FIE-Comps Bibliography

Research Questions:

1. **How do malicious browser extensions operate, and what are the most common attack techniques?**
2. **What techniques have been developed to detect or analyze malicious browser extensions, and how effective are they?**
3. **What static and dynamic indicators (permissions, APIs, network behavior, obfuscation) are most useful for flagging risky extensions?**
4. **What programs or tools already exist that can support building and evaluating security scanners?**

RQ1: Malicious Browser Extension Behavior & Attack Techniques

Robertson, Ben, and Guy Katzir. "Malicious Browser Extensions: An Overlooked Security Threat." Grip.security, July 2, 2025. https://www.grip.security/blog/browser-extensions-security-threat

- Gives a summary of the security threat browser extensions pose and a breakdown of the security breach. Also includes a helpful table of traditional control types that extension risks sneak past.
- I think this is useful because of the table it gives, which allows us to brainstorm starting points on what we should be evaluating with our scanner.

InstaTunnel. "Browser Extension Malware: The Trojan Horse in Your Dev Tools 🔧." Medium, October 24, 2025. https://medium.com/@instatunnel/browser-extension-malware-the-trojan-horse-in-your-dev-tools-c9efd4fd058d.

- Explains how malicious extensions operate and breaks down the anatomy of browser extension attacks. Also explains and expands on a few real world cases of when extensions turned malicious. Also explains what malicious extensions steal.
- I think this could be a very important source, because it gives a lot of information on common attacks, methods, and prevention methods.

Wolkstein, Eric. "Top 5 Browser Extension Security Risks and 5 Ways to Prevent Them." Seraphic Cyber Security, July 3, 2025.

https://seraphicsecurity.com/learn/browser-security/top-5-browser-extension-security-risks-and-5-ways-to-prevent-them/.

- Gives common security risks in browser extensions, best practises, and evaluates the overall security problem .
- I don't think this might be the most useful as it doesn't say anything new that isn't already covered with the sources above. It also doesn't have that much info.

McGraw, Zachary. "Uncovering Browser Extension Risk: What Users Should Know." Vision Computer Solutions, December 29, 2025. https://www.vcsolutions.com/blog/browser-extension-risk-what-every-user-must-know/.

- Gives a breakdown of how extensions compromise data privacy, and also how to identify signs of malicious browser extensions.
- Could be important because of its case studies.

Owasp.org. "Browser Extension Vulnerabilities - OWASP Cheat Sheet Series." Cheat Sheets Series Team, 2024. https://cheatsheetseries.owasp.org/cheatsheets/Browser_Extension_Vulnerabilities_Cheat_Sheet.html.

- Gives examples of code that extensions could use to be malicious.
- Could be important if we use Yara rules.

Singh, S., Varshney, G., Singh, T. K., Mishra, V., & Verma, K. (2025). A study on malicious browser extensions in 2025. arXiv preprint arXiv:2503.04292

- Talks about experiments that a group that bypassed security mechanisms of firefox and chrome. Their experiments could provide insights into browser extension risks.
- I think this could be useful, if we don't get enough information form the sources above. But I'm not sure how much.

RQ2: Detection and Analysis Techniques

Shahriar, Hossain, Komminist Weldemariam, Mohammad Zulkernine, and Thibaud Lutellier. "Effective detection of vulnerable and malicious browser extensions." Computers & Security 47 (2014): 66-84.
https://www.sciencedirect.com/science/article/pii/S0167404814000984

Summary:

Proposes an ML approach to classify and detect malicious browser extensions, building their model with Hidden Markov Model constructs and monitoring API calls & user interaction. The model identifies examples as benign, vulnerable, and malicious behaviors. It has a high accuracy, even detecting previously undetected malicious.

Thoughts:

Unless we want to follow an ML approach with our project, I don't see this article being too helpful in that regard (though a lot of detection methods use ML ideas). But the article does provide other helpful information. It explains the general layout and types of browser extensions with some examples of malicious and non-malicious code snippets. It also tackles the issue of finding malicious samples by defining rules to generate samples, which sounds extremely relevant to our proposed issue of actually finding samples.

Pantelaios, Nikolaos, Nick Nikiforakis, and Alexandros Kapravelos. "You've changed: Detecting malicious browser extensions through their update deltas." In Proceedings of the 2020 ACM SIGSAC conference on computer and communications security, pp. 477-491. 2020.
https://dl.acm.org/doi/abs/10.1145/3372297.3423343

Summary:

Proposes a novel two-step malicious extension approach by first using user feedback (comments & reviews) and then extension updates to identify clusters of MBEs (malicious browser extensions). Of flagged extensions, 44% of them were still available to download from the Chrome Store.

Thoughts:

Definitely introduces a sector of information that seems to be overlooked (public user feedback), which might be worth our attention when tallying the "potential maliciousness" of an extension. They also have open-source code and a public dataset of the malicious clusters they identified.

Kaushik, Keshav, Sakshi Aggarwal, Shambhavi Pandey, Shashank Mudgal, and Saksham Garg. "Investigating and Safeguarding the Web Browsers from Malicious Web Extensions." GRD Journal for Engineering 6, no. 10 (2021).
https://dl.acm.org/doi/pdf/10.1145/3372297.3423343

Summary:

Proposes “ExtAnalysis”, a user-friendly tool for investigating and analyzing browser extensions through permission requirements, API calls, and external Js. Advises users whether an extension is flagged as malicious or not.

Thoughts:

Seems like a student project to draw non-CS background users’ attention to malicious extensions by providing a clean GUI to unpack and analyze any given extensions. May not be groundbreaking, but provides a good idea for a potential structural layout of our project.

Kapravelos, Alexandros, Chris Grier, Neha Chachra, Christopher Kruegel, Giovanni Vigna, and Vern Paxson. "Hulk: Eliciting malicious behavior in browser extensions." In 23rd USENIX Security Symposium (USENIX Security 14), pp. 641-654. 2014. **https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/kapravelos**

Summary:

Proposes a dynamic analysis system that induces and identifies malicious behavior by probing unauthorized data access and page content tampering through HoneyPages and fuzzer drives. Analyzed 48k extensions and identified a number of malicious extensions, including one affecting 5.5 million users.

Thoughts:

The approach to detecting malicious extensions seems a little too complicated for us, but it could be worth investigating. The paper does answer other important questions, such as how to trigger malicious behavior and how to implement a honey trap if we choose to do so.

RQ3: Indicators of Risk (Permissions, APIs, Network, Obfuscation)

IritT. “Intro to Malware Analysis— SOC Level 1 -Digital Forensics and Incident Response — TryHackMe Walkthrough & Insights.” Medium, January 8, 2025. https://iritt.medium.com/intro-to-malware-analysis-fcd69f370391.

This source provides a nice introduction to malware and malware analysis by describing malware, its purpose, techniques of malware analysis, how to build sandboxes and how to actually analyze malware (Static Malware Analysis vs Dynamic Malware Analysis) in this environment. It is highly interactive and does explain obfuscation as an indicator of risk. This source is highly useful for our general understanding of malware and teaches us some practical uses of testing it.

Atak, Sinan Ugur. “Static Malware Analysis vs Dynamic Malware Analysis - Comparison Chart.” Malwation.com, 2024. https://www.malwation.com/blog/static-malware-analysis-vs-dynamic-malware-analysis-comparison-chart.

This source goes deeper into comparing Static Malware Analysis and Dynamic Malware Analysis. The article explains the two main techniques used in malware analysis—static and dynamic—and compares their strengths, weaknesses, and use cases in cybersecurity. It concludes that neither method alone is sufficient for modern malware defense. Static analysis helps with quick detection and classification, while dynamic analysis reveals true behavior and hidden actions. Combining both approaches gives security teams a stronger, more comprehensive understanding and defense capability. I think this gives us insight as to how to conduct analysis on malware when we get there.

Maria-Mădălina Andronache, Alexandru Vulpe, Corneliu Burileanu, Maria-Mădălina Andronache, Alexandru Vulpe, and Corneliu Burileanu. “Integrated Analysis of Malicious Software: Insights from Static and Dynamic Perspectives.” *Journal of Cybersecurity and Privacy* 5, no. 4 (November 10, 2025): 98–98. https://doi.org/10.3390/jcp5040098.
https://www.mdpi.com/2624-800X/5/4/98

RQ4:

https://spin.ai/blog/best-crxcavator-alternative-browser-extension-risk-assessment/

- Gives some background on CRXcavator and could be useful for us to finalize how we see our final product.
- Could be useful, but I also need to find more sources on how CRXcavtor works.

Scan-tastic! The Best Online Web Security Scanners for Your Website, Concertium, April 21, 2025.https://concertium.com/online-web-security-scanner/

- This article provides an overview of online web security scanners, explaining how they work, the types of vulnerabilities they detect, their benefits, limitations, and their role in compliance and cybersecurity strategy.
- This resource is very useful because it offers a practical perspective on vulnerability scanning tools, making it useful for understanding real-world web security practices, tool capabilities, and limitations relevant to a cybersecurity-focused project.