A Project report on

DARK-TRACER Malware Detection Using Spatiotemporal Patterns

A Dissertation submitted to JNTU Hyderabad in partial fulfillment of the academic requirements for the award of the degree.

Bachelor of Technology

in

Computer Science and Engineering

Submitted by

JANANI CHALAPATI
(20H51A05E2)
V.VENKATA SAI NATHA REDDY
(20H51A05M3)
MOHAMMED AWAIS KHAN
(20H51A05A0)

Under the esteemed guidance of
Ms. A. Mounika Rajeswari
(Associate Professor)

Department of Computer Science and Engineering

CMR COLLEGE OF ENGINEERING & TECHNOLOGY
(UGC Autonomous)
*Approved by AICTE *Affiliated to JNTUH *NAAC Accredited with A^+ Grade
KANDLAKOYA, MEDCHAL ROAD, HYDERABAD - 501401.

2020- 2024

CMR COLLEGE OF ENGINEERING & TECHNOLOGY

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD – 501401

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Major Project Phase I report entitled "**DARK-TRACER Malware Detection Using Spatiotemporal Patterns"** being submitted by Janani Chalapati (20H51A05E2), V. Venkata Sai Natha Reddy (20H51A05M3), Mohammed Awais Khan (20H51A05A0) in partial fulfillment for the award of **Bachelor of Technology in Computer Science and Engineering** is a record of Bonafide work carried out his/her under my guidance and supervision.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree.

Ms. A. Mounika Rajeswari
Associate Professor
Dept. of CSE

Dr. Siva Skandha Sanagala
Associate Professor and HOD
Dept. of CSE

ACKNOWLEDGEMENT

With great pleasure we want to take this opportunity to express my heartfelt gratitude to all the people who helped in making this project a grand success.

We are grateful to **Ms. A. Mounika Rajeswari,** Associate Professor, Department of Computer Science and Engineering for his valuable technical suggestions and guidance during the execution of this project work.

We would like to thank **Dr. Siva Skandha Sanagala,** Head of the Department of Computer Science and Engineering, CMR College of Engineering and Technology, who is the major driving forces to complete my project work successfully.

We are very grateful to **Dr. Vijaya Kumar Koppula**, Dean-Academics, CMR College of Engineering and Technology, for his constant support and motivation in carrying out the project work successfully.

We are highly indebted to **Major Dr. V A Narayana,** Principal, CMR College of Engineering and Technology, for giving permission to carry out this project in a successful and fruitful way.

We would like to thank the **Teaching & Non- teaching** staff of Department of Computer Science and Engineering for their co-operation

We express our sincere thanks to **Shri. Ch. Gopal Reddy**, Secretary, CMR Group of Institutions, for his continuous care.

Finally, we extend thanks to our parents who stood behind us at different stages of this Project. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project work.

Janani Chalapati	20H51A05E2
V.Venkata Sai Natha Reddy	20H51A05M3
Mohammed Awais Khan	20H51A05A0

TABLE OF CONTENTS

List of Figures

List of Tables

FIGURE NO.	TITLE	PAGE NO.

ABSTRACT

In an era of escalating global cyber threats, the imperative to discern and counteract malicious cyber activities has grown ever more pressing. The darknet, an uncharted realm within the IP address space, emerges as a promising frontier for the investigation and analysis of indiscriminate cyberattacks. Within this obscure domain, where legitimate communication is virtually nonexistent, we can discern peculiar patterns in the spatiotemporal flow of data that often signify the presence of malware-driven scanning operations. This research focuses on the early detection of these nefarious activities by homing in on anomalies within the spatiotemporal patterns manifest in darknet traffic data.

Our study builds upon previous efforts which have proposed algorithms designed to automatically estimate and pinpoint irregular spatiotemporal patterns in real-time darknet traffic. These algorithms are underpinned by three distinct machine learning techniques. In this endeavor, we have amalgamated these previously proposed methods into a singular framework, herein referred to as Dark-TRACER.

CHAPTER 1
INTRODUCTION

CHAPTER 1
INTRODUCTION

1.1. Problem Statement

In recent years, the Internet has witnessed a surge in indiscriminate cyberattacks, resulting in a growing challenge of analyzing these threats. Safeguarding the Internet's security necessitates swift identification of global cyberattack trends, their causes, the formulation of countermeasures, and worldwide threat disclosure. To achieve this, early detection of indiscriminate scanning attacks, often initiated by malware, before they escalate into pandemics is crucial.

1.2. Research Objective

Distinguishing malware scanning attacks amidst the vast volume of benign network traffic poses a formidable obstacle. In darknets, legitimate communication is absent, making indiscriminate scanning more conspicuous, resulting in a high signal-to-noise ratio. darknet traffic volume is exponentially increasing annually. Additionally, many communications possess unknown intent, including independent cyberattacks, benign scanning activities, and misconfigured communications. Our research focus centers on precisely differentiating these noisy communications from malicious attack activities

1.3. Project Scope & Limitations

The project's scope revolves around the development and evaluation of a cybersecurity tool named Dark-TRACER, designed to detect and analyze anomalous synchronization of spatiotemporal patterns within darknet traffic data. Considering the increasing global prevalence of cyberattacks, the tool aims to facilitate early detection of malware activities, offering a proactive approach to mitigate cybersecurity threats.

1.3.1. Scope

1. **Dark-TRACER Framework Development**: The project involves the integration of previously proposed algorithms into a single framework, Dark-TRACER. This framework combines three independent machine learning methods to automatically estimate and detect anomalies in darknet traffic patterns.

2. **Real-Time Monitoring**: The system is designed for real-time monitoring of darknet traffic data. The project emphasizes the analysis of large-scale darknet sensors, encompassing up to /17 subnet scales.

3. **Early Malware Detection**: Dark-TRACER aims to identify malware activities at an early stage, providing an advantage over traditional methods. The project evaluates the system's capability to detect malware activities before they are publicly disclosed by third-party security research organizations.

.

4. **Human Resource Evaluation**: The project incorporates an analysis of the human resources required to operate the Dark-TRACER framework daily. It quantifies the time and effort needed by two analysts to manage the system efficiently

1.3.2. Limitations

1. **Data Period**: The project is limited to the analysis of darknet traffic data observed between October 2018 and October 2020. Consequently, it may not address emerging threats or changing attack patterns beyond this timeframe.

2. **Counter Measures**: The project aims only at identify the malware at an early stage but does not encompass the counter measures necessary to eradicate the malware from a computer

3. **Algorithm Reliability**: The effectiveness of Dark-TRACER is based on the reliability and accuracy of the machine learning algorithms integrated into the framework. Limitations may arise if the algorithms' performance is inconsistent.

4. **Human Resource Scalability**: The evaluation of human resource requirements is based on a specific scenario and may not account for variations in workloads, expertise levels, or the complexity of detected threats.

CHAPTER 2
BACKGROUND WORK

CHAPTER 2
BACKGROUND WORK

2.1 Signature-based Detection

2.1.1 Introduction

Signature-based detection is one of the most traditional methods of identifying malware. It involves comparing the characteristics and patterns of files or network traffic against a database of known malware signatures. When a file or network activity matches a known signature, it is flagged as malicious. The main limitation of this approach is its inability to detect new or previously unseen malware, as it relies on known signatures.

2.1.2 Merits, Demerits and Challenges

Merits:

- **High Accuracy:** It is very effective in identifying known malware with well-defined signatures.
- **Low False Positives:** Signature-based detection tends to generate fewer false alarms compared to some other methods.
- **Low Overhead:** It imposes minimal system resource overhead, making it suitable for real-time scanning.

Demerits:

- **Ineffectiveness Against New Malware:** It cannot detect new or zero-day malware since it relies on predefined signatures.

- **Constant Updates:** Signature databases must be continually updated to remain effective.
- **Polymorphic and Encrypted Malware:** It struggles to detect polymorphic malware or malware that employs encryption to evade signature detection.

Challenges:

- **Zero-Day Threats:** The inability to detect previously unseen threats is a significant challenge.
- **Signature Management:** Managing a large and frequently updated database of signatures can be cumbersome.
- **Evasion Techniques:** Malware authors can employ various evasion techniques to bypass signature-based detection.

2.1.3 Implementation of Signature-based Detection

1. **Signature Creation:** Security experts or vendors create signatures for known malware. These signatures can be based on file hashes, specific patterns within files, or network traffic characteristics. The signatures are typically stored in a database.
2. **Scanning Files or Network Traffic:** Antivirus or intrusion detection systems scan files or network traffic in real-time or periodically. During the scan, the system checks for matches between the files or network traffic and the signatures in the database.
3. **Alert or Quarantine:** If a match is found, the system generates an alert or acts according to predefined rules. This can include quarantining or deleting the infected file.
4. **Signature Updates:** Regularly update the signature database to keep it current with the latest malware threats.

2.2 Heuristic Analysis

2.2.1 Introduction

Heuristic analysis involves identifying potentially malicious behavior based on predefined rules and heuristics. Instead of relying on specific signatures, this approach looks for suspicious activities or characteristics that might indicate malware. For example, it can detect unusual file behaviors, suspicious system calls, or unexpected network traffic patterns. While heuristics can be effective in identifying novel malware, they may also

2.2.2 Merits, Demerits, and Challenges

Merits:

- **Better at Detecting New Malware:** Heuristic analysis can identify malware with unknown or evolving characteristics.
- **Adaptive:** It can adapt to some extent as it uses behavioral patterns and heuristics.
- **Reduced False Negatives:** It may have fewer false negatives compared to signature-based methods.

Demerits:

- **Moderate False Positives:** Heuristic analysis can produce more false positives due to its reliance on behavioral patterns.
- **Limited Accuracy:** The effectiveness of heuristic methods depends on the quality of heuristics used.
- **Resource-Intensive:** Analyzing behavior can be computationally intensive, potentially impacting system performance.

Challenges:

- **Tuning Heuristics:** Determining the right set of heuristics and fine-tuning them can be a challenge.
- **Balancing Accuracy and False Positives:** Striking the right balance between accuracy and false positives can be difficult.
- **Advanced Malware Evasion:** Some advanced malware can obfuscate their behavior to evade heuristic detection.

2.2.3 Implementation of Heuristic Analysis

1. **Rule Development:** Security experts define rules and heuristics based on known malware characteristics and behaviors. Rules can cover activities like suspicious file behavior, unauthorized system changes, or unusual network traffic.

2. **Scanning and Monitoring:** Systems continually monitor files, processes, and network traffic for deviations from normal behavior. When a deviation is detected, the system evaluates it against the predefined rules.

3. **Alert or Quarantine:** If a behavior is deemed suspicious, the system generates an alert or acts, such as quarantining the suspicious element.

4. **Rule Updates:** Regularly update and refine the rules and heuristics to improve detection accuracy.

2.3 Behavior-based Analysis

2.3.1 Introduction

Behavior-based analysis focuses on monitoring the behavior of software and identifying any actions that deviate from normal, legitimate behavior. This approach is effective at detecting previously unknown malware, as it doesn't rely on static signatures or heuristics. Sandbox environments are often used to execute and observe the behavior of suspicious files or applications in a controlled setting. If the behavior is deemed malicious, the file or application is flagged as malware. Behavioral analysis can be resource-intensive and may produce false positives, but it is a valuable technique in modern threat detection.

2.3.2 Merits, Demerits, and Challenges

Merits:

- **Effective Against Zero-Day Threats:** It is highly effective at identifying previously unknown or zero-day malware.
- **Adaptive:** It can adapt to new threats by focusing on behavior rather than static characteristics.
- **Comprehensive Insight:** Provides a deep understanding of malware actions, aiding in threat analysis.

Demerits:

- **Higher False Positives:** Behavioral analysis can result in more false positives due to the complexity of legitimate software behavior.

- **Resource-Intensive:** Running applications in a sandbox or monitoring their behavior in real-time can consume system resources.
- **Complexity:** Setting up and managing a behavioral analysis system can be complex and costly.

Challenges:

- **Determining Malicious Intent:** Distinguishing malicious behavior from benign actions can be challenging.
- **Evasion Techniques:** Malware authors are constantly developing evasion techniques to avoid detection.
- **Scalability:** Managing and analyzing a large number of applications and behaviors can be a scalability challenge.

2.3.3 Implementation of Behavior-based Analysis

1. **Sandbox Environment:** Malware is executed in a controlled environment, commonly referred to as a sandbox, to observe its behavior without affecting the host system.

2. **Monitoring Behavior:** The system monitors and records various aspects of the malware's behavior, such as file system interactions, registry changes, network connections, and system calls.

3. **Analysis and Decision:** The behavior data is analyzed to determine if the observed actions are consistent with legitimate software or indicative of malicious behavior. Anomalies or malicious activities trigger alerts or further analysis.

4. **Alert Generation:** If the behavior analysis indicates malicious intent, an alert is generated, and appropriate action is taken, such as quarantining the malware.

5. **Machine Learning and AI:** Machine learning and artificial intelligence techniques are often used to improve the accuracy of behavior-based analysis by learning from historical data and patterns.

6. **Scalability and Resource Management:** Ensure that the system can handle the resource-intensive process of monitoring behavior at a scale while minimizing impact on the host system.

2.4 Proposed Framework

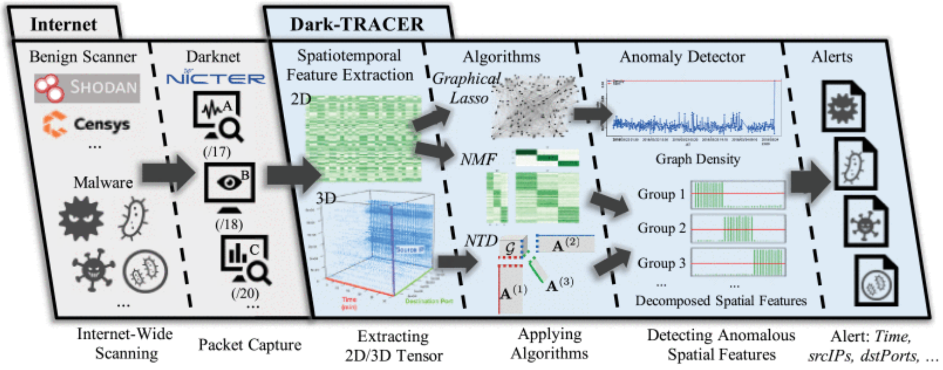

Figure 2.4.1 Illustration of the framework of *Dark-TRACER*

The overarching structure of Dark-TRACER is visually represented in Figure 1. This system employs three distinct algorithms, namely Graphical Lasso, Non-negative Matrix Factorization (NMF), and Non-negative Tensor Decomposition (NTD), to estimate the synchronicity of spatiotemporal features. The modules that integrate these algorithms are denoted as Dark-GLASSO, Dark-NMF, and Dark-NTD, respectively, to distinguish their unique functions.

By concentrating on synchronicity, Dark-TRACER offers several key advantages when compared to prevailing methods for detecting malware activity:

1. Mitigation of Benign Noise: Focusing on synchronicity enables us to diminish the impact of benign noise within darknet traffic. This, in turn, accentuates and isolates malicious communication, allowing for more effective identification.

2. Enhanced Detection Capabilities: The approach enhances our ability to detect malware activities that are traditionally challenging to trace through manual procedures. This includes small-scale, meticulously coordinated threats or those lacking conspicuous, explicit spikes in activity. Dark-TRACER can identify such anomalies in synchronized spatial features, thereby capturing these threats before they escalate into widespread malware infections.

3. Early Detection of Emerging Threats: Dark-TRACER excels in identifying malware activities that synchronize with other malicious operations during the initial stages of infection, when their scale is limited. This early detection capability provides a valuable opportunity to counteract and mitigate malware threats before they gain significant momentum.

CHAPTER 3

RESULTS AND DISCUSSION

CHAPTER 3
RESULTS AND DISCUSSION

3.1 **Performance Metrics**

In this section, we provide a comprehensive discussion and insight into the performance of our framework. First, we demonstrate the advantages of Dark-TRACER and provide a comprehensive comparison of each proposed module. Then, we discuss the potential concerns of our approach, such as adversarial attacks and the reduction of false-positive alerts. Finally, we present guidelines for the practical application of Dark-TRACER.

Advantages of Dark-TRACER

As mentioned in the introduction, by focusing on the synchronization of spatiotemporal patterns in darknet traffic, we have the following advantages:

1. **Trimming Unsynchronized and Noisy Communications**: Distinguishing between non-attack-related and attack-related communications from darknet traffic is a difficult task. In this paper, we focused on the fact that hosts infected with similar malware tend to compromise and scan in a synchronized spatiotemporal pattern. By estimating the synchronicity of spatiotemporal patterns in darknet traffic and eliminating unsynchronized communications from the scope of analysis, noisy communications are expected to be filtered out, and malicious communications can be highlighted.

2. **Detecting Malware Activities that are Conventionally Difficult to Detect**: Traditionally, malware activities have been detected based on changes in time-series data, such as the number of packets and the number of hosts. However, recent malware activities have become more diverse and sophisticated, making them hard to detect manually. Dark-TRACER can detect such traditionally hard-to-detect malware activities.
3. **Early Detection of Malware Activities in Real-Time**: Dark-TRACER is capable of detecting malware activities early and in real time. Even when the scale of malware activity is small, it can capture the signs of infection before it spreads widely. This allows for the early detection of orchestrated threats.

Comprehensive Comparison of Proposed Modules

In this section, we compare the proposed modules Dark-GLASSO, Dark-NMF, and Dark-NTD in terms of accuracy, cost, anomaly detection method, and spatial features.

1. **Accuracy**: Each module has its strengths and weaknesses. Dark-GLASSO has a high precision rate, making it practical for global malware activity analysis. Dark-NMF and Dark-NTD are effective in detecting local malware activities early. Combining these modules in Dark-TRACER provides a well-rounded approach to detection.
2. **Cost**: Dark-NMF is computationally inexpensive and does not require much preprocessing. Dark-GLASSO and Dark-NTD are computationally more expensive and require specific preprocessing. The cost of analysis depends on the specific module used.

3. **Anomaly Detection**: Dark-NMF and Dark-NTD decompose spatiotemporal features into latent frequent patterns and perform anomaly detection for each group of decomposed spatial features. Dark-GLASSO detects anomalies from all spatial features without decomposition. This influences the number of alerts generated.
4. **Spatial Features**: The choice of spatial features depends on the module. Dark-GLASSO handles host space, while Dark-NMF can handle both host and port space. Dark-NTD handles a three-dimensional spatiotemporal feature tensor from the beginning.

Considerations for Adversarial Attacks

We discuss possible adversarial attacks and how Dark-TRACER can respond to them. These attacks include attaching dummy scans, orchestrating multiple attacks, and slow stealth scan attacks. Dark-TRACER's ability to capture anomalous synchronizations makes it effective in detecting attacks even in the presence of these evasion techniques.

Reduction of False-Positive Alerts

False-positive alerts are primarily caused by synchronized scanning for investigative purposes. To address this issue, Dark-TRACER can distinguish between alerts caused by investigative scanners and those caused by malware activities. A simple rule is applied to reduce false positives, and further work is planned to develop a model for classifying or clustering scanners for investigative purposes.

CHAPTER 4
CONCLUSION

CHAPTER 4
CONCLUSION

In this study, we have addressed three distinct machine learning methods designed to automatically gauge the synchronization of spatiotemporal patterns within darknet traffic in real-time and identify anomalies. These methods are known as Dark-GLASSO, Dark-NMF, and Dark-NTD. Furthermore, we have put forth the concept of Dark-TRACER, an integrated framework that unifies all three methods into a single, comprehensive solution.

Our findings reveal that Dark-TRACER effectively addresses the limitations of each individual module, resulting in a remarkable recall rate for the detection of malware activities. Notably, it can detect these malware threats before they become publicly known through respected third-party security research organizations. Moreover, our study demonstrated that the daily operations required for threat detection could be executed by two analysts in reasonable time.

Currently, one of our primary challenges lies in the substantial number of false-positive alerts generated. However, our research indicates that even a straightforward rule-based approach can significantly mitigate false-positive alerts. Our forthcoming efforts will concentrate on reducing false positives by identifying the distinct markers of investigative scanners and constructing a model to track them. This reduction in false positives not only enhances the accuracy of threat detection but also reduces the associated analysis costs.

Furthermore, we are poised to automate the secondary collision analysis, which will provide valuable insights into the root causes and specific details behind the alerts identified by Dark-TRACER.

Ultimately, our vision is to deploy Dark-TRACER in real-world settings, allowing us to promptly and autonomously detect threats and malware activities in real time. This initiative will significantly bolster our cybersecurity defenses and support rapid response measures.

In conclusion, Dark-TRACER provides a robust framework for early detection of malware activities in darknet traffic, and its practical application can enhance cybersecurity efforts. By leveraging its various modules and conducting secondary analyses, organizations can effectively identify and respond to malware threats.

REFERENCES

REFERENCES

[1]. G. Gu, J. Zhang, and W. Lee, ‘‘BotSniffer: Detecting botnet command and control channels in network traffic,’’ in Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS), 2008, pp. 1–19.

[2]. M. Bailey, E. Cooke, F. Jahanian, A. Myrick, and S. Sinha, ‘‘Practical darknet measurement,’’ in Proc. 40th Annu. Conf. Inf. Sci. Syst., Mar. 2006, pp. 1496–1501.

[3]. J. Friedman, T. Hastie, and R. Tibshirani, ‘‘Sparse inverse covariance estimation with the graphical lasso,’’ Biostatistics, vol. 9, no. 3, pp. 432–441, Dec. 2007.

[4]. D. Lee and H. S. Seung, ‘‘Algorithms for non-negative matrix factorization,’’ in Proc. 13th Int. Conf. Neural Inf. Process. Syst. (NIPS), 2000, pp. 535–541.

[5]. Y.-D. Kim and S. Choi, ‘‘Nonnegative tucker decomposition,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2007, pp. 1–8.

[6]. C. Han, J. Shimamura, T. Takahashi, D. Inoue, M. Kawakita, J. Takeuchi, and K. Nakao, ‘‘Real-time detection of malware activities by analyzing darknet traffic using graphical lasso,’’ in Proc. 18th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun. (TrustCom), Aug. 2019, pp. 144–151.

[7]. C. Han, J. Shimamura, T. Takahashi, D. Inoue, J. Takeuchi, and K. Nakao, ‘‘Real-time detection of global cyberthreat based on darknet by estimating anomalous synchronization using graphical lasso,’’ IEICE Trans. Inf. Syst., vol. 103, no. 10, pp. 2113–2124, Oct. 2020.

[8]. C. Han, J. Takeuchi, T. Takahashi, and D. Inoue, ‘‘Automated detection of malware activities using nonnegative matrix factorization,’’ in Proc. IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun. (TrustCom), Oct. 2021.

[9]. H. Kanehara, Y. Murakami, J. Shimamura, T. Takahashi, D. Inoue, and N. Murata, ‘‘Real-time botnet detection using nonnegative tucker decomposition,’’ in Proc. 34th ACM/SIGAPP Symp. Appl. Comput., Apr. 2019, pp. 1337–1344.

GitHub Link

1.