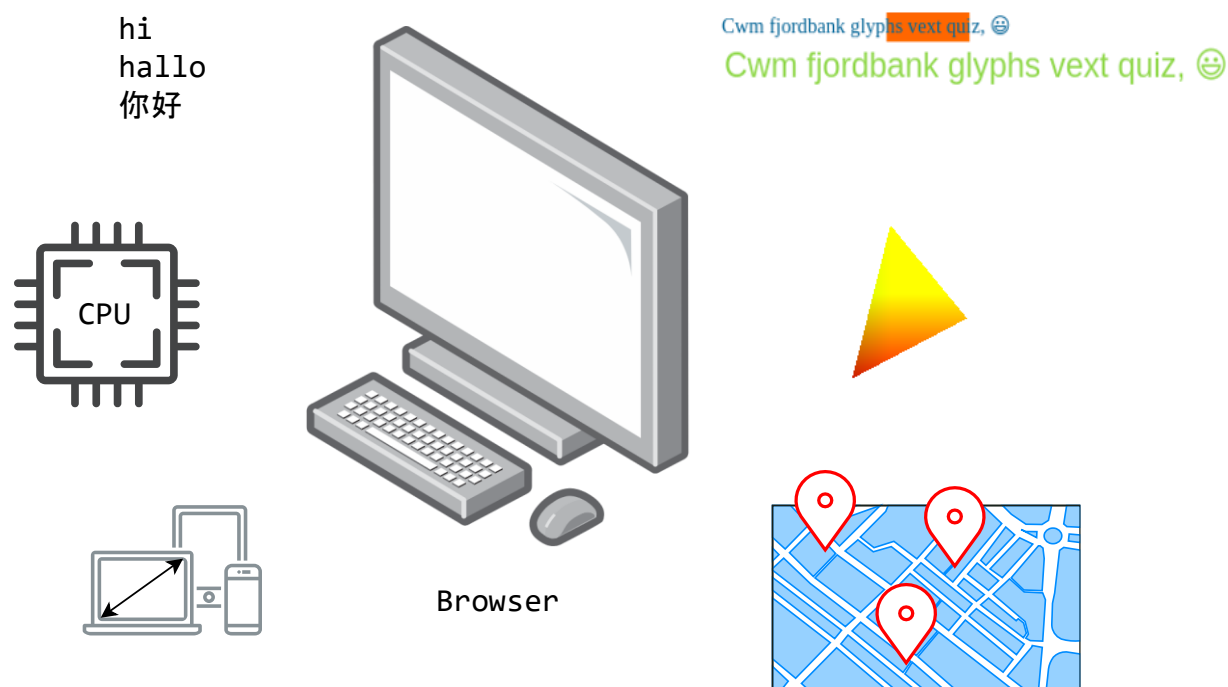
Investigate fingerprinting techniques on client browsers in order to study the current spread of this technology.

Create a way that enables privacy conscious users to gain insight into obfuscated fingerprinting activities.

Approach:

- Creating a browser extension that monitors current fingerprinting activity in real time
- Create a crawling environment for scanning large lists of websites
- Correlation of similarities of different fingerprinters
- Verify the privacy impact of other browser extensions and settings

Introduction – Web-Device Fingerprinting



Browsers have capabilities to interact with a multitude of system level components and software:

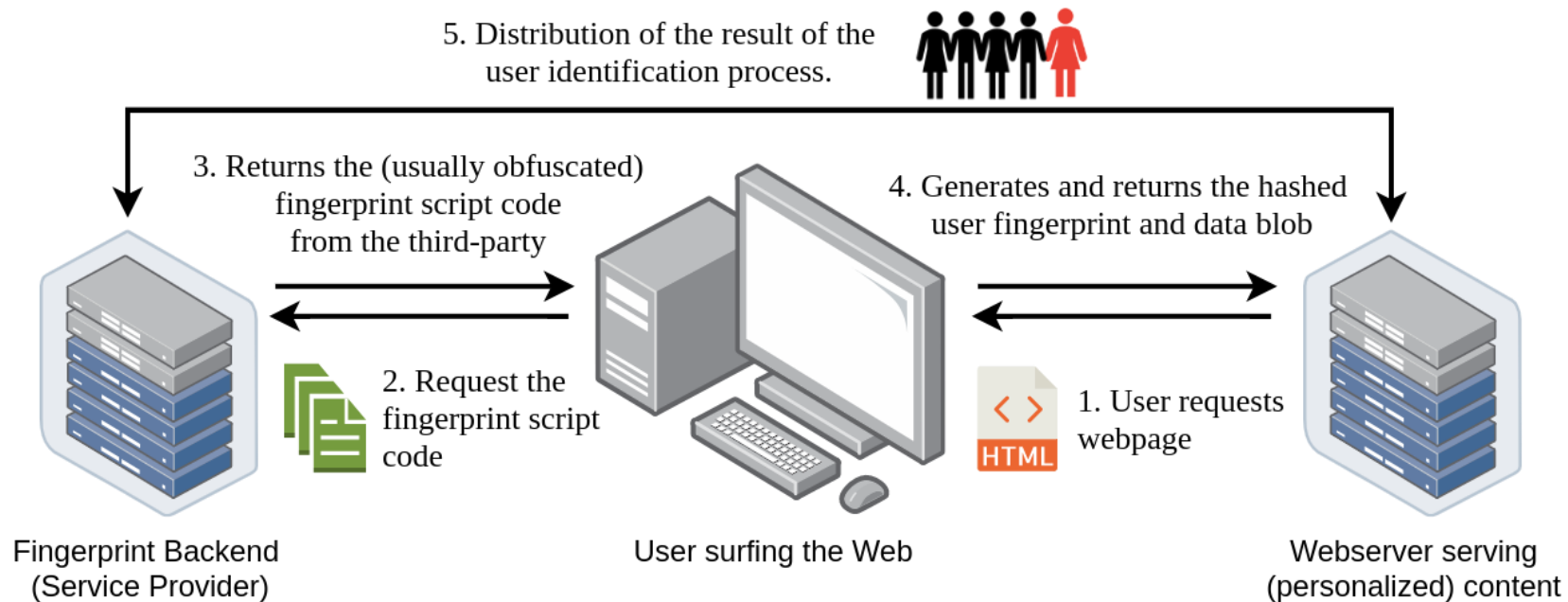
- Language settings
- System Information
- Screen settings
- Graphics cards, Audio processors
- Canvas API
- WebGL vendor and renderer
- Geolocation

= Every user transmits a lot of information and metadata while surfing the web.

Introduction – Web-Device Fingerprinting

- [69] Naoki Takei et al. “Web browser fingerprinting using only cascading style sheets”. In: *10th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA)*. IEEE. 2015, pp. 57–63.
- [70] Keaton Mowery and Hovav Shacham. “Pixel perfect: Fingerprinting canvas in HTML5”. In: *Proceedings of W2SP* (2012), pp. 1–12.
- [46] Pierre Laperdrix et al. “Morellian Analysis for Browsers: Making Web Authentication Stronger with Canvas Fingerprinting”. In: *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer. 2019, pp. 43–66.
- [6] Browser-fingerprinting using the AudioContext and Canvas API. <https://www.eff.org/de/deeplinks/2018/06/gdpr-and-browser-fingerprinting-how-it-changes-game-sneakiest-web-trackers>. Accessed: 2020-05-20.
- [32] Gabi Nakibly, Gilad Shelef, and Shiran Yudilevich. “Hardware fingerprinting using HTML5”. In: *arXiv preprint arXiv:1503.01408* (2015).
- [52] Nick Nikiforakis et al. “Cookieless monster: Exploring the ecosystem of web-based device fingerprinting”. In: *2013 IEEE Symposium on Security and Privacy*. IEEE. 2013, pp. 541–555.
- [50] Gunes Acar et al. “The web never forgets: Persistent tracking mechanisms in the wild”. In: *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. 2014, pp. 674–689.
- [5] Jonathan R Mayer and John C Mitchell. “Third-party web tracking: Policy and technology”. In: *2012 IEEE Symposium on Security and Privacy*. IEEE. 2012, pp. 413–427.
- [60] Peter Eckersley. “How unique is your web browser?” In: *International Symposium on Privacy Enhancing Technologies Symposium*. Springer. 2010, 1–18.
- [14] Antoine Vastel et al. “FP-STALKER: Tracking browser fingerprint evolutions”. In: *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2018, pp. 728–741.

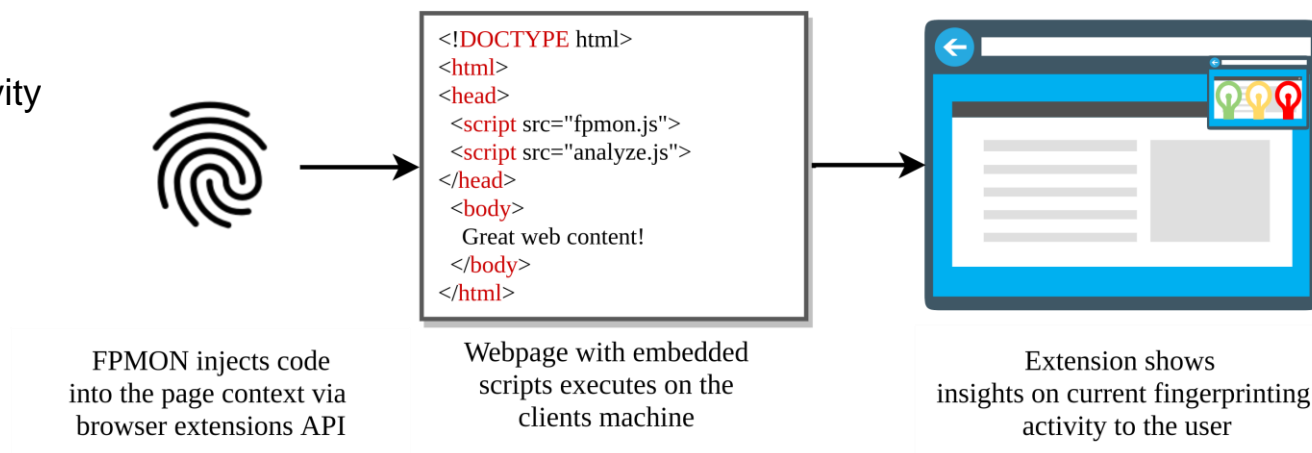
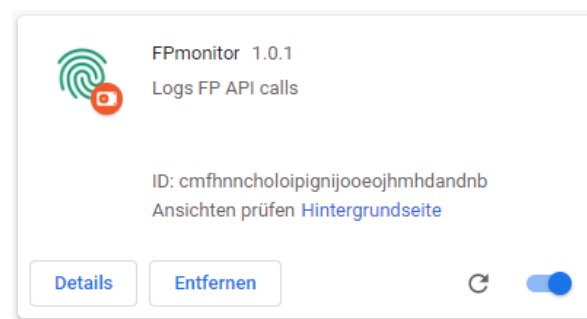
Introduction – Scope and Threats



Design – FPMON

How does FPMON work?

- Overriding JavaScript objects and properties
- Added a callback functionality which notifies and logs activity in realtime
- Visualizing fingerprinting activity and threat for the user
- Easy to use, usable for everyone
- Not impacting browser performance



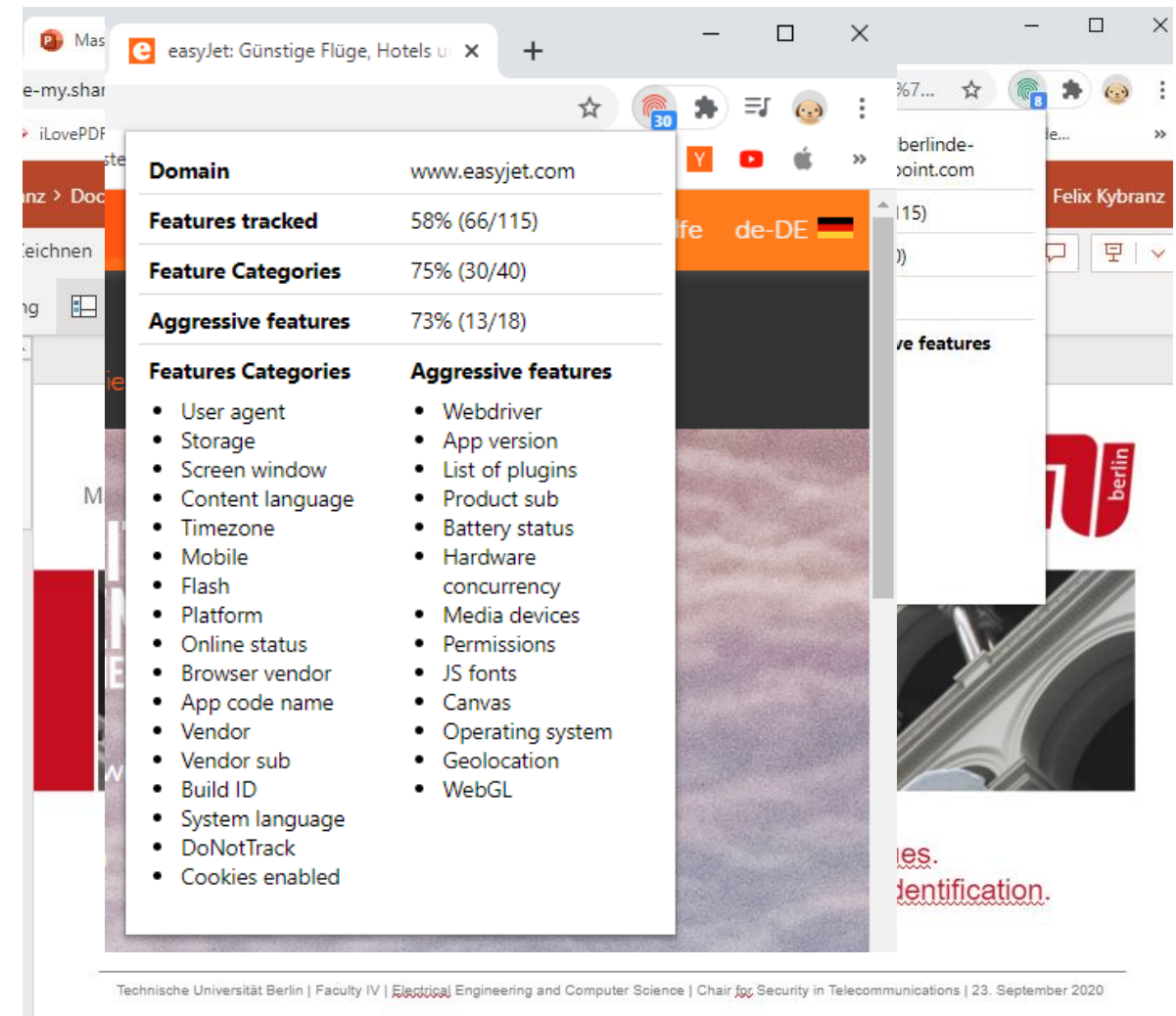
Design – FPMON

Sensitive Features:

- used in an intended manner
- necessary for user experience
- identifying a user needs several sensitive features simultaneously
- can be low and high level of entropy

Aggressive Features:

- used in an unintended way
- usually not used on sites for other purposes than fingerprinting
- single features can identify a user
- high level of entropy



Domain: www.easyjet.com	
Features tracked	58% (66/115)
Feature Categories	75% (30/40)
Aggressive features	73% (13/18)

Features Categories	Aggressive features
• User agent	• Webdriver
• Storage	• App version
• Screen window	• List of plugins
• Content language	• Product sub
• Timezone	• Battery status
• Mobile	• Hardware concurrency
• Flash	• Media devices
• Platform	• Permissions
• Online status	• JS fonts
• Browser vendor	• Canvas
• App code name	• Operating system
• Vendor	• Geolocation
• Vendor sub	• WebGL
• Build ID	
• System language	
• DoNotTrack	
• Cookies enabled	

Technical Identification.

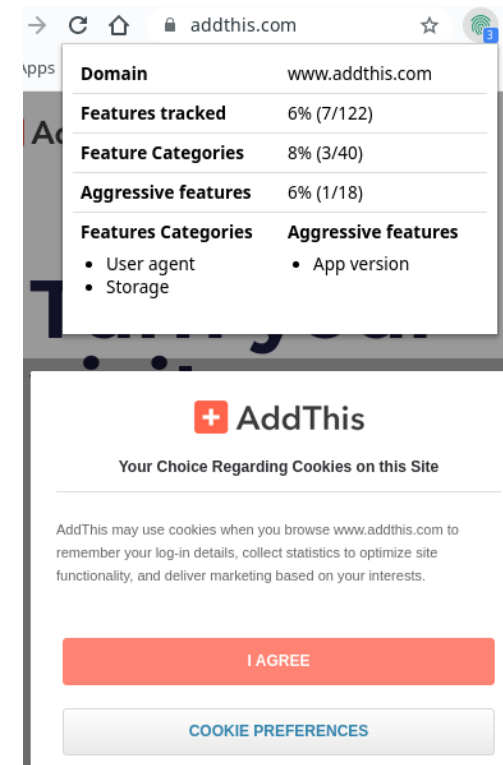
FPMON & FPCRAWL

FPMON can be further used to validate the **privacy impact** on:

- Cookiebanners
- Different browsers
- Privacy settings

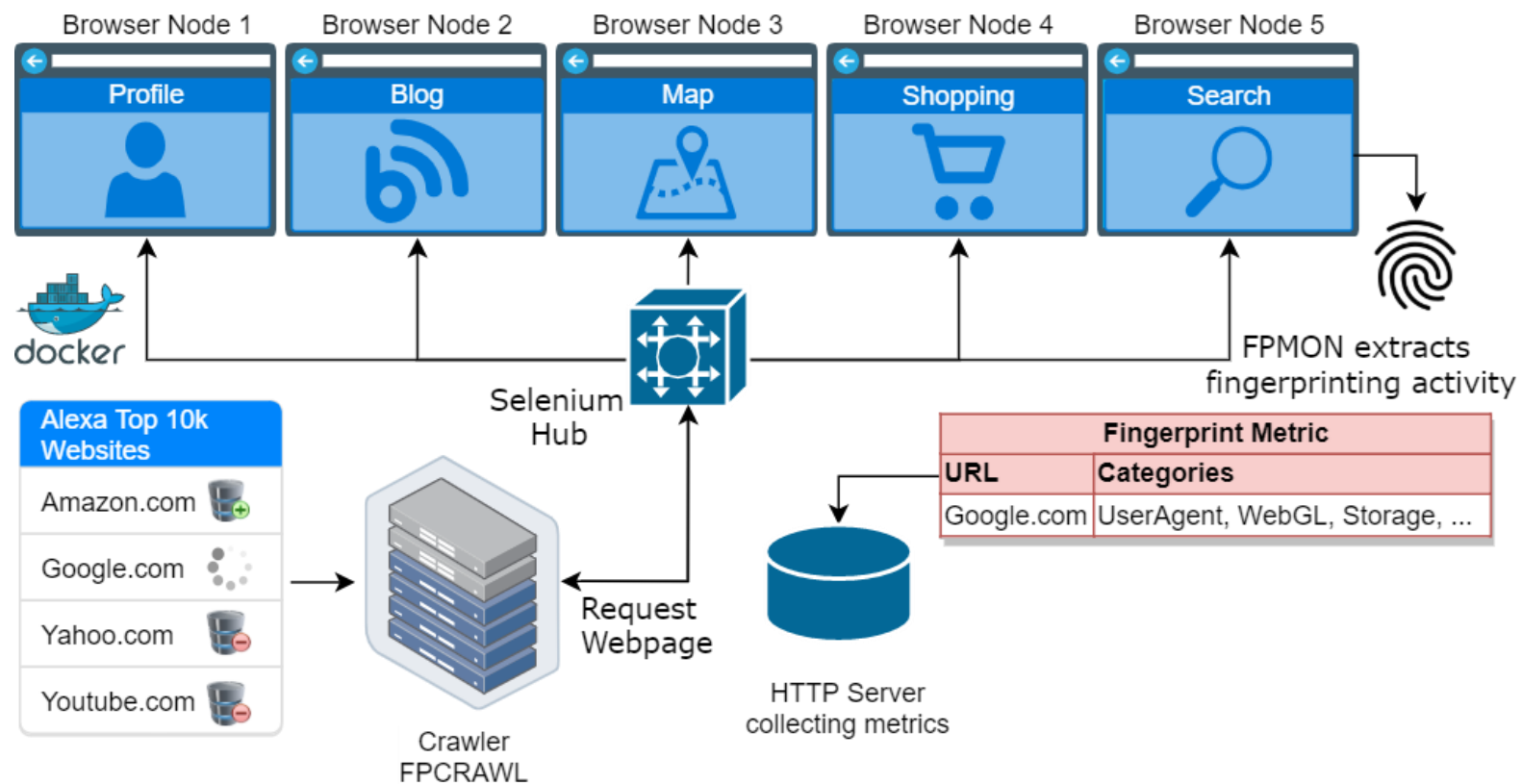
FPCRAWL crawling environment for automated scanning:

- Cookiebanner scan
- Extension scan
- Firefox standart vs. Firefox strict-mode
- Alexa 10k



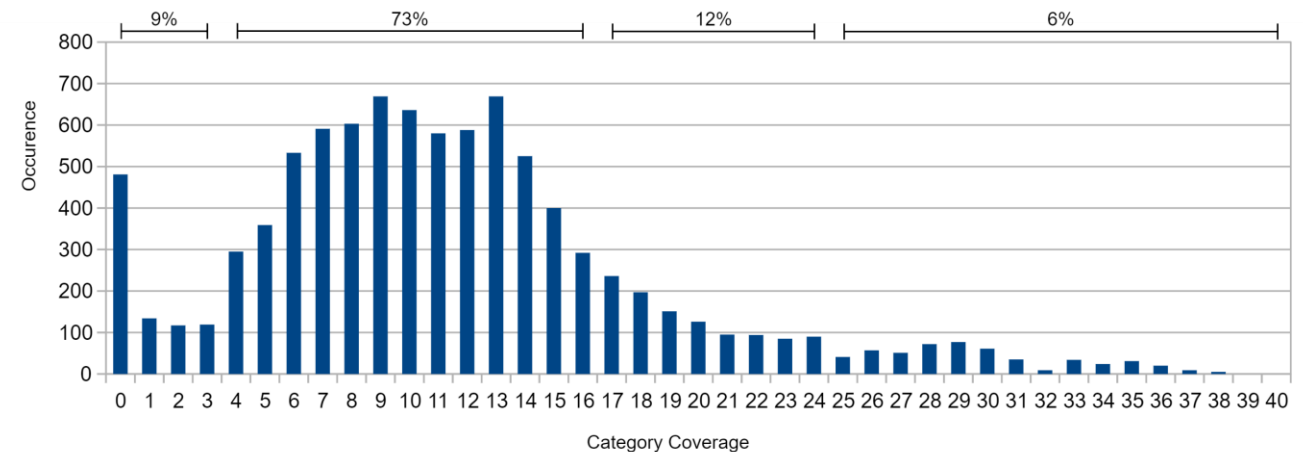
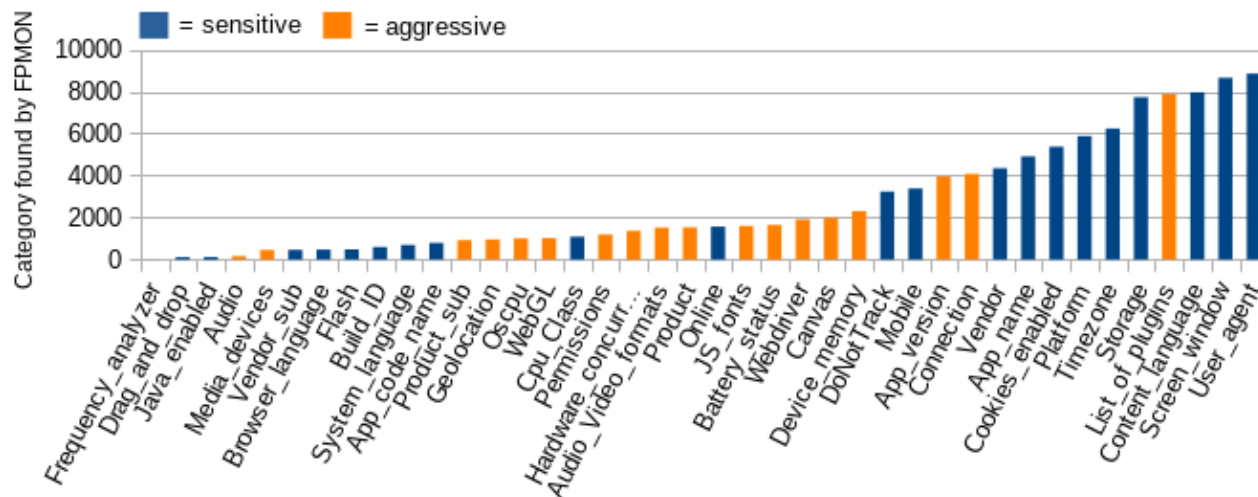
Domain	www.addthis.com
Features tracked	50% (61/122)
Feature Categories	85% (34/40)
Aggressive features	73% (13/18)
Features Categories	Aggressive features
<ul style="list-style-type: none"> User agent Storage Screen window Content language Browser vendor DoNotTrack Platform Product Vendor Vendor sub Build ID CPU class Mobile App code name Cookies enabled Java enabled Browser language System language Online status Drag and drop Timezone 	<ul style="list-style-type: none"> App version List of plugins Product sub Operating system Hardware concurrency Webdriver Media devices Permissions Canvas Geolocation Battery status Device memory Connection

Design – FPCRAWL Architecture



Evaluation – Result 10k Scan

- **91%** of websites use fingerprinting
- **18%** of them very aggressive



- **Highest** are the one used for enhance user experience
- **Middle** specifically used to aggressively fingerprint users
- **Lowest** rarely used and are edge cases mostly for browser functionality

Evaluation – Privacy impact on Extensions and Cookiebanners

Domain	Content Topic	Cookie Banner Deactivated		Cookie Banner Activated		Score Difference
nasdaq.com	Stock Market	28/40	70%	12/40	30%	- 40%
healthcare.gov	Healthcare	22/40	56%	21/40	53%	- 3%
medicare.gov	Healthcare	22/40	56%	21/40	53%	- 3%
vyprvpn.com	Privacy Service	19/40	48%	10/40	25%	- 23%
sba.gov	Government	16/40	40%	12/40	30%	- 10%
twitch.tv	Streaming	16/40	40%	14/40	35%	- 5%
reddit.com	Social Media	13/40	33%	9/40	23%	- 10%
spiegel.de	News	11/40	28%	18/40	45%	+ 17%
paypal.com	Finance	9/40	23%	10/40	25%	+ 2%
addthis.com	Fingerprinter	3/40	8%	34/40	85%	+ 77%
yahoo.com	Web Browsing	3/40	8%	9/40	23%	+ 15%
panopti-click.eff.org	Fingerprinter	1/40	3%	21/40	53%	+ 50%

Domain	Content Topic	Score Chrome	Ad-block	Duck-duckgo	Privacy Badger
metacafe.com	Video Sharing	95%	95%	95%	20%
easyjet.com	Flight Service	73%	73%	73%	73%
nasdaq.com	Stock Market	70%	70%	70%	68%
bankofamerica.com	Finance	65%	58%	65%	65%
savethechildren.org	Non-profit	63%	45%	23%	43%
nytimes.com	News	60%	58%	18%	60%
coinbase.com	Privacy Service	58%	58%	58%	58%
fingerprints.com	Fingerprinter	53%	53%	53%	53%
sba.gov	Government	40%	25%	10%	18%
theguardian.com	News	30%	23%	15%	18%

Privacy Impact validated



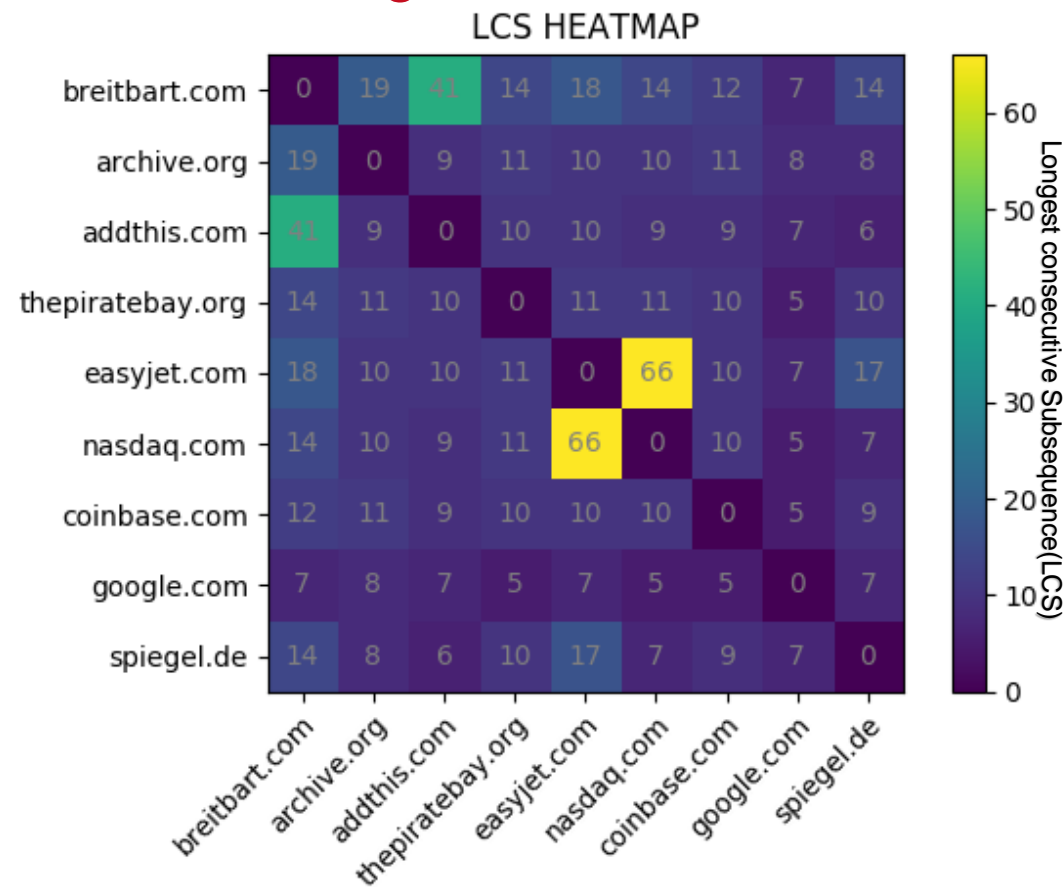
Privacy extensions protect users
(to a certain point)



Comprehensive defense
against fingerprinting



Evaluation – Signature Correlation



Domain	LCS	Cat.
globalnews.ca	79	34
express.co.uk	76	32
foxsports.com	76	32
nypost.com	75	31
moat.com	54	34
cbc.ca	49	33
oracle.com	49	33
codebeautify.org	45	33
breitbart.com	41	33
foursquare.com	40	30
... 70 more	38 – 79	20 – 34

(A) Signature correlation:
addthis.com

Domain	LCS	Cat.
dell.com	80	27
nike.com	80	27
vmware.com	80	27
adobe.com	77	27
foxnews.com	77	27
tiktok.com	77	27
ea.com	77	27
fedex.com	77	27
aeroflot.ru	77	27
easyjet.com	66	23
... 200 more	60 – 80	20 – 27

(B) Signature correlation:
nasdaq.com

Domain	LCS	Cat.
bild.de	39	5
zeit.de	39	5
motor-talk.de	39	5
stepstone.de	27	5
t-online.de	25	5
wetter.de	25	5
mobile.de	25	5
n-tv.de	25	5
tagesschau.de	25	5
heise.de	25	5
... 50 more	24 – 25	4 – 5

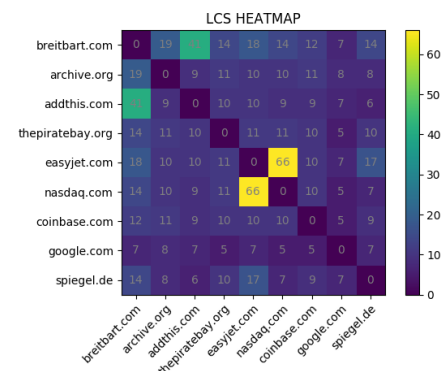
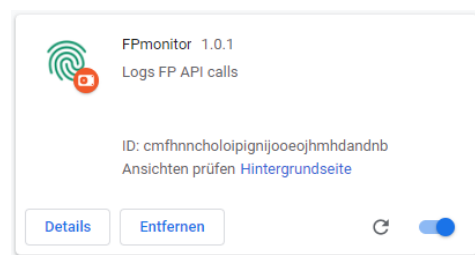
(A) Signature correlation:
spiegel.de




Domain	LCS	Cat.
aviasales.ru	93	20
tophotels.ru	91	20
torrentgalaxy.to	91	20
viatorrents.com	91	20
xhamsterlive.com	88	19
pccomponentes.com	88	19
chaturbate.com	88	19
anyporn.com	84	17
vk.com	72	16
coursera.org	70	16
... 140 more	40 – 91	9 – 20

(B) Signature correlation:
coinbase.com



Conclusion



- Revealed fingerprint activity, analyzed it and showed it to the user in real time. 
- Created an approach to actively monitor (any) JavaScript related functionality. 
- Scanned the web to analyze privacy for cookiebanners/browsers/fingerprint networks. 

Domain	www.addthis.com
Features tracked	50% (61/122)
Feature Categories	85% (34/40)
Aggressive features	73% (13/18)
Features Categories	Aggressive features
<ul style="list-style-type: none"> User agent Storage Screen window Content language Browser vendor DoNotTrack Platform Product Vendor Vendor sub Build ID CPU class Mobile App code name Cookies enabled Java enabled Browser language System language Online status Drag and drop Timezone 	<ul style="list-style-type: none"> App version List of plugins Product sub Operating system Hardware concurrency Webdriver Media devices Permissions Canvas Geolocation Battery status Device memory Connection



Future Work

- Improve granularity of detection mechanism
- Implement not yet covered fingerprinting methods
- Use transparent technologies in the www.



Thank you!

Prof. Dr. Jean-Pierre Seifert

Prof. Dr. Florian Tschorsch

Julian Fietkau

If you are further interested feel free to...

- + Read the thesis
- + Download the FPMON Browser Extension
- + Download the FPCRAWL CRAWLER
- + Ask me anything
- + Request the paper from Julian Fietkau

<https://github.com/kybranzf/thesis>

<https://github.com/kybranzf/fpmon>

<https://github.com/kybranzf/fpcrawl>

kybranz@win.tu-berlin.de

jfietkau@sec.t-labs.tu-berlin.de

Fingerprint Signature

```
var getCanvasFp = function (options) {  
  var result = []  
  var canvas = document.createElement('canvas')  
  canvas.width = 2000  
  canvas.height = 200  
  canvas.style.display = 'inline'  
  var ctx = canvas.getContext('2d')  
  ctx.rect(6, 6, 10, 10)  
  // [...]   
  ctx.fill()  
  ctx.fillStyle = 'rgb(255,0,255)'  
  ctx.arc(75, 75, 75, 0, Math.PI * 2, true)  
  ctx.fill('evenodd')  
  
  result.push('canvas fp: ' + canvas.toDataURL())  
  return result  
}
```

In FingerprintJS2 different objects initiate a canvas fingerprint:

- document.createElement('canvas')
- canvas.getContext('2d')
- canvas.toDataURL()

Canvas Signature:

canvas; canvas; canvas; canvas; canvas; canvas; canvas; canvas; canvas; canvas; canvas;

= 11x function calls from category canvas

Full Signature:

2xUserAgent; 10xLanguage; 25xCanvas; 7xScreen; 15xWebGL; 3xScreen; 2xGeolocation, ...

= has enough diversity to detect different fingerprinting scripts