Recorded Future®

Auto YARA: Automated Yara Rule Generation for High-confidence Threat Detection

Authors: Anders Hansson and Daniel Gillblad

Abstract

This white paper outlines the approach Recorded Future's Auto YARA system uses to generate YARA rules automatically. Our system aims to automate high-confidence detection capabilities while minimizing false positives, and our approach leverages advanced techniques to identify and extract unique patterns from malware samples, optimize these patterns through iterative refinement, and transparently document the rule-creation process. This white paper explains the overall process and validation of the system without revealing proprietary technical details, ensuring that customers understand and trust the generated rules.

Introduction

In today's fast-paced cybersecurity environment, rapid and accurate threat detection is essential. YARA is widely used by security professionals to identify malware based on distinctive file characteristics such as strings and binary patterns. Traditionally, developing effective YARA rules has involved significant manual effort, including detailed file analysis and trial-and-error refinements. Recorded Future's automated rule generator streamlines this process, saving time and reducing complexity while maintaining high standards of detection accuracy and returning few false positives.

The system's goal is to produce transparent, robust detection rules that can either be deployed directly or serve as a foundation for further manual refinement by experienced analysts.

Background and Motivation

Signature-based Detection

YARA rules (or signatures) enable pre-execution detection by analyzing file characteristics like extracted strings, file headers, and embedded code patterns. Although these signatures are highly effective, manually creating them can be both labor-intensive and prone to error, particularly when malware authors employ obfuscation or polymorphism to evade detection.

The Need for an Automated Approach

Automation in rule generation offers significant advantages:

Efficiency

Automation rapidly processes large sets of malware samples, reducing the amount of time experts spend on manual analysis.

Consistency

Iterative refinement ensures that the rules maintain a high level of accuracy across different malware families.

Transparency

By clearly documenting the pattern selection and optimization process, Recorded Future's Auto YARA system provides customers with confidence in the detection rules' reliability.

Methodology

Recorded Future's Auto YARA system employs a multi-step process that balances effective malware detection with transparency, without disclosing proprietary implementation specifics.

Pattern Identification

The pattern identification process begins when the system scans a collection of malware samples to extract common patterns and identify the longest common substring. The system chooses these patterns, which may include sequences of bytes or identifiable strings, based on their prevalence across malicious files and their absence from a curated set of benign files. The use of byte patterns in addition to ASCII strings is a key technique machines can apply efficiently, allowing an automatically generated rule to search through a large surface area of the file binary to identify rules. This technique ensures that the selected patterns strongly indicate malicious behavior.

Filtering and Refinement

Once the Auto YARA system identifies candidate patterns, it filters them to remove any that also appear in legitimate software. Given the computational expense of comparing against a high number of legitimate software samples, the system uses FM-index, a compressed full-text substring index based on the Burrows-Wheeler transfer (Ferragina; Manzini, 2005). This allows for exact substring matching with a guaranteed result.

The system uses an iterative approach to adjust the sensitivity of the extraction process, ensuring that the final set of patterns is both comprehensive and precise. This dynamic thresholding adapts to the variability found in real-world malware samples. The technique, which is widely used in bioinformatics, limits the time complexity of filtering out false positives to the length of each search string (or each pattern) rather than the size of the corpus.

Rule Optimization

Extracted patterns offer varying contributions to effective detection. Therefore, the system employs a heuristic scoring process to prioritize patterns that are most informative and reliable. It considers factors such as uniqueness, coverage across malware samples, and ease of interpretation.

To optimize the rule set, the system assigns heuristic scores (or cost) to the patterns. For instance, it penalizes patterns with very low entropy or overly long patterns, and it promotes patterns that match more binaries over those matching fewer binaries. Since explainability is important, the system also favors patterns containing ASCII-interpretable substrings.

We can represent the optimization problem using a bipartite graph, where one set of nodes corresponds to patterns and the other to binary files, as shown in Figure 1. Using a greedy, heuristic approach, the system identifies a subset of pattern nodes that collectively match all file nodes. This involves iterating through pattern nodes in decreasing order of cost and selecting them until all file nodes are covered. This method allows the system to compile an effective set of YARA rules, consisting of a minimal yet sufficient subset of patterns to match all malware binaries.

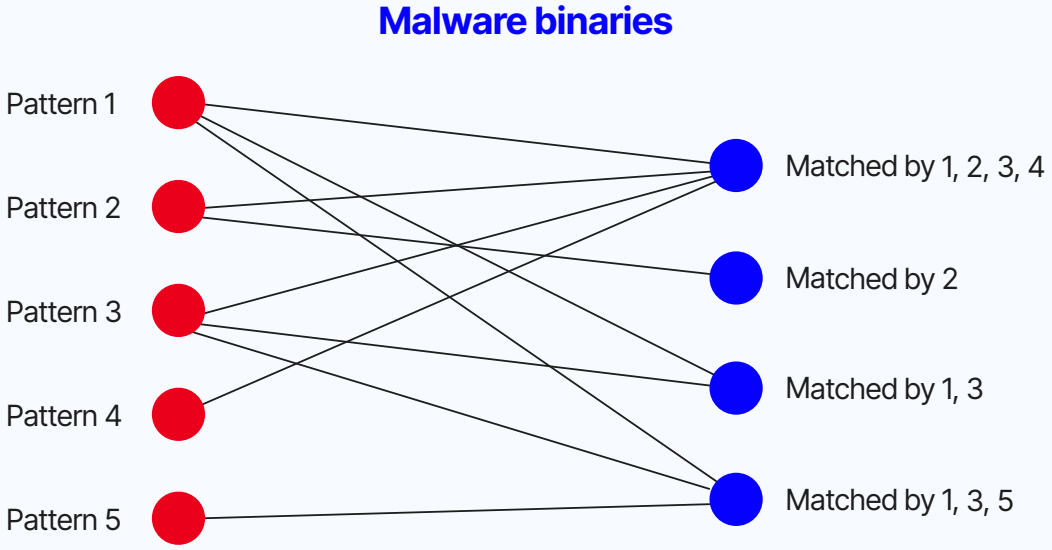

Figure 1: Bipartite graph exemplifying how patterns (on the left) match malware binaries (on the right).

Transparent Rule Generation

For each selected pattern, the system generates an atomic YARA rule. In cases where multiple patterns must be combined to ensure detection accuracy, the system creates composite rules. It appends metadata to each rule to document the number of samples it covers and other contextual details. This transparency provides an audit trail that helps analysts understand and trust the detection logic behind each rule.

Evaluation and Validation

We evaluated the system using a diverse set of malware families. For each family, we generated rules from a subset of samples and then tested them on a larger set of unseen files. Key aspects of our evaluation included:

High detection rates

The generated rules consistently identified a large proportion of malware samples across multiple families.

Low false positives

Rigorous filtering against a broad range of benign files ensured that the rules rarely triggered on non-malicious software.

Stability

Multiple tests have confirmed that the system produces consistent results, even when the sample set varies.

The following chart shows a comparison of YARA rules generated by Recorded Future's Auto YARA compared to a tool known as YarGen. In all cases, Auto YARA's detections had a higher success rate. Auto YARA's worst hit rate came when handling Metasploit, for which payloads are often automatically generated (rather than being hand-written) and encoded using the Shikata Ga Nai encoder, which employs polymorphic XOR additive feedback to ensure that the output of the encoder is different every time to evade signature-based detection. For 35 families, or 78% of all families, Recorded Future's Auto YARA achieved a hit rate greater than 0.8.

These results demonstrate the efficiency of a more robust pattern detection technique, and affirm that Recorded Future's automated approach is both effective in detecting malware and reliable across different threat scenarios.

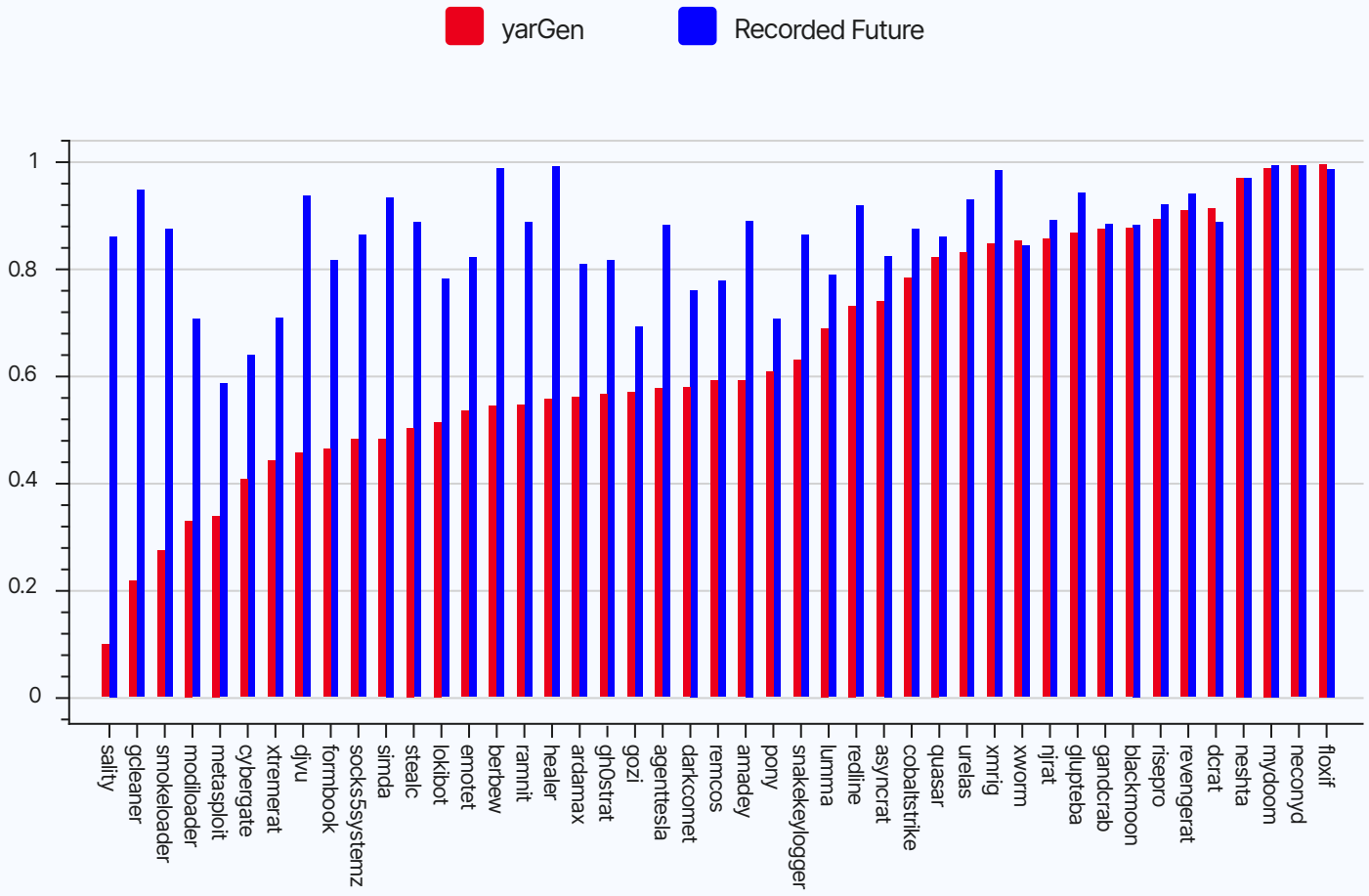

Figure 2: Comparison of YARA rules generated by YarGen and Recorded Future in terms of fraction of hits for 45 malware families.

Conclusion

Recorded Future's automated YARA rule-generation system offers a transparent, efficient, and robust solution for malware detection. By automating the extraction of unique patterns from malware and rigorously filtering them against benign files, it produces high-confidence detection rules that are ready for deployment. Our approach not only accelerates the response to emerging threats but also provides cybersecurity professionals with clear insights into the rule-generation process, fostering trust and confidence in the underlying detection mechanisms.

References

A selection of academic and industry sources that inform our methodology is available upon request. These include seminal works on efficient pattern matching and signature generation in cybersecurity research.

Recorded Future®