Document Engineering Issues in Malware Analysis

Charles Nicholas
University of Maryland, Baltimore County
Baltimore, Maryland 21250
nicholas@umbc.edu

ABSTRACT

We present an overview of the field of malware analysis with emphasis on issues related to document engineering. We will introduce the field with a discussion of the types of malware, including executable binaries, malicious PDFs, polymorphic malware, ransomware, and exploit kits. We will conclude with our view of important research questions in the field. This is an updated version of last year's tutorial, with more information about web-based malware and malware targeting the Android market.

CCS CONCEPTS

• **Security and privacy → Malware and its mitigation**; • **Applied computing →** *Document management and text processing*;

1 INTRODUCTION

Malware analysis has become an important field within the general area of cybersecurity. Skilled malware analysts are in high demand, and they are employed in cybersecurity firms, financial institutions, intelligence and law enforcement agencies, and other large organizations.

For many years, most malware was written for the Windows OS and the x86 architecture. Windows is still an important malware target, since so many PCs run it, but in recent years the amount of malware targeted to the mobile telephone, especially the Android, has grown enormously. Although it focuses on Windows XP, we have found that Sikorski's "Practical Malware Analysis" [1] is still the best single resource for this area.

2 TOOLS AND TECHNIQUES

In the tutorial we will present an overview of the field of malware analysis, with emphasis on topics we believe to be of special interest to the Document Engineering community. Teaching materials for Android malware are starting to become available, but for our purposes we will focus on the Windows environment, since that platform is more likely to be more familiar to more people.

Malware on the Windows platform is often, but by no means always, found in executable binaries. Malware can be examined in static form, e.g. by inspection of the PE header and the system call import table. Windows provides tools for such activity, and many third party tools do so as well. IDA is a powerful disassembler, which allows the analyst to examine a suspect binary in a variety of forms, including raw assembly code and call graphs. Basic IDA functionality can be augmented with plug-ins written in C or Python.

Malware can also be studied in dynamic form, that is, by running it and seeing what happens. OllyDbg is one of several powerful debuggers available for the Windows platform, which has gained a following among malware analysts. Dynamic analysis is usually done from the safe confines of a virtual machine, running under the auspices of VMWare, for example.

Some collections of malware specimens are available to researchers, and these will be used as examples as appropriate. Alas, there is no shortage of malware to be studied, since malware production is easily automated. Collecting malware specimens for analysis is an important sub-area, and anti-virus companies for example devote much effort to this.

As time permits, we will discuss recent and ongoing work in malware analysis-in-the-large, which (to us) refers to finding patterns and trends in collections of malware. Malware specimens can be subjected to cluster analysis, based on static and dynamic characteristics. Malware attribution is and will remain a difficult problem, for reasons which we will explain.

3 AUDIENCE PARTICIPATION

Tutorial participants are welcome to bring their own laptops. We recommend installing a virtual machine platform such as VMWare or Virtual Box, with virtual machines running Windows and Linux. Participants that have IDA Pro (the free version 5.0) and OllyDbg installed, as well as Microsoft's System Tools suite, may be able to run some examples with us. However, participants that choose to leave their laptops at home will be at no disadvantage.

Charles Nicholas is a professor of computer science at UMBC. He has been involved in the Document Engineering field for many years, and has recently turned his attention to the problems of malware analysis in the large. His recent work has considered questions related to storing, searching, and finding patterns in large collections of malware. He has taught a combined graduate-undergraduate course in malware analysis at UMBC for several years.

REFERENCES

[1] Michael Sikorski and Andrew Honig. 2012. *Practical Malware Analysis.* no starch press.

DocEng '17, September 04-07, 2017, Valletta, Malta

ACM ISBN 978-1-4503-4689-4/17/09.
https://doi.org/10.1145/3103010.3103027