

A Hands-on Digital Forensic Lab to Investigate Morris Worm Attack

Eric Xu Alex S. Xu Marriotts Ridge High School Marriotts Ridge High School Marriottsville, Maryland, U.S.A Marriottsville, Maryland, U.S.A exu1728@inst.hcpss.org axu1731@inst.hcpss.org Danny Ferreira Lin Deng University of Baltimore **Towson University** Baltimore, Maryland, U.S.A Towson, Maryland, U.S.A danny.ferreira@ubalt.edu ldeng@towson.edu Attack Machine Victim Machine worm.py (the first worm.py (again) worm.py (the badfile was was received worm.py was worm.py was last time) was created /executed/stopped executed read modified 08:10AM 2022-8-3 08:09 09:36AM overwrites

Figure 1: The Reconstructed Morris Worm Attacking Scenario in a Timeline.

ABSTRACT

We have developed a hands-on digital forensic lab to investigate the Morris Worm attack. In the poster, after the attack, we demonstrate a systematic approach to reconstructing the attack scenario by analyzing the worm's running processes, the networking communication used by running processes, and metadata of the files left on victims' machines.

INTRODUCTION

The Morris Worm was developed by Robert Tappan Morris in 1988. Despite Morris Worm being an old computer worm, it employs three key techniques to craft malware, including exploiting vulnerabilities, self-replication, and self-spreading. Using the Morris Worm as an example, we have developed a hands-on digital forensic lab to help students learn the fundamentals of digital forensic investigation, including identifying, collecting, and analyzing digital forensic evidence.

After simulating the Morris Worm attack with the SEED virtual machine [1], we focus on answering the following questions related

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SIGCSE 2023, March 15-18, 2023, Toronto, ON, Canada

© 2023 Copyright held by the owner/author(s). ACM ISBN 978-1-4503-9433-8/23/03. https://doi.org/10.1145/3545947.3576229

to cyber investigations: (1) What vulnerability was exploited? (2) How did the worm duplicate and spread? and (3) The reconstructed timeline of worm activities.

METHODS/ RESULTS / CONTRIBUTIONS

In our approach, we first use ps aux command to find suspicious processes running in the background. The discovered processes include bash (a running shellcode), python worm.py (the worm), and nc -lnv 9999 (a network utility). We also verify that a victim's machine was listening to the port 9999 for accepting worm.py. Finally, we monitor the timestamps (mtime, ctime, and atime) of worm.py and includingbadfile, a file that was associated with the worm. Figure 1 shows a reconstructed attacking scenario based on the timestamps of two files. The timeline reveals when the Morris Worm was created, viewed, and executed at the attack's machine, and when it was spread to the victim's machine. The lab materials are available at https://github.com/frankwxu/digital-forensics-lab.

ACKNOWLEDGMENTS

The work is partially supported by the U.S. National Science Foundation #2039289 and the Department of Justice #2019-DF-BX-K001.

REFERENCES

[1] Wenliang Du. 2011. SEED: hands-on lab exercises for computer security education. IEEE Security & Privacy 9, 5 (2011), 70-73.