Malicious Browser Extensions

Shishir Jindal, Tanmay Tiwari, Yogesh Biyani

Department of Computer Science and Engineering, Indian Institute of Technology Roorkee

Abstract

Through this paper, we project our work on the domain of Analysis of Malicious Browser Extensions for their classification and detection. There has been already a lot of work done on the detection of Native malware programs but no substantial research on Malicious extensions could be seen yet which brings with it one of the very challenging aspects faced by us, which is to collect an extensive dataset of malicious extensions in order to extract features correspondng to their threat-posing behaviour. Our research mainly focuses on the classification and detection of malicious extensions in which the most common maline behaviors and defence techniques are identified.

Introduction

Malware is one of the major security threats faced by the Internet today. The web browser is our main interface to the Internet. We have browser extensions, which can be easily added to the browser, offering functionalities such as changing the appearance of web pages, improving browsing security, blocking ads etc. Browser extensions are implemented with standard web technologies and are written by third parties. However, not all third parties have their best interest for the end-user. Malicious browser extensions are being leveraged in various types of attacks, ranging from data theft to spying. Due to a constant increase in volume, velocity and complexity of such malicious extensions, there arises an imperative need, now more than ever, to develop methodologies which can automatically compute the threat posed by a particular piece of malicious extension(to a victim machine) as soon as it appears in the wild.

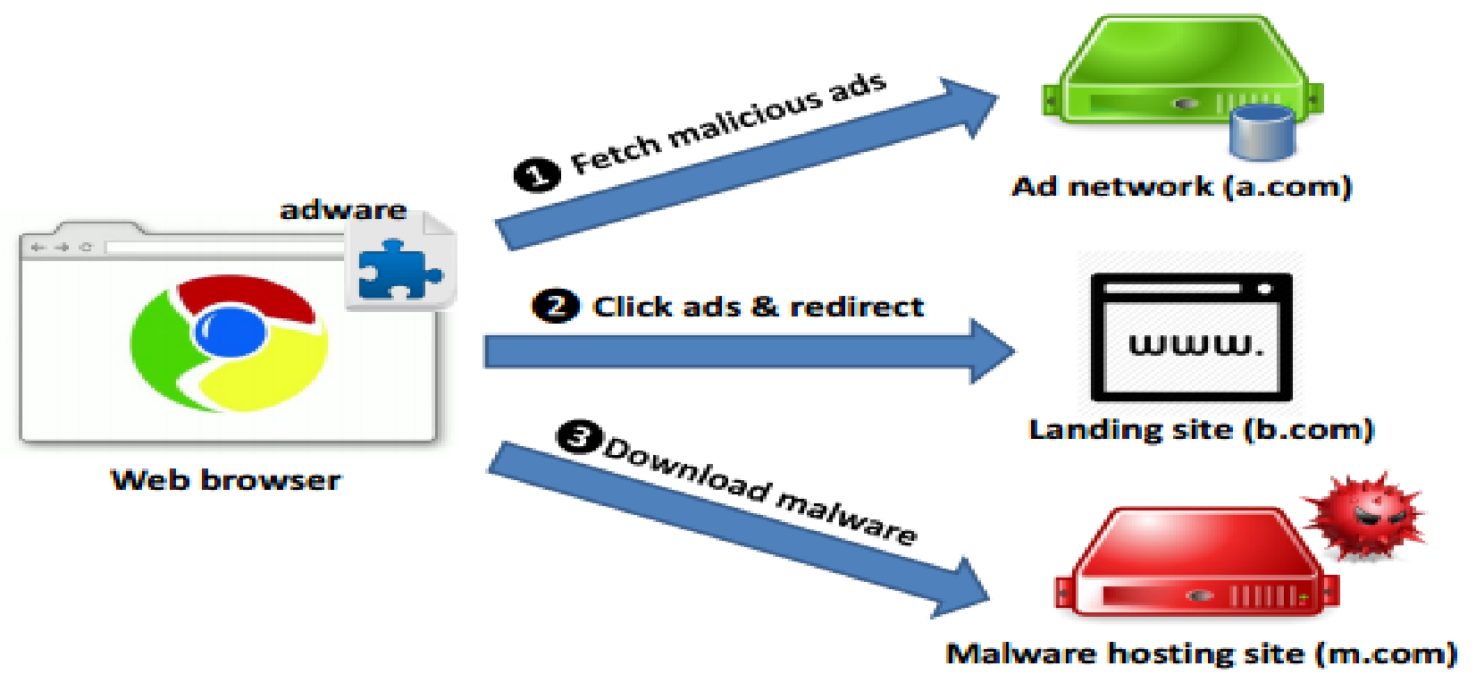

Figure: An Example of simple Malicious Extension

Dataset Collection

Dataset collection was a major challenge in this domain since there has not been any extensive research or analysis in this field so far. For this purpose, we developed a chrome extension scans for a link to an extension on the page and uploads it to a central repository. This extension was installed by some volunteers. Around 1600 extensions were collected over a total of 2 months. Also, 1300 benign extensions were collected from chrome's store.

Methodology

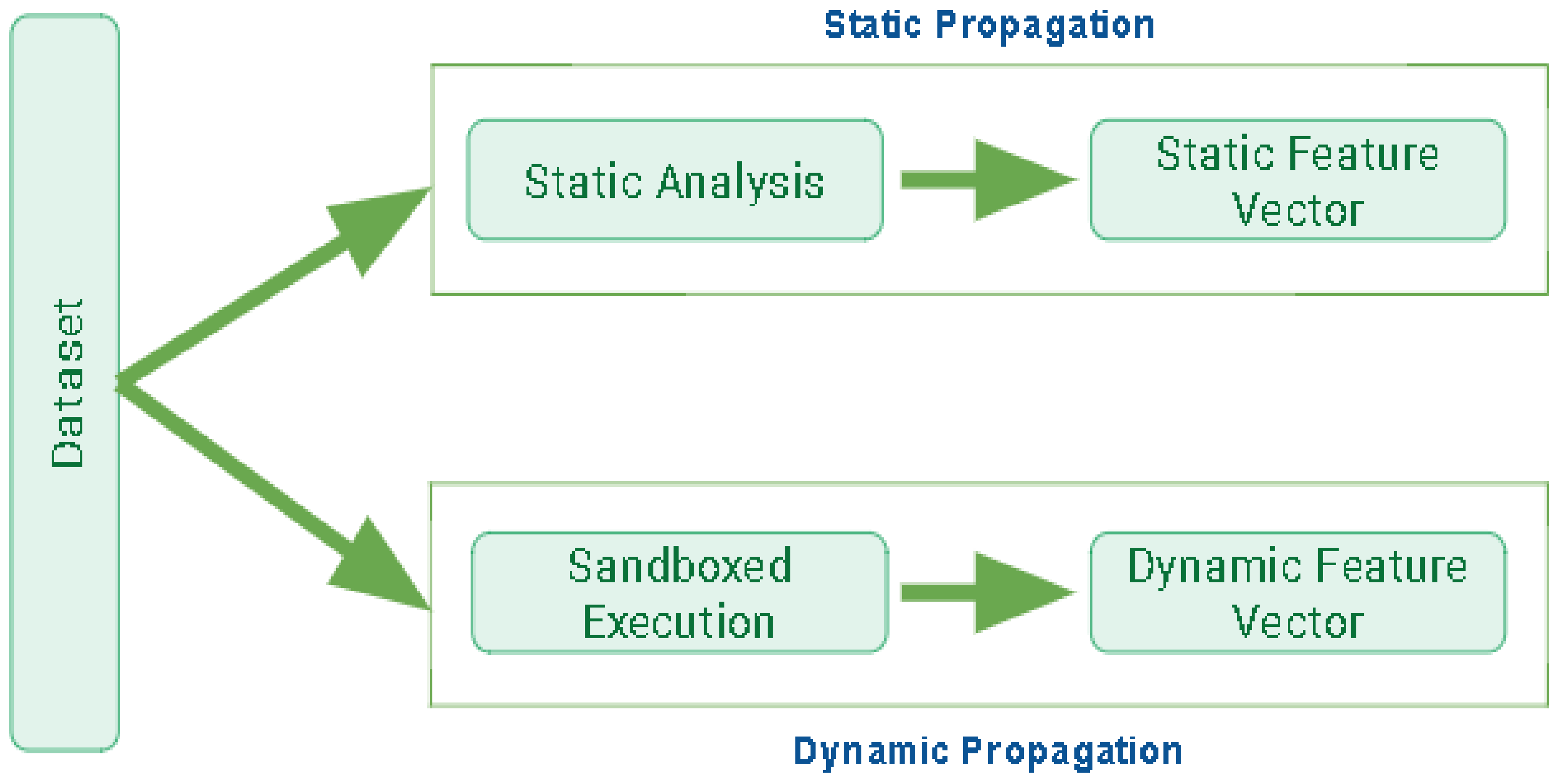

Figure: Flow Diagram

Analysis of an extension can be either static or dynamic.
Static analysis comprises of features extracted without actually executing it, eg. Source url, permissions etc.
For dynamic analysis, we need to run the extension in a sandboxed environment and log its behaviour, eg. data sent, page content modified, etc.
We will extract a feature vector corresponding to an extension using both static and dynamic analysis.
Next step is to determine an appropriate model and train it using the training set. We decided to use SVM because of its success in previous work on malware analysis.

SVM Classification

We will use various features in our model. For static analysis we will use features like the size of the js/binary files and the permissions the application is asking like opening tabs, some socket io permissions etc. For Dynamic analysis we will use features like the network logs captured, what all files the application is writing and reading in our system and the sequence of file execution etc.

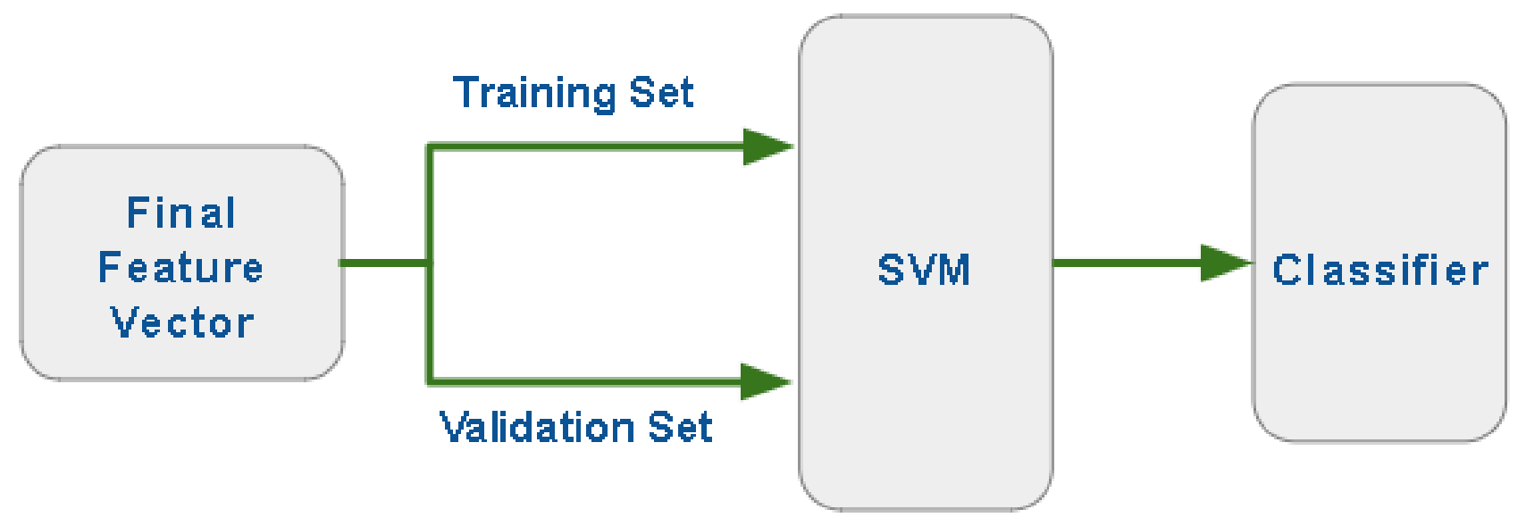

Figure: SVM Classifier on Dataset

Conclusion

We explore the novel model for Supervised Learning of features of the browser extensions using Support Vector Machine(SVM) classification Model. We dig into behaviour of malicious extensions(both static and dynamic malwares) by creating a sandboxed environment and learn patterns and repetitions based on the activities and executions performed by the maline extensions onto the target system. Finally, we were able to build a classifier which can predict wheter an extension is harmful or not and thus successfully classify the threat level.

References

[1] L. F. DeKoven, S. Savage, G. M. Voelker, and N. Leontiadis, "Malicious Browser Extensions at Scale." `https://cseweb.ucsd.edu/~ldekoven/publications/malicious_browser-extensions_at_scale.pdf/`.

[2] A. Guha, M. Fredrikson, B. Livshits, and N. Swamy, "Verified Security for Browser Extensions." `http://www.ieee-security.org/TC/SP2011/PAPERS/2011/paper008.pdf`.

[3] M. T. Louw, J. S. Lim, and V. Venkatakrishnan, "Extensible Web Browser Security." `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.86.8972&rep=rep1&type=pdf`.

[4] M. Dhawan and V. Ganapathy, "Analyzing Information Flow in JavaScript-based Browser Extensions." `https://www.cs.rutgers.edu/~vinodg/papers/acsac2009a/acsac2009a.pdf`.

[5] S. Bandhakavi, S. T. King, P. Madhusudan, and M. Winslett, "VEX: Vetting Browser Extensions For Security Vulnerabilities." `https://www.cs.rutgers.edu/~vinodg/papers/acsac2009a/acsac2009a.pdf`.