Chapter 2
AI-Powered Malware Detection and Behavioral Analysis

P. Selvakumar
http://orcid.org/0000-0002-3650-4548
Department of Science and Humanities, Nehru Institute of Technology, Coimbatore, India

Vaishali Rahate
http://orcid.org/0000-0001-6277-1685
Datta Meghe Institute of Management Studies, India

A. S. Deeppana
http://orcid.org/0009-0007-9840-5725
Karpagam Academy of Higher Education, India

Yukta Sawalkar
Datta Meghe Institute of Management Studies, India

C. John Paul
St. Joseph University, India

S. Murugaveni
http://orcid.org/0000-0003-2081-7803
SRM Institute of Science and Technology, India

ABSTRACT

In the rapidly evolving landscape of cybersecurity, malware continues to pose significant threats to individuals, enterprises, and critical infrastructures. Traditional signature-based detection techniques, though effective against known threats, fall short when confronted with sophisticated, polymorphic, and zero-day malware. This limitation has fueled research into more intelligent, adaptive detection mechanisms that can identify malicious software even when it exhibits novel patterns or obfuscation strategies. Static malware analysis, unlike dynamic analysis, focuses on examining the intrinsic attributes of executable files without executing them, making it safer, faster, and less resource-intensive. Static features typically include opcode

DOI: 10.4018/979-8-3373-8252-4.ch002

sequences, bytecode patterns, control flow graphs, API call frequency distributions, file headers, string literals, and metadata extracted from Portable Executable (PE) files or other binary formats.

INTRODUCTION

In the rapidly evolving landscape of cybersecurity, malware continues to pose significant threats to individuals, enterprises, and critical infrastructures. Traditional signature-based detection techniques, though effective against known threats, fall short when confronted with sophisticated, polymorphic, and zero-day malware. This limitation has fueled research into more intelligent, adaptive detection mechanisms that can identify malicious software even when it exhibits novel patterns or obfuscation strategies. Static malware analysis, unlike dynamic analysis, focuses on examining the intrinsic attributes of executable files without executing them, making it safer, faster, and less resource-intensive. Static features typically include opcode sequences, bytecode patterns, control flow graphs, API call frequency distributions, file headers, string literals, and metadata extracted from Portable Executable (PE) files or other binary formats. However, manually engineering features from these elements is labor-intensive, error-prone, and often fails to capture subtle patterns indicative of malware. Deep learning addresses these challenges by enabling automatic feature extraction, discovering complex nonlinear relationships within the data, and generating representations that are often more discriminative than handcrafted features.

By treating malware byte sequences as images or one-dimensional signal streams, CNNs can extract local and global patterns that signify malicious behavior. For instance, reshaping a binary file into a two-dimensional grayscale image allows CNNs to detect characteristic textures and structural anomalies, such as repeated sequences, packed sections, or unusual byte distributions that are indicative of obfuscation or encryption techniques commonly employed by malware authors. Such sequential modeling is crucial for understanding malware behavior at the instruction level, where specific instruction chains or API call sequences may reveal malicious intent, even in the absence of overt signatures. Hybrid architectures combining CNNs and RNNs have demonstrated superior performance in extracting both spatial and temporal features, effectively bridging the gap between structural analysis and sequential semantics. Autoencoders, another class of deep learning models, are particularly useful for unsupervised feature extraction and dimensionality reduction. By learning compact latent representations of high-dimensional malware data, autoencoders can capture essential characteristics while filtering out noise, thereby improving classification performance. Variational Autoencoders (VAEs) further enhance this