# Chapter 9
# Malware Analysis and Attribution in Human Threat Intelligence

**S. Manjula**
*Madanapalle Institute of Technology and Science, India*

**Amit Karbhari Mogal**
https://orcid.org/0009-0008-4183-2264
*MVP Samaj's CMCS College, Udoji Maratha Boarding Campus, India*

**C. S. Sandeep**
https://orcid.org/0000-0003-3269-7984
*Jawaharlal College of Engineering and Technology, India*

**L. R. Sujithra**
https://orcid.org/0009-0003-9806-268X
*Sri Eshwar College of Engineering, India*

**J. A. Jevin**
https://orcid.org/0009-0008-8511-9698
*Koneru Lakshmaiah Educational Foundation, India*

**Sachin Vasant Chaudhari**
https://orcid.org/0009-0005-8856-8905
*Sanjivani College of Engineering, India*

**V. Bhoopathy**
https://orcid.org/0000-0003-2175-6328
*Sree Rama Engineering College, India*

## ABSTRACT

*Increasing numbers of people using online resources without precautions is one of the main reasons hazardous computer apps evolve quickly. These days, Trojan horseback, malware, and worm authors do it for profit rather than fame. So, malware problem Computer security now prioritizes detection. Zero-day malware exploits undiscovered holes without detection. Deep learning models, which can analyze data in several dimensions and uncover features, are an effective threat detection and classification alternative. This chapter examines deep learning-based malware*

DOI: 10.4018/979-8-3373-4898-8.ch009

*recognition and categorization for zero-day vulnerabilities and attack authentication. While studying model architectures, data sources, evaluation criteria, and real-world applications, we discuss asymmetrical evasion, comprehensibility, and statistical imbalance. The article also explores how graph-based models, self-encoder RNN, and CNN understand unfamiliar virus behaviors.*

## 1. INTRODUCTION

Intentionally damaging the operating system kernel or other security-sensitive applications or data without the user's knowledge or agreement is the goal of malicious software. A wide variety of malicious software exist, including but not limited to computer viruses, worms, and PUP (Bilot et al., 2024). Numerous businesses and individuals across the globe are being impacted by cybercrime that employs this type of software. New assaults on the online have been created by undiscovered varieties of current malware that obscure their activity to elude detection. There have been several harmful activities on the web (Guo, 2023). Because there are no days between an unknown malware's first attack and its discovery, it is referred to as zero-day malware, which means fresh malware. This kind of attack is also known as a zero-day attack (Nikolopoulos & Polenakis, 2017). Static and dynamic analysis is the two primary methodologies to malware detection that are routinely used. In static analysis, features are extracted from a program's binary by using the file's syntax and structural attributes (Venkatraman & Alazab, 2018). In contrast, specific program activities can only be extracted by dynamic analysis of the file that is executed while the program is running.

Theoretically, static analysis benefits from information pertaining to structural qualities, such as sequences of bytes “signatures” and abnormalities in file content, make it faster and more effective than dynamic analysis. Runtime information, such a processor's state, might be used for dynamic analysis through the use of a control flow graph, which may make it more resistant to malware that obfuscates its messages. Several earlier investigations have found that combining the two methods yields superior outcomes. Malware authors use a plethora of metamorphic and polymorphic obfuscation strategies to stay undetected (Aboaoja et al., 2022). Among these methods are a number of approaches that involve inserting dead code, rearranging subroutines, transposing codes, replacing instructions, integrating codes, and reassigning registers (Zahoora et al., 2022). Furthermore, packers make sure that the code can only be examined during runtime by obfuscating the entire program (Malhotra & Bajaj, 2016).