Group Project Proposal

Project Title: How utilizing tools such as Wireshark and Suricata can be used as an effective way to implement a system to work as intrusion detection.

Proposers:

- Marissa “Mari” Turner - mlturner@student.uiwtx.edu
- Moses Chavez - mwchavez@student.uiwtx.edu
- Dylan Ponce - dyponce@student.uiwtx.edu

Project Objectives

1. Identify how to utilize and familiarize ourselves with tools such as Suricata, for example, we intend to use the tool to our preferences, in the sense that we will determine the rules for the data base and have Suricata carry out what we please based on those rules.

2. Determine how intrusion detection works on both small and larger scales. As mentioned, we will also be using Wireshark, this just to see how traffic within the network flows, and how you can manually stop/ see traffic that may be malicious.

Links

GitHub: https://github.com/mwchavez/IDS_DF_F2025_Project_MC_MT_DP

Summary of proposed project

For this project, we will be using tools such as Suricata and Wireshark to be able to implement an intrusion detection system that will work with the rules we set out in Suricata. For this project, we will first set our own rules that we came up with based on our previously knowledge, then we will switch to have rules that are based on a deep learning model, for the data we will use for this deep learning model, we will use the tool “malwaretrafficanalysis/net” (Slash is intended, just for the purpose of not linking anything.) From there to garner a better understanding of traffic analysis, we will use Suricata to see what rules we can make and how this affects a network. We will also be using Wireshark to analyze the packet, this will be elaborated on further.

Material List

- A system with Wireshark
- VM ware
- Security Onion
- Kali system run through the VM

- https://www.malware-traffic-analysis.net/index.html

- Private network, to be able to utilize security onion and its resources without anything blocking or interfering

IDS

- M1: Security Onion up, baseline & capture filters documented.
- M2: Initial Suricata rule pack + Zeek logs validated.
- M3: Hunt runbook + tuning (suppressions/thresholds) + metrics.
- M4: Live detection demo + short tuning report.

DF

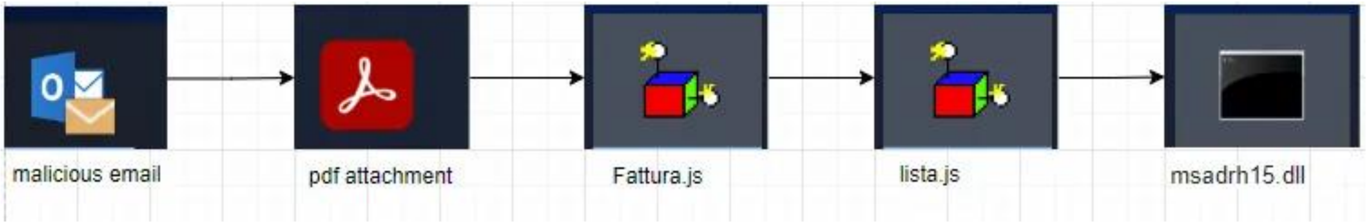
- M1: Case scope, chain-of-custody.
- M2: Wireshark reconstruction + artifact extraction.
- M3: (Optional) Memory/Disk triage and correlation to PCAP.
- M4: Final case report + IOC list + recommendations + demo.

Case Study

On May 14, 2021, our team was alerted to an intrusion within our system, we as the digital forensics team, and we as the intrusion detection team, were requested to study the case, documented in the GitHub is our findings (Digital Forensics) and our solutions to prevent this from happening in the future (Intrusion Detection).

The email was first sent to an employee on May 3rd, which was an excel sheet with the macros for URSNIF. On May 14th another email was sent, that was similar in nature. URSNIF is more commonly referred to as a banking trojan. Originally identified in 2006, there was a GitHub leak of the source code for this trojan.

Trojan trajectory path is as followed.



Credits of photo: *Ursnif, the banking trojan*. (2023). Acronis. https://www.acronis.com/en/tru/posts/ursnif-the-banking-trojan/

Primary Investigative Questions

- Are there any other malicious activities?

- What are the signs of the malware?

- Is there a data loss?

- What protocols were used, are they within the normal range of protocols used on a day-to-day basis?

- Timing of the incident?

- Was the malware trying to spread?

- What type of malware was identified?