WHITE PAPER

REVERSINGLABS

Modern Malware Analysis

Revisiting the Static vs Dynamic Debate

Introduction

Increasingly sophisticated malware routinely evades organizations' cyber defenses, lurking inside networks, often for months, only executing when it can cause the most damage. Even though the industry has developed various technologies to bolster detection and response, the situation persists. The discovery of successful attacks is still measured in months and containment in weeks, meaning the average dwell time from compromise to containment remains intolerably high.

The conventional approach for combating malware is not working. Organizations need to move beyond existing detection strategies and focus on filling the defensive gaps that allow zero-day, polymorphic, and evasive malware attacks to succeed. Worse, traditional approaches are too costly in the face of weak identification.

In this paper, we'll discuss the state of malware analysis, examining both static and dynamic analysis, the advantages and disadvantages of each technique, and how a new approach can overcome the shortcomings of each to drive higher levels of efficiency and efficacy in malware detection - all while reducing costs.

Traditional Static Analysis

Static analysis examines binary code without executing the program. Since there's no code execution, static malware analysis doesn't require a "live" environment. Moreover, by not executing a file, the analysis can occur rapidly across almost any file type.

Basic static analysis is straightforward and quick. It works by extracting technical indicators, such as file names, hashes, header data, and strings that can be used to determine whether that file is malicious. One benefit is that basic static analysis is fast. Unfortunately, it becomes ineffective against large, multi-layered binary files and provides little utility against complex file structures where code packing and obfuscation are used. In such cases where files are in a packed format, the actual analysis step must be preceded by a series of actions to deconstruct and de-obfuscate the file.

A more in-depth and considerably more complicated form of static analysis uses debuggers, disassemblers, de-obfuscators, and other specialized tools to identify and understand malware. It necessitates the use of multiple tools across many manual steps, in addition to requiring a high-level of expertise from the analyst, which makes it untenable in any real-time detection process. As part of an investigation process, this form of static code analysis is only available to the most well-funded security teams, which have the labs and skilled analysts required to complete the work.

Dynamic Analysis / Sandbox Environments

Dynamic analysis executes files in a safe environment (a sandbox) and captures how they interact with the system. The historical benefit of dynamic analysis and sandboxes has been to examine a file or application while it runs to observe its behavior in real-time. This greatly improves the ability to detect malicious actions, interactions, and potential security risks.

In practice, most security teams focus on dynamic analysis due to the perceived notion that only by running a file can its full capabilities be understood. This is why sandboxes have long been the "go-to" method for protecting against malware.

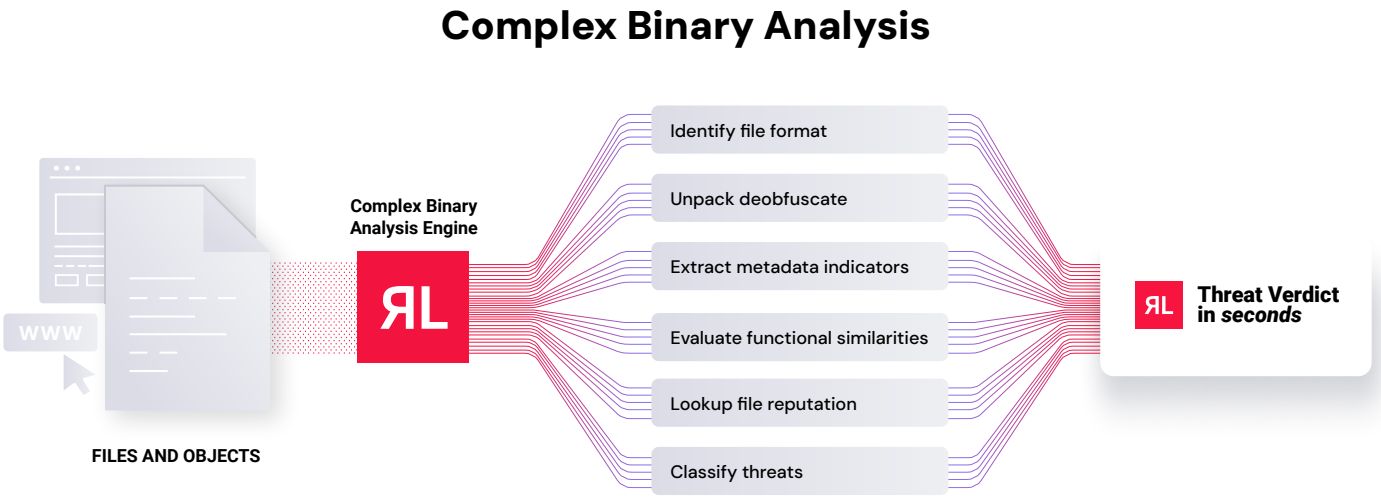
However, over time, the limitations of sandboxes/dynamic analysis have become quite clear.

- **File Type Limitations** – Dynamic analysis requires a file to execute a program, meaning non-executable files cannot be analyzed effectively.
- **File Size Constraints** – Large files significantly increase processing time, and in some cases, they may not be processed at all. This can result in a "fail open," where the analysis report might indicate "no malware found," but that's because the file couldn't be analyzed. This can leave the security team with a dangerous false sense of security, which can be detrimental to the organization.
- **High File Volume** – Large numbers of files slow down analysis workflows and require additional computational resources, making sandboxing cost-prohibitive.
- **High Cost Per File** – Dynamic analysis is significantly more costly than static, so blindly running every file in question through a sandbox, whether or not it is appropriate, can lead to unnecessarily high costs.
- **Easily bypassed** – Adversaries are known to build malware that can hide from or bypass sandbox execution, so the malware enters the network undetected.

Sandbox vendors have worked hard to overcome various limitations and deception techniques. Still, these efforts have further slowed dynamic analysis processing rates, generally precluding the use of a sandbox in real-time processes while increasing costs exponentially.

The Modern Approach: Moving Beyond Traditional Static and Dynamic Analysis

AI-driven complex binary analysis is an advanced static analysis that now automatically and recursively deconstructs binaries to discover a file's true intent and capabilities in seconds. And it overcomes the issues of file type limitations, allowing security teams to address large files and high file volumes at a lesser cost.



The complex binary analysis process begins by identifying the format of the file or object. Advanced file deconstruction means that once the object is identified, automated unpacking and de-obfuscation takes place with support for more than 400 packer formats (e.g., archives, installers, and compressors) along with the ability to identify over 4800 different types of files (e.g. PE/Windows, ELF/ Linux, Mac OS, iOS, Android, FLASH, media, documents, and more).

The deconstruction process fully dissects the object down to its base components. All child files and objects, along with all metadata and internal indicators are extracted, providing critical information not available from other tools for determining the intent and capabilities of a file or object. Discovered indicators may include embedded images, hidden archives, terminated processes, DLLs, APIs, file system changes, evasion techniques, hidden executables, improper writes, privacy intrusions, entropy level, and self-protection techniques.

All extracted indicators are then run against a series of advanced algorithms to determine their functional similarity to known attack techniques used by malware families so that practitioners can either recognize polymorphic malware or detect a new and unknown malware variant. A multi-factor threat classification process combined with real-time reputation data and intelligence provides a definitive threat verdict, enabling security teams to quickly, effectively, and confidently prioritize alerts and take action.

The RL Threat Repository

RL maintains the world's largest threat repository of malware and goodware, tracking 422+ billion files. It is 10 times the size of any other repository available today.

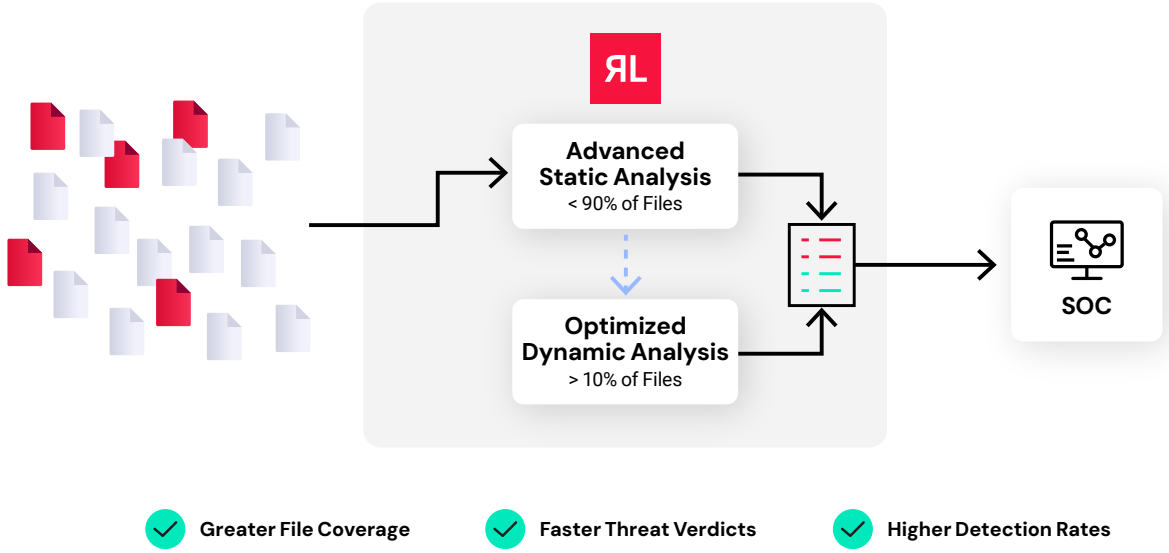
The file reputation check is a critical part of the process to quickly and accurately identify and classify both "known bad" and "known good" files. RL continuously processes billions of goodware and malware samples in its global Threat Repository to ensure the most up-to-date reputation status. The context of known bad files (malware) includes all the relevant threat details and contextual information so security teams can immediately take the appropriate response and containment actions. Likewise, identifying known good files is critical to reducing false positives and eliminating the time wasted chasing false alarms.

Better Results at Less Cost

The ultimate goal of security teams is to identify and mitigate threats as quickly and efficiently as possible, which is personified in RL's approach to malware detection.

RL's Advanced Malware Analysis Suite was developed around the principles of efficiency and efficacy. It all begins with our high-speed complex binary analysis, as outlined in the previous section. By employing this form of advanced static analysis at the start, 90% or more of files can be analyzed and classified up front - before the use of high cost dynamic solutions. And, because this analysis occurs pre-execution, results are provided in real time, which is crucial considering speed is of the essence when it comes to threat detection. Any remaining files requiring dynamic analysis can be sent to the sandbox for runtime processing. The result is significantly optimized malware analysis workflows.

Importantly, this approach overcomes limitations around file types, file sizes, and volumes of files inherent to traditional analysis methods, thus providing vastly increased file coverage and unmatched malware threat visibility.



ReversingLabs customers using the Advanced Malware Analysis Suite can experience up to 45% savings with 9.9x faster verdicts when compared to using a sandbox alone.

Conclusion

As sophisticated malware continues to evade existing detection tools and processes, security teams must adopt new technologies to keep pace. Complex binary analysis technology enables an entire roster of new detection and intelligence capabilities, this modern approach tips the scales back in favor of the security team in the malware arms race.

About ReversingLabs

ReversingLabs is the trusted name in file and software security. We provide the modern cybersecurity platform to verify and deliver safe binaries. Trusted by the Fortune 500 and leading cybersecurity vendors, RL Spectra Core powers the software supply chain and file security insights, tracking over 422 billion searchable files daily with the ability to deconstruct full software binaries in seconds to minutes. Only ReversingLabs provides that final exam to determine whether a single file or full software binary presents a risk to your organization and your customers.

To learn more about how advanced complex binary analysis can significantly improve your organization's security posture, please visit us on the web at reversinglabs.com

VISIT WEBSITE

WP-Rev-04.14.25

REVERSINGLABS

Worldwide Sales: +1.617.250.7518
sales@reversinglabs.com