Malware Signature Generation Framework for Threat Intelligence

Mr. B. Surya Narayana Reddy[1], G. Mamatha[2], P. Sindhu[3], B. Swathi[4], D. Manikanta[5]
[1]*Assistant Professor, Department of Cyber Security, Sphoorthy Engineering College, Hyderabad, Telangana, India*
[2,3,4,5]*Member, Department of Cyber Security, Sphoorthy Engineering College, Hyderabad, Telangana, India*

Abstract ***Malware attacks have been increasing over time, making it difficult for traditional detection systems to handle new and unknown threats. Most existing methods depend on already available signatures, which may not work when malware is slightly modified. In this paper, a simple malware signature generation framework is presented to support threat intelligence. The system focuses on analyzing files, extracting useful patterns, and organizing them into structured signatures. These signatures can later be used by detection systems to identify similar malicious behavior. The approach was tested using sample files to observe how patterns differ between suspicious and normal files. The results show that the system is able to generate meaningful signatures that can help in understanding and detecting potential threats. This work mainly focuses on the basic idea of signature generation and can be extended further in future.***

Index Terms **Malware detection, signature generation, threat intelligence, cybersecurity, pattern extraction**

I. INTRODUCTION

In today's digital environment, malware has become one of the most common and serious security threats. It can affect systems in different ways, such as data, damaging files, or interrupting normal operations. With new types of malware being developed regularly, it is becoming harder to detect them using traditional methods.

Most detection systems rely on predefined signatures to identify malicious files. While this works well for known threats, it does not perform effectively when new or slightly modified malware appears. Because of this, there is a need for approaches that can help in generating new signatures based on observed patterns.

This paper presents a basic framework for malware signature generation. The main idea is to analyze files, extract useful patterns, and convert them into structured signatures. These signatures can then be used as a reference for identifying similar threats. The focus of this work is not on complex detection techniques, but on building a simple and understandable approach for signature generation.

II. LITERATURE SURVEY

Several studies have been conducted in the field of malware detection and analysis. Traditional approaches mainly rely on signature-based detection, where known patterns are used to identify malicious files. While these methods are effective for known threats, they often fail to detect new or modified malware.

Researchers have also explored behaviour-based and heuristic approaches to overcome these limitations. These techniques focus on identifying suspicious activities rather than fixed patterns, making them more adaptable to new threats. However, such methods can be complex and may lead to false positives.

Some works have focused on automated signature generation, where patterns are extracted directly from malware samples. This reduces dependency on manually created signatures and improves detection capability.

In this paper, a simple approach is followed to understand the basic concept of signature generation by extracting patterns and organizing them into structured formats. The focus is on clarity and practical understanding rather than complex implementations.

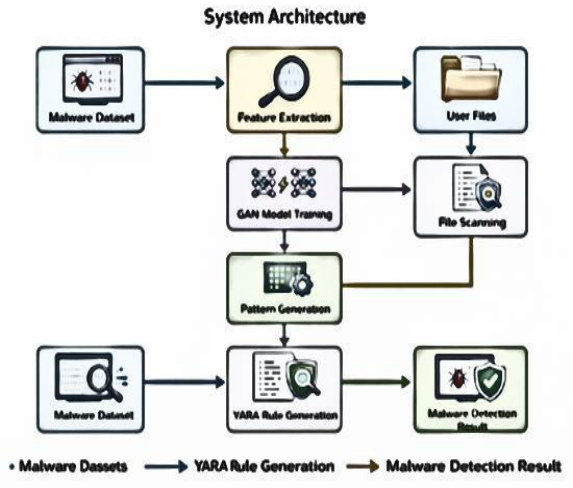
III. SYSTEM ARCHITECTURE

The system is designed in a simple and modular way so that each step can be understood clearly. It starts with taking a file as input, which can be either suspicious or normal.

Once the file is provided, it is analyzed to understand its content. The system then extracts patterns such as readable strings or identifiable data segments. These extracted patterns are further processed to remove common or unnecessary information.

After filtering, the remaining patterns are used to generate a structured signature. This signature represents the characteristics of the file and can be used later for analysis or detection purposes.

The output of the system is the generated signature, which can be stored or used for further processing.



IV. METHODOLOGY

The methodology followed in this work is simple and practical. First, files are selected for analysis. These files may include both suspicious samples and normal files, so that differences can be observed.

The system reads the file content and extracts patterns that are visible and meaningful. These may include strings or repeated data segments. Not all extracted patterns are useful, so a filtering step is applied to remove common or irrelevant ones.

After filtering, the system focuses on identifying patterns that are more unique to the file. These selected patterns are then used to create a structured signature.

The signature is designed in a way that it represents the key characteristics of the analysed file.

The overall process is kept simple to ensure clarity and ease of implementation, while still producing useful results.

V. MODULES

The proposed system is divided into several modules to simplify the overall process and improve understanding.

1. File Input Module

This module is responsible for taking the input file. The file can be either suspicious or a normal file used for comparison.

2. File Analysis Module

In this module, the content of the file is read and analysed. The system processes the data to identify useful information that can be used further.

3. Pattern Extraction Module

This module extracts patterns such as readable strings or identifiable data segments from the file. These patterns form the base for signature generation.

4. Pattern Filtering Module

Not all extracted patterns are useful. This module removes common, repeated, or irrelevant patterns to focus on more meaningful data.

5. Signature Generation Module

The filtered patterns are used to generate a structured signature. This signature represents the characteristics of the analysed file.

6. Output Module

The final generated signature is displayed or stored for further use in analysis or detection.

VI. ALGORITHM FOR SIGNATURE GENERATION

The signature generation process follows a sequence of steps that help in extracting and organizing useful patterns.

Algorithm Steps

1. Start the process
2. Take a file as input

3. Read the file content
4. Extract patterns such as strings
5. Remove common or repeated patterns
6. Select patterns that appear more significant
7. Generate a structured signature using selected patterns
8. Display or store the generated signature
9. End the process

Pseudo Code:

BEGIN

INPUT file

features ← extract_patterns(file)

filtered ← remove_common(features)

important_patterns ← select significant(filtered)

signature ← create_signature(important_patterns)

OUTPUT signature

END

VII. TESTING AND RESULTS

The system was tested using a small set of sample files to check how it performs. Both suspicious and normal files were used during testing.

The main aim of testing was to observe whether the system can extract meaningful patterns and generate useful signatures. It was also checked whether common patterns are removed properly during filtering.

From the results, it was observed that the system is able to generate structured signatures that reflect the content of the analysed files. While the system is basic, it still provides a clear idea of how signature generation can be done.

VIII. CONCLUSION

In this paper, a simple malware signature generation framework was presented. The system focuses on extracting patterns from files and organizing them into structured signatures. The results show that even a basic approach can help in identifying useful patterns that may assist in malware analysis.

This work mainly serves as a starting point for understanding signature generation. In future, the system can be improved by adding advanced techniques and integrating it with detection tools to make it more effective.

ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to the project guide and faculty members for their continuous support and valuable guidance throughout the development of this work. Their insights and suggestions greatly contributed to improving the quality of the project.

The authors also acknowledge Sphoorthy Engineering College, Nadergul, Hyderabad for providing the necessary resources and a supportive environment to carry out this work successfully.

REFERENCES

[1] M. Sikorski and A. Honig, Practical Malware Analysis, No Starch Press, 2012.
[2] P. Szor, The Art of Computer Virus Research and Defense, Addison-Wesley, 2005.
[3] Symantec Corporation, “Internet Security Threat Report,” 2020.
[4] Kaspersky Lab, “Cyber Threat Intelligence Report,” 2021.