Phishing for a title

Anonymous authors

ABSTRACT

ACM Reference Format:
Anonymous authors. 2024. Phishing for a title. In . ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

2 BACKGROUND

2.1 Evolution of Phishing over time

2.2 PhishingKits as a service

2.3 Tracking phishing in the wild

3 RELATED WORK

Measurement studies of phishing sites: Recent work in analizing phishing websites from phishing feeds includes Crawlphish [3], which uses a modified webkit engine to force execute all the paths in client-side javascript to identify cloaking behavior, Rods with Laser Beams [2] which uses a extension to capture set list of fingerprinting APIs, and identified the user of third party fingerprinting scripts in phishing pages that are not of the original target page, and Catching Phishers By Their Bait [1], which studied and identified phishing kits via manually crafted DOM and Javascript fingerprints.

Phishing kit analysis: Prior work has examined phishing kits to identify tradecraft within the ecosystem , studying similarity of kits observed , or in order to identify pages targetting a paticular theme.

Phishing page identification: Everyone under the sun has thrown ML at this problem, some state of the art work has looked at constructing knowlage based detection tools, and ofc, someome has done LLM To our knowlage, we are the first paper to automatically isolate, and craft detection fingerprints, and we are first to do it at the level that VV8 lets us do it.

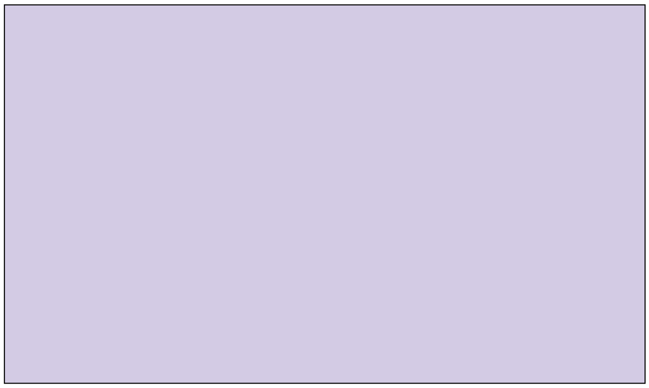

Figure 1: Crawling infrastructure

4 METHODOLOGY

4.1 Phishing Feeds and crawling

4.2 Data post processing

4.3 Client-side kit detector

4.4 Target identification and crawling

4.5 Interactive phishing

5 RESULTS

5.1 Javascript APIs

5.1.1 Fingerprinting APIs.

- Crypto.getRandomValues jumped from SMA of 20% to 80% of the daily traffic (184 domains to 810) February 8th 2024
- The usage of seed APIs used by Su at el.[**fptechniques-www23**] (well know APIs that trackers use) has been steadily decreasing from 30% of the daily traffic to 15%.

5.1.2 MDN API groups.

- We collected **134** MDN API categories
- HDBSCAN clustering of MDN API categories over time

5.1.3 Experimental APIs and WASM.

- We discover presese of 14 experimental MDN categories
- TODO: Do these categories go up or down over time?
- We observed using WASM modules for client-side bot detection via captchas

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

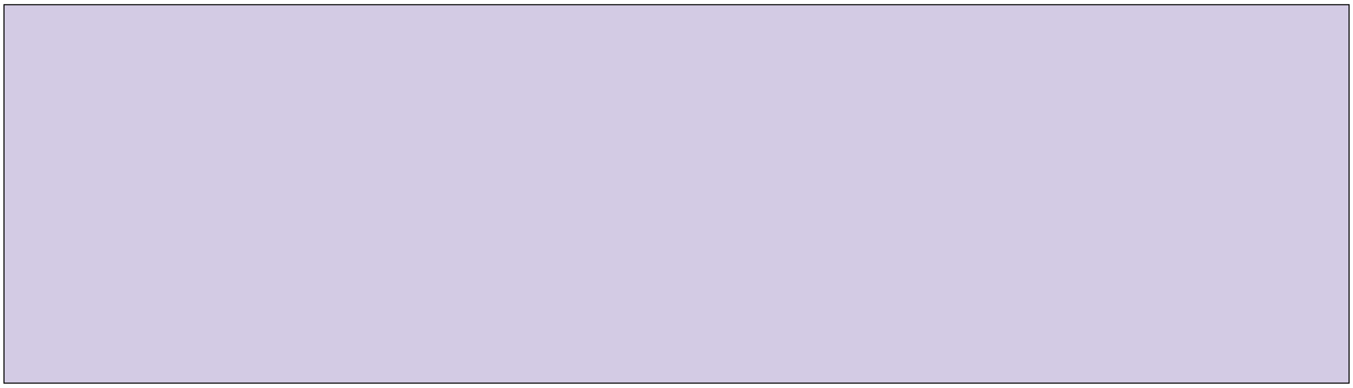

Figure 2: Server-side and Client-side cloaking over time

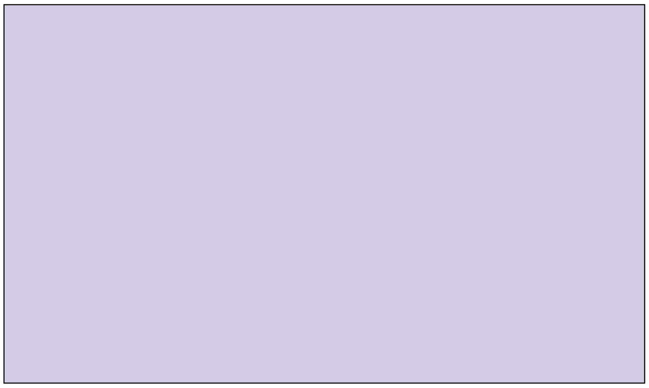

Figure 3: Crypto.getRandomValues API call over time

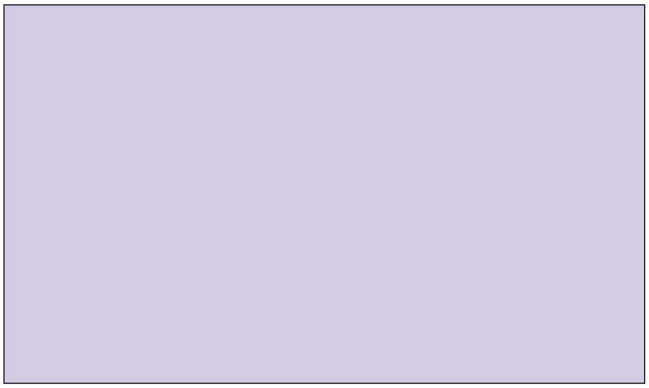

Figure 4: Seed FP APIs vs all FP APIs discovered by Su *et al.*

5.1.4 *Fingerprintability of Kits.*

- Using a set of MDN API categories as a metric, the mean infra-kit jaccard similarity XX (min: XX std: XX) while the mean intra-kit jaccard similarity was XX (min: XX std: XX)
- TODO: Compresison to real tranco pages
- TODO: Compresison to target pages

5.1.5 *Interactive JS logs.*

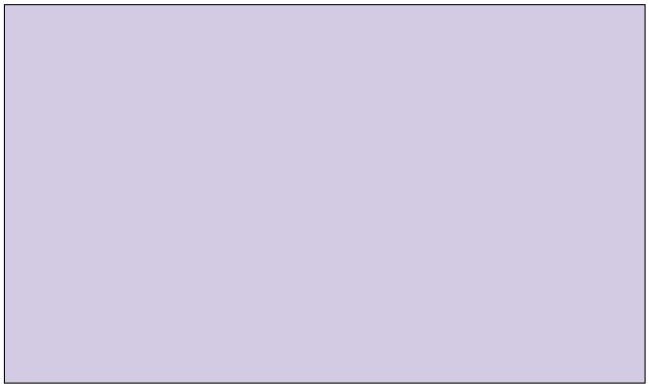

Figure 5: Clusters of seen kits

-

5.1.6 *Client-side cloaking tactics.*

-

5.1.7 *First Party/Third Party embedded/Third Party Scripts.*

5.2 Kit families

-

5.2.1 *Offline evaluation of detector.*

- We evaluate the detector on 70% of the domains that have a phishing kit attached
- We evaluate it on accuracy and ability to distinguish new kits. TODO THIS IS WHAT WE NEED TO DO

5.2.2 *Kit fragments and inheritence.*

-

5.2.3 *First Party script fragments and inheritance.*

- Using the SHA hash of scripts, we can detect scripts that some pages emmbed as first party while other load from a third party source.
-

5.3 Kit detection

5.3.1 Kit detection via javascript.

-

6 CASE STUDIES

6.1 Open Source client components

- Git directories in phishing kits
- Finding OSS components in client-side javascript via Github API

6.2 Mobile targets

- MDN distribution v. other sources
- APIs specific to mobile devices

6.3 Comparison to target pages

-

7 LIMITATIONS AND FUTURE WORK

7.1 Obfuscation and Flow analysis

7.2 Automated submissions

8 CONCLUSION

REFERENCES

[1] Hugo Bijmans, Tim Booij, Anneke Schwedersky, Aria Nedgabat, and Rolf van Wegberg. 2021. Catching Phishers By Their Bait: Investigating the Dutch Phishing Landscape through Phishing Kit Detection. In 30th USENIX Security Symposium (USENIX Security 21), 3757–3774. ISBN: 978-1-939133-24-3. Retrieved Mar. 12, 2024 from https://www.usenix.org/conference/usenixsecurity21/presentation/bijmans.

[2] Iskander Sanchez-Rola, Leyla Bilge, Davide Balzarotti, Armin Buescher, and Petros Efstathopoulos. 2023. Rods with Laser Beams: Understanding Browser Fingerprinting on Phishing Pages. In 32nd USENIX Security Symposium (USENIX Security 23), 4157–4173. ISBN: 978-1-939133-37-3. Retrieved Mar. 12, 2024 from https://www.usenix.org/conference/usenixsecurity23/presentation/sanchez-rola.

[3] Penghui Zhang et al. 2021. Crawlphish: large-scale analysis of client-side cloaking techniques in phishing. In *2021 IEEE Symposium on Security and Privacy (SP)*, 1109–1124. DOI: 10.1109/SP40001.2021.00021.