Intel-Based Macintosh Memory Forensics: Eavesdropping Attack Analysis Through Memory Dump

1st Farhan Muhammad Alif
School of Computing
Telkom University
Bandung, Indonesia
farhanmalif@student.telkomuniversity.ac.id

2nd Niken Dwi Wahyu Cahyani
School of Computing
Telkom University
Bandung, Indonesia
nikencahyani@telkomuniversity.ac.id

3rd Vera Suryani
School of Informatics
Telkom University
Bandung, Indonesia
verasuryani@telkomunivesity.ac.id

Abstract*—Memory forensics plays a critical role in cybersecurity, particularly in analyzing volatile memory during cyberattacks. This paper examines memory forensics analysis on Intel-based Macintosh systems targeted by remote attacks within shared networks. The study used a controlled setup involving an Intel-based Macintosh running a vulnerable PHP-based web application, DVWA (Damn Vulnerable Web Application). The system was attacked via SQL injection, command injection, and reflected Cross-Site Scripting (XSS) from a Kali Linux device over shared Wi-Fi. The attacks exploited application vulnerabilities to compromise the system, necessitating forensic examination. Memory dumps from the Mac device were analyzed using tools like the Volatility Framework to extract artifacts such as process details, network activity, and injected code. Memory artifacts were correlated with Wireshark packet analysis to uncover network-level evidence. The findings underscored the impact of remote attacks on system integrity and the effectiveness of memory forensic methods. Unique challenges in MacOS forensics include kernel-level access requirements, System Integrity Protection (SIP), compressed swapped memory, and address space layout randomization. These protections complicate forensic analysis, demanding specialized techniques. This study enhances knowledge of macOS memory forensics under remote attack scenarios, proposing methodologies to address evolving threats and highlighting the importance of volatile data analysis in system security.

Keywords*—Volatility Framework, RAM, Memory Forensic, DVWA, macOS.

I. INTRODUCTION

Memory forensics is a critical aspect of digital investigations, focusing on the collection and analysis of information from electronic devices, particularly volatile memory. Volatile memory plays a pivotal role in investigations as it contains metadata housing details about active processes, memory structures, and other transient data [1]. The acquisition and analysis of Random Access Memory (RAM) enable investigators to uncover malicious activities, such as active processes and network connections, that may not leave traces on a device's storage disk [17].

Despite the significant advancements in memory forensics, there remains a notable disparity in research coverage across operating systems. While Windows has been extensively studied, memory forensics on Macintosh devices has not received the same level of attention, even though Macintosh devices are widely adopted globally. This gap is particularly critical given the unique challenges posed by Macintosh memory architecture and security mechanisms. For example, macOS employs System Integrity Protection (SIP), which restricts memory acquisition tools unless they operate with kernel-level permissions [3]. Furthermore, macOS's memory management system uses compressed swap pages rather than traditional swap files, complicating the acquisition process [2]. The shift from x86 to ARM CPU architecture further limits the availability of suitable memory forensic tools for newer Macintosh devices [7]. These factors collectively create significant challenges for investigators seeking to uncover evidence from macOS systems.

This study addresses the research gap by focusing on the memory architecture and security features of macOS, specifically on Intel-based devices running macOS 10.12 and earlier versions. Through a comprehensive literature review, the study examines the intricacies of macOS memory forensics, including the acquisition and analysis of RAM on Intel-based systems. Additionally, it explores packet analysis of remote exploitation over a shared network, providing insights into macOS-specific forensic methodologies. By addressing these unique challenges, this research aims to advance the understanding of memory forensics on macOS and bridge the gap in existing knowledge compared to Windows and Linux systems.

II. RELATED WORK

Several publications discuss the memory forensic process and architecture of MacOS. Table I summarizes the literatures.

Table. 1 Summary of MacOS Forensics Literature

Researcher(s)	**Topic(s)**	**Platform(s)**
M. Manna, A. Case, A. Ali-Gombe, and G. G. Richard [6]	Decoding Swift and Objective-C runtime platform in MacOS	MacOS 10.15
A.Case, R. Maggio, M. Manna, and G. G. Richard [5]	MacOS Page Queues and Memory Architecture Analysis	MacOS 10.9 - MacOS 10.15
D. Sladovic, D. Topolcic, and D. Delija [10]	MacOS security chip and security features analysis alongside decoding attempt	MacOS 10.12