A Relevant Feature Identification Approach to Detect APTs in HTTPS Traffic

Abdou Romaric Tapsoba, Tounwendyam Frédéric Ouédraogo
UFR Sciences et Technologies, Université Norbert Zongo, Koudougou, Burkina Faso

***Abstract*—This study addresses the significant challenges posed by Advanced Persistent Threats (APTs) in modern computer networks, particularly their use of DNS to establish covert communication via command and control (C&C) servers. The advent of TLS 1.3 encryption further complicates detection efforts, as critical data within DNS over HTTPS (DoH) traffic remains inaccessible, and decryption would compromise user privacy. APTs frequently leverage Domain Generation Algorithms (DGAs), necessitating real-time detection solutions based on immediately accessible features within HTTPS traffic. Current research predominantly focuses on system-level behavioral analysis, often neglecting the proactive potential offered by Cyber Threat Intelligence (CTI), which can reveal malicious patterns through Techniques, Tactics, and Procedures (TTPs) and Indicators of Compromise (IoCs). This study proposes an innovative approach utilizing the MITRE ATT&CK framework to identify relevant features in the face of encryption and the inherent complexity of APT activities. The primary objective is to develop a robust dataset and methodology capable of detecting APT behaviors throughout their lifecycle, emphasizing a lightweight, cost-effective solution through passive monitoring of network traffic to ensure real-time detection. The key contributions of this research include an in-depth analysis of the encryption challenges in detecting DNS-based APTs, a thorough examination of APT attack strategies using DNS, and the integration of CTI to enhance detection capabilities. Moreover, this study introduces the KAPT 2024 dataset, generated by the KExtractor tool, and demonstrates the effectiveness of the detection model through experiments with a variety of machine learning algorithms. The results underscore the potential for this approach to significantly improve APT detection in encrypted network environments.**

Keywords*—*DNS over HTTPS; advanced persistent threats; machine learning; cyber threat intelligence; MITRE ATT&CK; domain generation algorithms

I. Introduction

Advanced Persistent Threats present significant challenges to security, representing a serious threat that demands thorough research and rigorous evaluation of effective detection techniques. These malicious actors, driven by various objectives ranging from espionage to service disruption, exploit sophisticated communication channels to establish connections with their Command and Control servers. Recent observations indicate that APT actors are increasingly leveraging DNS, even when encrypted, to establish these communications, thereby evading traditional detection methods.The use of DNS by APT actors as a communication channel can be detected through traffic analysis. However, existing machine learning-based methods for detecting malicious domains face significant challenges with HTTPS traffic because key features, such as textual and lexical domain information, NXDomain volumes, and other relevant data, are encrypted in TLS 1.3. The analysis of directly exploitable information within DoH traffic is hindered by the inaccessibility of a large amount of meaningful data, while decryption methods would compromise privacy. Another challenge, due to the evolving and stealthy nature of APT threats using DGA algorithms, is the responsiveness of malicious domain detection, which requires real-time analysis based on immediately accessible features in HTTPS traffic. The dynamic and evolving nature of these attacks necessitates immediate responsiveness, as the effectiveness of any security measure could be compromised without real-time detection [1].

Several techniques have been proposed in the literature to counter APT threats in general. Most current research on APT detection, based on machine and deep learning, focuses on behavioral analysis of attacks at the system level, thus neglecting crucial adversarial intelligence that could proactively contribute to threat prevention [2], [3]. Cyber Threat Intelligence has emerged as a potential solution to help organizations address the complex and stealthy nature of cyber threats [4]. The exploration of intelligence platforms involves extracting Techniques, Tactics, and Procedures and Indicators of Compromise on threats. TTPs and IoCs play a crucial role in identifying malicious behaviors and attack patterns specific to APTs. The term “tactics” refers to the method used by the APT to carry out the attack from start to finish. The “techniques” used by the APT during its attack describe its technological strategy to achieve its goals. Finally, the “procedure” of an APT describes the steps used by the attacker to achieve its objectives [5]. Many researchers have used machine learning to detect APT threats. However, these proposed methods do not consider detection at all levels of the APT lifecycle [6]. Additionally, the lack of datasets thoroughly exploiting the TTPs and IoCs provided by intelligence platforms does not proactively promote the detection of C&C domains submerged by DGAs, constituting a current research challenge [7]. Publicly available datasets can detect several levels of the cycle, but not the entirety of the phases, as many works have unfortunately not mentioned the use of persistence and stealth tools such as DGAs in the lifecycle. We aim to develop a system to analyze and detect malicious domains at every stage of the APT lifecycle using DNS, even when encrypted, as a communication channel. Our method must meet several key constraints, including ensuring privacy by avoiding the use of any decryption techniques and relying solely on clear-text features directly accessible from the traffic. Additionally, the solution should be lightweight and low-cost, requiring no equipment or installation on endpoints, and based on passive network traffic monitoring. Given the nature of APT threats, the method must also ensure efficiency and responsiveness, enabling quick reaction with real-time detection.

This study makes several significant contributions to the field of cybersecurity, particularly in the detection of APTs exploiting DNS. It provides a comprehensive analysis of the challenges posed by encryption in threat detection, addressing the limitations of current methods for detecting DGA attacks, which are increasingly complicated by the widespread encryption of communications. The research offers an in-depth examination of APT attack strategies, detailing the TTPs and IoCs employed by APTs to exploit DNS. Furthermore, it highlights the role of CTI platforms in enhancing detection capabilities by integrating relevant data sources and enriching detection features. The originality of this work lies in its innovative methodological approach, which combines Artificial Intelligence and Threat Intelligence to create a robust dataset, namely KAPT 2024. This dataset, coupled with importance level indicators, has been rigorously tested with various machine learning algorithms, demonstrating the effectiveness of the proposed architectural model in real-time, multi-class detection scenarios, thereby contributing significantly to the advancement of cybersecurity research.

The remainder of this paper is organized as follows. Section II provides a literature review on the subject. Section III discusses the contribution of threat intelligence in APT detection. Section IV details the steps of our proposed methodology, including the data sources used, feature extraction, and the multi-class classification module. Section V implements and evaluates the methodological approach. Finally, Section VI concludes the article and suggests future research directions.

II. Related Work

The literature review on Advanced Persistent Threats highlights the ongoing evolution of sophisticated attacks targeting computer systems and networks. Various techniques have been developed to counter APTs, with recent research emphasizing the predominant use of machine learning techniques in detecting these threats [5]. These studies underline the importance of continuous research to enhance APT detection capabilities and mitigate cybersecurity risks as threats evolve. For instance, Weiwu Ren *et al.* have analyzed the effectiveness of deep learning for precise and real-time APT detection [8], while Nkiruka Eke *et al.* proposed a hybrid model combining deep and machine learning techniques for more effective detection [9]. Manuel Miguez *et al.* contributed by proposing a cyber kill chain model and early detection methodology for APTs, stressing the importance of a strategic defense approach [10]. APT attacks, meticulously planned by malicious actors, generally involve several stages. While these actors may exhibit distinct characteristics, the phases of their attacks are typically similar, differing mainly in the tactics and techniques used in each phase. Alshamrani *et al.* categorized APT attacks into five stages: Reconnaissance, Establish Foothold, Lateral Movement/Stay Undetected, Exfiltration/Impediment, and Post-Exfiltration/Post-Impediment [11]. The authors argue that these stages can represent any APT attack, regardless of the objective. Several studies [2], [7], [9] in the literature have drawn inspiration from this schema presented in [11]. By leveraging this cycle, as done in the present study, it is possible to detect and prevent APT attacks by understanding the techniques and tactics employed by attackers [2].

CTI plays a crucial role in compiling a comprehensive database on APT behavior. CTI helps manage indicators of compromise, techniques, tactics, and protocols associated with various APT groups. Building on this foundation, Abir Dutta *et al.* proposed the integration of machine learning with a threat intelligence platform [12]. Other works [13], [14] have also emphasized the importance of CTI in enhancing enterprise resilience. Researchers like Yinghai Zhou *et al.* [15] and Nan Sun *et al.* [3] have explored the use of CTI to counter APT attacks by employing automatic extraction and analysis methods for CTI information. Additional studies [16], [17] have presented a framework for sharing CTI that incorporates machine learning models. The challenges of effective threat intelligence sharing are explored in [18], while the works mentioned in [19] propose a trust taxonomy for sharing threat information. Among the most widely used intelligence platforms is the MITRE ATT&CK matrix, which catalogues a comprehensive set of TTPs used by adversaries in each phase of their attacks [3], [5]. The matrix's database is continuously updated with contributions from the research community, making it a cornerstone for APT threat intelligence. Certain studies underscore the increasing importance of integrating threat intelligence and machine learning for defending against cyberattacks [20], [21].

While existing research underscores the importance of machine learning and CTI in detecting APT threats, it also reveals significant limitations that need to be addressed. The necessity of improving machine learning models by exploring CTI platforms for enhanced resilience is evident. However, the exploitation of TTPs and IoCs remains insufficient, as the publicly available training datasets do not comprehensively cover all stages of the APT lifecycle. Consequently, many of the proposed methods fall short of detecting APT threats across all lifecycle stages, particularly when it comes to identifying C&C domains obfuscated by DGAs. This gap highlights the need for further research and the development of innovative methodologies that can overcome these challenges and provide more effective APT detection systems.

III. Contribution of CTI in Detecting APTs

Cyber Threat Intelligence has emerged as a potential solution for companies to address the complex and stealthy nature of cyber threats [13]. Exploring intelligence platforms involves extracting TTPs and IoCs about threats. TTPs and IoCs play a crucial role in identifying malicious behaviors and attack patterns specific to APTs. TTPs describe how attackers achieve their objectives, while IoCs provide concrete evidence of an intrusion, such as IP addresses, file hashes, or malicious domain names. Various tactics, techniques, and procedures are used at each stage of an APT attack, which progresses to the next stage. Given that the TTP attribute allows profiling an APT actor, it is relevant to consider it as a constituent element of a specific detection technique, and it can be used to anticipate and identify APT attacks early [22].

By anticipating the tactics used by attackers, security teams can strengthen their defenses, identify weak points, and develop appropriate detection and response strategies. In-depth knowledge of TTPs also allows for better targeting of security investments and maximizing the effectiveness of protection tools and technologies. In this context, the use of the MITRE ATT&CK matrix proves to be a valuable asset for security professionals. The ATT&CK matrix provides a comprehensive view of the TTPs used by attackers, categorized by attack stage and target platform. By integrating ATT&CK matrix data into their security strategies, organizations can gain a deep understanding of the tactics employed by APTs, as well as the IoCs associated with each stage of the attack. MITRE ATT&CK has been used to represent APT TTPs as it provides extensive knowledge of adversary tactics and techniques based on real-world observations. The ATT&CK knowledge base has been widely used as a foundation for developing specialized threat models and techniques in the business, government, and cybersecurity sectors worldwide [5]. The choice of MITRE as a source of information on TTPs and IoCs stems from its reputation for excellence in the field of cybersecurity. In addition to the ATT&CK matrix, MITRE offers a multitude of free and paid resources, such as technical reports, analysis tools, and training programs. This wealth of resources makes MITRE an essential partner for organizations seeking to strengthen their security posture and protect against APT threats. Compared to methods proposed in the literature, the MITRE platform is better suited to APT threat intelligence, which describes TTPs in a canonical form and can more accurately extract the TTPs summarized in CTI reports [15].

One of the major aspects influencing the accuracy of Machine Learning models is the search for discriminatory features [25]. Indeed, the features designated in one APT detection solution are not necessarily applicable to another solution. The MITRE platform, by providing TTPs and IoCs about APT threats, helps us designate relevant features on the behaviors and techniques used by attackers. By analyzing this data, we can identify the specific patterns and signals associated with APT attacks, allowing us to determine which features are most likely to be relevant for detecting these attacks during each phase. For example, if an IoC indicates that an APT attack used a specific technique to compromise a system, we might select features related to that technique to strengthen our detection model. Similarly, TTPs can guide us toward the features that are most representative of the attack methods used by APTs, enabling us to better target our analysis and defense efforts. The feature selection process begins by identifying representative techniques by exploring the different phases of the APT attack (Initial Access, Execution, Persistence, Privilege Escalation, Defense Evasion, Credential Access, Discovery, Lateral Movement, Collection, Command and Control, Exfiltration, Impact) and identifying the specific techniques at each phase relevant to the type of targeted attacks. Each phase of the cycle identified on the platform is then translated into measurable features in network data. For example, the “Data Exfiltration” phase will have measurable characteristics such as statistics related to the volume of data transferred, the frequency of outgoing connections, etc. There has been immense interest in exploiting CTI, specifically for proactive cybersecurity defense. By exploring this unique dimension of cybersecurity, this research provides new insights and opens avenues for innovation in combating persistent and sophisticated threats, thereby making a significant contribution to the advancement of the field. In the following section, we will delve into the architectural model of our approach for identifying relevant features in detail.

IV. Proposed Methodology

We have already explored intelligence platforms to identify the most significant features that can contribute to the detection of APT threats and designate data sources covering all phases of the APT lifecycle in the previous section. In this section, we present the architecture used in this study, illustrated in Fig. 1. We proceed with feature extraction to construct the KAPT-2024 dataset, leveraging the selected data sources. We conclude the process by training our models with several supervised learning algorithms.

A. Data Source

The accuracy of classification models inevitably relies on the quality of the data. Exploring intelligence platforms allows us to observe the various tools, TTPs, and IoCs used by APT actors during each phase of the APT lifecycle, thereby indicating the different data sources widely referenced in the literature. These data sources, in pcap format, are selected based on the tools used to generate the traffic. These tools enable us to map the threats from each public data source according to the APT lifecycle. This is essentially what will be discussed in this subsection. In total, four data sources will be analyzed to constitute our own dataset.

1) CSE-CIC-IDS2018 on AWS: CSE-CIC-IDS2018[1], developed in collaboration between the Communications Security Establishment (CSE) and the Canadian Institute for Cybersecurity (CIC), is designed to generate a diverse and comprehensive benchmark dataset for intrusion detection. Among the simulated attacks are scenarios such as network infiltration from the inside, HTTP denial-of-service attacks, web application attack collection, brute-force attacks, and attacks based on recent vulnerabilities such as Heartbleed. CSE-CIC-IDS2018 dataset mainly covers the initial compromise, lateral movement, and camouflage phases of the APT threat lifecycle. Included attacks such as brute-force attacks, web attacks, and port scans illustrate a strong focus on the initial compromise phase, crucial for gaining initial access to target systems. Although this dataset is useful for analyzing these critical stages, it does not cover the entire APT threat lifecycle, omitting phases such as persistence, privilege escalation, defense evasion, data exfiltration, and final impact [23].

2) UNSW-NB15: The UNSW-NB15[2] dataset was specifically designed to evaluate Intrusion Detection Systems, making it a valuable tool for threat detection and prevention. With a variety of source files and a wide range of simulated attacks, UNSW-NB15 provides a more realistic and comprehensive representation of modern network traffic compared to some previous datasets like NSL-KDD. The UNSW-NB15 dataset covers several phases of the APT threat lifecycle. Fuzzers and Analysis attacks are associated with the Reconnaissance phase, where attackers gather information about potential targets. Backdoors, DoS, Exploits, and Shellcode attacks fall under the Initial Compromise phase, where attackers exploit vulnerabilities to gain access to target systems. Generic and Worms attacks are linked to the Lateral Movement phase, allowing attackers to move within the compromised network. Although the dataset provides a good representation of critical

[1] https://www.unb.ca/cic/datasets/ids-2018.html

[2] https://research.unsw.edu.au/projects/unsw-nb15-dataset

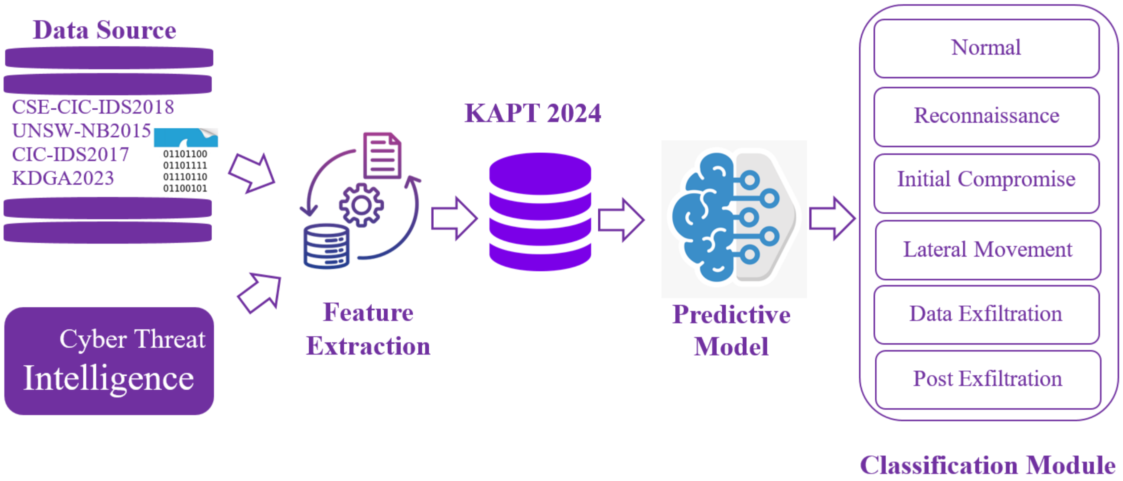


Fig. 1. The methodological structure of the proposed method.

phases, it does not explicitly cover all phases, such as data exfiltration and post-exfiltration [24].

3) CIC-IDS 2017: The CIC IDS 2017[3] dataset is designed for the evaluation of Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS). It contains both benign network traffic and common attacks, thus providing a realistic representation of real-world data. The attacks included in the dataset cover a wide range, including Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web Attack, Infiltration, Botnet, and DDoS. CIC-IDS 2017 covers the Initial Compromise phase with web attacks such as XSS (Cross-Site Scripting) and SQL injection attacks, as well as brute-force attacks on services like FTP and SSH. The Lateral Movement phase is represented by infiltration attacks and the use of botnets to move laterally within the network, with port scans (PortScan) being possible. The Camouflage phase includes DoS attacks like Hulk, GoldenEye, slowloris, slowhttptest, and DDoS, which seek to conceal malicious activities. However, the dataset does not seem to explicitly cover the data exfiltration phase, which is also an essential component of the APT attack lifecycle [23].

4) KDGA-Insight23: The KDGA-Insight23[4] dataset [25] is specifically designed for real-time analysis of DNS traffic, focusing on detecting malicious activities such as Domain Generation Algorithms, particularly in the context of using DNS over HTTPS. It includes 36 features extracted from pcap files, which are used to distinguish different types of DNS traffic, including DoH and non-DoH traffic, DoH-Tunnel and non-Tunnel traffic, as well as DGA and non-DGA traffic. This dataset can contribute to APT threat detection, especially regarding DGA attacks. In the context of DGA attacks, the initial compromise of a host machine is a fundamental step, corresponding to the first two stages of the APT cycle: acquiring initial access and establishing an initial foothold. Subsequently, setting up a tunnel through the Command and Control corresponds to a later stage of the APT cycle, usually associated with lateral movement within the network and exfiltration of sensitive data. This step aims to establish secret communication between the compromised machine and the C&C server, thereby allowing the attacker to access and control the network more broadly. As for the use of DGA, it may be considered in the camouflage or persistence phase, where attackers deploy sophisticated techniques to evade detection and maintain their network access over the long term. The KDGA-Insight23 dataset provides valuable information for APT threat detection, focusing particularly on DGA attacks and the use of tunnels via C&C. These aspects are closely related to several stages of the APT attack lifecycle, thereby enhancing its relevance in the cybersecurity context.

In conclusion, the use of the four datasets CSE-CIC-IDS2018, CIC-IDS2017, UNSW-NB15, and KDGA-Insight23 in our project is of crucial importance for several reasons. Firstly, each dataset offers a unique perspective on threats and potential attacks encountered in the modern cybersecurity landscape. CSE-CIC-IDS2018 and CIC-IDS2017 provide a variety of real-world attacks, allowing for the testing and evaluation of Intrusion Detection Systems effectiveness in detecting common attacks such as DoS, brute-force attacks, and web attacks. On the other hand, UNSW-NB15 focuses on detecting malicious activities related to DNS traffic, offering valuable insight into detecting DNS-based attacks, including DGA attacks. Lastly, KDGA-Insight23 specifically focuses on detecting DGA attacks in the context of using DNS over HTTPS, making it particularly relevant for detecting camouflage and persistence activities associated with APT attacks. By combining these four datasets, we are able to broadly cover the entire lifecycle of APT attacks, from reconnaissance to long-term persistence in the network. Each dataset contributes to filling the gaps of the others in terms of coverage of specific attack types and camouflage techniques used by attackers. The combined use of these four datasets allows us to benefit from a comprehensive and balanced overview of potential threats in the cybersecurity domain, thereby enhancing our ability to develop and evaluate robust and effective Intrusion Detection Systems against APT attacks. Table I represents our dataset, encompassing all phases of the APT threat lifecycle and enabling threat detection at each stage of the cycle. By extracting these features from the data sources explored in this

[3] https://www.unb.ca/cic/datasets/ids-2017.html
[4] https://github.com/artapsoba/KDGA-Insights

TABLE I. APT CYCLE DESCRIPTION AND ATTACK/TOOLS

APT Cycle	Description	Attack/Tools
Reconnaissance	Network reconnaissance, identifying vulnerabilities	PortScan
Initial compromise	Establishing a foothold in the network through various techniques	Brute Force, Sql Injection, XSS, FTP-Patator, SSH-Patator
Lateral Movement	Discovering the internal network through compromised systems and taking control of critical devices	Infiltration attack, Bot ARES
Data Exfiltration	Transferring data from local machines in the network to C&C servers, locations, or remote users	Iodine, Dnscat2, Dns2tcp
Post Exfiltration	Persisting the exfiltration process, disabling other critical components, and destroying evidence to ensure clean removal from the organization's network	DoS GoldenEye, DoS Hulk, DoS Slowhttptest, DoS slowloris, DoS Heartbleed, DDoS LOIC, DGA

subsection, our dataset stands out for its ability to capture and analyze the various aspects of APT attacks through these 87 selected features.

B. Feature Extraction and Data Pre-Processing

Lexical and textual data have largely lost their relevance due to traffic encryption. Information related to DNS, HTTP, and TLS layers, which has been successfully used in the literature, is now encrypted. The widespread adoption of traffic encryption presents fewer opportunities for security professionals and represents one of the major challenges today. This study addresses the issue of respecting user privacy while maintaining an optimal level of security. It seeks to demonstrate the effectiveness of Machine Learning methods using only the information directly accessible from DoH traffic. Feature extraction from network packets is a crucial step in network data analysis, particularly for APT detection. In this process, packets are grouped by flow to capture network interactions between specific IP addresses and ports, thereby defining forward (incoming) and backward (outgoing) flows. This grouping allows for detailed and granular characterization of data flows, facilitating the analysis of suspicious network behaviors. Forward and backward flows are used to distinguish communication directions, which is essential for identifying potential attack patterns such as unusual data transfers or suspicious responses. By analyzing features such as packet lengths, inter-arrival times (IAT), and TCP flags (PSH, URG), traffic patterns can be better understood, and anomalies indicating a threat can be detected. This approach enables the identification of not only the overall characteristics of flows but also the directional nuances that might indicate malicious activities, making the analysis more precise and relevant for APT detection. The dataset is labeled based on the tools used to generate the traffic. We extract a total of 87 features that comprehensively cover the phases of the APT lifecycle. Depending on the tools used to generate the data sources utilized in this study, we have grouped the raw data into six categories (Normal, Reconnaissance, Initial Compromise, Lateral Movement, Data Exfiltration, and Post Exfiltration) as outlined in Table I. Preprocessing involves cleaning the dataset of all its outliers. The transformations applied to this dataset include digitization, normalization, imputation of missing values, and feature selection. Normalization involves changing the range of values from a large range to a smaller one, typically [0, 1] or [-1, 1] [26]. In this study, data normalization was conducted using the *MinMaxScaler()* method from the Sklearn library, followed by data imputation, which involved removing rows with missing values. The next step focused on selecting the most significant features. This process began with analyzing the correlation between features using a heatmap, which helped identify and remove highly correlated, redundant columns, thereby simplifying the model and improving its performance. The study then employed Recursive Feature Elimination (RFE) to optimize feature selection, retaining only the most relevant variables. This approach reduced noise, improved predictive accuracy, and stabilized performance with around 55 features, leading to a more efficient and accurate model.

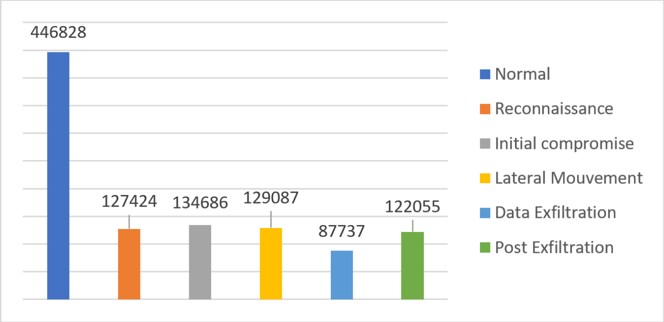


Fig. 2. Collected dataset.

C. Dataset KAPT 2024

The KAPT24 dataset is designed to address the challenges posed by APT threats. It is structured around the complete lifecycle of APT threats, covering the phases of Reconnaissance, Initial Compromise, Lateral Movement, Data Exfiltration, and Post-Exfiltration. The primary objective of this dataset is to provide a comprehensive solution for detecting APT threats by leveraging features extracted directly from HTTPS traffic. To construct this dataset, we utilized intelligence platforms such as MITRE ATT&CK to identify relevant Techniques, Tactics, and Procedures and Indicators of Compromise. This information was crucial in selecting features that effectively capture malicious behaviors. This approach allows for the detection of suspicious activities in a non-intrusive manner, making the dataset valuable for research and the development of new threat detection techniques. The data is collected and classified into different phases of the APT threat lifecycle, as illustrated in Fig. 2. This dataset includes the following categories: Normal (446,828 samples), Reconnaissance (127,424 samples), Initial Compromise (134,686 samples), Lateral Movement (129,087 samples), Data Exfiltration (87,737 samples), and Post Exfiltration (122,055 samples).

A thorough analysis is conducted to select the most relevant features, those that demonstrate a significant ability to discriminate between normal traffic and malicious traffic related to APTs. The choice of features and the method of grouping by flow are motivated by the need to capture the complex dynamics and abnormal behaviors that characterize APT attacks. The forward and backward flows provide a detailed view of network interactions, thus facilitating the identification of anomalies typical of different phases of an APT attack. The features of the KAPT24 dataset, summarized in Table II, include essential information for network flow analysis and APT threat detection.

TABLE II. EXTRACTED FEATURES

Feature Group	**Features**
Flow Identification	F01: FlowID, F02: SrcIP, F03: DstIP, F04: SrcPort, F05: DstPort, F06: Protocol, F07: Timestamp
Flow Duration and TTL	F08: Fl_Duration, F09: TTL, F10: DistinctTTLValue
Packet Counts	F11: Tot_Fwd_Pkts, F12: Tot_Bwd_Pkts, F13: TotLen_FwdPkts, F14: TotLen_BwdPkts
Packet Length Statistics	F15: Fwd-Pkt_Len_Max, F16: Fwd-Pkt_Len_Min, F17: Fwd-Pkt_Len_Mean, F18: Fwd-Pkt_Len_Std, F19: Bwd-Pkt_Len_Max, F20: Bwd-Pkt_Len_Min, F21: Bwd-Pkt_Len_Mean, F22: Bwd-Pkt_Len_Std
Flow Rates	F23: Flow_Byts_sec, F24: Flow_Pkts_sec
Inter Arrival Times	F25: Flow_IAT_avg, F26: Flow_IAT_Std, F27: Flow_IAT_Max, F28: Flow_IAT_Min, F29: Fwd_IAT_Tot, F30: Fwd_IAT_avg, F31: Fwd_IAT_Std, F32: Fwd_IAT_Max, F33: Fwd_IAT_Min, F34: Bwd_IATTot, F35: Bwd_IAT_avg, F36: Bwd_IAT_Std, F37: Bwd_IAT_Max, F38: Bwd_IAT_Min
Flag Counts	F39: Fwd_PSH_Flags, F40: Bwd_PSH_Flags, F41: Fwd_URG_Flags, F42: Bwd_URG_Flags, F54: FIN_Flag_Cnt, F55: SYN_Flag_Cnt, F56: RST_Flag_Cnt, F57: PSH_Flag_Cnt, F58: ACK_Flag_Cnt, F59: URG_Flag_Cnt, F60: CWE_Flag_Cnt, F61: ECE_Flag_Cnt
Header and Byte Metrics	F43: Fwd_Header_Len, F44: Bwd_Header_Len, F45: Fwd_Byts_sec, F46: Bwd_Byts_sec, F47: Fwd_Pkts_sec, F48: Bwd_Pkts_sec
Ratios and Averages	F62: Down_Up_Ratio, F63: Pkt_Size_Avg, F64: Fwd_Seg_Size_Avg, F65: Bwd_Seg_Size_Avg, F66: Fwd_Byts_blk_Avg, F67: Fwd_Pkts_blk_Avg, F68: Fwd_Blk_Rate_Avg, F69: Bwd_Byts_blk_Avg, F70: Bwd_Pkts_blk_Avg, F71: Bwd_Blk_Rate_Avg
Subflows	F72: Subflw_Fwd_Pkts, F73: Subflw_Fwd_Byts, F74: Subflw_Bwd_Pkts, F75: Subflw_Bwd_Byts
Initial Window Metrics	F76: Init_Fwd_Win_Byts, F77: Init_Bwd_Win_Byts
Active and Idle Times	F78: Fwd_Act_Data_Pkts, F79: Fwd_Seg_Size_Min, F80: Active_Mean, F81: Active_Std, F82: Active_Max, F83: Active_Min, F84: Idle_Mean, F85: Idle_Std, F86: Idle_Max, F87: Idle_Min

D. Classification Module

To evaluate the performance of our classification model, we employed six state-of-the-art algorithms. Each of these algorithms was selected for its distinct capabilities to handle complex data and deliver accurate results in various contexts. Using these supervised learning algorithms, we constructed detection models based on the features extracted from the KAPT24 dataset. This dataset, rich in information and meticulously annotated, served as the foundation for training our models. Through this training, the models have acquired the ability to effectively predict APT threats at each stage of their lifecycle. Specifically, our classification models were designed to identify and categorize malicious activities into six distinct categories: Normal (Stage 0), representing benign or normal activities that pose no threat; Reconnaissance (Stage 1), involving information-gathering activities where the attacker searches for vulnerable entry points; Initial Compromise (Stage 2), which marks the phase where the attacker successfully compromises the target system initially; Lateral Movement (Stage 3), referring to movement within the network, allowing the attacker to navigate and extend access to other systems; Data Exfiltration (Stage 4), where the attacker extracts sensitive information from the target network; and Post Exfiltration (Stage 5), encompassing post-exfiltration activities that typically include attempts to cover up traces of the attack or maintain access to the compromised system.

The integration of these algorithms into our classification module has enhanced the accuracy and reliability of threat detection. Each algorithm brings a unique approach to data analysis, capturing various nuances of malicious behaviors. For example, Random Forests and Support Vector Machines provide robust perspectives in terms of classification, while ensemble algorithms like XGBoost, LGBM, and CatBoost optimize performance through sophisticated aggregation methods. The Multi-Layer Perceptron leverages neural networks' capabilities to model complex relationships between features. Similarly, Koala *et al.* employed a comparable approach by using multiple machine learning algorithms to analyze application behavior through event traces in detecting security vulnerabilities [**?**]. The use of these cutting-edge algorithms has enabled the creation of a sophisticated and effective classification model, capable of detecting and categorizing APT threats across all phases of their lifecycle, ensuring proactive and robust defense against advanced cyber threats. Building on the proposed methodology, the next section will focus on the practical implementation and evaluation of this approach, demonstrating its effectiveness in real-world scenarios through comprehensive experimentation and analysis.

V. IMPLEMENTATION AND EVALUATION OF THE APPROACH

In this section, we transition from the theoretical foundation provided in the proposed methodology to the practical implementation of our approach. The main objective of implementing this approach is to demonstrate its real-world applicability, especially given the evolving and adaptive nature of DGAs. To this end, we developed a Flask application using Python, capable of capturing real-time network traffic, extracting key features, and analyzing them through a pre-trained classification model integrated into the system. For packet manipulation, we utilized the Scapy library, which allows for efficient packet sniffing and feature extraction. The system is designed to operate in real-time, predicting whether the sniffed packet is benign or malicious, thus offering an immediate response to potential threats.

Additionally, we integrated a graphical user interface (GUI) using Tkinter to streamline user interaction, making the system more accessible and user-friendly for practical deployment. This real-time capability is crucial for staying ahead of the constantly evolving DGA tactics and bolstering cybersecurity defenses.

In this section, we will thoroughly evaluate the relevance of the extracted features using performance indicators to ensure the system's predictions are both accurate and reliable. Finally, we will conclude by presenting the results of various performance metrics that validate the robustness of our approach, highlighting its capacity to detect APT threats in real-time environments.

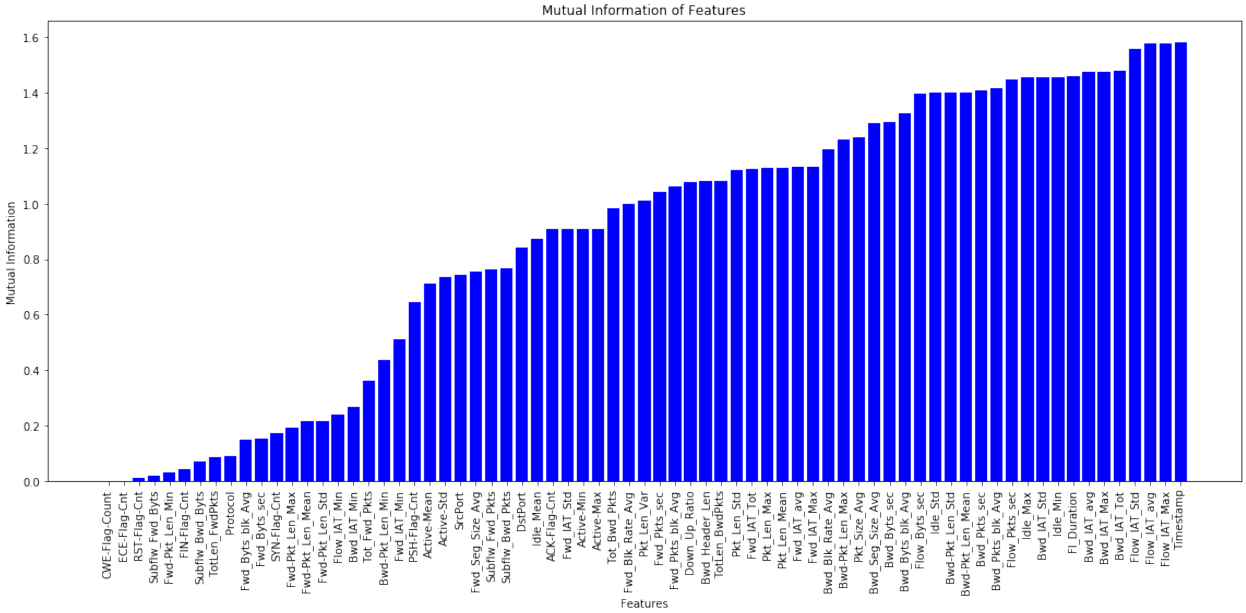


Fig. 3. Feature importance using mutual information.

A. Relevance of Features Analysis

Mutual information is a measure that captures nonlinear and independent relationships in data distribution, making it more flexible than methods like Pearson correlation or ANOVA. Unlike Pearson correlation, which only measures linear relationships, or chi-square tests, limited to categorical variables, mutual information is robust to monotonic transformations and useful for both continuous and categorical variables. It also outperforms model-based feature importance methods in machine learning by avoiding bias towards features with more levels. Fig. 3 presents the mutual importance of features in the KAPT 2024 dataset, used to address challenges posed by APT. The vertical bar chart displays features on the horizontal axis and their mutual importance values on the vertical axis. This visualization highlights that virtually all features have notable mutual importance, suggesting that the feature set is relevant for detecting APT threats.

In summary, we can conclude that the selected features contribute, to varying degrees, to the classification process of the models. This provides a solid starting point for training our models with supervised learning algorithms, whose very satisfactory results are detailed in the following subsection.

B. Results Analysis

In this subsection, we analyze the results obtained from applying various machine learning algorithms to our dataset. The confusion matrix is a crucial metric for evaluating classification models in machine learning, particularly in the context of multi-class classification problems [30]. It provides a detailed view of the model's performance by showing not only the number of correct predictions but also the types and quantities of errors made. From the confusion matrix, various performance metrics such as precision, recall, F1 score, and accuracy can be calculated for each phase individually, offering a more nuanced evaluation of the model's performance. It also helps identify if certain classes are systematically under-predicted or over-predicted, which is particularly useful in imbalanced datasets where some classes may dominate. By using ensemble algorithms such as XGBoost, CatBoost, and LightGBM, as well as algorithms like Random Forests, SVMs, and MLPs, the classification performances of these algorithms are compared and evaluated in a multi-class approach for each of the 6 stages of the APT lifecycle. To ensure the integrity of the evaluation, we followed a methodical approach by mixing samples from each class before splitting them into distinct training and test sets. The results of these evaluations are concisely summarized in Fig. 4, which represents the average metrics across all phases of the cycle [31].

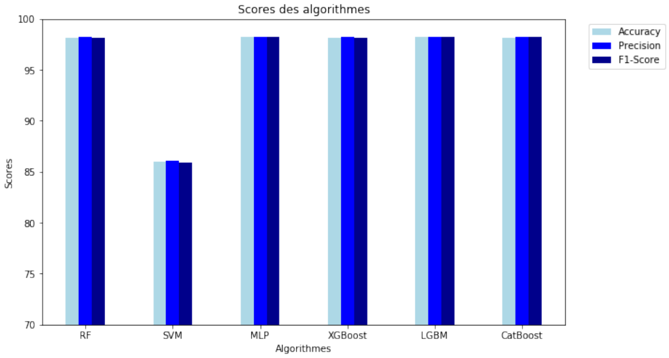
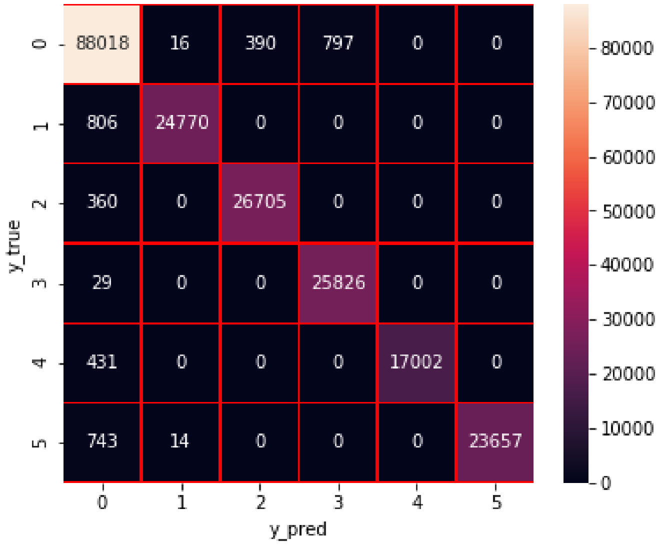


Fig. 4. Simulation results.

The graph shows that the MLP, LGBM, XGBoost, CatBoost and RF algorithms achieve very high and similar scores in terms of Accuracy, Precision, and F1-Score, indicating their effectiveness for this dataset.

Regarding the results from the confusion matrix analysis, let's focus on the MLP case, which offers very satisfactory



y_true \ y_pred	0	1	2	3	4	5
0	88018	16	390	797	0	0
1	806	24770	0	0	0	0
2	360	0	26705	0	0	0
3	29	0	0	25826	0	0
4	431	0	0	0	17002	0
5	743	14	0	0	0	23657

Fig. 5. Confusion matrix for MLP.

results. The confusion matrix for the MLP algorithm (Fig. 5) shows robust overall performance in classifying the different classes, with high numbers on the diagonal representing correct predictions. For example, class 0 is well predicted with 88,018 instances correctly classified, while classes 2 and 3 also show high classification rates, with 26,705 and 25,826 instances correctly classified, respectively. However, some classification errors, though minor, are present, as indicated by the off-diagonal values. False positives, such as the 797 instances of class 0 incorrectly classified as class 3, and false negatives, like the 806 instances of class 1 predicted as class 0, highlight the limitations of the algorithm.

The results obtained from our study have exceeded our expectations, confirming the effectiveness of our proposed solution. Our approach aims to provide a cost-effective and privacy-conscious method that is highly responsive to the evolving nature of APT threats targeting DNS. Initially, we identified 87 plaintext features that are directly accessible without the need for third-party equipment to decrypt data. Subsequently, we developed a lightweight application for capturing traffic, which demonstrated its capability to analyze network traffic in real-time and detect APT threats effectively. Furthermore, the integration of the MITRE framework has provided a comprehensive understanding of APT behaviors, enabling proactive threat detection. This makes our approach not only more efficient and effective but also lightweight and secure, addressing critical concerns in cybersecurity.

VI. Conclusion

The absence of significant features due to encryption in TLS, the limited exploitation of intelligence platforms in the search for proactive solutions, and the lack of training data covering all stages of the APT lifecycle underscore the importance of a balanced understanding of adversaries' behaviors, capabilities, and intentions for effective defense against APT threats targeting DNS. This study sought to develop a comprehensive approach for analyzing and detecting malicious domains throughout the entire APT lifecycle. The method had to meet several constraints: strict privacy compliance, lightweight and low operational cost, as well as optimal efficiency and responsiveness. To address this challenge, we developed a distinctive feature extraction module by analyzing TTPs and IoCs from APT threats using the MITRE ATT&CK matrix, thus contributing to the identification of features and data sources for building a dataset covering all phases of the APT lifecycle. Another major contribution of this study lies in the focus on detecting C&C servers and tools such as DGAs within the APT lifecycle.

Our experiments were conducted using six machine learning algorithms enabling a thorough evaluation of our approach's performance in a multi-class framework. This novel approach, which integrates intelligence platforms and importance indicators, has proven effective in detecting APTs throughout their entire lifecycle. The results obtained open promising perspectives for the continuous improvement of threat detection systems. Our future work will focus on establishing an intelligence platform aimed at sharing threat information within a trust circle for stakeholders with common challenges and strategies. Community sharing allows for alerting others about the occurrence of a probable attack and benefiting from feedback on how to counter a threat.

References

[1] A. R. Tapsoba, T. F. Ouédraogo, and W.-B. S. Zongo, "Analysis of Plaintext Features in DoH Traffic for DGA Domains Detection," in *Information Technology and Systems*, Á. Rocha, C. Ferrás, J. Hochstetter Diez, and M. Diéguez Rebolledo, Eds., Cham: Springer Nature Switzerland, 2024, pp. 127–138. doi: 10.1007/978-3-031-54235-0_12.

[2] A. Al Mamun, H. Al-Sahaf, I. Welch, et S. Camtepe, "Advanced Persistent Threat Detection: A Particle Swarm Optimization Approach", in *2022 32nd International Telecommunication Networks and Applications Conference (ITNAC)*, Wellington, New Zealand: IEEE, nov. 2022, pp. 1-8. doi: 10.1109/ITNAC55475.2022.9998358.

[3] N. Sun et al., "Cyber Threat Intelligence Mining for Proactive Cybersecurity Defense: A Survey and New Perspectives", *IEEE Commun. Surv. Tutor.*, vol. 25, no 3, pp. 1748-1774, 2023, doi: 10.1109/COMST.2023.3273282.

[4] C. Gan, J. Lin, D.-W. Huang, Q. Zhu, et L. Tian, "Advanced Persistent Threats and Their Defense Methods in Industrial Internet of Things: A Survey", *Mathematics*, vol. 11, no 14, p. 3115, juill. 2023, doi: 10.3390/math11143115.

[5] N. I. Che Mat, N. Jamil, Y. Yusoff, et M. L. Mat Kiah, "A systematic literature review on advanced persistent threat behaviors and its detection strategy", *J. Cybersecurity*, vol. 10, no 1, p. tyad023, janv. 2024, doi: 10.1093/cybsec/tyad023.

[6] G. Yan, Q. Li, D. Guo, et B. Li, "AULD: Large Scale Suspicious DNS Activities Detection via Unsupervised Learning in Advanced Persistent Threats", *Sensors*, vol. 19, no 14, p. 3180, juill. 2019, doi: 10.3390/s19143180.

[7] J. Al-Saraireh et A. Masarweh, "A novel approach for detecting advanced persistent threats", *Egypt. Inform. J.*, vol. 23, no 4, pp. 45-55, déc. 2022, doi: 10.1016/j.eij.2022.06.005.

[8] W. Ren et al., "APT Attack Detection Based on Graph Convolutional Neural Networks", *Int. J. Comput. Intell. Syst.*, vol. 16, no 1, p. 184, nov. 2023, doi: 10.1007/s44196-023-00369-5.

[9] H. N. Eke et A. Petrovski, "Advanced Persistent Threats Detection based on Deep Learning Approach", in *2023 IEEE 6th International Conference on Industrial Cyber-Physical Systems (ICPS)*, Wuhan, China: IEEE, mai 2023, pp. 1-10. doi: 10.1109/ICPS58381.2023.10128062.

[10] M. Miguez et B. Sassani (Sarrafpour), "Feature-based Systematic Analysis of Advanced Persistent Threats", *AI Comput. Sci. Robot. Technol.*, vol. 2, mai 2023, doi: 10.5772/acrt.21.

[11] A. Alshamrani, S. Myneni, A. Chowdhary, et D. Huang, “A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities”, *IEEE Commun. Surv. Tutor.*, vol. 21, no 2, pp. 1851-1877, 2019, doi: 10.1109/COMST.2019.2891891.

[12] A. Dutta et S. Kant, “An Overview of Cyber Threat Intelligence Platform and Role of Artificial Intelligence and Machine Learning”, in *Information Systems Security*, vol. 12553, S. Kanhere, V. T. Patil, S. Sural, et M. S. Gaur, Éd., Lecture Notes in Computer Science, vol. 12553. Cham: Springer International Publishing, 2020, pp. 81-86. doi: 10.1007/978-3-030-65610-2_5.

[13] S. Saeed, S. A. Suayyid, M. S. Al-Ghamdi, H. Al-Muhaisen, et A. M. Almuhaideb, “A Systematic Literature Review on Cyber Threat Intelligence for Organizational Cybersecurity Resilience”, *Sensors*, vol. 23, no 16, p. 7273, août 2023, doi: 10.3390/s23167273.

[14] S. Kumar, B. P. Singh, et V. Kumar, “A Semantic Machine Learning Algorithm for Cyber Threat Detection and Monitoring Security”, in 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India: IEEE, déc. 2021, p. 1963-1967. doi: 10.1109/ICAC3N53548.2021.9725596.

[15] Y. Zhou, Y. Tang, M. Yi, C. Xi, et H. Lu, “CTI View: APT Threat Intelligence Analysis System”, Secur. Commun. Netw., vol. 2022, p. 1-15, janv. 2022, doi: 10.1155/2022/9875199.

[16] D. Preuveneers et W. Joosen, “Sharing Machine Learning Models as Indicators of Compromise for Cyber Threat Intelligence”, J. Cybersecurity Priv., vol. 1, no 1, p. 140-163, févr. 2021, doi: 10.3390/jcp1010008.

[17] X. Zhang, X. Miao, et M. Xue, “A Reputation-Based Approach Using Consortium Blockchain for Cyber Threat Intelligence Sharing”, Secur. Commun. Netw., vol. 2022, p. 1-20, août 2022, doi: 10.1155/2022/7760509.

[18] A. Ramsdale, S. Shiaeles, et N. Kolokotronis, “A Comparative Analysis of Cyber-Threat Intelligence Sources, Formats and Languages”, Electronics, vol. 9, no 5, p. 824, mai 2020, doi: 10.3390/electronics9050824.

[19] T. D. Wagner, E. Palomar, K. Mahbub, et A. E. Abdallah, “A Novel Trust Taxonomy for Shared Cyber Threat Intelligence”, Secur. Commun. Netw., vol. 2018, p. 1-11, juin 2018, doi: 10.1155/2018/9634507.

[20] W. Tounsi et H. Rais, “A survey on technical threat intelligence in the age of sophisticated cyber attacks”, Comput. Secur., vol. 72, p. 212-233, janv. 2018, doi: 10.1016/j.cose.2017.09.001.

[21] M. Gschwandtner, L. Demetz, M. Gander, et R. Maier, “Integrating Threat Intelligence to Enhance an Organization’s Information Security Management”, in Proceedings of the 13th International Conference on Availability, Reliability and Security, Hamburg Germany: ACM, août 2018, p. 1-8. doi: 10.1145/3230833.3232797.

[22] A. R. Tapsoba et T. Frederic Ouedraogo, “Evaluation of supervised learning algorithms in binary and multi-class network anomalies detection”, in 2021 IEEE AFRICON, Arusha, Tanzania, United Republic of: IEEE, sept. 2021, p. 1-6. doi: 10.1109/AFRICON51333.2021.9570886.

[23] I. Sharafaldin, A. Habibi Lashkari, et A. A. Ghorbani, “Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization”:, in Proceedings of the 4th International Conference on Information Systems Security and Privacy, Funchal, Madeira, Portugal: SCITEPRESS - Science and Technology Publications, 2018, p. 108-116. doi: 10.5220/0006639801080116.

[24] N. Moustafa et J. Slay, “UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)”, in 2015 Military Communications and Information Systems Conference (MilCIS), nov. 2015, p. 1-6. doi: 10.1109/MilCIS.2015.7348942.

[25] A. R. Tapsoba, T. F. Ouédraogo, M. B. Diallo, et W.-B. S. Zongo, “Toward Real Time DGA Domains Detection in Encrypted Traffic”, in *Proceedings of the 7th International Conference on Networking, Intelligent Systems and Security*, in NISS ’24. New York, NY, USA: Association for Computing Machinery, August 2024, pp. 1-8. doi: 10.1145/3659677.3659684.

[26] A. R. Tapsoba, T. F. Ouédraogo, et A. E. Ouédraogo, “Relevance of the Gaussian classification on the Detection of DDoS Attacks”, in 2022 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Suzhou, China: IEEE, oct. 2022, p. 42-49. doi: 10.1109/CyberC55534.2022.00018.

[27] R. Battiti, “Using Mutual Information for Selecting Features in Supervised Neural Net Learning”, Neural Netw. IEEE Trans. On, vol. 5, p. 537-550, août 1994, doi: 10.1109/72.298224.

[28] H. Liu, L. Liu, et H. Zhang, “Feature Selection Using Mutual Information: An Experimental Study”, in PRICAI 2008: Trends in Artificial Intelligence, T.-B. Ho et Z.-H. Zhou, Éd., Berlin, Heidelberg: Springer, 2008, p. 235-246. doi: 10.1007/978-3-540-89197-0_24.

[29] Gouayon Koala, Didier Bassolé, Telesphore Tiendrebeogo, and Oumarou Sié. “Software Vulnerabilities’ Detection by Analysing Application Execution Traces.” *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 6, January 2023. doi:10.14569/IJACSA.2023.01406136. Licensed under CC BY 4.0.

[30] M. Heydarian, T. E. Doyle, et R. Samavi, “MLCM: Multi-Label Confusion Matrix”, IEEE Access, vol. 10, p. 19083-19095, 2022, doi: 10.1109/ACCESS.2022.3151048.

[31] H. Suryotrisongko, Y. Musashi, A. Tsuneda, et K. Sugitani, “Robust Botnet DGA Detection: Blending XAI and OSINT for Cyber Threat Intelligence Sharing”, IEEE Access, vol. 10, p. 34613-34624, 2022, doi: 10.1109/ACCESS.2022.3162588.