TRAKR: A framework to collect, process and analyze diverse APT files and reports

Alejandro Barreiro-Morante, Daniel García-Algora, Alejandro Carlos del-Rio-Álvarez, Jorge Blasco
Department of Computer Systems, Universidad Politécnica de Madrid
Calle Alan Turing s/n, Madrid, Spain
alejandro.bmorante@upm.es, d.galgora@upm.es, ac.delrio@upm.es, jorge.blasco.alis@upm.es

***Abstract*—Advanced Persistent Threats (APTs) are a significant challenge for the cybersecurity industry, requiring continuous monitoring and analysis. Cybersecurity professionals and academics share data about these threats via datasets, Cyber Threat Intelligence (CTI) reports and automatic exchanges via APIs and other less public means. In this paper we present Threat Reconnaissance and APT Knowledge Repository (TRAKR) , a framework designed to collect, process, and analyze CTI reports alongside binary samples related to APT incidents. TRAKR integrates multiple data sources, enabling correlation between threat reports and malware artifacts. We show its initial analysis capabilities, highlighting its potential for uncovering trends within APT operations and issues in how APT data is being collected. Our initial results show TRAKR's ability to enhance APT intelligence by bridging the gap between structured reports and technical malware analysis.**

***Index Terms*—APTs , Malware, CTI.**

Tipo de contribución: *Investigación en desarrollo*

I. Introduction

Advanced Persistent Threats (APTs) are now a common element of geopolitical warfare and differ significantly from traditional malware threats. They are carried out by adversaries with plenty of resources that pursue their objectives over extended periods of time, reacting to defender's mitigation techniques and trying to keep access to compromised systems for as long as possible. Although the term APT was coined 20 years ago to allow the US military to discuss the details of the intrusion with civilian experts [1], the term APT is now widely used across academia and industry.

The rapid increase in the amount and sophistication of these attacks has led to several initiatives to share and exchange information about these attacks. These range from standards to allow automated exchange of incident data such as STIX [2] to reports being published by organisations and cybersecurity vendors. These reports provide details on information such as the impact they had on their victim's systems, the threat actor's profile, and behavioral indicators such as tactics, techniques and procedures that may be identifiable. This information proves valuable not only for preemptive purposes, but also for attribution when dealing with unknown attacks [3].

In this paper, we present Threat Reconnaissance and APT Knowledge Repository (TRAKR), a system that enables the collection, processing, and analysis of Cyber Threat Intelligence (CTI) reports and binary file samples related to APTs incidents. TRAKR collects information from a variety of sources and allows to create processing pipelines that combine information from threat reports and features extracted from malware files. TRAKR can be updated at regular intervals and can be expanded easily with new sources of information and information extractors.

II. Background

APTs have undergone substantial evolution since the term was introduced by the United States Air Force in 2006. Initially conceived to facilitate discussions regarding intrusion activities between military and civilian entities, the concept has since expanded to encompass highly sophisticated, state-sponsored cyber operations with significant geopolitical ramifications [1]. The National Institute of Standards and Technology (NIST) characterizes an APT as "an adversary with sophisticated levels of expertise and significant resources, allowing it through the use of multiple different attack vectors to generate opportunities to achieve its objectives" [4]. Typically, these objectives involve establishing prolonged access to targeted information technology infrastructures to persistently exfiltrate data or disrupt critical systems.

APTs differ fundamentally from conventional cyber threats in several dimensions. While traditional attacks often involve single actors targeting unspecified systems for immediate financial gain, APTs are executed by well-resourced, organized groups against specific organizations or governmental institutions for strategic advantages [5]. The Stuxnet attack in 2010, which targeted Iran's nuclear program by manipulating programmable logic controllers (PLCs), exemplifies this evolution—demonstrating unprecedented sophistication in a cyber weapon designed for targeted physical impact through prolonged, stealthy operation [6].

Integrating malware samples with corresponding threat intelligence can enhance APT analysis. This integration can provide context that binary analysis alone cannot offer revealing strategic intentions, attribution details, and operational patterns that characterize specific threat actors. In fact, as observed in [7], this correlation enables analysts to identify similarities between campaigns and develop effective protective measures against similar attacks.

The analysis of APT attack vectors must extend beyond traditional executable files. Document-based malware, mobile applications, and fileless components have become increasingly prevalent in sophisticated attacks [8]. These vectors often serve as initial infection mechanisms, leveraging widespread use of office documents and inherent vulnerabilities in common applications. Processing these diverse file types provides additional correlation points between seemingly disparate campaigns and enhances detection capabilities against evolving APT methodologies.

III. TRAKR

This section presents TRAKR, a system designed for the integration and examination of APT malware samples along with their corresponding threat intelligence narratives. Figure 1 provides an overview of TRAKR's pipeline. It employs a multi-faceted approach to data collection, processing, and analysis, establishing correlations between different cybersecurity information sources. Contrary to most academic datasets, TRAKR is frequently updated with new reports and binary samples.

A. Crawling

The crawling component collects cyber threat intelligence and binaries from heterogeneous sources. TRAKR employs specialized crawlers adapted to each intelligence source's unique structure and accessibility constraints. We classify the data sources of our crawling component in four types: datasets, APIs, web sources, and proprietary sources. Table I provides a summary of the data sources available in the current version of TRAKR. The following section elaborates on the specifics of each of these.

1) Datasets: TRAKR combines academic and dynamic online datasets to ensure analytical rigor and relevance to current threats. Academic sources—characterized by their structural consistency—provide 33,976 samples [9, 10, 11, 12], but they typically exhibit a latency that limits their utility for tracking emerging threats (i.e., they tend to be static snapshots taken at a specific point in time). To address this, TRAKR integrates live sources such as VX-Underground[1], contributing 35,657 binaries and 2,102 reports[2] that reflect emerging APT techniques with minimal delay.

2) APIs: Public APIs deliver regulary updated threat intelligence through standardized query protocols. TRAKR utilizes the VirusTotal API to obtain 768 binary samples not present in other datasets, along with their corresponding metadata.

3) Web Sources: Many threat reports are published via CTI provider websites (e.g., FireEye, Mandiant, CrowdStrike) in formats designed for human consumption rather than automated parsing. Despite extraction challenges, these sources remain critical as they contain nuanced intelligence often overlooked in structured sources. TRAKR collects reports from 14 web sources, totaling 22,784 reports.

4) Proprietary Sources: TRAKR can incorporate proprietary datasets subject to confidentiality constraints, enriching the framework with specialized knowledge typically unavailable in public domains.

Through the integration of these diverse data sources, TRAKR establishes a basis for correlating disparate pieces of threat intelligence. In total, TRAKR includes 23 different sources of information, 6 of which are static and 17 dynamic. Table I provides an overview of these data sources from which reports and binaries are obtained.

B. Data Processing

The processing pipeline of TRAKR consists of two parallel processes (one for threat reports and another for binary files) that converge in a unified correlation engine.

[1]https://vx-underground.org/apt

[2]As of March 10, 2025.

Table I
SUMMARY OF DATA SOURCES USED BY TRAKR AS OF MARCH 2025.

Name	Type	# binaries	# reports
APTMalware [9]	Dataset	3594	-
ADAPT it! [10]	Dataset	6134	-
APT-class [11]	Dataset	20242	-
vx-underground	Dataset	35657	2102
dAPTs [12]	Dataset	3238	-
VirusTotal	API	768	-
APT Cybercampaign	Dataset	-	1585
APT_REPORT	Dataset	-	340
APTs -database	Dataset	-	419
APTnotes	Dataset	-	688
ORKL Archive	CTI Website	-	13267
Microsoft	CTI Website	-	106
Kaspersky Securelist	CTI Website	-	310
Blackberry	CTI Website	-	286
Cyber Operations Tracker	CTI Website	-	1608
Sentinel Labs	CTI Website	-	34
Talos intelligence	CTI Website	-	57
The DFIR Report	CTI Website	-	164
Checkpoint research	CTI Website	-	397
Cyberseason	CTI Website	-	100
Cyberhint	CTI Website	-	36
Labs k7	CTI Website	-	20
ETDA Encyclopedia	CTI Website	-	4481

1) Reports: Reports are obtained in different forms depending on their source (pdf or html files). We extract the textual information via Python libraries. Some raw texts are obtained from their web source parsing the request's result with *BeautifulSoup*, while others required downloading .pdf files. These files were parsed using the *pdfplumber* library in order to extract the raw text they contain.

Indicators of Compromises (IoCs) are extracted using *iocsearcher* [13], which fetches relevant information by seeking patterns with regular expressions, like IP addresses, hashes, domain names, etc. Both the date the report was collected and its URL are also stored as attributes for traceability and reproducibility purposes.

2) Binary Files: The diverse data sources ingested by TRAKR can yield duplicate binaries from multiple origins. To address this, TRAKR employs a unified schema that uses cryptographic hashes as unique identifiers and normalizes heterogeneous metadata—including contextual attributes, attribution details, and malware labels—into a single record that preserves all available intelligence while eliminating redundancy.

Our framework leverages the VirusTotal API for multi-engine detection reports and employs AVClass2 [14] for taxonomic categorization, which normalizes vendor labels to reduce naming convention inconsistencies. For robust file type identification, we use Magika [15], which has demonstrated high accuracy in recent research [8] analyzing various file formats in APT operations.

TRAKR extracts indicators across the full spectrum of malware artifacts in APT operations, including Windows executables, document-based threats, mobile applications, and fileless malware components. This comprehensive approach reflects the evolving tactics of contemporary APT campaigns, where non-executable files like Microsoft Office documents often serve as initial infection vectors due to their widespread use and inherent vulnerabilities [8].

a) Executable Binary Analysis: TRAKR employs *capa* [16] to identify malware capabilities by analyzing code struc-

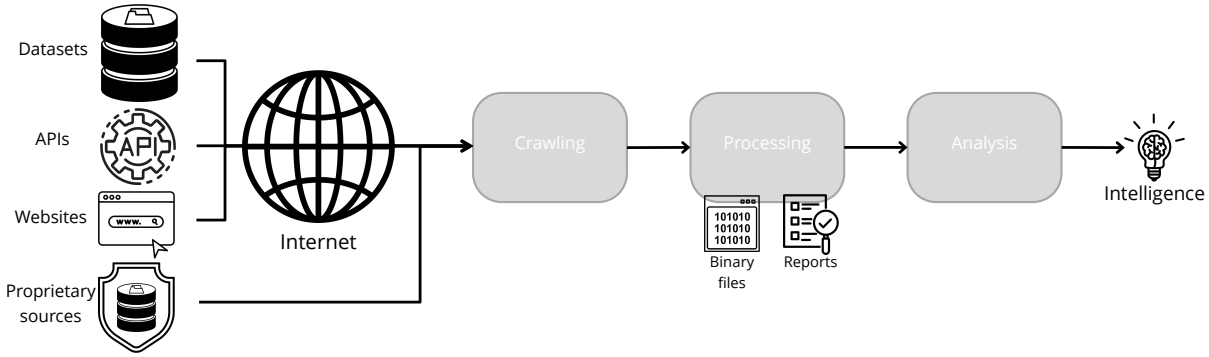

Figure 1. Overview of TRAKR

tures and API references without execution. *Radare2* [3] extracts additional indicators such as entry points and imported libraries, while *Floss* [4] deobfuscates strings, which are further processed with *iocsearcher* [13] to extract and categorize IoCs.

b) Document-based Analysis: Documents are a prevalent form of infection, TRAKR incorporates *Oletools* [17] to extract information from Microsoft *Oli2*, *RTF*, and *OpenXML* file formats, detecting features like encryption and embedded VBA macros.

c) Compressed Files: On numerous occasions, a malicious file will include other files embedded within. TRAKR extracts the files contained within others for a subsequent analysis. In its current version, TRAKR has specialized extractors for *zip* files and derivatives, and for document files from the Office suite such as *docx*, *pptx* or *odt* files, among others.

C. Analysis

TRAKR's processing component transforms raw binary files and CTI reports into structured, correlatable data. Its analysis component integrates these sources to link binaries with CTI reports, individually and together. This section introduces TRAKR's initial analsis capabilities.

a) Reports: TRAKR aggregates and analyzes CTI reports from a diverse set of sources, regarding threat actor activity, tactics, and observed incidents. By normalizing and cross-referencing information from multiple reports, TRAKR can help identify recurring patterns, correlate IoCs between campaigns, and detect changes in adversary behavior over time. The centralization of reports from both structured and unstructured sources can be advantageous in obtaining a detailed assessment of the threat landscape, enhancing the ability to track long-term trends in APT operations.

b) Malware Files: TRAKR is being developed to support the creation of binary similarity data structures that can cross-relate different malware samples based on shared components. As a work in progress, we are exploring the application of deep learning techniques to identify potential code fragments, libraries, or development patterns shared within and between APT campaigns. This ongoing effort aims to facilitate the discovery of previously unrecognized connections between seemingly disparate threat actors. In parallel, we are investigating the use of compilation timestamps to construct chronological patterns of development activity. However, this must be approached with caution, as such timestamps can be manipulated by sophisticated attackers to hinder accurate attribution.

c) Cross-correlation: TRAKR performs cross-referencing between cryptographic hashes mentioned in threat reports and those present in malware repositories, establishing concrete links between samples and their contextual descriptions. Beyond basic hash matching, the framework allows to derive intelligence from the collected data through the different processing steps detailed in the previous component.

TRAKR utilizes *capa* to align capabilities extracted from binary files with the MITRE ATT&CK framework, thereby standardizing behavioral annotations within a unified taxonomy. This correlation verifies the observed binary activities against the TTPs described in threat reports, allowing new insights to be derived from the elaboration of those reports and the malware they refer to.

In addition, TRAKR integrates IoCs from both threat reports and malware samples to create sets of indicators for identified campaigns or actors. By examining relationships between different IoCs types, the system can identify infrastructure patterns that may not be evident when examining indicators in isolation.

IV. Early Results

This section provides some early results obtained by applying the analysis components of TRAKR on data obtained up to March 10th, 2025.

A. Binary Dataset Overview

Our analysis of malware samples collected through TRAKR reveals interesting notes about the current state of APT malware collection. After applying the de-duplication process described in Section III, we mapped the distribution of samples across our source repositories to identify patterns of overlap and uniqueness.

Figure 2 presents a visual representation of the intersection between malware samples collected from various repositories. The diagram illustrates an interesting phenomenon: Despite analyzing approximately 47,821 total samples, the majority appear uniquely in individual repositories, with a small overlap between sources.

This limited intersection, 27 samples across all five repositories, suggests that current APT malware collection efforts

[3]https://github.com/radareorg/radare2
[4]https://github.com/mandiant/flare-floss

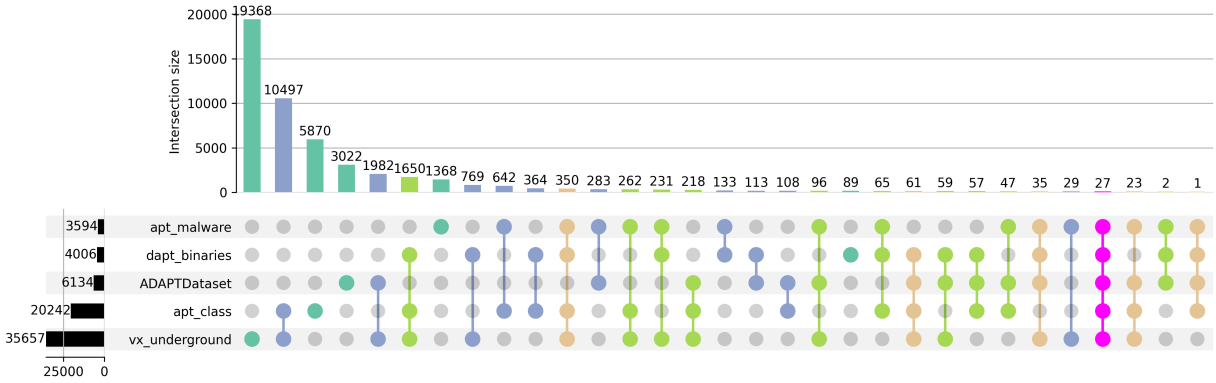

Figure 2. Venn diagram illustrating the intersection of malware files collected from various sources by TRAKR.

operate in relative isolation, with each repository capturing distinct segments of the threat landscape. The substantial number of unique samples per source indicates divergent collection methodologies and potentially different targeting priorities among cybersecurity researchers and organizations.

The minimal overlap demonstrates that relying on any single source, regardless of its reputation or size, would result in substantial blind spots in APT coverage. This supports TRAKR's multi-source integration approach, synthesizes disparate data repositories.

B. Malware Taxonomic Analysis

1) File Type Distribution: Analysis of file types in our dataset reveals diverse attack vectors employed in APT campaigns. As shown in Table II, Windows executable files (*pebin*) constitute 73.08% of collected samples, reflecting Windows' prevalence in enterprise environments.

Table II
DISTRIBUTION OF THE 8 MOST COMMON FILE TYPES IN MALWARE SAMPLES COLLECTED BY TRAKR

File Type (Magika)	# Files	Proportion
pebin	34948	73.08%
apk	2253	4.71%
doc	1007	2.11%
docx	926	1.94%
rtf	920	1.92%
elf	848	1.77%
xls	722	1.51%
zip	714	1.49%

Document formats (*rtf*, *doc*, *docx*, and *xls*) collectively represent a significant portion of the dataset, confirming their increasing use as sophisticated attack vectors in APT operations, consistent with recent findings [8]. The presence of mobile malware (*apk*), Linux executables (*elf*), and compressed archives (*zip*) demonstrates the multi-platform nature and complex deployment mechanisms of modern APT campaigns.

This diversity underscores a critical requirement for APT analysis frameworks: analytical capabilities must extend beyond conventional executable analysis to include specialized processing for document-based, mobile, and other non-traditional vectors to avoid critical blind spots in detection and characterization efforts.

2) APT Group Attribution Distribution: Analysis of APT group attributions across our dataset reveals different patterns in the threat landscape captured by TRAKR. As shown in Table III, APT17, a threat actor attributed to China, represents the largest proportion of samples at 7.79%. This is followed by SIG45 (5.59%) and the North Korean attributed actor Lazarus Group (3.75%).

Table III
DISTRIBUTION OF TOP 10 APT GROUPS ASSOCIATED WITH MALWARE FILES COLLECTED BY TRAKR

APT Group	# Files	Proportion	Country
apt17	2137	7.79%	China
sig45	1533	5.59%	Unknown
lazarus group	1028	3.75%	North Korea
sig9	749	2.73%	Unknown
sig17	748	2.73%	Israel
Gorgon Group	741	2.70%	Pakistan
sig25	683	2.49%	South Korea
oceanlotus	668	2.44%	Vietnam
Lazarus	516	1.88%	North Korea
Gamaredon	446	1.63%	Russia

The attribution distribution highlights two challenges in APT analysis. First, inconsistent naming conventions appear throughout the dataset, exemplified by "Lazarus Group" and "Lazarus" being treated as distinct entities despite referring to the same threat actor. Second, a substantial portion of the samples are associated with groups of uncertain origin (e.g. SIG45, SIG9), reflecting the difficulties in attribution.

Future work will improve APT group and country attribution by integrating additional data sources to develop an ontology system that enables more accurate and consistent identification of threat actors and their respective countries.

C. Cross-Dataset Correlation Results

From our CTI repository containing 26,000 scraped reports, we extracted 111,649 unique file hashes (40,310 MD5, 19,913 SHA1, and 51,426 SHA256 hashes). When correlated against our dataset, containing 47,821 file samples, we identified intersections of 12,208 matching MD5 hashes (30.3% of our repository's MD5 collection), 4,829 matching SHA1 hashes (24.3% of SHA1s), and 12,203 matching SHA256 hashes (23.7% of SHA256s). It is reasonable to expect reports not mentioning a large part of the hashes we obtain from files. This is because of two main reasons. First, reports tend to

mention the most relevant files observed within an attack. Second, we perform unpacking of several filetypes, resulting in additional files that may not have been included within reports. A less expected result, is the large number of hashes of each kind that we were not able to map to files within our binary dataset. This could be because these hashes refer to benign tools being used during APT attacks (also known as living-off-the-land [18]. We plan to investigate this behavior in more detail in the future.

V. Conclusions

In this paper we present TRAKR, a framework for collecting, processing and analyzing APT-related cyber threat intelligence reports and binary files. TRAKR's adaptable architecture supports data from 23 sources and analysis of 13 different file types beyond executables, with regular updates to maintain currency. Our initial analysis reveals significant fragmentation in APT research, with datasets largely operating in isolation. Approximately one-quarter to one-third of malware specimens discussed in threat reports are represented in common malware repositories, while also highlighting the significant volume of potentially novel or underreported threats documented exclusively in our intelligence corpus. This highlights the need for comprehensive frameworks that integrate diverse sources. Future work will focus on expanding TRAKR's data collection capabilities and developing new analytical methods to derive insights from the collected intelligence.

Acknowledgments

This work has been supported by the MATM project (C127/23), with the collaboration of the Spanish Institute for Cybesecurity (INCIBE), and the Recovery, Transformation and Resilience Plan funded by the European Union (Next Generation).

References

[1] I. Jeun, Y. Lee, and D. Won, "A practical study on advanced persistent threats," in *Computer Applications for Security, Control and System Engineering*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 144–152.

[2] O. C. T. I. C. T. Committee, "Stix version 2.1: Structured threat information expression," 2021, accessed: 2025-03-17. [Online]. Available: https://docs.oasis-open.org/cti/stix/v2.1/stix-v2.1.html

[3] Z. Li, J. Zeng, Y. Chen, and Z. Liang, "Attackg: Constructing technique knowledge graph from cyber threat intelligence reports," in *European Symposium on Research in Computer Security*. Springer, 2022, pp. 589–609.

[4] National Institute of Standards and Technology. Apt definition(s). Accessed: 2025-03-19. [Online]. Available: https://csrc.nist.gov/glossary/term/apt

[5] P. Chen, L. Desmet, and C. Huygens, "A study on advanced persistent threats," in *Communications and Multimedia Security: 15th IFIP TC 6/TC 11 International Conference, CMS 2014, Aveiro, Portugal, September 25-26, 2014. Proceedings 15*. Springer, 2014, pp. 63–72. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-662-44885-4_5

[6] M. Baezner and P. Robin, "Stuxnet," ETH Zurich, Tech. Rep., 2017. [Online]. Available: https://www.research-collection.ethz.ch/bitstream/handle/20.500.11850/200661/1/Cyber-Reports-2017-04.pdf

[7] B. Tang, J. Wang, Z. Yu, B. Chen, W. Ge, J. Yu, and T. Lu, "Advanced persistent threat intelligent profiling technique: A survey," *Computers and Electrical Engineering*, vol. 103, p. 108261, 2022.

[8] A. Saha, J. Blasco, and M. Lindorfer, "Exploring the malicious document threat landscape: Towards a systematic approach to detection and analysis," in *2024 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE, 2024, pp. 533–544.

[9] C. Boot, A. C. Serban, and E. Poll, "Supervised learning for state-sponsored malware attribution."

[10] A. Saha, J. Blasco, L. Cavallaro, and M. Lindorfer, "Adapt it! automating apt campaign and group attribution by leveraging and linking heterogeneous files," in *Proceedings of the 27th International Symposium on Research in Attacks, Intrusions and Defenses*, 2024, pp. 114–129.

[11] J. Gray, D. Sgandurra, L. Cavallaro, and J. Blasco Alis, "Identifying authorship in malicious binaries: Features, challenges & datasets," *ACM Computing Surveys*, vol. 56, no. 8, pp. 1–36, 2024.

[12] G. Laurenza and R. Lazzeretti, "daptaset: A comprehensive mapping of apt-related data," in *Computer Security: ESORICS 2019 International Workshops, IOSec, MSTEC, and FINSEC, Luxembourg City, Luxembourg, September 26–27, 2019, Revised Selected Papers 2*. Springer, 2020, pp. 217–225.

[13] J. Caballero, G. Gomez, S. Matic, G. Sánchez, S. Sebastián, and A. Villacañas, "The rise of goodfatr: A novel accuracy comparison methodology for indicator extraction tools," *Future Generation Computer Systems*, vol. 144, pp. 74–89, 2023.

[14] S. Sebastián and J. Caballero, "Avclass2: Massive malware tag extraction from av labels," in *Proceedings of the 36th Annual Computer Security Applications Conference*, 2020, pp. 42–53.

[15] Y. Fratantonio, E. Bursztein, L. Invernizzi, M. Zhang, G. Metitieri, T. Kurt, F. Galilee, A. Petit-Bianco, and A. Albertini, "Magika content-type scanner," *Magika Content-type Scanner*, 2023.

[16] W. Ballenthin, M. Raabe, F. Team *et al.*, "capa: Automatically identify malware capabilities," *Fire Eye Threat Research Blog*, 2020.

[17] P. Lagadec, "oletools: Python tools to analyze microsoft ole2 files (structured storage, compound file binary format) and ms office documents," 2025, accessed: 2025-03-03. [Online]. Available: https://github.com/decalage2/oletools

[18] F. Barr-Smith, X. Ugarte-Pedrero, M. Graziano, R. Spolaor, and I. Martinovic, "Survivalism: Systematic analysis of windows malware living-off-the-land," in *2021 IEEE Symposium on Security and Privacy (SP)*, 2021, pp. 1557–1574.