

MALICIOUS SOFTWARE IDENTIFICATION THROUGH OPERATING SYSTEM MONITORING

Marcus F. Botacin (IC), André R.A. Grégio (PQ), Paulo L. de Geus (PQ)

Abstract

Desktop malicious programs are moving to different file formats and applying evasive techniques in order to continue their successful infections among end users computers, as well as to avoid being monitored by dynamic analysis systems. Publicly available malware dynamic analysis systems do not address these new variants: malicious programs encapsulated in CPL and Mono file formats, compiled for 64-bits systems and those that require a reboot to exhibit their malicious behavior have been forgotten, turning the remediation process difficult. This work introduces a dynamic analysis system able to handle the mentioned issues, providing information about the behavior of these other than ordinary malware samples and discuss the challenges faced in the design and implementation of a system like this when dealing with the security features present in modern Windows systems (NT 6.x kernel version).

Key words: Computer Security, Malicious Software, Malware Analysis

Introduction

Desktop malware continue to threaten end users on a daily basis. Despite all security mechanisms and defensive tools, phishing e-mail messages increase, thousands of malware variants arise, and operating systems protection are subverted by the very same old mechanisms. Besides that, malware developers usually apply the already known techniques of obfuscation, packing, modularity, multi-ciphering and polymorphism.

In terms of protection and defensive mechanisms, Microsoft Windows 8 is claimed to be the most secure version of this operating system due to the new security features, but there is a lack of studies about their effectivity.

To develop the analyzer, we studied these new security mechanisms, such as Kernel Patch Protection, Session Isolation and Driver Signing. The architecture proposed is based on callbacks and file system filters, implemented as a kernel driver, running on scalable Virtual Machines.

The system is able to monitor Registry, Process, Filesystem and network activities, monitoring different filetypes, such as PE+, CPL and Mono.

Results and Discussion

As far as we know, the presented system is the only one capable of analyzing PE+ files and providing a 64-bits malware analysis environment. 64-bits malware samples were submitted to the publicly available Web interface of popular

dynamic analysis systems and none of them were able to complete the analysis.

Besides that, It is also important to note that most labels provided by the antivirus software were based on general heuristics, which, albeit informing the user about the maliciousness of the files, do not provide information regarding the damage caused by them. Dynamic analysis systems such as the one presented in this paper are complementary to antivirus software, providing information about the behavior of malicious samples and making the identification of suspicious programs possible for those that do not have coded heuristics or signatures.

Conclusions

In this work, we introduced a dynamic analysis to monitor forgotten malicious programs, i.e., those malware that are not being currently addressed by available systems. We discussed the obstacles faced to implement such analysis system for 64-bits Windows system with security features and details about our proposed architecture.

We showed that our system is able to analyze PE, PE+, CPL, Mono and reboot-dependent malware, and that we extracted suspicious behaviors from these samples execution. This way, we provide a broader ability to analyze malware and to identify malignity even in unknown binaries.

Acknowledgement

Work under PIBIC-CNPQ grant number 118998/2014-5.

¹Botacin, M.; Afonso, V.; Geus, P. L. e Grégio, A.. *SBSEG, Belo Horizonte* 2014,.