Network Monitoring using Traffic Dispersion Graphs (TDGs)

Aditya Mohan Gupta
Computer Science
UCR
Riverside, CA, USA
agupt166@ucr.edu

Ankit Laxmikant Yadav
Computer Science
UCR
Riverside, CA, USA
ayada023@ucr.edu

Pournima Shinde
Computer Science
UCR
Riverside, CA, USA
pshin024@ucr.edu

Pragati Mahesh Sukhija
Computer Science
UCR
Riverside, CA, USA
psukh002@ucr.edu

People Contributions :

Info				Reading	Bonus	Effort			Attendance		
Name	**SID**	**Email**	**Cell***	**Book**	**Papers**	**Total**	**Dev**	**Comm/ Writing**	**In person**	**Zoom in real time**	**Recording**
Aditya Gupta	**862467699**	**agupt166@ucr.edu**	**9515361758**	**85**	**100**	**25**	**25**	**25**	**15**	**5**	**1**
Ankit Yadav	**862466907**	**ayada023@ucr.edu**	**9515489083**	**85**	**100**	**25**	**25**	**25**	**15**	**5**	**1**
Pournima Shinde	**862468394**	**pshin024@ucr.edu**	**9515368448**	**85**	**100**	**25**	**25**	**25**	**15**	**5**	**1**
Pragati Sukhija	**862468396**	**psukh002@ucr.edu**	**9514059549**	**85**	**100**	**25**	**25**	**25**	**15**	**5**	**1**

Aditya took charge of collecting custom data for video conferencing scenarios and finding online datasets for DDoS attacks, alongside the crucial task of data cleaning. He was responsible for writing the code that built TDGs for video conferencing and DDoS attacks. In the report, Aditya wrote the cover page, introduction, problem definition, and solution sections.

Ankit also contributed to custom data collection for DDoS attacks and gaming sessions. He coded the TDG construction for these two scenarios. For the report, Ankit prepared the cover page, abstract, results, and discussion sections.

Pournima's role included gathering custom data for video streaming and finding DDoS datasets online. She was involved in data cleaning and wrote the code to create TDGs for video streaming. In the report, she contributed to the introduction, solution description, results, and conclusion.

Pragati focused on collecting custom data for the Port Scanning attack and developed the foundational code to build TDGs for all scenarios. She wrote the specific code for Port Scanning attacks and contributed to the introduction, results, discussion, and conclusion in the report.

Presentation Slide: Link

1. ABSTRACT

We leverage Traffic Dispersion Graphs (TDGs) to analyze network traffic under various scenarios such as gaming sessions, live streaming, Zoom calls and identify potential security threats like DDoS and port scanning attacks, while also employing TDG to investigate the flow of different protocols. By utilizing public datasets as well as proprietary data collected via Wireshark, our Python-based tool, built within a Jupyter Notebook environment, offers users the capability to generate, visualize, and interpret TDGs. Employing libraries like NetworkX for graph construction and Matplotlib for visualization, coupled with Pandas for data manipulation, our efficient solution encompasses approximately 400 lines of code. The main challenge tackled was the extraction of relevant packet data for specific use cases, which is pivotal in discerning meaningful patterns within the TDGs. This multifaceted perspective not only provides actionable insights to strengthen network defenses but also enhances the monitoring of typical user activities for optimized performance and security.

2. INTRODUCTION

In the ever-expanding digital landscape, the analysis of network traffic stands as a crucial endeavor, offering insights into system performance, anomalies, and potential threats. The overall aim of this report is to delve into the realm of network traffic analysis, with a primary focus on utilizing Traffic Dispersion Graphs (TDGs) as a powerful tool for visualization and understanding. This introduction sets the stage by providing a concise overview of the problem motivation, the significance of our work, and the main contributions we offer to the field.

2.1 The Problem

By collecting traffic data and transforming it into Traffic Dependency Graphs (TDGs) for the purpose of analyzing traffic flows in specific scenarios such as Zoom calls, gaming, live streaming, as well as identifying attacks like DDoS and port scanning and traffic flow across various protocols, we intend to assess the effectiveness of TDGs as a holistic instrument for deciphering and illustrating network traffic information. Identifying and understanding complex network interactions and threats in various network scenarios like online gaming, video conferencing, and streaming services.

Input: Network traffic data, collected both from public datasets and private data using tools like Wireshark for specific scenarios.

Output: Traffic Dispersion Graphs (TDGs) that visualize network interactions, with patterns that can indicate normal or anomalous behavior.

Assumptions: The traffic data collected is sufficient to represent typical network behavior for the scenarios in question. The patterns discerned from the TDGs accurately reflect the underlying network events, and the anomalies detected correlate with potential security threats or network issues.

2.2 Previous Work

In network security and analysis, Traffic Dispersion Graphs (TDGs) have been utilized to understand complex network interactions[1][2]. Previous work in this area has introduced the use of TDGs, applying graph theory to visualize and analyze network traffic behaviors, and implementing measures such as degree distribution, maximum degree, and dK-2 distance for a more nuanced approach[2].

In the context of network monitoring, prior research has set the groundwork by defining TDGs with edges that represent network interactions. This work has been pivotal in moving beyond traditional packet content or flow statistics methods, thus revolutionizing network traffic analysis by focusing on host interactions. By implementing advanced filtering techniques for graph refinement, these studies have demonstrated the potential of TDGs to reveal distinct patterns for a variety of network applications and behaviors.

2.3 Our Contribution

In our study, we harnessed the capabilities of Traffic Dispersion Graphs (TDG) and their intrinsic properties to map out and visualize different network activities. We captured network traffic through Wireshark and generated custom datasets tailored to our specific case study. To enrich our analysis and expand our viewpoint, we integrated publicly accessible datasets with our proprietary data, employing this comprehensive visualization to scrutinize various network phenomena. Our analysis encompassed a diverse range of scenarios, including DDoS attacks, port scanning attack, gaming sessions, Zoom calls, and traffic flow across different protocols, leveraging TDGs as a pivotal tool in our exploration.

3. Background and Problem Definition Model

In this section, we provide the foundational background necessary to understand the scope and context of our work, along with a clear definition of the problem we aim to address. We also outline the model and framework that underpin our approach to network traffic analysis.

3.1 Definitions, Scenario, Model

Network traffic analysis investigates data flows to improve system performance, spot irregularities, and identify security threats. We use TDGs to visualize and make sense of network actions, with nodes and edges illustrating the traffic patterns and connections within the network.

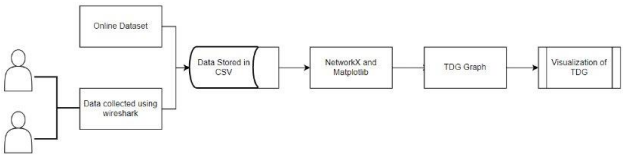

Fig: Our project model

Data is gathered from two sources: an online dataset[8][6] and direct capture using Wireshark. Once collected, the data is stored in a CSV (Comma-Separated Values) format. Following storage, the data is processed using Python for studying graphs and networks, alongside Matplotlib, a plotting library for creating static, interactive, and animated visualizations in Python. The processed data is used to construct a Traffic Dependency Graph (TDG), which is subsequently visualized to aid in the analysis of network traffic.

3.2 Weaknesses of Previous Work

Previous studies have utilized Traffic Dispersion Graphs (TDGs) mainly for network traffic analysis[1][2], focusing on specific aspects or using traditional methods. Our research builds on this by applying TDGs to unique datasets, like those from Zoom calls and YouTube, aiming to make the findings more relatable to readers. We offer a thorough approach to network traffic analysis, exploiting TDGs' unique strengths to gain a full understanding of network behaviors and patterns. Our work includes identifying port scan activities, a feature not covered in earlier studies, and examining all network protocol types. This broadens the scope from merely detecting anomalies in DDoS attacks[7][4] to offering a wider perspective on network security threats.

3.4 Assumptions and Limitations

Our work operates under certain assumptions and limitations. While we strive to provide a comprehensive framework for network traffic analysis, our approach may have inherent limitations in terms of scalability and real-time analysis capabilities. However, we believe that these limitations do not diminish the significance of our contributions but rather serve as opportunities for future research and refinement.

4 Idea/Approach

To facilitate our analysis, we have used information, definitions and concepts from previous studies where necessary, ensuring clarity and precision in our approach. TDGs are visual representations of network traffic flows, with nodes representing network entities and edges indicating traffic flow between them. By utilizing Python for coding, along with the NetworkX library for TDG generation, and the support of Matplotlib for visualization, we have developed a robust framework for network traffic analysis.

Our approach works by first collecting and filtering diverse datasets, including custom datasets from sources like Zoom calls and Gaming, and publicly available datasets. We then preprocess the data and generate TDGs, capturing network interactions and traffic patterns. Fine points and clarifications in our methodology include considerations for data preprocessing techniques, TDG generation algorithms, and visualization strategies to effectively communicate our findings.

Overall, our effort has culminated in a framework for network traffic analysis, leveraging the unique capabilities of TDGs to offer valuable insights into network behaviors and patterns. Through our approach and meticulous attention to detail, we aim to contribute to the advancement of the field and provide practical solutions for enhancing network security measures and system performance.

5 Experimental Results

5.1 DDoS Attack Traffic Analysis

We developed the TDG using the 2011 VAST Mini Challenge's public dataset alongside our own custom dataset. Our analysis revealed striking similarities between the two TDGs. Both exhibited a high degree of consistency in the source IP addresses and an elevated level of TCP

traffic. Additionally, the TDGs displayed a star-like structure, with potential victims situated at the core and central nodes characterized by the highest in-degrees. This pattern could serve as a hallmark for identifying DDoS attacks. Please note, a directed edge signifies the transfer of packets from a source node to a destination node.

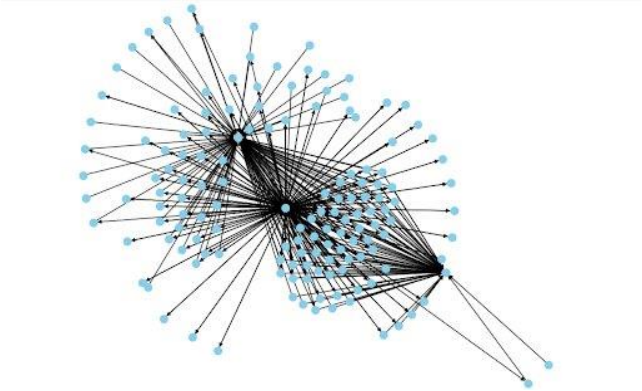

Fig: TDG before DDoS attack on 2011 VAST data

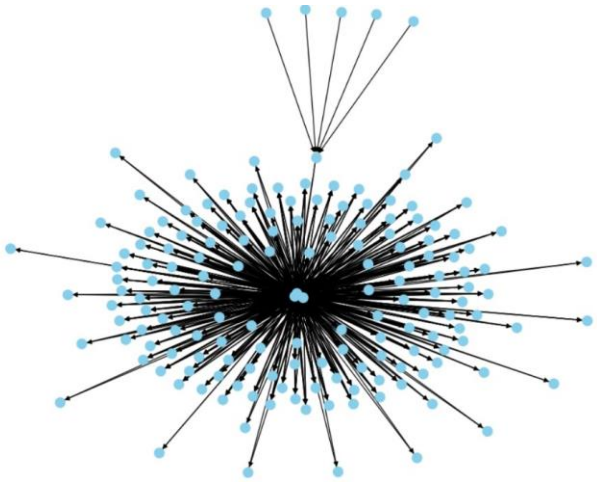

Fig: TDG after DDoS attack on 2011 VAST data

IP (Node)	In degree (Before)	In degree (After)
192.168.1.14	65	142
192.168.1.2	112	134
192.168.1.6	111	152

Table: In-degrees before/after DDoS

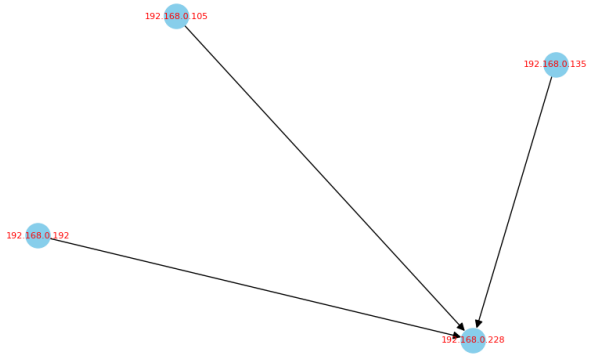

Fig: DDoS attack (custom dataset)

5.2 Zoom Call Traffic Analysis

In our small-scale experiment involving Zoom calls on three hosts within the same network, we noticed that the hosts do not interact directly. Instead, their communication is facilitated through a central server. This conclusion was drawn from observing that the IP addresses of our 3 laptops belong to the same subnet, whereas the IP address of the central node is under a different subnet mask. Please note that in the diagram below, a directed edge represents the flow of data from the source to the destination. In our observation, we identified bidirectional communication.

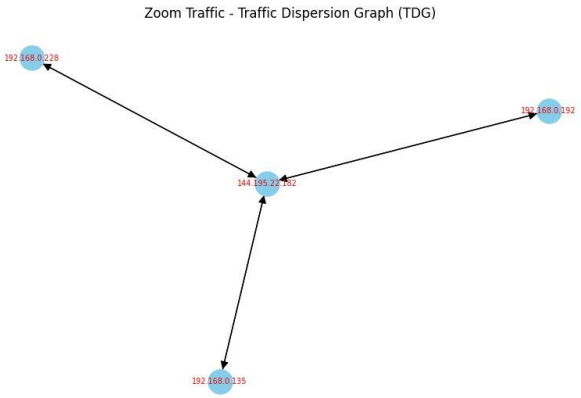

Fig: Zoom call – TDG

5.3 Gaming Traffic Analysis

The Traffic Dispersion Graph (TDG) depicted below represents the network activity for the online multiplayer game TeamFight Tactics, derived from an online 5G traffic dataset. It illustrates numerous user nodes connected to a

central server node, with no direct communication between the user nodes themselves. This centralized network topology is characteristic of client-server models, where all user interactions with the game are mediated through the server. The TDG suggests the server concurrently hosts multiple games, evidenced by the number of user nodes exceeding the eight-player limit of a single game session. Importantly, each connection between the server and user nodes is bidirectional, indicative of continuous data exchange — such as player actions and game state updates. This pattern is expected in multiplayer games requiring real-time synchronization.

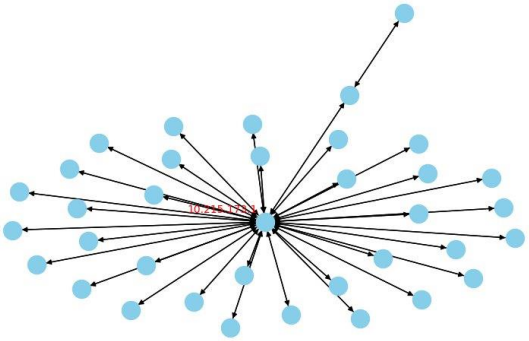

Fig: Gaming(TeamFight Tactics) – TDG

5.4 Port Scan Attack Analysis

An attacker may probe a target machine's ports to discover open or listening ports, which could then be leveraged for malicious purposes. We attempted to detect such port scanning activities through TDG. Utilizing netcat (nc), we conducted a port scan from one host to another on two occasions: initially targeting specific ports and subsequently performing a comprehensive scan of all ports. The findings are illustrated in the figures below. Here, an edge signifies the transfer of a TCP SYN packet from the source to the destination port. We noted a significant uniformity in the source IP addresses along with elevated levels of TCP traffic originating from the same source but directed towards various ports.

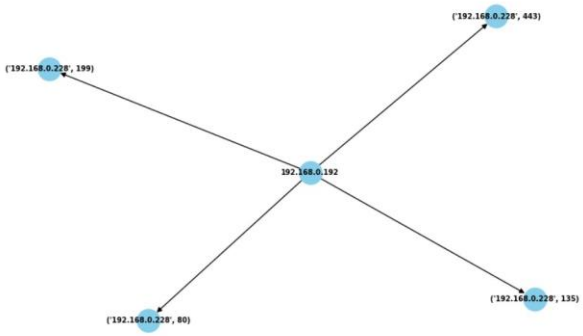

Fig: Port-Scan (specific ports) – TDG

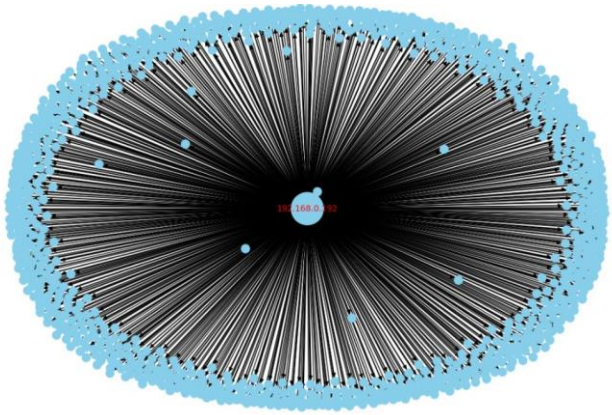

Fig: Full port scan – TDG

5.5 TDG to visualize traffic with different protocols

We utilized Traffic Dependency Graphs (TDG) to examine the traffic flow of various protocols. Traffic data was captured using Wireshark on the University Village Tower (UVT) network. In the TDG, each edge indicates a connection established from one node to another, with distinct colors representing different protocols. We noted that the central node within the cluster on the right corresponds to UVT WiFi (100.64.11.41).

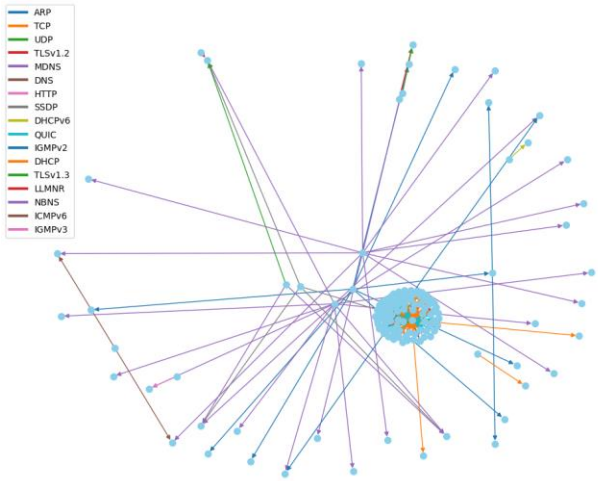

Fig: UVT network – TDG

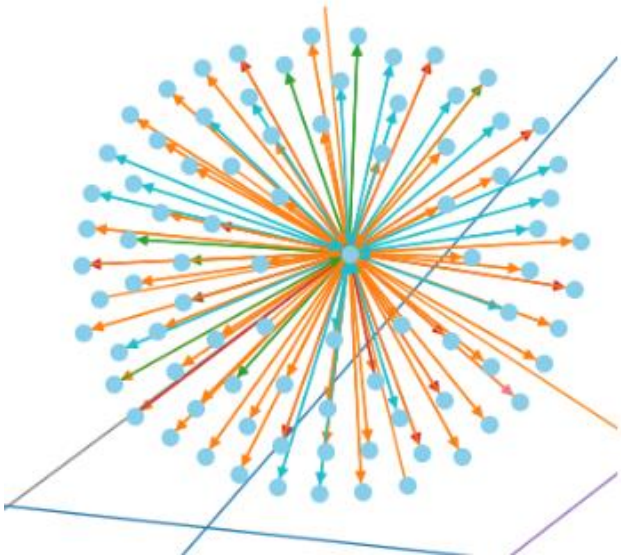

Fig: Cluster in previous TDG zoomed-in

6. Discussion

In the discussion section, we delve deeper into the implications of our findings, highlighting the significance of our work, its relation to other pieces of research, and potential practical applications.

6.1 Overview of Section:

Our discussion begins by emphasizing the importance of our work in the context of network traffic analysis and security. We then explore the connections between our findings and existing research in the field, identifying areas of alignment and divergence. Finally, we discuss potential practical applications of our methodology and offer insights into future research directions.

6.2 Significance of Work:

This work expanded our evaluation of effectiveness of TDGs across diverse scenarios. We found that TDGs are versatile, suitable for visualizing traffic from specific applications as well as analyzing large-scale networks. They proved valuable in detecting potential port scans and DDoS attacks and in illustrating the traffic flow across various protocols. The project deepened our understanding of network behaviors.

6.3 Relation to Other Pieces of Work:

Our work builds upon previous research in network traffic analysis, particularly in the utilization of TDGs for visualization and analysis. The previous papers like [1] helped in building upon their idea of leveraging TDG and techniques from them were used on our custom datasets. Some existing studies also explored specific aspects of network traffic or employed traditional methods. Our approach offers a holistic perspective, leveraging TDGs to capture diverse network activities and interactions.

6.4 Possible Practical Applications:

The practical implications of our findings are significant, with potential applications in various domains. For network administrators, our methodology offers a valuable tool for detecting security threats, such as DDoS attacks and port scanning attempts. Additionally, our insights into network behaviors can inform the optimization of network infrastructure and improve the overall reliability and performance of systems.

6.5 Future Research Directions:

In future developments, the challenge of identifying key patterns within the dense data of TDGs could be addressed by incorporating machine learning for automated pattern detection. Expanding the scope of investigation to include further scenarios such as Spear Phishing Attacks, Man in the Middle Attacks, and Ransomware Traffic is also possible. Additionally, there is potential to establish a system that provides real-time TDGs and traffic analysis within a live operational context.

7. Related Work

In the field of network traffic monitoring and anomaly detection, significant contributions have been made to understand and diagnose abnormal behaviors in network traffic. Among these, the concept of Traffic Dispersion Graphs (TDGs) plays a pivotal role.

One notable study, "Network Monitoring using Traffic Dispersion Graphs (TDGs)" by Marios Iliofotou and others[1], proposes the use of TDGs to monitor, analyze, and visualize network traffic. TDGs represent the social behavior of hosts in a network by modeling "who talks to whom," offering a novel perspective on traffic analysis. This approach not only aids in detecting unwanted applications that obfuscate their traffic but also enhances the understanding of network-wide interactions, providing a robust framework for monitoring network operations.

Another significant contribution is the paper "Traffic Dispersion Graph Based Anomaly Detection" by Do Quoc Le, Taeyoel Jeong, H. Eduardo Roman, and James Won-Ki Hong[2]. This work introduces a novel approach to detect anomalous network traffic using graph theory concepts like degree distribution, maximum degree, and dK-2 distance. The authors employ Traffic Dispersion Graphs (TDG) to model network traffic over time, enabling the detection of anomalies and identification of attack patterns in network traffic. This method has been validated using network traces from POSTECH and CAIDA[3], demonstrating its effectiveness in improving the reliability of anomalous traffic detection.

Both studies underscore the importance of graph-based methods in enhancing the accuracy and efficiency of network monitoring and anomaly detection, thereby contributing to the broader field of network security and management.

8. Effort

In documenting our effort for this project, it is essential to acknowledge the significant challenges and learning opportunities we encountered throughout the development process.

One of the primary challenges was sourcing relevant and current datasets. With heightened privacy regulations, the availability of recent packet movement data has significantly diminished. The task of finding a dataset that was both recent and relevant required exhaustive research and considerable time, as most of the initially discovered datasets were outdated.

Data collection using Wireshark, while initially straightforward, presented its own set of complexities. To ensure purity of the data, particularly for tasks like video streaming, we had to isolate the network activity. This involved turning off all other devices connected to the network and ensuring that only the application under test was active. This step was critical in reducing noise and avoiding the collection of extraneous data. For certain applications, we had to capture network activity from each computer and merged it to produce a unified TDG.

Pattern recognition within the constructed TDGs proved to be another strenuous endeavor. Identifying meaningful insights from the graphs required a deep understanding of both the underlying network behaviors and the graphical representations of these behaviors.

Our technical acumen was significantly expanded by the need to master new Python libraries. None of our team members had prior experience with NetworkX, necessitating a deep dive into its documentation and various coding examples to harness its full potential. Matplotlib also posed a learning curve for effective data visualization.

Initially, the aesthetics of the TDGs left much to be desired; nodes were congested, and the overall clarity of the graphs was lacking. We invested considerable effort in adjusting settings and fine-tuning the visual elements to achieve a balance between technical accuracy and visual clarity. This step was crucial in ensuring that the TDGs were not only analytically valuable but also comprehensible and visually informative.

9. Conclusion

In conclusion, we have explored the realm of network traffic analysis, with a focus on utilizing Traffic Dispersion Graphs (TDGs) for visualization and understanding. Our study has contributed by developing a robust methodology for generating and analyzing TDGs from diverse datasets, identifying patterns and anomalies within network traffic, and demonstrating the versatility of TDGs in capturing various network activities.

The significance of our work lies in its potential to offer valuable insights into network traffic dynamics, enabling

practitioners to make informed decisions and enhance network security measures. Practical implications include guidance for effectively leveraging TDGs in network analysis and security enhancement efforts.

Looking forward, future work should focus on refining and extending our methodology, exploring new applications for TDGs. The challenge of identifying key patterns within the dense data of TDGs could be addressed by incorporating machine learning for automated pattern detection. Additionally, there is potential to establish a system that provides real-time TDGs and traffic analysis within a live operational context.

REFERENCES

[1] Marios Iliofotou, Prashanth Pappu, Michalis Faloutsos, Michael Mitzenmacher, Sumeet Singh, and George Varghese. 2007. Network monitoring using traffic dispersion graphs (TDG). In Proceedings of the 7th ACM SIGCOMM conference on Internet measurement (IMC '07). Association for Computing Machinery, New York, NY, USA, 315–320. https://doi.org/10.1145/1298306.1298349

[2] Do Quoc Le, Taeyoel Jeong, H. Eduardo Roman, and James Won-Ki Hong. 2011. Traffic dispersion graph based anomaly detection. In Proceedings of the 2nd Symposium on Information and Communication Technology (SoICT '11). Association for Computing Machinery, New York, NY, USA, 36–41. https://doi.org/10.1145/2069216.2069227

[3] Hick, P., Aben, E., Claffy, K. and Polterock, J. 2007. The CAIDA DDoS Attack 2007 Dataset. http://www.caida.org/data/passive/ddos-20070804dataset.xml (accessed on 2011-05-10).

[4] D. Q. Le, T. Jeong, H. E. Roman and J. Hong, "Traffic dispersion graph based anomaly detection", Proc. of the 2nd Sym. on Infor. and Comm. Tech., pp. 36-41, Oct. 2011.

[5] Karagiannis, Thomas & Papagiannaki, Konstantina & Faloutsos, Michalis. (2005). BLINC: Multilevel Traffic Classification in the Dark. Computer Communication Review - CCR. 35. 229-240.

[6]https://visualdata.wustl.edu/varepository/VAST%20Challenge%202011/challenges/MC1%20-%20Characterization%20of%20an%20Epidemic%20Spread/

[7] S. Cheung et al. The Design of GrIDS: A Graph-Based Intrusion Detection System. UCD TR-CSE-99- 2, 1999

[8]https://www.kaggle.com/datasets/kimdaegyeom/5g-traffic-datasets