NETWORK SUPPORT FOR IP TRACEBACK

by

Semih Balkı

B.S., Industrial Engineering, Sabanci University, 2019

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Electrical and Electronics Engineering

Boğaziçi University

2024

ACKNOWLEDGEMENTS

Completing this thesis has been an immensely rewarding journey, not just academically but personally, and it would not have been possible without the support and encouragement of many. First and foremost, I extend my deepest gratitude to Prof. Emin Anarım, my thesis advisor, whose guidance, patience, and expertise have been invaluable throughout this process. His insights and suggestions have profoundly shaped this work, and his support has been a constant source of motivation. I owe a debt of gratitude to my peers, Arda Bayer and Ahmetcan Acar, whose contributions were instrumental in the development of the Compressed Edge Fragment Sampling Algorithm. Their collaboration in writing the code and troubleshooting technical challenges was pivotal to the success of this project. To my best friend, whose unwavering support, understanding, and encouragement have been my stronghold. Your belief in my abilities has been a constant source of strength and motivation. Thank you for being there for me through the ups and downs of this academic endeavor. My family deserves special mention for their endless love, support, and sacrifices. Your constant encouragement and belief in my potential have been the backbone of my journey. Thank you for providing me with the strength and peace of mind needed to pursue my goals. To my dog, Paşa, thank you for being my loyal companion and for the countless moments of joy and comfort you have provided me during this stressful period. Your presence has been a source of relaxation and happiness, reminding me of the importance of balance and well-being. This thesis represents not just my academic efforts but the collective support and belief of all those mentioned. I am immensely grateful to each one of you for your invaluable contribution to my journey. Thank you from the bottom of my heart.

ABSTRACT

NETWORK SUPPORT FOR IP TRACEBACK

The increasing frequency and complexity of Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks pose significant threats to network security and integrity. This thesis presents a sophisticated method for IP traceback by enhancing the Compressed Edge Fragment Sampling (CEFS) algorithm. This improved CEFS method employs an innovative packet marking technique that integrates a secret key-sharing mechanism among servers for accurate packet origin authentication. This approach involves encrypting packet identifiers using a shared secret key among the involved servers, significantly improving the efficiency of tracing the origins of malicious packets.

Comparative evaluations with other IP traceback methods, such as ICMP Traceback and Ingress Filtering, reveal that the revised CEFS algorithm is more efficient and effective, particularly in scenarios following an attack. By providing more effective countermeasures against DoS and DDoS attacks, this thesis contributes valuable solutions that bolster network resilience against these widespread threats. This research not only advances cybersecurity technology but also delivers a viable framework for enhancing network security architectures in practical settings.

ÖZET

IP GERİ İZLEME İÇİN AĞ DESTEĞİ

Ağ güvenliğini ve bütünlüğünü koruma konusunda, Sıkıştırılmış Kenar Parça Örnekleme (CEFS) algoritmasını geliştirerek IP geri izlemesi için gelişmiş bir yöntem sunan bu tez, giderek artan ve karmaşıklaşan Hizmet Engelleme (DoS) ve Dağıtılmış Hizmet Engelleme (DDoS) saldırılarının ortaya çıkardığı önemli tehditlere çözüm getirmektedir. Bu geliştirilmiş CEFS yöntemi, paket kökenlerini doğru bir şekilde doğrulamak için sunucular arasında gizli anahtar paylaşım mekanizması entegre eden yenilikçi bir paket işaretleme tekniği kullanmaktadır. Bu yaklaşım, ilgili sunucular arasında paylaşılan gizli bir anahtar kullanılarak paket tanımlayıcılarının şifrelenmesini içermekte olup, kötü amaçlı paketlerin kökenlerinin etkin bir şekilde izlenmesini büyük ölçüde iyileştirmektedir.

ICMP Geri İzleme ve Giriş Filtreleme gibi diğer mevcut IP geri izleme teknikleri ile yapılan karşılaştırmalı analizler, saldırı sonrası senaryolarda değiştirilmiş CEFS algoritmasının daha üstün verimlilik ve etkinlik gösterdiğini ortaya koymaktadır. DoS ve DDoS saldırılarının etkilerini daha etkili bir şekilde azaltarak, bu tez, bu yaygın tehditlere karşı ağ direncini artıran değerli çözümler sunmaktadır. Bu araştırma, sadece siber güvenlik teknolojisini ilerletmekle kalmayıp, aynı zamanda pratik ortamlarda ağ güvenlik altyapılarını iyileştirmek için uygulanabilir bir çerçeve de sunmaktadır.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

c	Number of distinct routers or paths at the same distance from the victim
d	Path length, indicates the number of hops between the attacker and the victim
$E(X)$	The expected number of packets required for path reconstruction
k	Number of non-overlapping fragments
p	Probability of a router marking a packet
p_{auth}	Represent the probability of successfully authenticating a legitimate packet marking
p_{forge}	Represent the probability of an attacker successfully forging a packet marking that passes the secret key validation.
q	Interval for updating the secret key
R	A specific router in the network
w	Number of columns in Δ_q
α	Complexity factor, related with w
Δ_q	Secret key matrix
Σ	Secret key vector
τ	ID field as vector
Ω	Hash value vector

LIST OF ACRONYMS/ABBREVIATIONS

AMS	Advanced Marking Schemes
CEFS	Compressed Edge Fragment Sampling
DDoS	Distributed Denial of Service
Dist	Distance
DoS	Denial of Service
DPM	Deterministic Packet Marking
E-CEFS	Enhanced Compressed Edge Fragment Sampling
FPR	False Positive Rate
ICMP	Internet Control Message Protocol
IP	Internet Protocol
ISP	Internet Service Provider
MAC	Message Authentication Code
Mod. CEFS	Modified CEFS
Orig. CEFS	Original CEFS
PPM	Probabilistic Packet Marking
Pkt. Sav. M-E	Packet Saving Modified vs. Enhanced CEFS
Pkt. Sav. O-E	Packet Saving Original vs. Enhanced CEFS
Pkt. Sav. O-M	Packet Saving Original vs. Modified CEFS
Red. M-E	Reduction between Modified vs. Enhanced CEFS
Red. O-E	Reduction between Original vs. Enhanced CEFS
Red. O-M	Reduction between Original vs. Modified CEFS

1. INTRODUCTION

1.1. Overview of Internet Security Challenges

The advent of the digital era has transformed the Internet into a linchpin of global communication, commerce, and entertainment. However, this ubiquitous connectivity brings with it a Pandora's box of cybersecurity challenges that continually evolve, becoming more sophisticated and damaging. Among these, Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks stand out for their ability to exploit the very architecture of the Internet to render services inaccessible. The methodology behind these attacks is deceptively simple, yet their impact can be devastating, ranging from significant financial losses to the erosion of trust in digital platforms. The foundational vulnerability these attacks exploit lies in the Internet Protocol's (IP) lack of source authentication, a design oversight that dates back to the earliest days of network design. This flaw allows attackers to use IP spoofing to masquerade as legitimate entities, making it challenging to differentiate malicious from benign traffic. Highlighted by Robert T. Morris in 1985, this vulnerability underscores the ongoing battle between evolving threat vectors and cybersecurity defenses, a dynamic that is central to the discourse on Internet security.

1.2. Significance of IP Traceback in Cybersecurity

The rise in the frequency and sophistication of DDoS attacks has catalyzed the development of defensive mechanisms aimed at not only mitigating the impact of these attacks but also identifying their origins. IP Traceback stands as a beacon in this effort, offering a systematic approach to trace back the source of malicious traffic. This methodology is not just a tool for post-attack analysis; it is a cornerstone for developing proactive defenses, shaping the way for more resilient network architectures and security protocols. The evolution of IP Traceback, from rudimentary packet marking to advanced hybrid methods, mirrors the broader narrative of cybersecurity: a con-

stant arms race between attackers and defenders. This pursuit of more effective IP Traceback methods is driven by the necessity to uphold the integrity and reliability of online services, ensuring they remain accessible in the face of relentless cyber threats.

1.3. Objectives of the Thesis

This thesis embarks on a journey to dissect the complex landscape of IP Traceback within the context of DDoS attack mitigation. It aims to navigate through the various methodologies developed over the years, critically analyzing their design, implementation, and effectiveness. The objectives are multi-fold:

(i) Elucidate the Technical Underpinnings: This involves a deep dive into the mechanics of IP Traceback strategies, unraveling how they function to pinpoint the source of DDoS attacks. This exploration seeks to clarify the role these strategies play in the broader cybersecurity defense mechanism, highlighting their potential to fortify network security.

(ii) Assess Limitations and Strengths: By evaluating existing IP Traceback methods, this thesis endeavors to shed light on their operational challenges and limitations, while also recognizing their strengths. This comprehensive assessment aims to provide valuable insights into the practicality of these methods, informing future developments in the field.

(iii) Contribute to Cybersecurity Discourse: The ultimate goal is to contribute to the ongoing dialogue on cybersecurity solutions, advocating for the innovation and adoption of more effective and resilient IP Traceback techniques. Through this work, the thesis aspires to underscore the importance of IP Traceback in the arsenal of tools available to combat DDoS attacks, encouraging further research and innovation in cybersecurity.

By pursuing these objectives, the thesis positions IP Traceback as a critical element in the defense against DDoS attacks, spotlighting its significance in enhancing Internet security and inspiring continued progress in the field of cybersecurity.

2. BACKGROUND AND LITERATURE REVIEW

2.1. The Nature of DoS and DDoS Attacks

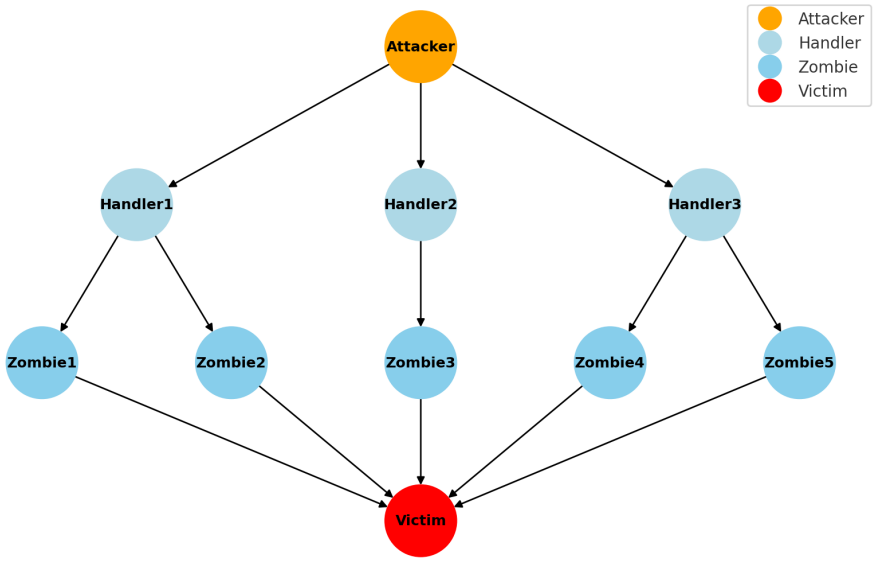

Figure 2.1. Zombie Army.

Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks represent some of the most formidable challenges in the realm of cybersecurity. These attacks fundamentally disrupt the normal functioning of networked services, denying access to legitimate users and compromising the integrity and availability of essential digital resources. The nuanced nature of these attacks, leveraging both the inherent vulnerabilities of the internet protocol (IP) and the vast scale of interconnected systems, necessitates a comprehensive understanding and robust defensive strategies to mitigate their impact [14], [37], [38].

2.1.1. Denial of Service (DoS) Attacks: A Closer Look

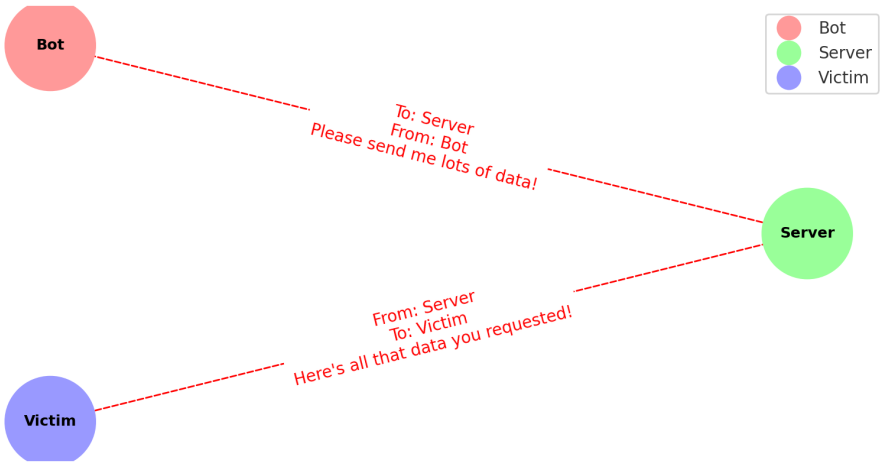

Figure 2.2. IP Spoofing.

A Denial of Service attack is a malicious attempt to interrupt the availability of a target server, service, or network by overwhelming it with a flood of illegitimate requests. This inundation is designed to exhaust the resources of the victim, rendering it incapable of fulfilling legitimate requests. The simplicity and effectiveness of DoS attacks have made them a popular tool for attackers looking to disrupt services for a variety of motivations, ranging from financial gain to ideological objectives [32].

The mechanics of a DoS attack involve the generation of vast amounts of fake traffic directed towards the victim. This can be achieved through various methods, including exploiting vulnerabilities in server software or overwhelming the bandwidth or computational resources of the target. The consequences of such attacks can be severe, leading to prolonged downtime, compromised system integrity, and significant financial and reputational damage to the affected entities [19], [12].

2.1.2. The Evolution into Distributed Denial of Service (DDoS) Attacks

DDoS attacks represent an evolution of the traditional DoS attack, where the malicious traffic is generated not from a single source but from a distributed network of compromised systems, often referred to as a "botnet" or "zombie army". These compromised systems, which can number in the thousands or even millions, are controlled by the attacker and used to generate a massive volume of requests that can overwhelm even well-prepared targets [39], [40].

The distributed nature of these attacks makes them particularly challenging to defend against. The attack traffic can come from diverse geographical locations and across different network paths, complicating efforts to filter out malicious traffic without affecting legitimate users. Furthermore, the use of compromised legitimate systems as part of the botnet adds a layer of complexity to the defense, as blocking these sources outright can lead to collateral damage to innocent users caught in the crossfire [41], [42].

2.1.3. IP Spoofing: A Key Enabler of DDoS Attacks

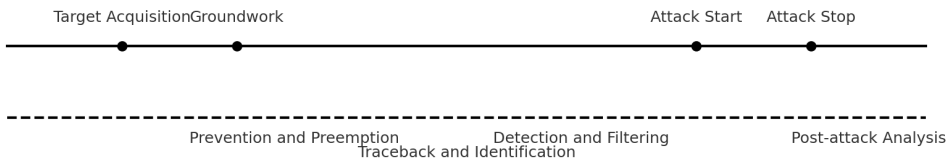

Figure 2.3. DDoS attack timeline.

A critical technique often employed in the execution of DDoS attacks is IP spoofing. This involves the falsification of the source address in IP packets, making the traffic appear to come from a trusted or unrelated source. IP spoofing complicates the

task of distinguishing between legitimate and malicious traffic, and effectively masks the true origin of the attack, thereby protecting the attacker from direct identification and retaliation [43], [44].

The vulnerability of the TCP/IP protocol to such spoofing attacks has been known for decades, yet it remains a potent tool in the arsenal of cyber attackers. This vulnerability underscores the fundamental challenge of securing the open and interconnected nature of the internet, where trust and authentication mechanisms can be exploited for malicious purposes [45], [19].

2.1.4. The Role of IP Traceback in Countering DDoS Attacks

In response to the challenges posed by IP spoofing and DDoS attacks, the cybersecurity community has developed various strategies for tracing the source of malicious traffic, known collectively as IP traceback methods. These techniques aim to identify the origins of attack traffic, even in the presence of spoofing, thereby enabling targeted countermeasures and providing evidence for legal action against the perpetrators [46], [47].

IP traceback involves a range of methodologies, from packet marking schemes that embed path information into packets as they traverse the network, to logging approaches that record packet paths at strategic points across the internet. Each method has its trade-offs in terms of accuracy, resource overhead, and feasibility of implementation, reflecting the complexity of navigating the decentralized and heterogeneous nature of the global internet infrastructure [16].

2.1.5. Conclusion: A Multi-faceted Approach to Mitigating DoS and DDoS Attacks

The nature of DoS and DDoS attacks, with their ability to exploit the fundamental principles of the internet's architecture, presents a persistent threat to the stability and reliability of networked services. Combating these attacks requires a multifaceted approach that includes robust security architectures, real-time monitoring and response capabilities, and the deployment of advanced techniques like IP traceback to deter attackers and minimize the impact of attacks when they occur [48], [39].

As the digital landscape continues to evolve, so too will the tactics of cyber attackers. The ongoing development of more sophisticated defensive technologies, coupled with international cooperation and legal frameworks targeting cybercrime, will be crucial in ensuring the resilience of the internet against the scourge of DoS and DDoS attacks.

2.2. EXISTING DEFENSE MECHANISMS AGAINST DOS/DDOS ATTACKS

IP Traceback represents a critical component in the arsenal of network defense mechanisms, designed to counteract the pervasive threats posed by Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. The core objective of IP Traceback is to accurately determine the origin of malicious traffic, despite the deliberate obfuscation attempts by attackers. This endeavor, however, is significantly complicated by techniques such as IP spoofing, where attackers disguise the true source of their packets by falsifying the packet's source IP address. The consequence of this deception is that the malevolent packets, before arriving at their intended target, traverse through a series of routers—often compromised devices known as zombies—that further mask the trail back to the attacker. Each hop in this journey through the

network infrastructure adds layers of complexity to the traceback process, challenging the defenders' ability to pinpoint the origin of the assault.

The intricacies of the IP Traceback process have led to the development of several methodologies, each aiming to overcome the limitations and exploit different aspects of network behavior and architecture to trace back to the source of the attack. These methodologies include Ingress Filtering, Link Testing, Logging, and ICMP Traceback, each possessing unique attributes and operational principles suited to different scenarios and attack patterns.

2.2.1. Ingress Filtering

Ingress Filtering is a network security measure aimed at mitigating certain types of cyber attacks, notably spoofing attacks which are often a precursor to Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. As delineated in the work by N. M. Tenali and B. S. Jyosyula [48], this process involves configuring routers to reject packets with source addresses that are deemed illegitimate or suspicious. The fundamental premise here is that routers, given the proper configuration and capabilities, can effectively differentiate between legitimate and illegitimate source addresses, thereby preventing malicious packets from infiltrating the network.

The operational mechanics of ingress filtering require routers to expend resources to inspect and analyze each incoming packet. This scrutiny involves a comparative assessment against predefined criteria to ascertain the legitimacy of the packet's source address. Given the resource-intensive nature of this process, routers tasked with performing ingress filtering must possess adequate computational power and capacity to manage the analysis without degrading network performance.

Despite its utility in enhancing network security, the practical deployment of ingress filtering presents certain challenges, particularly in the context of large-scale networks or enterprise environments. The high volume of packet traffic and the com-

plexity of network configurations in such contexts may render ingress filtering less feasible or effective. The logistical and technical demands of implementing and maintaining ingress filtering at scale can be prohibitive, limiting its applicability primarily to smaller customer networks where packet traffic is relatively low, and the potential for address ambiguity is minimized.

One of the direct implications of employing ingress filtering is the potential for inadvertently blocking legitimate packets. This scenario can arise in situations where legitimate servers, for various reasons, utilize spoofed IP addresses for communication. The indiscriminate rejection of packets based on source address legitimacy without consideration of such nuances could lead to unintended disruptions in service and communication within the network. This limitation highlights a critical trade-off inherent in ingress filtering—the balance between enhancing security against spoofing and DoS attacks and ensuring the uninterrupted flow of legitimate traffic.

Given these considerations, ingress filtering emerges as a viable security measure predominantly for smaller-scale networks or in environments where the clarity of legitimate versus illegitimate traffic is more pronounced. Its efficacy in thwarting DoS attacks, while significant, is tempered by the operational and logistical constraints associated with its implementation. As such, while ingress filtering represents a valuable tool in the cybersecurity arsenal, its deployment must be carefully considered within the broader context of network architecture, traffic characteristics, and security objectives.

In summary, ingress filtering serves as a targeted approach to enhancing network security by preventing the ingress of packets with spoofed or otherwise illegitimate source addresses. Its application, while beneficial in specific contexts, is bounded by practical limitations related to network size, traffic volume, and the potential for inadvertently blocking legitimate communications. As networks continue to evolve and diversify, the role of ingress filtering in the cybersecurity landscape will likely remain that of a niche strategy, complemented by a suite of other security measures designed

to address the multifaceted challenges of protecting digital infrastructures from cyber threats.

2.2.2. Link Testing

Link testing, as an innovative approach to IP traceback, represents a critical advancement in the realm of cybersecurity, specifically in the efforts to trace and mitigate the impact of cyber attacks, including Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. This technique, first proposed by H. Burch and B. Cheswick [39], is grounded in the principle of systematically investigating the path of attack traffic through the network, starting from the router closest to the victim and moving upstream to identify the source of malicious traffic.

The essence of link testing lies in its methodical approach to tracing the origins of attack traffic by analyzing the upstream links that have propagated the attacker's traffic. This process is executed in a recursive manner, moving from the victim's network upstream towards the source, thereby narrowing down the potential origins of the attack. Two primary methodologies underpin the concept of link testing: input debugging and controlled flooding, each with its distinct operational mechanism and applications.

Input debugging is predicated on the victim's ability to replicate a hostile packet that mimics the characteristics of the attack traffic. This replicated packet is then sent back upstream to the router from which the original attack packet was received. Upon receipt, the router conducts a comparison between the replicated packet and the signatures of incoming attack packets across its upstream links. This comparative analysis enables the identification of specific upstream routers through which the attack traffic is flowing. The recursive application of this method across multiple routers holds the potential to unveil the attack's path back to its source, thereby identifying the attacker. The merits of input debugging include its compatibility with existing network protocols, minimal bandwidth consumption, and the absence of a requirement

for new protocol introductions. However, the efficacy of this method hinges on the cooperation of Internet Service Providers (ISPs) and the continuity of the attack during the traceback process.

Contrary to input debugging, controlled flooding does not rely on packet replication and comparison. Instead, this method entails measuring the intensity of attack traffic on upstream links. By systematically increasing or modulating traffic on these links, it is possible to observe variations in attack intensity, thereby identifying links that contribute to the propagation of attack traffic. Similar to input debugging, controlled flooding offers the advantage of easy implementation without necessitating new protocol introductions. Despite its simplicity, this approach faces limitations in its application against DDoS attacks, necessitates knowledge of the network's structure, and requires the attack to be ongoing during the implementation of the technique.

Both input debugging and controlled flooding embody the strategic essence of link testing as a means to trace cyber attacks to their sources. While each method offers distinct advantages and faces particular limitations, their contributions to cybersecurity are undeniable. They provide valuable mechanisms for identifying the pathways through which cyber attacks are launched, thereby enabling targeted defensive actions and facilitating efforts to hold attackers accountable.

In conclusion, link testing, through its input debugging and controlled flooding methodologies, presents a nuanced approach to IP traceback. While challenges remain in their practical application, particularly in the context of widespread DDoS attacks and the dependency on active attack schemes, the foundational principle of tracing attack traffic through network links marks a significant step forward in cybersecurity efforts. The ongoing refinement and adaptation of these techniques, coupled with enhanced cooperation among network operators and ISPs, hold promise for more effective mitigation of cyber threats in the digital age.

2.2.3. Logging

The IP traceback problem, a significant challenge in the realm of cybersecurity, is central to the efforts of identifying the origins of malicious network traffic, particularly in the context of Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. A promising approach to addressing this issue, as discussed by Sager [43] and Stone [45], involves logging packets at strategic points within the network. This method proposes the recording of packet information at certain routers which are determined to be critical junctures for traffic flow, thereby creating a repository of data that can later be analyzed to trace the path of packets through the network.

The concept of logging as a solution to the IP traceback challenge offers several distinct advantages. Primarily, it operates within the existing network and protocol infrastructure, requiring no introduction of new protocols or modifications to current network operations. This seamless integration ensures that the traceback process can be conducted without disrupting normal network functions or requiring extensive reconfiguration of network elements.

Furthermore, logging allows for IP traceback to be conducted in the post-attack phase. This retrospective analysis capability is particularly valuable, as it enables network administrators and security professionals to dissect attack vectors and origins without the pressure of real-time analysis or the need for immediate response during an ongoing attack. The absence of additional traffic generated by the traceback process itself is another notable benefit, as it ensures that the investigation does not contribute to network congestion or interfere with legitimate network traffic.

Despite these advantages, the logging approach to IP traceback is not without its challenges. The foremost concern is the significant storage overhead required to log all packet information traversing the monitored routers. The sheer volume of data generated by modern networks, combined with the need to retain detailed packet logs for analysis, necessitates substantial storage capacity, potentially imposing considerable

costs and logistical complexities.

Privacy concerns also emerge as a critical consideration in the logging methodology. The comprehensive logging of packet information inevitably captures a wide array of data, including potentially sensitive content. This raises questions regarding user privacy and the legal implications of storing and analyzing such data, necessitating careful consideration of privacy protections and compliance with data protection regulations.

Moreover, the utility of logged data for IP traceback is inherently time-bound. The necessity to refresh or overwrite logs after a certain period, due to storage constraints or policy requirements, imposes a limited window for analysis. This temporal limitation means that the effectiveness of the logging approach in supporting post-attack investigations is contingent upon timely analysis and the availability of recent log data.

In summary, while logging presents a viable method for addressing the IP traceback problem, offering a non-intrusive, protocol-agnostic solution capable of supporting detailed post-attack analysis, it also poses significant challenges. These include the need for extensive storage capacity, considerations regarding user privacy, and the time-sensitive nature of log data availability. Addressing these challenges will be key to realizing the full potential of logging as a tool for enhancing network security and resilience in the face of cyber threats.

2.2.4. ICMP Traceback

The ICMP based IP traceback scheme, as initially introduced by Taylor, Leech, and Bellovin [44], represents a significant innovation in the realm of cybersecurity, aimed at enhancing the capability to track and analyze the path of packets through a network, particularly in the context of mitigating Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. This approach hinges on the strategic use

of Internet Control Message Protocol (ICMP) packets, also referred to as trace packets, which are generated by routers as they process network traffic.

The fundamental premise of the ICMP traceback strategy is the selective generation of trace packets from a subset of the passing traffic, as opposed to generating a trace packet for every packet that traverses the router. Given the voluminous nature of network traffic, particularly in large-scale or high-throughput environments, the generation of trace packets for every passing packet is deemed impractical. Therefore, a sampling strategy is employed, such as generating a trace packet for every 20,000th packet, to ensure manageability and minimize additional network load. This selective approach balances the need for traceback capability with considerations for network performance and overhead.

The trace packets generated under the ICMP traceback scheme are imbued with critical information that facilitates the later analysis and traceback of packet paths. This information includes, but is not limited to, timestamps indicating the time of packet processing, details about adjacent routers through which the packet has passed, as well as IP and MAC addresses relevant to the packet's journey. This rich dataset enables comprehensive post-attack analysis, providing investigators and network administrators with the necessary insights to trace back and understand the origins and paths of attack traffic.

One of the notable advantages of the ICMP traceback approach is its compatibility with existing network protocols and architectures. It seamlessly integrates into the network infrastructure without necessitating the introduction of new protocols or extensive modifications to network operations. Moreover, its implementation does not require intensive collaboration with Internet Service Providers (ISPs), further enhancing its appeal as a practical and accessible option for network defense.

However, the ICMP traceback scheme is not devoid of challenges and limitations. A significant concern is the potential for misuse or exploitation by attackers. In the absence of robust authentication mechanisms for trace packets, malicious actors may generate spoofed trace packets to obfuscate their activities or mislead investigators, undermining the reliability of traceback efforts. Additionally, the generation of trace packets, albeit selectively, introduces additional traffic into the network, which, during large-scale attacks, could compound network congestion and impede normal operations.

Another critical consideration is the reliance of ICMP traceback on specific router functionalities, such as input debugging, which may not be uniformly available across all router models and architectures. This variability in router capabilities can limit the applicability and effectiveness of the ICMP traceback method, restricting its deployment to environments where compatible router technology is in place. Furthermore, the successful implementation of ICMP traceback in diverse network settings may necessitate the development and deployment of key distribution solutions to ensure the integrity and authentication of trace packets.

In conclusion, the ICMP based IP traceback scheme offers a promising avenue for enhancing network security and forensic capabilities, particularly in the context of identifying and analyzing the sources of cyber attacks. Despite its potential, the practical deployment of this scheme must carefully navigate the challenges of authentication, additional network traffic, and router compatibility. Continued advancements in technology and cybersecurity practices are expected to further refine and expand the utility of ICMP traceback, solidifying its role as a valuable tool in the ongoing effort to secure digital networks against the evolving landscape of cyber threats.

2.3. Limitations of Current Approaches

We delve into the complexities and challenges inherent in the prevailing methodologies for IP traceback, a critical aspect of cybersecurity aimed at identifying the sources of malicious network traffic. These methodologies, including Ingress Filtering,

Link Testing, Packet Logging, and ICMP Traceback, each present unique advantages in the quest to enhance network security. However, they also embody distinct limitations that constrain their efficacy and practicality across diverse network environments. This section examines these limitations, drawing upon insights from the preceding discussions on each approach.

Ingress Filtering has emerged as a proactive measure to prevent packets with illegitimate source addresses from entering a network. Despite its utility in thwarting spoofing attempts, its application is predominantly viable in smaller, customer-focused networks due to the substantial resource demands and potential for ambiguity in address legitimacy in larger enterprise or global network contexts. The necessity for routers to possess sufficient computational resources to analyze each packet introduces a scalability challenge, rendering Ingress Filtering less feasible for widespread adoption [43], [45].

Link Testing, while innovative in its approach to recursively identifying the upstream paths of attack traffic, is contingent upon active cooperation from Internet Service Providers (ISPs) and the persistence of the attack during the analysis phase. The requirement for the attack to remain active for effective traceback significantly limits its utility, especially in scenarios where attackers employ brief, sporadic, or complex multi-vector attacks [39].

Packet Logging offers a retrospective analysis capability, logging packets at strategic routers to facilitate post-attack traceback. However, this method's viability is marred by substantial storage overhead, necessitating the logging of immense volumes of packet information. Privacy concerns further complicate Packet Logging, as the comprehensive capture of packet contents raises significant data protection and user privacy issues. Moreover, the finite nature of log storage, necessitating periodic refreshing or deletion of logs, imposes a temporal limitation on analysis efforts [43], [45].

ICMP Traceback proposes a selective packet marking strategy using ICMP packets to encode path information, circumventing the need for new protocols. While this method reduces the need for ISP cooperation and allows for post-attack analysis, it introduces additional network traffic through the generation of trace packets. The lack of robust authentication for these trace packets presents an opportunity for attackers to generate spoofed trace packets, potentially misleading traceback efforts. Furthermore, ICMP Traceback's effectiveness is contingent upon router support for specific features like input debugging, which may not be universally available [44].

The collective examination of these methodologies underscores a recurring theme of trade-offs between effectiveness, resource demands, privacy considerations, and operational feasibility. While each approach contributes valuable strategies to the domain of IP traceback, their limitations highlight the need for continued innovation and development of more holistic, scalable, and secure solutions. As cyber threats evolve in complexity and scale, so too must the methods employed to trace and mitigate these threats, ensuring the resilience and integrity of digital networks in the face of relentless cyber adversaries.

2.4. Evolution Towards IP Traceback Methods

The trajectory of development and refinement in strategies devised to trace the origins of malicious network traffic, particularly in the face of Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. The landscape of IP traceback has witnessed significant innovations, from ingress filtering to ICMP Traceback, each method aiming to enhance the capability of network administrators to identify and mitigate the sources of cyber threats. This section explores the evolutionary path of these methodologies, underscoring the advancements and the shifting paradigms in tackling the challenges of cybersecurity.

Ingress Filtering, as one of the earliest approaches, laid the foundational principle of preemptively blocking packets with suspicious or illegitimate source addresses at the network's edge. This method's simplicity and directness in curbing spoofing attempts marked a crucial first step in the IP traceback journey. However, its applicability was soon recognized to be limited by scalability issues and the intricate challenge of distinguishing between legitimate and spoofed addresses in complex network environments [43], [45].

Link Testing introduced a more dynamic approach, focusing on tracing the upstream path of attack traffic through recursive testing. This method represented a shift towards a more investigative and analytical framework for IP traceback, although it was hampered by practical challenges such as the need for ongoing attacks and cooperation from ISPs for effective implementation [39].

The adoption of Packet Logging marked a significant evolution, emphasizing the collection and analysis of packet data across network nodes. This method's retrospective analysis capability offered valuable insights into attack paths, albeit at the cost of substantial storage requirements and emerging concerns regarding privacy and data protection [43], [45].

ICMP Traceback emerged as an innovative solution, leveraging ICMP packets for selective marking and facilitating post-attack traceback without introducing new protocols. This approach underscored the growing emphasis on efficiency and minimal network disruption, addressing some of the limitations of earlier methods by reducing the reliance on ISP cooperation and minimizing additional network traffic [44].

The evolution towards modern IP traceback methods reflects a continuous balancing act between effectiveness, practicality, and resource optimization. Each phase of development has contributed to a deeper understanding of the complexities involved in tracing cyber attacks, leading to more sophisticated and nuanced solutions. The progression from static filtering techniques to dynamic, investigative methodologies

illustrates the cybersecurity field's adaptive response to the ever-changing threat landscape.

Moreover, this evolutionary journey highlights the critical importance of innovation in addressing the inherent challenges of cybersecurity. As attackers employ increasingly sophisticated strategies to evade detection, the development of IP traceback methods must similarly evolve, embracing advanced technologies and approaches to stay ahead of threats.

In conclusion, the evolution towards IP traceback methods encapsulates a significant aspect of the broader efforts to enhance network security and resilience. From the foundational steps of ingress filtering to the nuanced strategies of ICMP Traceback, each method has paved the way for future advancements, setting the stage for continued innovation in the relentless battle against cyber threats. The trajectory of these developments not only reflects the ingenuity and adaptability of cybersecurity practices but also underscores the imperative for ongoing research, collaboration, and technological advancement to safeguard digital infrastructures in an increasingly interconnected world.

2.5. Conclusion

Throughout this exploration, we have traversed the landscape of cybersecurity solutions developed to counteract the pervasive threats of Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. The journey from foundational methods such as Ingress Filtering to more sophisticated strategies like ICMP Traceback underscores the dynamic and responsive nature of cybersecurity efforts in addressing the multifaceted challenges posed by malicious network traffic.

Ingress Filtering, as an initial foray into IP traceback, offered a straightforward approach to blocking packets with illegitimate source addresses. Despite its simplicity, the method's limitations in scalability and applicability in complex network environ-

ments highlighted the need for more adaptable and nuanced approaches. Link Testing and Packet Logging further advanced the traceback capabilities by enabling more detailed analysis of attack paths, albeit with challenges related to practical implementation, resource demands, and privacy considerations.

The introduction of ICMP Traceback marked a significant evolution, providing a method that balanced efficiency with the capability for detailed post-attack analysis without necessitating extensive cooperation from Internet Service Providers (ISPs) or introducing new network traffic burdens. This method, alongside its predecessors, exemplifies the ongoing effort to refine IP traceback techniques to be both effective and practical within the diverse architectures of modern digital networks.

The exploration of these methodologies reveals a consistent theme: the balancing act between the desire for comprehensive traceback capabilities and the practical constraints of network performance, privacy, and scalability. As cyber threats continue to evolve in sophistication and scale, so too must the strategies employed to trace and mitigate these threats. The future of IP traceback lies in the continued innovation and integration of advanced technologies, from machine learning algorithms to decentralized blockchain networks, to enhance the precision, efficiency, and security of cyber defense mechanisms.

Moreover, the journey through IP traceback methodologies underscores the importance of collaboration among cybersecurity professionals, network operators, ISPs, and policymakers. Developing standards, fostering international cooperation, and ensuring alignment with legal and ethical considerations are crucial for advancing the effectiveness and acceptance of IP traceback techniques.

In conclusion, the quest for robust IP traceback methods is an integral component of the broader cybersecurity landscape, reflecting the adaptive and innovative spirit of those dedicated to protecting the integrity and availability of digital infrastructures. The evolution of traceback techniques from simple filtering to advanced packet marking

and logging illustrates a commitment to overcoming the challenges posed by cyber adversaries. As we look to the future, the continued refinement of these methods, coupled with a collaborative and multidisciplinary approach to cybersecurity, will be paramount in ensuring a safer digital world for all.

3. IP TRACEBACK TECHNIQUES

	Management overhead	Network overhead	Router overhead	Distributed capability	Post-mortem capability	Preventative/ reactive
Ingress filtering	Moderate	Low	Moderate	N/A	N/A	Preventative
Link testing	Moderate	Low	Moderate	N/A	N/A	Preventative
Input debugging	High	Low	High	Good	Poor	Reactive
Controlled flooding	Low	High	Low	Poor	Poor	Reactive
Logging	High	Low	High	Excellent	Excellent	Reactive
ICMP Traceback	Low	Low	Low	Good	Excellent	Reactive
Marking	Low	Low	Low	Good	Excellent	Reactive

Figure 3.1. Comparison of methods [19].

In the domain of cybersecurity, the imperative to trace and mitigate the deleterious effects of Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks has rendered the development of robust IP traceback mechanisms a focal point of research. Within this context, packet marking has surfaced as a pivotal technique, offering a nuanced solution to the constraints imposed by more conventional methods.

The foundational principle of packet marking, as postulated by Burch and Cheswick [39], involves the strategic annotation of packets with trail information. This information is intricately woven into the packet's journey, enabling victims to reconstruct the assault's trajectory post-attack and trace back to the progenitor. This paradigmatic shift from traditional approaches like Ingress Filtering, which blocks packets based on static rules, and Link Testing, which tracks packet flow through active network cooperation, positions packet marking as a technique that synergistically integrates with the packets' transit, rendering it inherently capable of facilitating post-incident investigations.

As expounded in Figure 3.1, packet marking stands out within a comparative framework evaluating critical criteria instrumental for the operational viability of IP traceback mechanisms. The parameters scrutinized include overheads associated with management, network, and routers, alongside capabilities in distributed systems and post-mortem analysis.

Foremost, when adjudicating management overhead, packet marking delineates a compelling advantage. Unlike the resource-intensive Ingress Filtering, which necessitates continuous rule-set evolution, and Logging, which demands expansive data management, packet marking operates autonomously. It eschews the need for constant oversight, conferring upon system administrators the latitude to concentrate on strategic security imperatives rather than on the minutiae of ongoing management.

Concerning network overhead, methods like Controlled Flooding are encumbered by their reliance on an a priori understanding of network topography, a necessity that engenders considerable network strain. Packet marking transcends this requirement, functioning with agility and adaptability, uninhibited by the structural complexities of network configurations.

Router overhead is pivotal; as networks scale, the resource demand on routers intensifies. Techniques such as Input Debugging and Logging levy a heavy tax on routers, demanding extensive computational capacity. Conversely, packet marking imposes a negligible overhead, enabling its seamless integration into existing network infrastructures without necessitating additional resource allocation.

The efficacy of packet marking is further accentuated in distributed environments—a key consideration in contemporary, expansive attack vectors. Unlike Controlled Flooding, whose efficacy is compromised by its foundational dependence on pre-existing network blueprints, packet marking's distributed capability is inherent, allowing for scalable and robust deployment.

In the arena of post-mortem capability, packet marking distinctly outperforms Link Testing. While Link Testing's utility evaporates with the cessation of an attack, packet marking provides invaluable forensic capabilities, offering insights into the attack long after its conclusion.

In the dialectic of preventative versus reactive measures, packet marking aligns with the reactive paradigm. However, it's pivotal to acknowledge that while Ingress Filtering operates preventatively, it often demands impractical trade-offs, especially in dynamic network environments where flexibility and scalability are paramount.

In summation, packet marking distinguishes itself as a versatile and efficient technique within the arsenal of IP traceback methods. By skirting the pitfalls that beleaguer alternative approaches, packet marking affords a balanced, strategic mechanism poised to address the complexities of tracing cyber onslaughts in an increasingly interconnected world.

3.1. Probabilistic Packet Marking (PPM)

In the analytical landscape of IP traceback methodologies delineated in the compendious chapter on IP Traceback Techniques, an essential dichotomy surfaces, categorizing the techniques into probabilistic and deterministic packet marking. This demarcation transcends theoretical musings, extending its influence to the pragmatics of implementation, efficacy, and the adaptability of the traceback process in the crucible of real-world application.

The paradigm of Probabilistic Packet Marking (PPM) emerges as a sophisticated stratagem within this discourse. PPM entails the stochastic annotation of packets with incremental path information—referred to as 'marking information'—as they traverse the network from the origin to the intended victim. While individually inconclusive, these fragments cumulatively coalesce to manifest a probabilistic contour of the attack vector. This culmination of marked packets engenders a probabilistic map, ostensibly delineating the trajectory of the attack.

The acumen of PPM is encapsulated in its reliance on the law of large numbers, which posits that precision in the reconstructed attack path escalates commensurately with the increment in the volume of marked packets. Moreover, PPM's implementation

is characterized by a minimal footprint on network performance, eschewing the need for comprehensive knowledge of the network's configuration and forgoing the necessity of active router collaboration. Nonetheless, PPM is not impervious to imperfections, as packet loss, routing disparities, and the stochastic nature of packet marking could necessitate the accrual of a considerable corpus of packets to refine traceback accuracy.

Within the purview of PPM, several avant-garde techniques have been developed, each augmenting the domain with nuanced innovations:

- Compressed Edge Fragment Sampling: Conceived by Savage et al. [19], this methodology encodes the 'edge'—the interstice between two IP addresses within the network's fabric. The objective is to distill and condense these edge fragments, thereby reducing the exigencies of storage and computation without compromising the facility to trace the attack's origin.
- Advanced Marking Schemes: The brainchild of Song and Perrig [46], these schemas, encompassing Advanced Marking Scheme I and II, alongside an Advanced Authenticated Scheme, refine the marking process. By introducing intricate encoding mechanisms for path information and, notably in the authenticated variant, providing safeguards to uphold marking integrity, these schemes fortify the network against spoofing exploits.
- Algorithms for Reconstructing DDoS Attack Graphs: Pelleg et al. [49] tackle the exigency of delineating attack graphs amidst DDoS onslaughts. Their algorithms capitalize on probabilistic markings to reconstruct the attack's architecture, shedding light on not only the source but also the expansive nature of the threat.

The confluence of these methodologies underlines the vibrant dynamism inherent in PPM as an active field of inquiry and refinement. Each technical iteration propels the discipline towards more robust and efficient IP traceback capabilities, buttressing the cybersecurity infrastructure against the perpetually shifting panorama of cyber hostilities.

3.1.1. Compressed Edge Fragment Sampling

The technique of Compressed Edge Fragment Sampling represents a pivotal innovation in the realm of Probabilistic Packet Marking (PPM), providing a methodical solution to the challenge of efficiently encoding the pathway traversed by packets during a cyber attack.

Encoding Each Edge Using XOR. Central to this technique is the utilization of the XOR operation to encode the 'edge', defined as the connection between two consecutive IP addresses within the packet's path. Traditional methods that involve recording both IP addresses are replaced by the XOR operation, effectively maintaining the intrinsic characteristics of the edge while minimizing the informational footprint required for each packet. This method not only ensures a more efficient use of the available space within the IP header but also retains the ability to reconstruct the path by exploiting the reversible property of the XOR operation. The process involves marking each packet at various routers along its path with an XOR of the router's IP address and the IP address of the previous router. For instance, if a packet travels through routers with IP addresses labeled as a, b, c, and then d, at each router, the packet is marked with the result of the current router's IP address XORed with the previous router's IP address (e.g., at router b, the packet is marked with $a \oplus b$). This marking continues until the packet reaches its destination, known as the 'victim' in this context. Upon arrival, the victim uses the same XOR operation to decode and reconstruct the entire path from the sequence of encoded addresses. This reconstruction is feasible due to the reversible nature of the XOR operation, allowing the original sequence of router addresses to be systematically retrieved. Although the visual demonstration of this method is omitted, the described technique highlights the use of XOR for efficient path encoding and simplifies path tracking within network packets.

Subdividing Edge-Id into Number k. Given the limited bit-space in the IP header, specifically the 16-bit identification field, the technique proposes a subdivision of the edge identifier (edge-id) into k fragments. This subdivision allows routers to mark packets with a randomly chosen fragment of the edge-id, subsequently facilitating the reconstruction of the complete edge-id at the destination. To ascertain the integrity of the fragments upon reconstruction, $\log_2 k$ bits are appended as an offset to each fragment. This method is crucial in reducing the per-packet space requirement, thus enabling a more scalable approach to packet marking. The process of edge-id subdivision and marking is facilitated by encoding the address and its hash into several segments. Initially, the address is processed to create a hash value. Both the address and its hash are then interleaved bit by bit to form a single sequence. This interleaved sequence is then subdivided into k fragments, which are sent into the network. Each fragment comprises bits from both the address and its hash, maintaining data integrity and allowing for the original data to be reconstructed at the destination by reversing the interleave process.

Adding Error Detection Code. Considering the potential for collisions in edge-id fragments—owing to their non-uniqueness—an error detection code is incorporated within the marking process. This code significantly diminishes the likelihood of erroneously reconstructing the edge-id from overlapping fragments, thereby enhancing the accuracy of the traceback mechanism. The application of an error detection code is a proactive measure to ensure the fidelity and reliability of the packet marking technique. The depicted process involves combining k fragments from the network, which are then passed through a BitDeinterleave process to separate the interleaved bits of the address and its hash. These are further processed to confirm if they match the expected hash of the address, a crucial step in validating the integrity of the reconstructed data. If the hashes match, the original address is accepted; otherwise, it is rejected. This method effectively reduces the risk of path reconstruction errors, thus maintaining the integrity of data transmission across network paths.

In summation, the Compressed Edge Fragment Sampling method significantly contributes to the body of knowledge in IP Traceback, representing a quantum leap in the effective tracing of cyber attacks. Through its intelligent use of XOR encoding, packet space optimization, and error detection, this technique fortifies the network's ability to pinpoint the origins of malevolent traffic, thus bolstering its defensive capabilities in the ever-escalating arms race against cyber threats.

3.1.2. Advanced Marking Scheme I, II, and Advanced Authenticated Scheme

In the arena of IP traceback, the shortcomings of the Fragment Marking Scheme (FMS), particularly its inability to differentiate disjoint paths, render it suboptimal for robust attack path reconstruction – especially under Distributed Denial of Service (DDoS) conditions. FMS is especially challenged by a high false positive rate attributable to its insufficient encoding capabilities. This was evidenced by Song and Perrig's [46] observation that reconstructing paths with 25 distributed attackers could extend to several days. The quintessential challenge herein is the development of an algorithm capable of efficient, accurate, and authenticated encoding within the constraints of the 16 available bits in the IP identification field.

Advanced Marking Scheme I (AMS-I). AMS-I innovates by hashing the router's IP address using two distinct hash functions into an 11-bit structure, $h(R_i)$, and appending a 5-bit distance field, cumulatively conforming to the 16-bit limit. The schema is designed such that each packet carries an 11-bit hash of the router's IP address, $h(R_i)$, derived by applying a hash function. This hash is coupled with a 5-bit field that represents the distance (in hops) from the source or a significant node, allowing an effective tracking of packet paths up to 32 hops, which studies suggest is sufficient to represent most internet paths. Together, these fields utilize the full capacity of the 16-bit identification field available in the IP header. This marking structure not only minimizes the space required within the packet but also enhances the traceability and integrity verification of the path, as the hash ensures that even if packets are reordered

or dropped, the path can be reconstructed up to the point of disruption. AMS-I has demonstrated its capacity to detect up to 50 distributed attack sites, a significant enhancement over previous methods. This methodology efficiently combines path length and hash integrity checks into a compact representation, optimizing both space and functional utility in network traffic management.

Advanced Marking Scheme II (AMS-II). AMS-II extends the approach by employing a set of 2^w independent hash functions to minimize collision probability, described as $\frac{1}{2^{11 \cdot m}}$. When marking a packet, a router selects one hash function from the set, necessitating that the victim at the reconstruction phase identifies the hash function used. This methodology has proven capable of identifying up to 1500 distributed attack sites. The approach is illustrated by the packet marking and path reconstruction process. Each router along the packet's path marks the packet by applying an XOR operation between the router's IP address and the previously marked address. This process continues sequentially down the path until it reaches the victim. For reconstruction, the victim decodes the path by sequentially reversing the XOR operations, starting from the final mark and the known victim's address, progressing backwards to deduce the original sequence of router addresses involved in the path. This enables a precise reconstruction of the packet's path despite potential overlaps and collisions in hash values, thanks to the independent set of hash functions used at each step.

Advanced Authenticated Scheme. Addressing the vulnerability of packet marking algorithms to compromised routers, the Advanced Authenticated Scheme applies a Message Authentication Code (MAC) per marking, utilizing HMAC-MD5 for enhanced efficiency compared to 1024-bit RSA signing. This scheme significantly strengthens the integrity of the traceback process.

The advanced marking schemes, with their nuanced approach to encoding and authentication, present a comprehensive solution to the challenges inherent in IP traceback, heralding an era of enhanced precision in cyber-attack detection and analysis.

3.1.3. Algorithms for Reconstructing DDoS Attack Graphs

In the realm of cyber security, the development of algorithms for reconstructing DDoS attack graphs constitutes a significant stride towards the understanding and mitigation of such pervasive threats. Two principal algorithms, the Basic and the Heuristic algorithms, are at the forefront of this endeavor.

The Basic algorithm operates on the principle of probabilistic marking, wherein each packet is independently marked with a probability p, given p is below a predefined threshold. This probability serves as the cornerstone of the marking strategy. Upon receipt of a marked packet, it is discerned to contain an edge segment of the attack graph, complete with a distance field. Nevertheless, a marked packet does not guarantee the accurate identification of the attacker's IP address, which could be obfuscated or forged, leading to an approximate traceback problem. For example, if packet fragments v_0, v_1, and v_2 are collected, one might postulate that v_2 corresponds to the attacker, albeit this might not reflect reality.

The Heuristic algorithm introduces an additional layer of complexity by implementing a waiting mechanism upon the acquisition of a complete subpath P'. The length of this subpath, denoted by $j = L(P')$, is instrumental in determining the subsequent course of action. If P' does not constitute the entirety of the path P, there exists at least one additional edge segment that remains unmarked. The probability that none of the ℓ packets was marked by this elusive edge is quantified as $(1 - p(1-p)^j)^\ell$. The collection of packets persists until this probability falls beneath a negligible threshold ϵ. When this condition is met, P' is accepted as the complete attack path, termed ϵ-full timed.

The methodologies applied in the analysis are well-illustrated through comprehensive statistical assessments, highlighting the reconstruction of paths using the expected number of packets based on formulas from both Savage et al. and Saurabh et al., particularly for scenarios with $n = 4$ routers. This comparison elucidates the precision and

effectiveness of various algorithms in reconstructing network paths under controlled conditions. The outcomes demonstrate that the expected number of packets required for accurate path reconstruction varies significantly across different algorithms, thereby influencing their practical applicability in real-world scenarios.

Further, a success rate comparison among the algorithms reveals their varying degrees of efficiency in identifying the sources of distributed denial of service (DDoS) attacks. These success rates are quantified through extensive simulations, indicating that while some algorithms exhibit near-perfect success rates, others struggle with lower reliability, thus affecting their adoption for security-sensitive environments.

Moreover, the probability distribution of achieving full subpath lengths at any iteration was meticulously analyzed for a scenario involving a single attacker with $n = 25$ routers. This analysis employed a logarithmic scale to detail the probability across different subpath lengths, showcasing the challenges and potential of using advanced probabilistic models to predict the success rates of path reconstruction under various network conditions.

These analyses collectively underscore the algorithms' efficacy and potential in tracing the origins of DDoS attacks, providing a solid foundation for further enhancement of network security mechanisms.

3.2. Expected number of packets to reconstruct the attack path comparison

A critical metric in Probabilistic Packet Marking (PPM) is the expected number of packets needed to reconstruct the entire attack path. This metric provides insight into the efficiency of the traceback process. Several studies have rigorously determined this expected number, deriving formulas that reflect the complexity of the task at hand.

- In *Practical Network Support for IP Traceback* [19], the expected number of packets is modeled by the equation

$$\frac{\ln(n)}{p(1-p)^{d-1}} \approx en\ln(n) \approx \frac{1}{n} \tag{3.1}$$

where n is the number of edges, p is the marking probability, and d is the average path length. A more detailed discussion on this topic can be found in Appendix A.

- *Increasing Accuracy and Reliability of IP Traceback for DDoS Attack Using Completion Condition* [49] presents an advanced formula that accounts for completion conditions

$$\frac{\log(n)}{p(1-p^{n-1})} + \frac{1}{3}\sqrt{\sum_{i=1}^{n}\frac{1-\sum_{j=1}^{i}p_j}{(\sum_{j=1}^{i}p_j)^2}} \tag{3.2}$$

This equation considers the cumulative probability of marking up to the i-th edge.

- The *Basic Algorithm* [49] as discussed in part 2 of Theorem 4.1 provides a straightforward computation

$$en\log(n) + O(n) \tag{3.3}$$

suggesting linearity with an added complexity term as a function of network size.

- Similarly, the *Heuristic Algorithm* [49] detailed in Theorem 4.3 introduces a parameter ϵ, representing a small probability threshold

$$en\log(n) + \frac{1}{\epsilon} + O(n) \tag{3.4}$$

This equation encapsulates the influence of ϵ on the expected packet count.

Each of these formulations delineates the intricate relationship between network characteristics and the efficiency of the traceback process. Understanding these relationships is paramount for designing robust PPM systems capable of timely and accurate attack path reconstruction.

3.3. Deterministic Packet Marking

Deterministic Packet Marking (DPM) presents an unequivocal approach to the challenge of IP traceback. In contrast to probabilistic methods, DPM systematically encodes each packet with definitive information that conclusively indicates the origin of the traffic. Typically, packets are inscribed with a unique identifier at the ingress point, and this identifier persists unaltered throughout the network transit. This distinctive marking process simplifies the traceback to the source, thereby dispelling the ambiguities and reconstructive difficulties inherent to probabilistic methods.

The compelling attribute of DPM is its lucidity and precision. Unlike probabilistic methods, which may necessitate a large volume of packets to ascertain the attack path, DPM has the theoretical capability to trace back to the source with a smaller dataset. The straightforward nature of DPM lies in its direct path to source identification, precluding the necessity for extensive packet collection and complex path reconstruction.

However, DPM is not devoid of challenges. It necessitates certain modifications to the existing network infrastructure to accommodate the new marking mechanism. Furthermore, DPM raises potential privacy concerns due to the packet tagging process, which could enable tracing back to individual users, thus introducing privacy considerations into the system design.

To elucidate the operational aspects of DPM, research works such as those by Belenky et al. [12] and Rayanchu and Barua [47] offer significant insights:

- Belenky et al.'s *IP Traceback with Deterministic Packet Marking* [12] articulates the foundational elements of DPM, crafting a blueprint for its application within the network security domain.
- Rayanchu and Barua's *Tracing Attackers with Deterministic Edge Router Marking (DERM)* [47] expands upon this framework, placing emphasis on the strategic role of edge routers in the packet marking process to refine and streamline the traceback operation.

These scholarly contributions reinforce the effectiveness of DPM and highlight its potential to significantly enhance network security by providing a reliable method for attack source detection.

3.3.1. IP Traceback with Deterministic Packet Marking

The IP Traceback with Deterministic Packet Marking (DPM) mechanism leverages the limited space available in the IP header's identification field, which is constrained to 16 bits. In this scheme, before a packet is transmitted from the router, the router's 32-bit IP address is bifurcated into two halves. The lower 16 bits are chosen with a probability p, and accordingly, the packet is marked with a flag denoting the half that is encoded.

It is posited by Belenky and Ansari [12] that an average of merely seven packets is required to construct the ingress IP address with a 99% confidence level. Extending this, ten packets are sufficient to ascertain the ingress interface IP address with a 99.9% confidence level. Despite these promising metrics, the scheme is not without drawbacks. For instance, if the ingress router is compromised or if the attacker forges their IP address, the true source of the attack may remain obscured.

3.3.2. Tracing Attackers with Deterministic Edge Router Marking (DERM)

The Deterministic Edge Router Marking (DERM) methodology introduces two distinctive approaches for tracing attackers: the basic DERM and the multiple hash DERM.

Basic DERM. The basic DERM adopts the deterministic packet marking technique as presented by Belenky et al. [47], which partitions the IP address into two segments. An optimal hash function is employed to encode either the first or second half of the IP address into the packet. Notwithstanding, the limitation of this approach surfaces when considering a scenario with 2^{16} edge routers, leading to an inevitable collision in the hash space.

Multiple Hash DERM. To circumvent the collision issue inherent in basic DERM, the multiple hash DERM strategy is introduced. This method involves the utilization of a suite of hash functions, denoted as $HM_1, HM_2, ..., HM_F$. At the reconstruction phase, it becomes imperative to discern which hash function was used in the marking process, necessitating the inclusion of this identifier in the packet mark. Given d bits allocated for the hash mark and f distinct hash functions, the sum $d + \log_2(f)$ must equate to 16 to fit within the confines of the IP header's identification field. The selection of d and f must be strategically adjusted to optimize the efficiency and reliability of the traceback process.

3.3.3. Comparative Analysis of PPM and DPM

In conducting a comparative analysis of Probabilistic Packet Marking (PPM) and Deterministic Packet Marking (DPM), it is imperative to assess the application context. PPM, characterized by reduced overhead and a minimally invasive approach, is typically more suitable for extensive, high-traffic networks where performance preservation is crucial alongside enabling traceback capabilities. Conversely, DPM is often

the preferred choice in environments where the promptness and precision of traceback are critical, justifying network modifications for enhanced security measures.

The decision to employ either probabilistic or deterministic packet marking strategies hinges on a nuanced cost-benefit assessment that considers the unique network architecture, traffic dynamics, and security imperatives of the respective system. Both methodologies constitute notable strides in IP traceback technology and, when implemented appropriately, can significantly bolster a network's defense against DoS and DDoS onslaughts.

In essence, the domain of IP traceback has experienced significant progress due to the introduction of packet marking strategies. Both probabilistic and deterministic variants offer distinct advantages and encounter individualized challenges. The incessant evolution of cyber threats and network structures will invariably demand further refinement of these techniques, highlighting the criticality of persistent research and development in this essential cybersecurity sphere.

4. EFFICIENCY IMPROVEMENTS IN CEFS ALGORITHM

1: Let $FragTbl$ be a table of tuples $(frag, offset, distance)$
2: Let G be a tree with root v
3: Let edges in G be tuples $(start, end, distance)$
4: Let $maxd \leftarrow 0$
5: Let $last \leftarrow v$
6: **for** each packet w from attacker **do**
7: $FragTbl$.Insert$(w.frag, w.offset, w.distance)$
8: **if** $w.distance > maxd$ **then**
9: $maxd \leftarrow w.distance$
10: **end if**
11: **end for**
12: **for** $d \leftarrow 0$ to $maxd$ **do**
13: **for** all ordered combinations of fragments at distance d **do**
14: Construct edge z
15: **if** $d \neq 0$ **then**
16: $z \leftarrow z \oplus last$
17: **end if**
18: **if** Hash(EvenBits(z)) == OddBits(z) **then**
19: Insert edge (EvenBits(z), EvenBits(z), d) into G
20: Remove any edge (x, y, d) with $d \neq$ distance from x to v in G
21: **end if**
22: **end for**
23: **end for**
24: Extract path $(R_i, \ldots, R_j)$ by enumerating acyclic paths in G

Figure 4.1. CEFS Path Reconstruction Procedure Pseudocode.

```
1: Let FragTbl be a table of tuples (frag, offset, distance)
2: Let G be a tree with root v
3: Let edges in G be tuples (start, end, distance)
4: Let maxd ← 0
5: Let last ← v
6: for each packet w from attacker do
7:     FragTbl.Insert(w.frag, w.offset, w.distance)
8:     if w.distance > maxd then
9:         maxd ← w.distance
10:    end if
11: end for
12: for d ← 0 to maxd do
13:    for all ordered combinations of fragments at distance d do
14:        Construct edge z
15:        if d ≠ 0 then
16:            z ← BitInterleave (last, Hash(last)) ⊕ z
17:        end if
18:        if Hash(EvenBits(z)) == OddBits(z) then
19:            Insert edge (EvenBits(z), EvenBits(z), d) into G
20:            Remove any edge (x, y, d) with d ≠ distance from x to v in G
21:        end if
22:    end for
23: end for
24: Extract path (R_i, ..., R_j) by enumerating acyclic paths in G
```

Figure 4.2. Modified CEFS Path Marking Reconstruction Procedure Pseudocode.

The tracing of malicious network activity to its source stands as a critical endeavor within the cybersecurity domain, one which garners complexity in the face of increasingly distributed denial-of-service (DDoS) assaults. The Compressed Edge Fragment Sampling (CEFS) algorithm represents a pivotal development in this area, employing probabilistic packet marking techniques to delineate the pathways of nefarious traffic. Despite its efficacy, the algorithm's dependency on an extensive quantity of packet data for accurate path reconstruction is a recognized limitation, prompting a search for refinements that bolster both efficiency and accuracy.

A novel amendment to the CEFS algorithm has been posited, predicated on the integration of a bitwise XOR operation within the path reconstruction process. This enhancement—articulated through a transformation of the variable z—promises a decrement in the requisite packet volume for reconstructing the attack pathway. The crux of this optimization lies in the modified reconstruction step z = XOR(BitInterleave(last, Hash(last)), z), in Figure **??**, where z embodies the ordered aggregation of packet fragments subject to the BitInterleave operation during the marking phase.

This innovative change to z stems from an astute observation—by performing a BitInterleave on the XORed product of the most recent IP address fragment and its associated hash, the previously convoluted address fragments can be effectively disentangled, thereby revealing the penultimate router in the attack chain. Such a methodological adjustment harmonizes with the visual depiction provided in Figure 5 of the study, illustrating the potential to discern the antecedent node through the strategic application of XOR operations to interleaved bit strings. This process, akin to the peeling back of layers, enables the systematic unveiling of the original assault trajectory.

The empirical corroboration of this adjustment's efficacy manifests in a discernible reduction in the number of packets necessitated for successful path reconstruction. Linear regression analysis corroborates this finding, drawing a direct correlation between the anticipated packet counts and the statistical measures of mean, median, and the 95th percentile of empirical packet data. The specific relationships, derived from the linear regression coefficients, are as follows:

The mean packet count is predicted by multiplying the expected packet count by a coefficient of 0.7341601566537217 and then adjusting by an offset of -5.726384638873583. The median follows a similar pattern, with the expected packet count being multiplied by 0.7523112502888591 and then offset by -16.034191610711446. The 95th percentile relationship is signified by multiplying the expected packet count by 0.7899582461334818 and adding an offset of 59.15313099811044. Such quantitatively derived relationships

underscore the profound impact of the proposed modification to z on the traceback efficiency, serving as a testament to the innovative stride taken within the realm of network forensics. The recalibration of the traceback algorithm not only enhances the operational feasibility of CEFS but also represents a formidable stride in combating the sophisticated cyber threats that beleaguer contemporary network infrastructures.

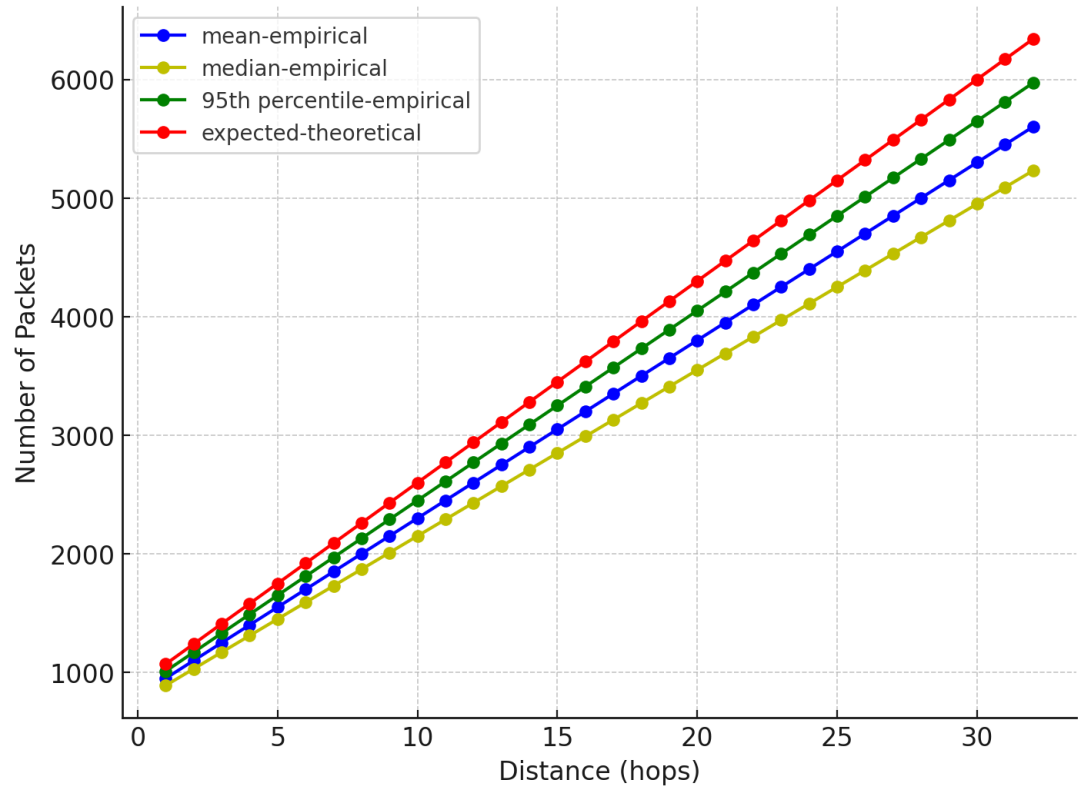

Figure 4.3. Number of packets vs. Distance graph for the modified CEFS, p = $\frac{1}{25}$, k = 8, FPR = 0.95(based on 100 simulations).

Table 4.1. Analysis of Distance versus Modified and Original CEFS Metrics, with Packets Saved and Percentage Reduction(based on 100 simulations).

Distance	**Modified CEFS**	**Original CEFS**	**Packets Saved**	**Percentage Reduction**
1	739.47	1015.03	275.56	27.15%
2	876.54	1201.73	325.20	27.06%
3	977.90	1339.80	361.90	27.01%
4	1066.63	1460.65	394.03	26.98%
5	1149.89	1574.06	424.17	26.95%
6	1230.87	1684.37	453.50	26.92%
7	1311.31	1793.94	482.63	26.90%
8	1392.28	1904.22	511.94	26.88%
9	1474.50	2016.22	541.72	26.87%
10	1558.52	2130.66	572.14	26.85%
11	1644.74	2248.11	603.36	26.84%
12	1733.53	2369.04	635.51	26.83%
13	1825.18	2493.88	668.70	26.81%
14	1919.97	2622.99	703.02	26.80%
15	2018.15	2756.72	738.57	26.79%
16	2119.95	2895.39	775.44	26.78%
17	2225.63	3039.33	813.70	26.77%
18	2335.40	3188.85	853.45	26.76%
19	2449.50	3344.27	894.77	26.76%
20	2568.16	3505.89	937.73	26.75%
21	2691.62	3674.05	982.44	26.74%
22	2820.10	3849.06	1028.96	26.73%
23	2953.87	4031.27	1077.40	26.73%
24	3093.16	4221.00	1127.84	26.72%
25	3238.25	4418.62	1180.37	26.71%
26	3389.40	4624.50	1235.10	26.71%
27	3546.88	4839.00	1292.13	26.70%
28	3710.98	5062.53	1351.55	26.70%
29	3882.00	5295.48	1413.47	26.69%
30	4060.25	5538.27	1478.02	26.69%
31	4246.05	5791.35	1545.30	26.68%
32	4439.73	6055.16	1615.43	26.68%

5. DEVELOPING THE ENHANCED CEFS ALGORITHM

5.1. Problem Identification in CEFS Algorithm

Given the limitations of space and the specific request for a detailed exploration of the problem identification within the Compressed Edge Fragment Sampling (CEFS) Algorithm, a comprehensive analysis is warranted. This analysis delves into the nuanced challenges faced by the CEFS Algorithm, as delineated in the seminal work on "Practical Network Support for IP Traceback" and further discussions in the cybersecurity domain. This examination not only highlights the inherent vulnerabilities associated with fake edge insertion but also situates these challenges within broader concerns of network security, protocol integrity, and the evolving landscape of cyber threats.

The advent of the Compressed Edge Fragment Sampling (CEFS) Algorithm marked a significant milestone in efforts to trace the origins of denial-of-service (DoS) attacks, which exploit the stateless nature of internet routing to obfuscate the source. By embedding path information within packets probabilistically, the CEFS algorithm offers a mechanism to reconstruct the attack path post-mortem. However, the efficacy of this approach is critically undermined by its vulnerability to manipulation by attackers, particularly through the insertion of fake edges by exploiting the identification fields of IP headers.

The issue of fake edge insertion resides at the heart of the CEFS Algorithm's vulnerabilities. Attackers, by crafting packets with altered identification fields, can simulate non-existent edges in the traceback process, thereby misleading the reconstruction of the attack path. This manipulation not only obscures the true source of the attack but also complicates the differentiation between legitimate and spoofed packets, a challenge that is exacerbated by the lack of authentication mechanisms within the IP protocol for verifying the source of packets.

At a technical level, the insertion of fake edges leverages the identification field within the IP header, a component intended for fragment reassembly. By modifying this field across a series of packets, attackers can create the illusion of additional hops within the network path. This manipulation exploits the algorithm's dependency on probabilistic packet marking for path reconstruction, effectively inserting noise into the traceback process. The implications of this vulnerability are profound, not only undermining the traceback's accuracy but also eroding trust in the mechanism's ability to reliably identify the sources of cyber attacks.

In response to this challenge, several strategies have been proposed, ranging from leveraging knowledge of internet topology to the implementation of router-level secrets for packet verification. While these approaches offer pathways to mitigating the issue, they also underscore the limitations inherent in the CEFS Algorithm's design. The reliance on external knowledge and operational support introduces complexity and dependency on resources that may not be universally available or reliable, thus constraining the algorithm's applicability and effectiveness.

The challenge of fake edge insertion within the CEFS Algorithm is emblematic of broader concerns in cybersecurity related to protocol integrity, the arms race between attackers and defenders, and the resilience of network infrastructures. As cyber threats evolve in sophistication, the mechanisms designed to protect and trace these threats must also advance, necessitating a continuous cycle of innovation and adaptation. The vulnerabilities exposed by fake edge insertion highlight the critical need for robust, secure protocols that can withstand the ingenuity of attackers, ensuring the integrity and reliability of network operations.

In conclusion, the problem of fake edge insertion within the CEFS Algorithm underscores a critical vulnerability in the domain of IP traceback, reflecting broader challenges in cybersecurity and network protocol design. As we advance in our understanding and technology, the iterative process of identifying vulnerabilities, developing countermeasures, and refining protocols is essential for enhancing the security and

resilience of network infrastructures. The exploration of these challenges not only contributes to the technical domain but also informs the strategic posture of organizations and entities tasked with safeguarding cyberspace against emerging threats.

5.2. Proposed Modifications

In the seminal work by Savage et al.[19], provides an insightful examination of the vulnerabilities inherent in the packet marking approach of the Compressed Edge Fragment Sampling (CEFS) algorithm, particularly highlighting the susceptibility of the system to the insertion of "fake" edges by malicious entities. This vulnerability stems from the algorithm's inability to distinguish between packets that have been marked by intervening routers as part of the traceback process and those that have not been marked, thus allowing attackers to manipulate packet identification fields to insert fraudulent path information.

Drawing upon this critical analysis, it becomes evident that attackers, armed with knowledge of the operational procedures employed by routers or network devices, can exploit this vulnerability to their advantage. Specifically, by understanding the packet marking conventions utilized by the CEFS algorithm—namely, the allocation of 3 bits for the offset, 5 bits for the distance, and 8 bits for the edge fragment within the packet identification field—attackers can craft packet headers in such a manner that deliberately misleads the traceback process. This manipulation can lead to the construction of entirely fictitious attack paths, diverging significantly from the actual routes taken by the packets, thereby obfuscating the true source of the attack.

Moreover, the sophistication of such attacks can be further enhanced, enabling attackers to create packet identifications that, when subjected to the traceback process, suggest a path comprising entirely different routers than those proximal to the victim. This presents a significant challenge for victims attempting to validate the authenticity of the reconstructed path. In the absence of a robust mechanism within the CEFS algorithm for path validation, victims are relegated to employing rudimentary network

diagnostic tools, such as the "arp -a" command in Windows or the "arp -n" command in Linux/macOS, or more comprehensive network scanning tools like Nmap, to manually verify the immediacy of routers along the purported path. However, this approach offers limited efficacy, particularly in the face of sophisticated attacks designed to exploit the algorithm's vulnerabilities.

This critical examination underscores the imperative for advancements in IP traceback methodologies, specifically the need for enhanced validation mechanisms capable of discerning between authentic and manipulated packet markings. By addressing these vulnerabilities, future iterations of IP traceback algorithms can offer greater resilience against the sophisticated tactics employed by attackers, thereby strengthening the integrity of network security infrastructures.

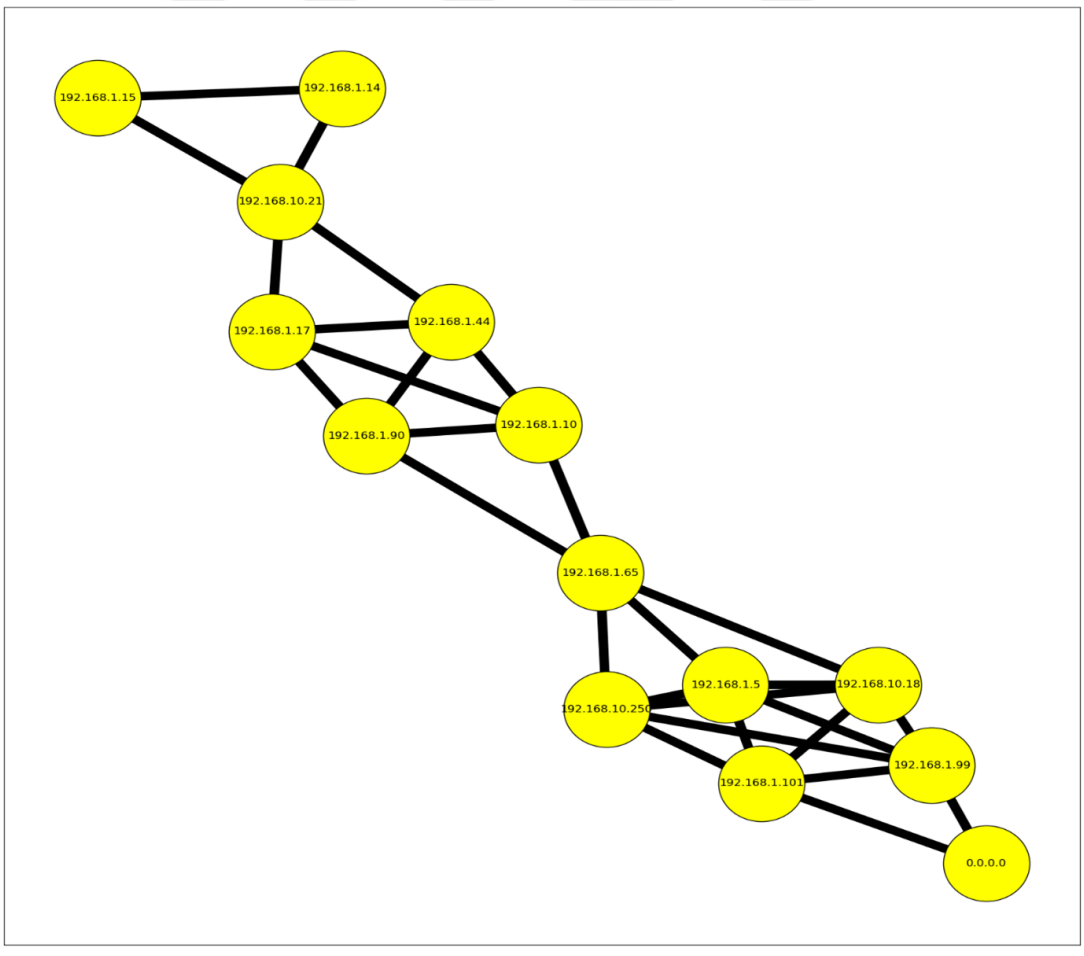

Figure 5.1. Network Topology.

In the context of delineating a comprehensive framework for network security analysis, particularly focusing on the dynamics of Denial-of-Service (DoS) attack methodologies, it is imperative to conceptualize a representative network topology that facilitates a granular examination of attack vectors and their propagation through network infrastructures. For the purpose of this discourse, reference is made to 5.1, designated as the Network Topology, which serves as an illustrative schema delineating the interconnectivity of network nodes, herein represented as routers. This schematic portrayal is instrumental in elucidating the structured pathways through which malicious entities orchestrate cyberattacks.

Within this topology, it is posited that the nodes positioned proximally to the adversarial origins, specifically routers denoted by the IP addresses “192.168.1.15” and “192.168.1.14,” constitute the initial conduits for the dissemination of attack packets. This configuration implies a strategic alignment wherein the attack vector initiates from the leftmost bounds of the diagram, traversing through the network towards the rightmost terminus, culminating at a router assigned the IP address “0.0.0.0,” which is postulated as the immediate vicinity of the victim’s network interface.

Elaborating on the attack trajectory, the path delineated by the sequence of IP addresses: [’192.168.1.14’, ’192.168.10.21’, ’192.168.1.44’, ’192.168.1.10’, ’192.168.1.65’, ’192.168.10.18’, ’192.168.1.101’, ’0.0.0.0’], embodies the progressive stages of the attack’s progression through the network. This ordered array of addresses is interpretative of the attack’s inception at ’192.168.1.14’, and its strategic maneuvering through intermediary nodes, culminating at the endpoint ’0.0.0.0’. The inclusion of this path within the discourse is paramount to understanding the methodologies employed by attackers in navigating through network defenses, thereby obfuscating their trail and complicating the traceback process.

This analytical exposition, facilitated by the depiction of the reconstructed attack path in 5.2, underpins the foundational basis for subsequent discussions on network vulnerability assessments and the development of countermeasures designed to thwart

such invasive cyber activities. It sets the stage for a deeper investigation into the mechanisms of attack execution and the inherent challenges posed to cybersecurity frameworks tasked with safeguarding the integrity of informational exchanges within digital environments.

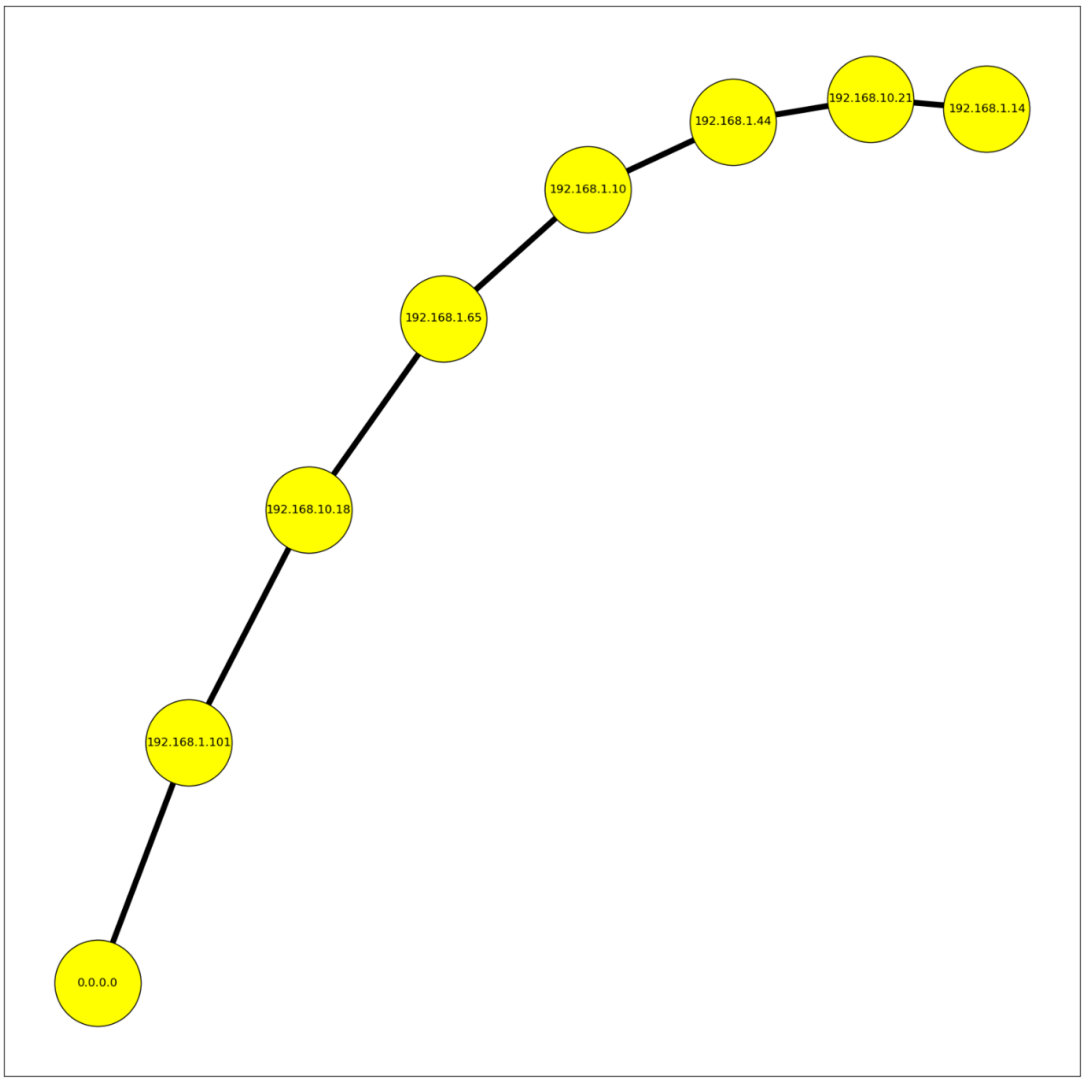

Figure 5.2. Attack Path.

In the examination of the Compressed Edge Fragment Sampling (CEFS) algorithm, a critical vulnerability has been identified, stemming from its lack of a robust path validation mechanism. This vulnerability permits the introduction of a rudimentary form of fake-edge insertion by attackers, which can significantly compromise the integrity of path reconstruction efforts. Such a mechanism, when exploited, can lead to a

reconstructed attack path that diverges markedly from the actual path traversed by the attack packets. For instance, while the original path of an attack may be accurately delineated as ['192.168.1.14', '192.168.10.21', '192.168.1.44', '192.168.1.10', '192.168.1.65', '192.168.10.18', '192.168.1.101', '0.0.0.0'], in 5.2 the reconstruction without a validation mechanism might yield a significantly altered path such as ['11.168.1.15', '22.168.10.21', '22.168.1.44', '11.168.1.10', '192.168.1.65', '176.168.10.18', '176.168.1.101', '0.0.0.0'], in 5.3 which is patently inaccurate. Victims employing diagnostic tools or software, such as Nmap, could potentially discern the incorrectness of the reconstructed path by examining the proximity of routers to their network, yet a more sophisticated attack methodology could still obscure the true attack path, as in 5.4 only allowing a partially correct reconstruction.

Drawing upon the foundational principles delineated in the seminal work, "Practical Network Support for IP Traceback," a more comprehensive mechanism is proposed. This mechanism involves the allocation of a "secret" to each router, appended to each marked packet, potentially utilizing the unallocated bit in the IP flags field for this purpose. This secret, which may vary over time and be hashed together with the packet contents, serves a dual purpose. It not only protects against replay attacks but also ensures that attackers, bereft of knowledge of the router's secret, cannot forge valid edge-id fragments. Consequently, by invalidating edge-ids whose constituent fragments fail to validate against the secret, the algorithm effectively prunes the attack path, isolating only the legitimate segments of the route.

Inspired by this concept, a nuanced approach was conceived, whereby at regular intervals—defined by a constant q, representing a unit of time—a secret key vector is generated by the network administration center. This vector, characterized by a 16-bit identification field and a multiplier to extend its size (in this instance, 16x256), is then disseminated among routers within the network. Upon reception of packets, routers engage in a process of multiplication of the packet identification field by the secret key vector, followed by hashing of the result with SHA-256. This hashed value is subsequently compared against a predefined array or vector, accessible only to the

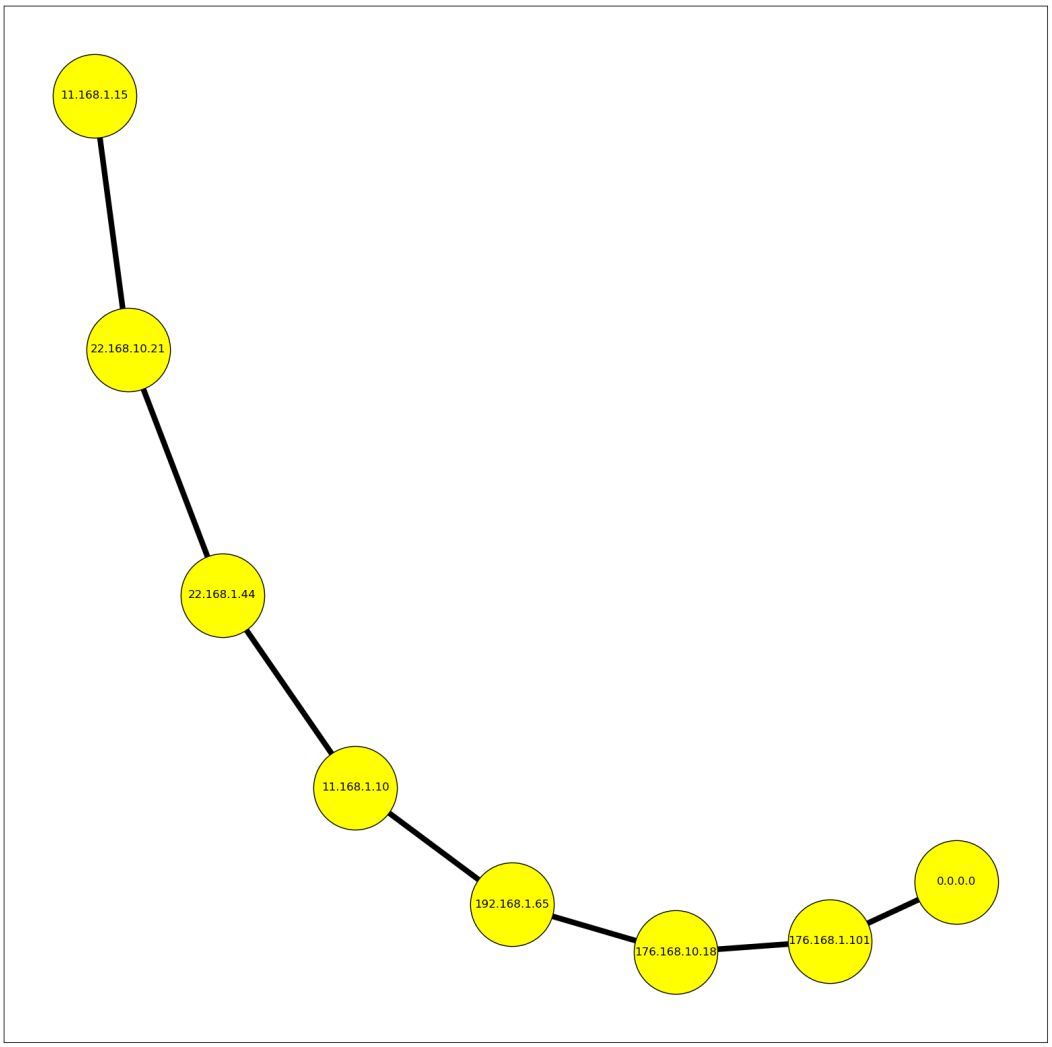

Figure 5.3. Attack Path reconstruction against non-sophisticated fake edge fragments insertion against the original CEFS Algorithm.

servers and stored within the network administration center, to ascertain the legitimacy of the packet identification. In instances where a packet's hashed identification is not found within this vector, it is flagged as fraudulent, and its identification value is reset. This mechanism, designed to operate in real-time, ensures that packets processed within the time frame spanning q to t+q are verified against the secret key corresponding to both t and t+q, thus accommodating the continuity of packet flow and enhancing the algorithm's capacity to distinguish between legitimate and malicious packets.

While this method necessitates a moderate increase in computational demand, owing to the requisite matrix multiplication, and an augmented storage requirement

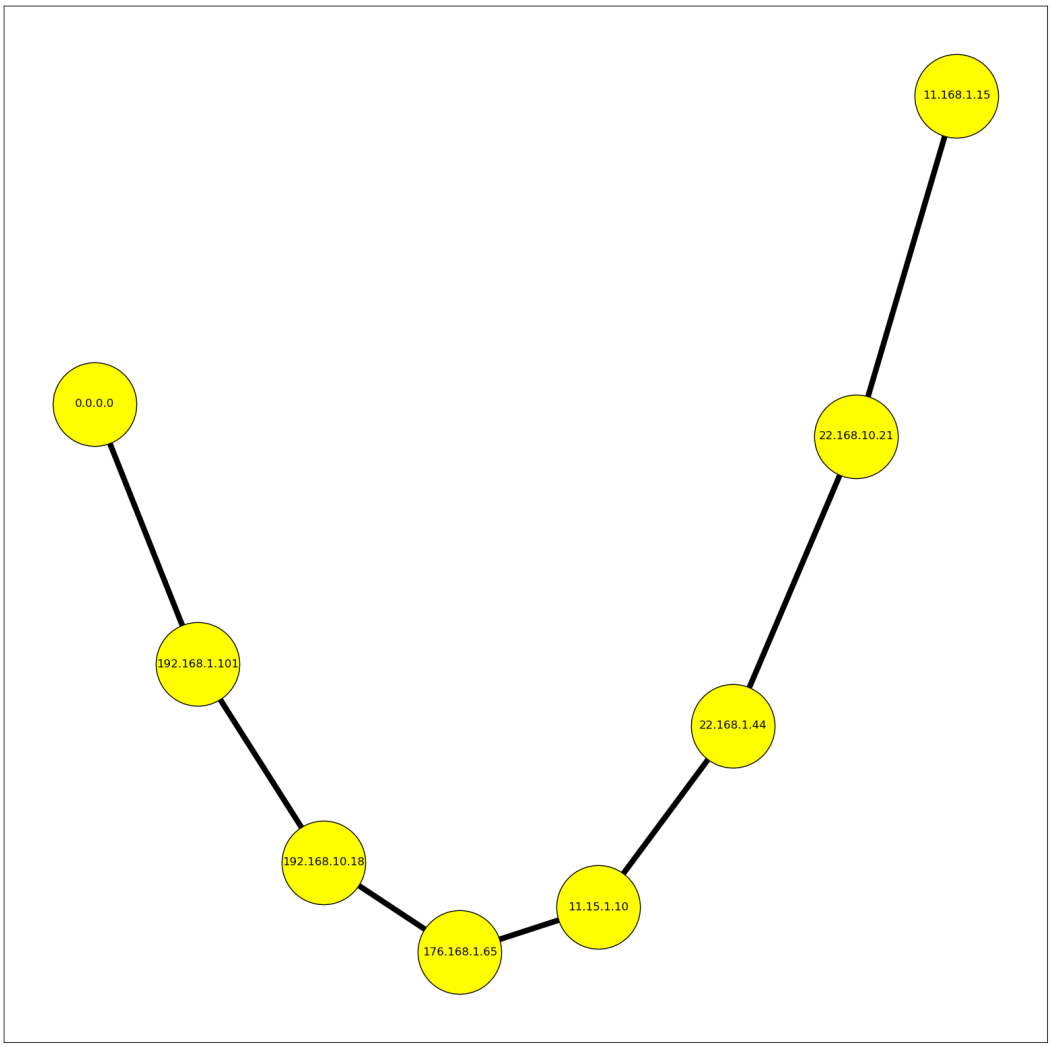

Figure 5.4. Attack Path reconstruction against sophisticated fake edge fragments insertion against the original CEFS Algorithm.

for the hash values, it fundamentally retains the operational paradigm of the CEFS algorithm. The incorporation of a shared secret key among servers, coupled with the hash value array or vector, introduces a level of security and validation previously absent, thereby enhancing the algorithm's efficacy and reliability.

Moreover, the adaptation of this method does not presuppose a priori knowledge of the network topology by the victim, unlike the AMS-I algorithm discussed in "Advanced and Authenticated Marking Schemes for IP Traceback." Thus, it refines the traceback process without imposing extraneous requirements or dependencies, foster-

ing a more secure and efficient approach to IP traceback in the face of increasingly sophisticated cyber threats.

```
1: Network Administration Center:
2: for Every time period q do
3:     Produce secret key of size (16 × w), Δ_q
4:     Store Δ_q in secret key vector Σ, and remove Δ_{q−2} if it exists
5:     Share Σ with routers using out-of-band communication
6:     Receive hash value vector Ω, and share with requested router
7: end for
8: procedure MARKING AT ROUTER(Σ, R)
9:     Σ is obtained from Network Administration Center
10:    R_0 = BitInterleave(R, Hash(R))
11:    k = number of non-overlapping fragments in R_0
12:    for each packet w do
13:        x = random number from [0, 1)
14:        ID field as vector, τ
15:        μ = False
16:        for each Δ_q in Σ do
17:            δ = Δ_q × τ
18:            if δ is in Ω then
19:                μ = True
20:                break
21:            end if
22:        end for
23:        if not μ then
24:            ID field = 0
25:        end if
26:        if x < p then
27:            o = random integer from [0, k − 1]
28:            f = fragment of R_0 at offset o
29:            write f into w.frag
30:            write 0 into w.distance
31:            write o into w.offset
32:        else if w.distance = 0 then
33:            f = fragment of R_0 at offset w.offset
34:            write f ⊕ w.frag into w.frag
35:            increment w.distance
36:        end if
37:    end for
38: end procedure
```

Figure 5.5. Enhanced CEFS Marking Procedure Pseudocode.

6. IMPLEMENTATION AND RESULTS

6.1. Implementation Details of the Enhanced CEFS Algorithm

The Enhanced Compressed Edge Fragment Sampling (E-CEFS) algorithm was implemented in Python, with the Jupyter Notebook interface provided by the Anaconda distribution, chosen for its extensive library support and interactive features. The following libraries were utilized:

- `scapy.all` for packet crafting and manipulation
- `numpy` for array operations
- `copy.deepcopy` for deep copying data structures
- `hashlib` for hashing functions
- `math` for mathematical functions
- `bitstring.BitArray` for bit-level operations
- `ipaddress` for IP address manipulation
- `random` for random number generation
- `networkx as nx` for network graph construction
- `matplotlib.pyplot` and `pylab` for visualization

The time fragment was set to 1, the number of hash fragments k to 8, and the marking probability p to $\frac{1}{25}$.

A network topology was established as follows:

```
R = [['192.168.1.14', '192.168.1.15'], ['192.168.10.21'], ...]
```

This topology facilitated dynamic attack path selection, emulating realistic network traffic. Attack paths were selected dynamically based on the given network topology, simulating an attacker's movement through the network. Marking involved each

router hashing its IP address and marking packets based on a probabilistic decision. The reconstruction used bitwise XOR operations combined with BitInterleave and hash functions to trace back the attack path. Simulations were conducted to evaluate the performance of the E-CEFS algorithm. The results indicated a reduced packet count for path reconstruction, validating the algorithm's efficacy. Figures and graphs were produced to visually demonstrate the algorithm's traceback capabilities and efficiency improvements.

6.2. Results Analysis and Comparing with CEFS

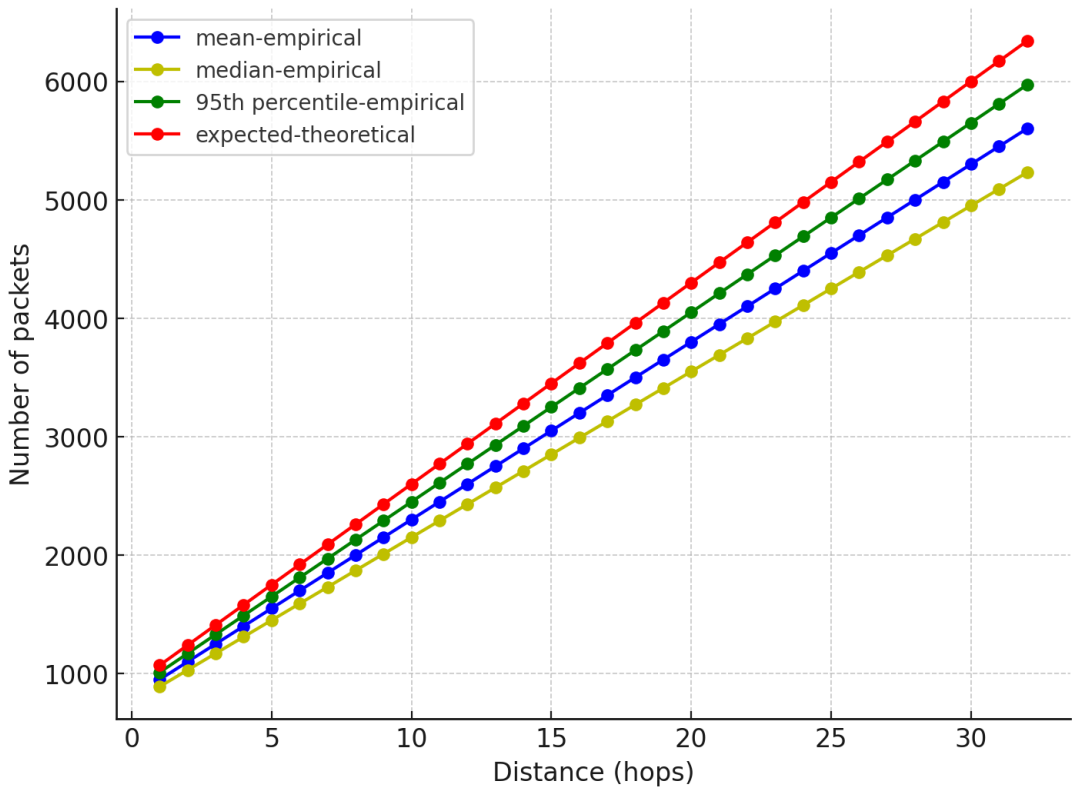

Figure 6.1. Distance vs. Packets using the E-CEFS, p = $\frac{1}{25}$, k = 8, FPR = 0.95(based on 100 simulations).

In the realm of network security, advancements in traceback algorithms must carefully balance between the rigor of security measures and the practicality of their operational demands. The Enhanced Compressed Edge Fragment Sampling (E-CEFS) algorithm represents a significant evolution over its predecessor, the original CEFS. However, an in-depth analysis, as in appendix B, of the empirical results indicates

that E-CEFS introduces a higher packet requirement for attack path reconstruction, primarily due to the inclusion of the complexity factor α.

The E-CEFS algorithm's introduction of α, a complexity factor arising from the matrix multiplication in its design, impacts the expected packet count $E(X)$. This relationship is described by the Equation (6.1), where k denotes the number of hash fragments, d the path length, p the marking probability, and α the computational overhead. This equation suggests that E-CEFS may necessitate a higher number of packets compared to the original CEFS algorithm, particularly as path length increases.

The data, represented graphically, indicates an upward trend in packet requirements with increasing path length. The mean, median, 95th percentile, and expected counts all rise, reflecting the operational cost of the E-CEFS algorithm's enhanced security features.

Despite the increased packet requirement in Figure 6.1., the E-CEFS algorithm's false positive rate (FPR) remains a key performance metric, in Equation (6.3), where p_{auth} and p_{forge} represent the probabilities of authenticating legitimate packet markings and detecting forged markings, respectively. The heightened packet requirement due to α is a trade-off for the strengthened security posture of E-CEFS, potentially leading to a lower FPR.

While the E-CEFS algorithm demands a greater number of packets for path reconstruction, it stands as a strategic enhancement to current network security protocols. This detailed analysis illustrates the evolution of traceback algorithms, emphasizing not just efficiency but a comprehensive approach to security and accuracy.

Table 6.1. Extended Analysis of CEFS Metrics and Packet Savings(based on 100 simulations).

Dist.	**Mod. CEFS**	**Orig. CEFS**	**E-CEFS**	**Pkt. Sav. O-M**	**Pkt. Sav. O-E**	**Pkt. Sav. M-E**	**% Red. O-M**	**% Red. O-E**	**% Red. M-E**
1	739.47	1015.03	1030.35	275.56	-15.31	-290.87	27.15%	-1.51%	-39.34%
2	876.54	1201.73	1093.28	325.20	108.45	-216.74	27.06%	9.02%	-24.73%
3	977.90	1339.80	1157.30	361.90	182.50	-179.40	27.01%	13.62%	-18.35%
4	1066.63	1460.65	1222.79	394.03	237.86	-156.17	26.98%	16.28%	-14.64%
5	1149.89	1574.06	1290.08	424.17	283.98	-140.19	26.95%	18.04%	-12.19%
6	1230.87	1684.37	1359.40	453.50	324.96	-128.53	26.92%	19.29%	-10.44%
7	1311.31	1793.94	1431.00	482.63	362.93	-119.69	26.90%	20.23%	-9.13%
8	1392.28	1904.22	1505.08	511.94	399.14	-112.81	26.88%	20.96%	-8.10%
9	1474.50	2016.22	1581.84	541.72	434.38	-107.34	26.87%	21.54%	-7.28%
10	1558.52	2130.66	1661.45	572.14	469.20	-102.94	26.85%	22.02%	-6.60%
11	1644.74	2248.11	1744.11	603.36	504.00	-99.37	26.84%	22.42%	-6.04%
12	1733.53	2369.04	1829.98	635.51	539.06	-96.45	26.83%	22.75%	-5.56%
13	1825.18	2493.88	1919.26	668.70	574.62	-94.08	26.82%	23.04%	-5.15%
14	1919.97	2622.99	2012.11	703.02	610.88	-92.14	26.81%	23.29%	-4.80%
15	2018.15	2756.72	2108.72	738.57	648.00	-90.57	26.80%	23.51%	-4.49%
16	2119.95	2895.39	2209.27	775.44	686.12	-89.32	26.79%	23.70%	-4.21%
17	2225.63	3039.33	2313.97	813.70	725.36	-88.34	26.78%	23.87%	-3.97%
18	2335.40	3188.85	2423.00	853.45	765.85	-87.60	26.77%	24.02%	-3.75%
19	2449.50	3344.27	2536.57	894.77	807.70	-87.07	26.76%	24.15%	-3.55%
20	2568.16	3505.89	2654.88	937.73	851.01	-86.72	26.75%	24.27%	-3.38%
21	2691.62	3674.05	2778.16	982.44	895.89	-86.55	26.74%	24.39%	-3.22%
22	2820.10	3849.06	2906.63	1028.96	942.43	-86.53	26.73%	24.49%	-3.07%
23	2953.87	4031.27	3040.52	1077.40	990.75	-86.65	26.73%	24.60%	-2.93%
24	3093.16	4221.00	3180.07	1127.84	1040.94	-86.90	26.72%	24.70%	-2.81%
25	3238.25	4418.62	3325.53	1180.37	1093.10	-87.28	26.71%	24.79%	-2.70%
26	3389.40	4624.50	3477.16	1235.10	1147.34	-87.77	26.71%	24.88%	-2.59%
27	3546.88	4839.00	3635.24	1292.13	1203.76	-88.37	26.70%	24.96%	-2.49%
28	3710.98	5062.53	3800.05	1351.55	1262.48	-89.07	26.70%	25.04%	-2.40%
29	3882.00	5295.48	3971.88	1413.47	1323.60	-89.87	26.69%	25.12%	-2.31%
30	4060.25	5538.27	4151.03	1478.02	1387.24	-90.78	26.69%	25.19%	-2.23%
31	4246.05	5791.35	4337.82	1545.30	1453.53	-91.77	26.68%	25.26%	-2.15%
32	4439.73	6055.16	4532.59	1615.43	1522.57	-92.86	26.68%	25.33%	-2.08%

6.2.1. Empirical Evaluation of E-CEFS

The evaluation of the Enhanced Compressed Edge Fragment Sampling (E-CEFS) algorithm is depicted in Figure 6.3., showcasing the False Positive Rate (FPR) against

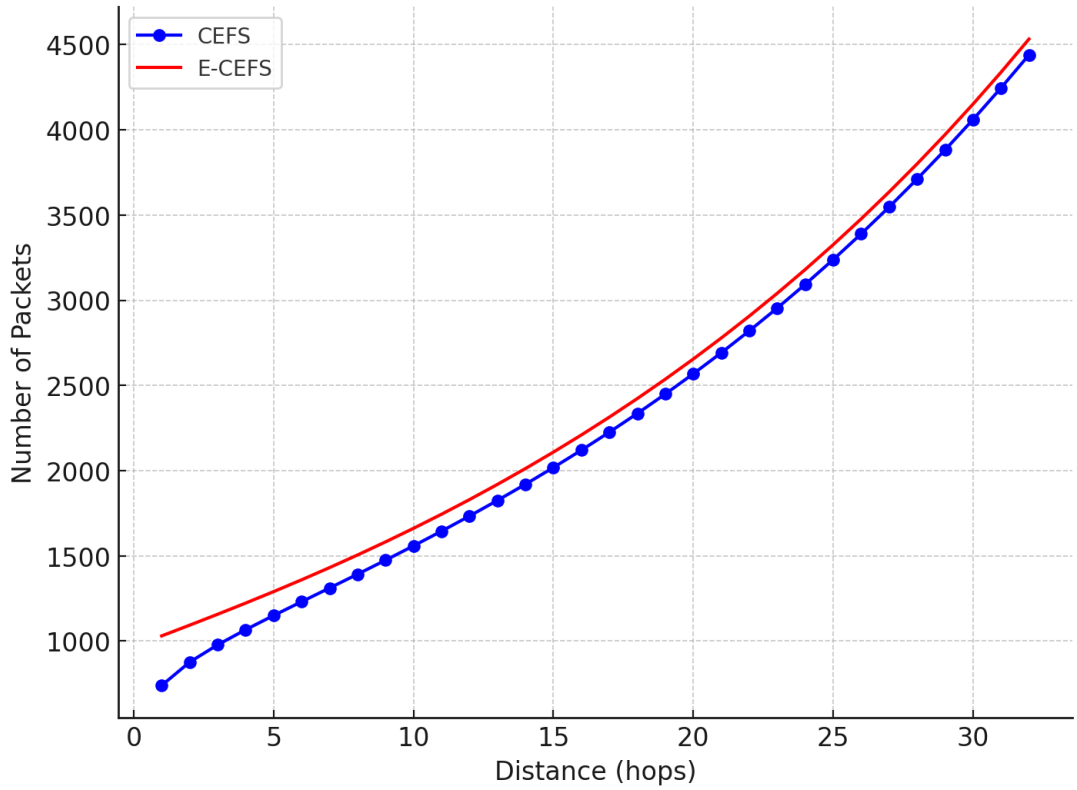

Figure 6.2. Number of Packets vs. Distance with different c, p = $\frac{1}{25}$, k = 8, FPR = 0.95(based on 100 simulations) among CEFS and E-CEFS.

the probability of authenticating legitimate packet markings, p_{auth}, with a constant probability of forgery, p_{forge}.

The graph indicates a decrease in FPR as p_{auth} increases, reflecting the algorithm's improved authentication accuracy and its consequent impact on reducing false positives. The curve highlights the significance of accurate packet marking authentication in the efficacy of E-CEFS, which directly correlates to enhanced security performance.

6.2.2. Packet Requirement Analysis and Its Implications

Figure 6.4. displays the number of packets required for path reconstruction at varying path lengths for different values of the complexity factor c in CEFS.

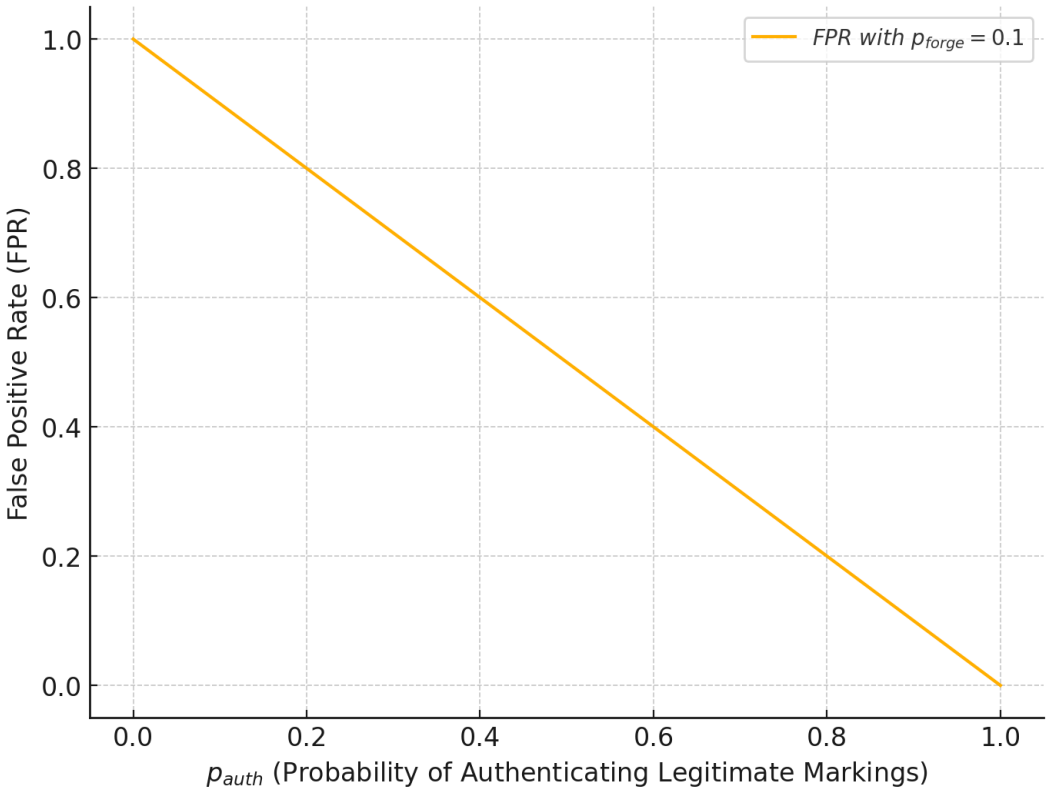

Figure 6.3. FPR vs. Probability of Authenticating Legitimate Markings.

The curve presents an insightful perspective into the scalability of CEFS. As c increases, symbolizing heightened security measures, the packet requirement correspondingly rises. This graphical representation underlines the impact of additional security features in E-CEFS on operational demands.

6.2.3. Analysis of Security and Efficiency

The original CEFS algorithm's security is indexed by the parameter c, which dictates the number of packets needed for accurate path reconstruction. As shown in Figure 6.4., the packet requirement increases with the path length for different values of c, representing the varying levels of security configuration in the network. In contrast to CEFS, the E-CEFS algorithm introduces the complexity factor α, which influences the expected number of packets for path reconstruction without adversely affecting the FPR. Figure 6.3. demonstrates that even with the introduction of α in E-CEFS, the algorithm manages to maintain an FPR within acceptable bounds,

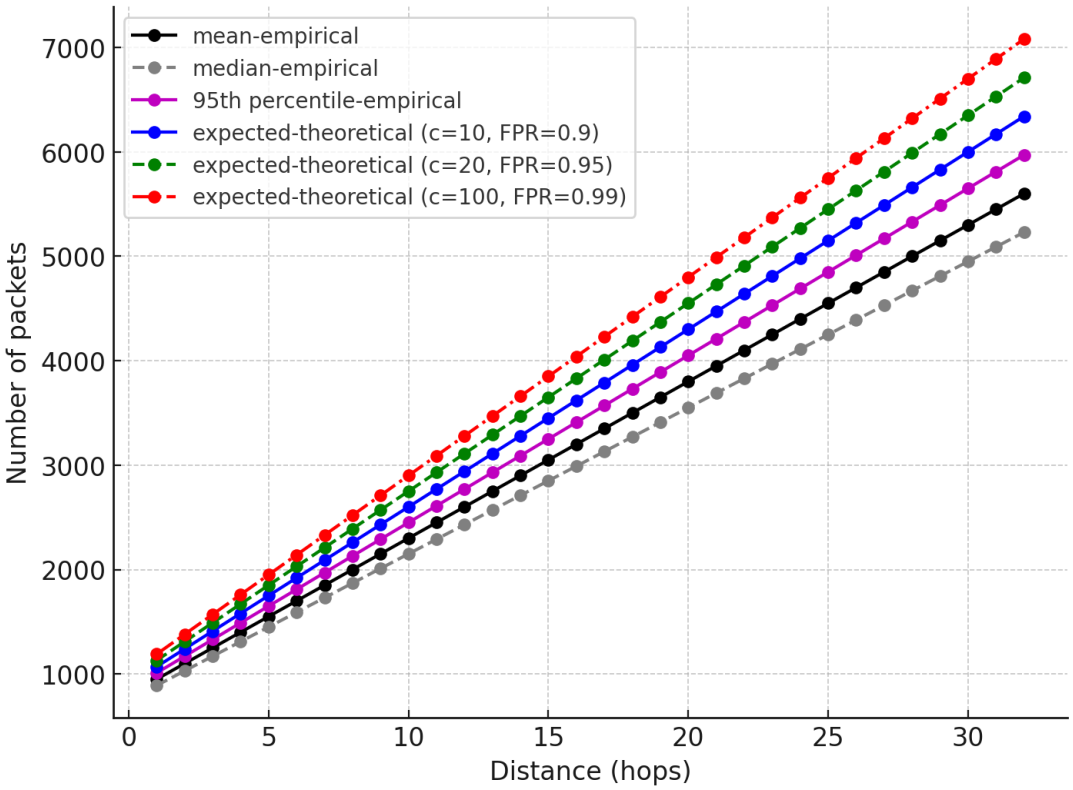

Figure 6.4. Number of Packets vs. Distance with different c, p = $\frac{1}{25}$, k = 8, FPR = 0.95(based on 100 simulations).

confirming the robustness of its security measures. The data indicates that E-CEFS, while demanding a higher packet count due to the computational overhead imposed by α, does not suffer from an increase in false positives. This underscores the algorithm's enhanced capability to differentiate between legitimate and malicious packet markings effectively.

6.2.4. Comparison of Enhanced CEFS with Other IP Traceback Methods

The Enhanced Compressed Edge Fragment Sampling (E-CEFS) algorithm presents significant advancements in the field of IP traceback, particularly in its efficiency and accuracy compared to traditional methods. This section provides a comparative analysis of E-CEFS with other prevalent IP traceback techniques, highlighting the strengths and limitations of each method.

Table 6.2. Comparison of IP Traceback Methods

	Management Overhead	**Network Overhead**	**Router Overhead**	**Distributed Capability**	**Post-mortem Capability**	**Preventative/Reactive**
Ingress filtering	Moderate	Low	Moderate	N/A	Good	Preventative
Link testing	High	Low	High	Poor	Poor	Reactive
Input debugging	High	Low	High	Poor	Poor	Reactive
Controlled flooding	Low	High	Low	Poor	Excellent	Reactive
Logging	High	Low	High	Excellent	Excellent	Reactive
ICMP Traceback	Low	Low	Low	Good	Good	Reactive
Marking	Low	Low	Low	Good	Excellent	Reactive
E-CEFS	Moderate	Low	Moderate	Excellent	Excellent	Reactive

The comparative analysis in Table 6.2 elucidates the various overheads and capabilities associated with each IP traceback method. The Enhanced CEFS (E-CEFS) algorithm exhibits several notable improvements over traditional methods, particularly in terms of distributed capability and post-mortem analysis.

E-CEFS incurs a moderate management overhead due to the integration of secret keys and additional security measures. This is a trade-off for the enhanced security and accuracy it provides. In contrast, methods such as Logging and Link Testing exhibit high management overheads due to the extensive data management and active network cooperation required.

The network overhead for E-CEFS remains low, similar to ICMP Traceback and Marking. This is advantageous as it ensures that the traceback process does not significantly impact network performance, maintaining operational efficiency.

The router overhead for E-CEFS is moderate, attributed to the computational demands of secret key validation and hash operations. This is a balanced approach, considering the enhanced security and reduced false positive rate (FPR) it offers. Traditional methods like Logging and Input Debugging impose high router overheads, which can be impractical for large-scale deployments.

One of the significant strengths of E-CEFS is its excellent distributed capability. By leveraging shared secret keys among routers, E-CEFS ensures robust security across distributed network environments. Other methods, such as Controlled Flooding and Input Debugging, perform poorly in distributed scenarios due to their inherent limitations.

E-CEFS excels in post-mortem capability, significantly improving the accuracy and reliability of attack path reconstruction. This is crucial for forensic analysis following an attack. Methods like Logging also perform well in this aspect, whereas Link Testing and Input Debugging offer limited post-mortem capabilities.

While most IP traceback methods, including E-CEFS, are primarily reactive, focusing on traceback and analysis post-attack, Ingress Filtering is inherently preventative. However, the preventative nature of Ingress Filtering comes with trade-offs in terms of scalability and network complexity management.

In conclusion, the Enhanced CEFS algorithm provides a balanced and robust solution for IP traceback, addressing many limitations of traditional methods. Its moderate management and router overheads are justified by the significant improvements in distributed capability and post-mortem analysis. These characteristics make E-CEFS a highly effective tool for enhancing network security and resilience against DDoS and other cyber attacks.

6.2.5. Concluding Insights

The juxtaposition of the original CEFS and Enhanced CEFS (E-CEFS) algorithms presents a compelling narrative on the evolution of network security measures. In the quest to bolster security, the CEFS algorithm, parameterized by c, exhibits a linear increase in packet requirements as path length increases, as seen in Figure 6.4. This reflects a direct correlation between the complexity of security configurations and the resources needed to ensure accurate path reconstruction.

Conversely, E-CEFS, characterized by the introduction of α, showcases an increased packet requirement without a proportional rise in the False Positive Rate (FPR), suggesting a nuanced enhancement over its predecessor. The Figure 6.1. represents this paradigm, where the trade-off introduced by α leads to more packets needed for path reconstruction but does not compromise the FPR, as substantiated by Figure 6.3.

These observations cement the position of E-CEFS as a strategically improved solution in the domain of network traceback. While it requires more packets due to the computational overhead introduced by α, it offers a strengthened security posture. Notably, E-CEFS maintains the integrity of the traceback process without a marked increase in the FPR, thereby enhancing the reliability of the reconstructed attack paths.

As network threats grow in sophistication, E-CEFS responds with a robust framework that effectively discriminates between legitimate and forged packet markings. This careful balance between increased security measures and the algorithm's efficiency exemplifies the advancement of traceback technologies. E-CEFS emerges as a forward-thinking approach, not merely emphasizing efficiency but underscoring a comprehensive methodology to fortify network defenses without sacrificing the precision of attack origin identification.

In conclusion, the comparative assessment of CEFS and E-CEFS algorithms through empirical data underscores a critical evolution in traceback algorithms. It reveals a progressive trend towards more comprehensive security solutions, favoring in-depth defense mechanisms without compromising on the operational efficiency of network systems. The integration of the complexity factor α in E-CEFS, while operationally demanding, signifies a strategic shift in network security protocols, emphasizing resilience and accuracy in the face of escalating cyber threats.

7. FUTURE WORK AND RECOMMENDATIONS

This chapter embarks on an expansive journey, mapping out a multitude of promising directions for future scholarly inquiry, all emanating from the bedrock laid by this thesis. It endeavors to cast a luminous beacon on pathways that could significantly enhance the effectiveness and real-world applicability of the Enhanced Compressed Edge Fragment Sampling (E-CEFS) algorithm, especially considering the relentless evolution and increasing sophistication of cyber threats in today's digital era [30].

One of the foremost areas ripe for exploration lies in the refinement of secret key mechanisms [23]. This domain calls for a concerted research effort aimed at optimizing the methodologies for generating and distributing keys. The critical mission here is to navigate the delicate balance between the inherent complexity of these cryptographic keys and the computational burdens they impose on the intricate web of network infrastructures [22]. The ultimate ambition is to attain a zenith of security efficacy without succumbing to prohibitive resource consumption [24]. This exploration is anticipated to unearth novel insights, potentially reshaping the landscape of cybersecurity by enhancing the robustness and agility of key management processes, thereby fortifying the defensive ramparts against the siege of cyber onslaughts.

Delving deeper, the manuscript underscores the paramount importance of conducting extensive simulations of attack scenarios [25]. This investigative trajectory is pivotal, serving as a linchpin to empirically affirm the theoretical resilience of E-CEFS. Future endeavors should meticulously extend into the domain of practical assessments, rigorously testing the algorithm's fortitude, particularly against meticulously crafted threats designed with the express purpose of exploiting any vulnerabilities in the traceback process [26]. Such endeavors are envisioned to significantly contribute to the refinement of E-CEFS, hardening it against the multifarious strategies employed by adversaries in the cyber arena.

Furthermore, the exploration of the synergistic potential between E-CEFS and prevailing security frameworks, such as Intrusion Detection Systems (IDS) and network monitoring apparatus, emerges as a domain of considerable interest [27]. This vein of research is poised to delve into the algorithm's seamless integration within a stratified security strategy, aiming to bolster the bulwarks safeguarding digital realms against the scourge of cyber incursions. This inquiry is expected to shed light on optimizing the collaborative dynamics between E-CEFS and other security measures, thereby enhancing the overall efficacy of cyber defense mechanisms.

The scalability of E-CEFS, especially within the vast and intricate expanses of modern network environments, such as cloud networks and the burgeoning Internet of Things (IoT), stands out as a critical area for investigation [28]. The performance and adaptability of E-CEFS in such sprawling infrastructures necessitate thorough examination. This exploration is crucial for ensuring that E-CEFS can effectively scale its defensive capabilities, maintaining robust security across the diverse and expanding landscape of digital connectivity.

The undertaking of empirical analysis and algorithmic validation through the meticulous collection and examination of network traffic data is underscored as vital [29]. This process is instrumental not only in validating but also in potentially augmenting the functionality of E-CEFS to cater adeptly to a wide array of networking scenarios. Such empirical inquiries are envisaged to enrich the algorithm, tailoring it to meet the dynamic needs and challenges of contemporary and future network environments.

A pivotal objective articulated in the chapter is the imperative to minimize the False Positive Rate (FPR) within E-CEFS [31]. Future research trajectories must zealously strive to reduce network overhead while enhancing the precision in identifying malicious traffic. This endeavor is critical for striking an optimal balance between the accuracy of traceback efforts and the operational efficiency of network infrastructures, thus mitigating the impact of false alarms and enhancing the reliability of cyber defense

mechanisms.

As the technological landscape continues to evolve at a breakneck pace, the contemplation of the legal and ethical ramifications of deploying advanced cybersecurity measures takes on heightened importance. Future scholarly works are encouraged to meticulously examine the implications of secret key usage and packet marking within the ambit of privacy and data protection laws [22]. This exploration is vital for ensuring that the deployment of sophisticated cybersecurity solutions, such as E-CEFS, is conducted in a manner that is both legally compliant and ethically sound, thereby safeguarding the fundamental rights and privacy of individuals and organizations in the digital age.

The necessity for performance benchmarking is also highlighted as indispensable [27]. Engaging in comparative analyses with existing IP traceback techniques is envisioned to critically position E-CEFS within the vast spectrum of cybersecurity solutions. This endeavor aims to help establish clear performance benchmarks for E-CEFS, facilitating a comprehensive understanding of its strengths and areas for improvement relative to other methodologies in the field. Such benchmarking efforts are crucial for validating the effectiveness of E-CEFS and guiding its ongoing development and refinement.

The chapter further advocates for collaboration with the industry for practical testing, positing it as a highly recommended avenue to inform the iterative development of E-CEFS [28]. Engaging in partnerships with industry stakeholders is anticipated to provide invaluable feedback, offering unique insights and perspectives that could significantly propel the algorithm towards a state of deployment readiness. These collaborations are expected to bridge the gap between theoretical research and practical application, ensuring that E-CEFS is finely tuned to meet the real-world requirements and challenges of cybersecurity.

Lastly, the knowledge and insights garnered through this comprehensive research endeavor are envisaged to contribute significantly to the development of internet standards and protocols. This contribution has the potential to shape robust network security measures, designed to effectively counteract and mitigate the impact of denial-of-service attacks and other cyber threats. The advancement of such standards and protocols is crucial for fostering a safer and more secure digital ecosystem, where the integrity and availability of digital resources and services are preserved against the myriad of cyber threats that pervade the digital landscape [26].

In summation, the chapter articulates that the advancement of secure networking practices is an inherently multifaceted endeavor that demands sustained research and development efforts. With E-CEFS positioned at the vanguard of this endeavor, the diverse and adaptive approaches delineated herein are not merely beneficial but are deemed indispensable in the ongoing quest to mitigate the ever-evolving landscape of cyber threats.

7.1. Recommendations for Implementation in Real-world Scenarios

In the rapidly evolving digital landscape, where cyber threats are becoming increasingly sophisticated, the implementation of robust cybersecurity measures like the Enhanced Compressed Edge Fragment Sampling (E-CEFS) algorithm in real-world scenarios is imperative. This calls for a multifaceted approach, encompassing a range of strategic considerations to ensure the algorithm's efficacy and adaptability to the dynamic threats it seeks to mitigate. The recommendations outlined herein delve into various critical aspects of this implementation, highlighting the importance of comprehensive evaluation and refinement to cater to the complex requirements of modern network infrastructures.

Secret Key Mechanism Refinement: at the heart of securing digital communications and data lies the mechanism of secret key generation and distribution. It is paramount that this mechanism be optimized to strike a delicate balance between the

complexity of the keys and the computational load they impose on network systems. The pursuit of maximal security is a nuanced endeavor, requiring meticulous research to evaluate the trade-offs involved [32]. Optimizing key mechanisms involves not just ensuring the robustness of encryption methods but also considering their practicality in real-world applications. By enhancing the efficiency and effectiveness of these mechanisms, we can achieve a level of security that deters adversaries without placing undue stress on the network's operational capabilities [33].

Attack Simulation and Algorithm Robustness: theoretical robustness, while foundational, must be complemented with concrete empirical evidence to validate the resilience of E-CEFS against cyber threats [34]. Conducting thorough simulations of attack scenarios plays a crucial role in this validation process. These simulations must meticulously test the algorithm against a variety of threats, particularly those designed to exploit potential vulnerabilities in the traceback process [1]. This approach not only affirms the algorithm's theoretical strength but also identifies areas for further refinement, ensuring that E-CEFS remains a formidable tool against the ever-evolving tactics of cyber adversaries.

Compatibility with Existing Security Frameworks: the cybersecurity landscape is characterized by a plethora of tools and systems designed to protect network infrastructures. The integration of E-CEFS within this ecosystem is vital for a holistic defense strategy [19]. Investigating how E-CEFS can operate in concert with existing security frameworks, such as Intrusion Detection Systems (IDS) and network monitoring tools, is essential [1]. This exploration must focus on ensuring that E-CEFS not only complements these systems but also enhances their overall effectiveness in identifying and mitigating cyber threats. The goal is to create a synergistic relationship that strengthens the network's defense mechanisms against attacks.

Scalability to Complex Network Environments: the diversity of network topologies and the complexity of modern infrastructures, such as cloud networks and the Internet of Things (IoT), present unique challenges [19]. E-CEFS's ability to adapt

and perform effectively across these varied environments is critical. This necessitates exhaustive examination of the algorithm's scalability and performance, ensuring it can be seamlessly integrated into different network settings without compromising on efficiency or effectiveness [35].

Empirical Analysis and Algorithmic Validation: the validation and continuous improvement of E-CEFS hinge on the collection and analysis of traffic data across various network scenarios [36]. This empirical analysis is crucial for assessing the algorithm's performance and adjusting its parameters to optimize its functionality in real-world settings. By engaging in this rigorous validation process, the algorithm can be fine-tuned to address the specific challenges and requirements of different network environments, enhancing its precision and reliability in identifying and mitigating cyber threats.

Reduction of False Positive Rates: a significant challenge in the deployment of cybersecurity measures is the minimization of false positive rates. High FPRs can lead to unnecessary network overhead and the misidentification of legitimate traffic as malicious. Developing strategies to reduce FPR in E-CEFS is therefore paramount [36]. This involves a careful balance between maintaining high levels of traceback accuracy and minimizing the impact on network performance. By achieving this balance, E-CEFS can provide a more reliable and efficient means of identifying and addressing cyber threats without hindering network operations.

Legal and Ethical Implications: the implementation of any cybersecurity measure, including E-CEFS, must be conducted with a keen awareness of its legal and ethical implications [32]. This is especially true regarding the use of secret keys and packet marking techniques, which must be scrutinized in light of privacy concerns and data protection laws. Ensuring that the deployment of E-CEFS adheres to these legal and ethical standards is essential for maintaining the trust and confidence of users and stakeholders. It also serves to safeguard the integrity of the data and communications it seeks to protect, ensuring that cybersecurity efforts do not inadvertently infringe

upon the rights and privacy of individuals.

In summary, the effective implementation of E-CEFS in real-world scenarios requires a comprehensive and nuanced approach that addresses a spectrum of strategic considerations. From optimizing key mechanisms and conducting attack simulations to ensuring compatibility with existing security frameworks and considering legal and ethical implications, each aspect plays a crucial role in enhancing the algorithm's functionality and applicability. By meticulously addressing these recommendations, we can fortify our digital infrastructures against the multitude of cyber threats that loom in the digital age, ensuring a safer and more secure cyber landscape for all.

8. CONCLUSION

The realm of cybersecurity stands perpetually on the precipice of innovation, driven by the relentless evolution of cyber threats that challenge the integrity and resilience of network infrastructures. Within this dynamic landscape, the thesis has carved a niche, addressing the complex challenge of IP traceback in the context of mitigating the impact of Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. Through the development of an Enhanced Compressed Edge Fragment Sampling (CEFS) algorithm and the exploration of cutting-edge strategies such as the implementation of secret key sharing among servers for packet origin authentication, this research has made significant strides in advancing the cybersecurity domain.

The core of this thesis revolves around a novel approach that integrates cryptographic principles with IP traceback methodologies, thereby introducing a level of sophistication and security previously unattained in the field. By meticulously comparing this enhanced method with traditional mechanisms like ICMP Traceback and Ingress Filtering, the research delineates a clear trajectory of advancement, demonstrating superior computational efficiency and the ability to maintain operational efficacy in the aftermath of cyber attacks. This comparative analysis not only highlights the strengths of the proposed Enhanced CEFS algorithm but also critically examines the limitations inherent in existing methodologies, thereby shedding light on potential avenues for further research and development.

In delving into the theoretical underpinnings and practical applications of the Enhanced CEFS algorithm, this thesis has provided a comprehensive exploration of its mathematical foundations and empirical performance. The rigorous validation process, underpinned by mathematical analysis and empirical evaluation, attests to the algorithm's efficacy in tracing and mitigating DoS and DDoS attacks. Furthermore, the innovative use of secret keys among servers emerges as a pivotal feature, enhancing the security framework by ensuring the authenticity of packet origins and significantly

diminishing the feasibility of packet identification manipulation by malicious actors.

The implications of this research extend beyond the immediate enhancements to IP traceback techniques, offering a broader perspective on the cybersecurity challenges faced by network infrastructures. The findings underscore the critical need for continued innovation and adaptation in cybersecurity practices to counteract the evolving spectrum of cyber threats. Moreover, the introduction of cryptographic elements into IP traceback methodologies opens new horizons for securing network operations against an increasingly sophisticated array of cyber attacks.

Looking forward, this thesis not only contributes to the academic and practical discourse on cybersecurity but also sets the stage for future research endeavors. The exploration of further cryptographic integration, the optimization of algorithmic efficiency, and the expansion of empirical evaluations across diverse network environments represent key areas for continued investigation. As the digital landscape evolves, so too must the strategies employed to protect and secure it. The work presented herein lays a foundational stone in this ongoing journey, offering insights, methodologies, and a vision for the advancement of network security in the face of relentless cyber adversities.

REFERENCES

1. Bhavani, Y., Janaki, V., & Sridevi, R., "Survey on Packet Marking Algorithms for IP Traceback," *Oriental Journal of Computer Science & Technology*, vol. 10, no. 2, pp. 507-512, 2017.

2. Li, Q., "A Model of IP Traceability Dynamic Collaboration Technology in the Denial of Service Attack," *IEEE International Conference on Communication Technology*, Beijing, China, 2012.

3. Yin, H., & Li, J., "An Efficient Probabilistic Packet Marking Scheme (NOD-PPM)," in S.K. Katsikas et al. (Eds.), *ISC 2006, LNCS 4176*, Springer-Verlag Berlin Heidelberg, Berlin, Germany, pp. 373-382, 2006.

4. Peng, T., Leckie, C., & Ramamohanarao, K., "Adjusted Probabilistic Packet Marking for IP Traceback," in E. Gregori, M. Conti, A.T. Campbell, G. Omidyar, & M. Zukerman (Eds.), *Networking 2002: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications, Lecture Notes in Computer Science*, vol. 2345, Springer, Berlin, Heidelberg, pp. 697-708, 2002.

5. Gao, Z.Q., & Ansari, N., "A Practical and Robust Inter-Domain Marking Scheme for IP Traceback," *Computer Networks*, vol. 51, no. 3, pp. 732-750, 2007.

6. Gong, C., Le, T., Korkmaz, T., Sarac, K., "Single Packet IP Traceback in AS-Level Partial Deployment Scenario," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '05)*, St. Louis, MO, USA, pp. 3-8, 2005.

7. Chao, G., & Kamil, S., "Toward a Practical Packet Marking Approach for IP Traceback," *International Journal of Network Security*, vol. 8, no. 3, pp. 271-281,

2009.

8. Zhang, M., Zhao, H., "New Scheme for IP Traceback," *Computer Engineering and Applications*, vol. 47, no. 30, pp. 83-85, 97, 2011.

9. Wang, X.-J., Xiao, Y.-L., & Wei, S.-J., "A Stochastic Model of Attack Path Reconstruction Problem for IP Traceback," *Transactions of Beijing Institute of Technology*, no. 2, pp. 168-172, 2011.

10. Wu, R.H., Wu, L., "Research on the Packet Marking Algorithm for IP Trackback," *Journal of Hunan University (Natural Sciences)*, no. 06, pp. 39-43, 2011.

11. Lin, B., Yang, B., Wu, P., Mao, J., "Discussion of IP Traceback Issues: Evaluation and Deployment," *Digital Technology Application*, no. 3, pp. 22-28, 2012.

12. Belenky, A., & Ansari, N., "IP Traceback with Deterministic Packet Marking," *IEEE Communications Letters*, vol. 7, no. 4, pp. 162-164, 2003.

13. Gong, C., & Sarac, K., "IP Traceback Based on Packet Marking and Logging," in *Proceedings of the IEEE International Conference on Communications (ICC 2005)*, Seoul, Korea (South), vol. 2, pp. 1043-1047, 2005.

14. Morris, R. T., "A Weakness in the 4.2BSD Unix TCP/IP Software," *Technical Report*, AT&T Bell Laboratories, Murray Hill, New Jersey, 1985.

15. Dean, D., Franklin, M., & Stubblefield, A., "An Algebraic Approach to IP Traceback," *ACM Transactions on Information and System Security*, vol. 5, no. 2, pp. 119-137, 2002.

16. Özen, M. S., "Internet Protocol Traceback Using Dynamic Changing Packet Marking Probability Method," M.S. Thesis, Anadolu Üniversitesi, Elektrik-Elektronik

Mühendisliği Anabilim Dalı, 2008.

17. Dong, Q., Banerjee, S., Adler, M., Hirata, K., "Efficient Probabilistic Packet Marking," in *Proceedings of the 13th IEEE International Conference on Network Protocols (ICNP'05)*, Boston, MA, USA, pp. 62-73, 2005.

18. Xiang, Y., Zhou, W., & Guo, M., "Flexible Deterministic Packet Marking: An IP Traceback System to Find the Real Source of Attacks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 4, pp. 567-580, 2009.

19. Savage, S., Wetherall, D., Karlin, A., Anderson, T., "Practical Network Support for IP Traceback," in *Proceedings of SIGCOMM'00: Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, Stockholm, Sweden, pp. 295-306, 2000.

20. Ghosh, R. K., & Mohanty, H. (Eds.), "Distributed Computing and Internet Technology," Springer, Berlin, Heidelberg, 2005.

21. Saurabh, S., & Sairam, A. S., "Increasing Accuracy and Reliability of IP Traceback for DDoS Attack Using Completion Condition," *International Journal of Network Security*, vol. 18, no. 2, pp. 224-234, 2016.

22. Zhou, X., "Scheme Optimization in Key Management with Cryptographic Methods," *Applied Mechanics and Materials*, vol. 20-23, pp. 539-545, 2010.

23. Oliveira, P. F., & Barros, J., "A Network Coding Approach to Secret Key Distribution," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 414-423, 2008.

24. Li, G., Sun, C., Xu, W., De Renzo, M. D., & Hu, A., "On Maximizing the Sum Secret Key Rate for Reconfigurable Intelligent Surface-Assisted Multiuser Systems,"

IEEE Transactions on Information Forensics and Security, vol. 17, no. 1, pp. 211-225, 2022.

25. Scarani, V., Bechmann-Pasquinucci, H., Cerf, N., Dušek, M., Lutkenhaus, N., & Peev, M., "The Security of Practical Quantum Key Distribution," *Reviews of Modern Physics*, vol. 81, no. 3, pp. 1301-1350, 2009.

26. Fung, C., Ma, X., & Chau, H., "Practical Issues in Quantum-Key-Distribution Postprocessing," *Physical Review A*, vol. 81, no. 1, 012318, 2009.

27. Fossier, S., Diamanti, E., Debuisschert, T., Villing, A., Tualle-Brouri, R., & Grangier, P., "Field Test of a Continuous-Variable Quantum Key Distribution Prototype," *New Journal of Physics*, vol. 11, 045023, 2008.

28. Zhang, Z., Zhao, Q., Razavi, M., & Ma, X., "Improved Key-Rate Bounds for Practical Decoy-State Quantum-Key-Distribution Systems," *Physical Review A*, vol. 95, no. 1, 012333, 2016.

29. Ren, K., Su, H., & Wang, Q., "Secret Key Generation Exploiting Channel Characteristics in Wireless Communications," *IEEE Wireless Communications*, vol. 18, no. 4, pp. 6-12, 2011.

30. Eriksson, T., Trinh, P. V., Endo, H., Takeoka, M., & Sasaki, M., "Secret Key Rates for Intensity-Modulated Dual-Threshold Detection Key Distribution Under Individual Beam Splitting Attacks," *Optics Express*, vol. 26, no. 16, pp. 20409-20419, 2018.

31. Ren, K., Zeng, K., & Lou, W., "A New Approach for Random Key Pre-Distribution in Large-Scale Wireless Sensor Networks," *Wireless Communications and Mobile Computing*, vol. 6, no. 3, pp. 307-318, 2006.

32. Singh, K., Singh, P., Kumar, K., "A Systematic Review of IP Traceback Schemes for Denial of Service Attacks," *Computers Security*, vol. 56, pp. 111-139, 2016.

33. Roy, S., Chawla, H., & Sairam, A. S., "IP Traceback in Dynamic Networks," in S. Nandi, D. Jinwala, V. Singh, V. Laxmi, M. Gaur, P. Faruki (Eds.), *Security and Privacy. ISEA-ISAP 2019. Communications in Computer and Information Science*, vol. 939, Springer, Singapore, 2019.

34. Wang, X., Reeves, D. S., "Robust Correlation of Encrypted Attack Traffic Through Stepping Stones by Manipulation of Interpacket Delays," in *Proceedings of the 10th ACM Conference on Computer and Communications Security*, Washington, DC, USA, pp. 20-29, 2003.

35. Adler, M., "Trade-Offs in Probabilistic Packet Marking for IP Traceback," *Journal of the ACM*, vol. 52, no. 2, pp. 217-244, 2002.

36. Goodrich, M. T., "Efficient Packet Marking for Large-Scale IP Traceback," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, Washington, DC, USA, pp. 117-126, 2002.

37. "Internetworking and the Internet Protocol," Available online: `http://software-engineer-training.com/internetworking-and-the-internet-protocol/`, accessed 2022.

38. "What Is IP Spoofing?" Cloudflare, Available online: `https://www.cloudflare.com/learning/ddos/glossary/ip-spoofing/`, accessed 2022.

39. Burch, H., Cheswick, B., "Tracing Anonymous Packets to Their Approximate Source," unpublished paper, 1999.

40. Cheswick, B., Burch, H., Branigan, S., "Mapping and Visualizing the Internet,"

in *Proceedings of the USENIX Annual Technical Conference*, San Diego, CA, USA, 2000.

41. Govindan, R., Tangmunarunkit, H., "Heuristics for Internet Map Discovery," in *Proceedings of the IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, Tel Aviv, Israel, vol. 3, pp. 1371-1380, 2000.

42. Claffy, K., "Internet Measurement and Data Analysis: Topology, Workload, Performance, and Routing Statistics," presented at the NAE'99 Workshop, Los Angeles, CA, USA, 1999.

43. Sager, G., "Security Fun with OCxmon and cflowd," presentation at the Internet 2 Working Group, 1998.

44. Taylor, T., Leech, M., Bellovin, S., "ICMP Traceback Messages," available online: `https://tools.ietf.org/html/draft-ietf-itrace-04`, accessed 2022.

45. Stone, R., "CenterTrack: An IP Overlay Network for Tracking DoS Floods," to appear in *Proceedings of the 2000 USENIX Security Symposium*, Denver, CO, 2000.

46. Song, D. X., Perrig, A., "Advanced and Authenticated Marking Schemes for IP Traceback," in *Proceedings of the IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, Anchorage, AK, USA, vol. 2, pp. 878-886, 2001.

47. Rayanchu, S. K., Barua, G., "Tracing Attackers with Deterministic Edge Router Marking (DERM)," in R. K. Ghosh H. Mohanty (Eds.), *Distributed Computing and Internet Technology, ICDCIT 2004, Lecture Notes in Computer Science*, vol. 3347, Springer, Berlin, Heidelberg, pp. 400-409, 2004.

48. Tenali, N. M., Jyosyula, B. S., "IP Traceback Scenarios," *Global Journal of Computer Science and Technology*, vol. 13, pp. 17-23, 2013.

49. Barak-Pelleg, D., Shavitt, Y., Shir, E., "Algorithms for Reconstructing DDoS Attack Graphs Using Probabilistic Packet Marking," arXiv preprint arXiv:2304.05123, 2023.

APPENDIX A: UNDERSTANDING NETWORK PATH RECONSTRUCTION THROUGH THE COUPON COLLECTOR'S PROBLEM

A.1. Overview of the Coupon Collector's Problem

The coupon collector's problem is a seminal paradigm within probability theory that inquires into the requisite number of random samples (termed coupons) needed from a collection comprising n distinct entities to ensure at least one instance of each entity is obtained. It is established that the expected number of samples essential escalates in proportion to $n \cdot (\ln(n) + O(1))$, wherein $O(1)$ symbolizes a constant whose significance diminishes with increasing n.

A.2. Analogy within the Document's Framework

Within the ambit of the referenced document, the metaphor of a "coupon" finds a parallel in a unique fragment necessary for piecing together the network path. In this scenario, n is analogously represented by kd, symbolizing the aggregate number of distinct fragments requisite for completion (with k fragments delineated per edge across d edges).

Expounding upon the foundational principles and contextual backdrop elucidated up to page 302, we embark on a rigorous mathematical derivation of the expression $\frac{k \ln(kdc)}{p(1-p)^{d-1}}$ and elucidate its pivotal role in facilitating the network path's reconstruction with a probability of $1 - \frac{1}{c}$.

A.3. Rigorous Mathematical Derivation

The initiation of the derivation process acknowledges the imperative for k fragments per edge-id to meticulously reconstruct an attack path, contemplating an attacker positioned d hops away alongside a marking probability p. This methodology resonates with the edge sampling technique, albeit with a nuanced approach that necessitates multiple fragments for an exhaustive edge identification, thereby enhancing the data precision required for accurate path reconstruction.

Deriving from the probabilistic marking principle and the necessity to aggregate fragments from each of the d edges, the expected number of packets $E(X)$ is inferred through the lens of the coupon collector's problem. This conundrum suggests that for the acquisition of one instance of each type among n unique items, the expected trial count is $n(\ln(n) + O(1))$.

This analogy underscores that to efficaciously accumulate all essential fragments for path reconstruction (akin to gathering all coupon types), the expected packet count proportionally increases with k (fragments per edge) and d (edge count), modulated by the marking probability p and the incremental diminution in marking efficacy with distance, encapsulated by $p(1-p)^{d-1}$, culminating in the equation

$$E(X) < \frac{k \ln(kdc)}{p(1-p)^{d-1}} \tag{A.1}$$

A.4. Elucidation of the Probability $1 - \frac{1}{c}$

The probability $1 - \frac{1}{c}$ for the successful reconstruction of the path originates from an understanding of the intrinsic dynamics of probabilistic collection dilemmas. This term succinctly captures the concept of "oversampling" necessary to mitigate the probabilistic nature of packet marking and ensure a comprehensive fragment collection for accurate path delineation.

The expression $1-\frac{1}{c}$ represents the probabilistic boundary beyond which the aggregated data suffices for path reconstruction. As c increases, the probability approaches 1, denoting heightened confidence in path reconstruction. This model leverages the logarithmic scale inherent in the coupon collector's paradigm, where the quest to secure the last few unique fragments (or, in this context, the necessary final edge fragments for path reconstruction) demands exponentially more samples. Hence, the formula integrates the expected augmentation in packet sampling requisite for attaining a specified confidence level in path reconstruction, harmonizing the probabilistic nature of the marking with the combinatorial challenge of compiling a comprehensive set of unique fragments critical for accurate path identification.

A.5. Optimal value for p

Given the function

$$f(p) = \frac{k\ln(kdc)}{p(1-p)^{d-1}} \tag{A.2}$$

The derivative of $f(p)$ with respect to p is obtained through the quotient rule and chain rule in calculus

$$f'(p) = \frac{d}{dp}\left(\frac{k\ln(kdc)}{p(1-p)^{d-1}}\right) \tag{A.3}$$

After applying the differentiation rules, we find

$$f'(p) = -\frac{k\ln(kdc)(1-p)^{1-d}}{p^2} - \frac{(1-d)k\ln(kdc)}{(1-p)^d p} \tag{A.4}$$

Setting the derivative equal to zero for solving p

$$-\frac{k\ln(kdc)(1-p)^{1-d}}{p^2} - \frac{(1-d)k\ln(kdc)}{(1-p)^d p} = 0 \tag{A.5}$$

The solution yields

$$p = \frac{1}{d} \tag{A.6}$$

APPENDIX B: MATHEMATICAL ANALYSIS OF E-CEFS

Δ_q, of dimensions $16 \times w$ is shared among routers within the network. This key is dynamically updated at regular intervals to ensure the security of packet markings.

The probability of a router marking a packet is represented by p. The path length, denoted by d, indicates the number of hops between the attacker and the victim. The algorithm's efficiency and reliability hinge on these parameters, alongside the computational complexity introduced by the secret key validation, denoted by α.

The expected number of packets, $E(X)$, required for path reconstruction is influenced by the marking probability p, the path length d, and the complexity factor α. The formula for $E(X)$ is given by

$$E(X) < \frac{k \ln(kdc + \alpha)}{p(1-p)^{d-1}}, \tag{B.1}$$

where q is the interval for updating the secret key, and α encapsulates the additional computational overhead due to the secret key validation process.

The computational complexity, $O(\alpha)$, introduced by E-CEFS includes operations related to secret key generation, packet validation, and path reconstruction

$$O(\alpha) = O(qw) + O(\text{hash}), \tag{B.2}$$

where $O(qw)$ represents the complexity of generating and distributing the secret key matrix, and $O(\text{hash})$ accounts for the complexity of hashing operations in packet validation.

E-CEFS significantly enhances the IP traceback process by integrating a robust packet validation mechanism. This analysis underscores the algorithm's potential in improving network security through advanced traceback capabilities.

B.1. False Positive Rate in E-CEFS

The false positive rate in the original CEFS algorithm is defined as $1 - \frac{1}{c}$, indicating the likelihood of incorrectly identifying a packet's origin or path. In E-CEFS, the introduction of a secret key validation mechanism aims to reduce this rate by authenticating packet markings.

Assuming a robust and secure secret key mechanism, the false positive rate in E-CEFS can be significantly lowered. This rate primarily depends on the effectiveness of the secret key validation in distinguishing between legitimate and forged packet markings.

B.1.1. Derivation

Let:

- p_{auth} represent the probability of successfully authenticating a legitimate packet marking,
- p_{forge} represent the probability of an attacker successfully forging a packet marking that passes the secret key validation.

The false positive rate (FPR) in E-CEFS can be expressed as

$$FPR = 1 - p_{\text{auth}} + p_{\text{forge}} \tag{B.3}$$

In an ideal robust system, p_{auth} is close to 1, and p_{forge} is close to 0.

Thus, the expression simplifies to

$$FPR \approx p_{\text{forge}} \tag{B.4}$$

The theoretical framework suggests that E-CEFS improves over CEFS in terms of security and reliability by reducing the false positive rate to approximately p_{forge}. However, empirical validation based on the specifics of the secret key mechanism implementation is necessary to determine the exact false positive rate.

APPENDIX C: COMPARISON OF CEFS AND E-CEFS

Given the formulas:

- Original CEFS

$$E(X) < \frac{k \ln(kdc)}{p(1-p)^{d-1}} \tag{C.1}$$

- Enhanced CEFS (E-CEFS)

$$E(X) < \frac{k \ln(kdc + \alpha)}{p(1-p)^{d-1}} \tag{C.2}$$

C.1. Mathematical Analysis

To demonstrate that adding α (where $\alpha > 0$) increases the expected number of packets $E(X)$, consider the logarithmic components of both formulas. The logarithmic function is monotonically increasing, implying that

$$kdc + \alpha > kdc \tag{C.3}$$

Applying the logarithmic function to both sides gives

$$ln(kdc + \alpha) > \ln(kdc) \tag{C.4}$$

Multiplying by k (assuming $k > 0$) maintains the inequality

$$k \ln(kdc + \alpha) > k \ln(kdc) \tag{C.5}$$

Substituting these results back into the original formulas for $E(X)$ leads to

$$E(X)_{\text{E-CEFS}} < \frac{k \ln(kdc + \alpha)}{p(1-p)^{d-1}} > \frac{k \ln(kdc)}{p(1-p)^{d-1}} = E(X)_{\text{CEFS}} \tag{C.6}$$

This inequality clearly shows $E(X)_{\text{E-CEFS}} > E(X)_{\text{CEFS}}$, indicating that the enhanced version requires more packets.

C.2. Reasons Behind Increased Packets in E-CEFS

(i) Increased Complexity: The term α represents additional complexities or overheads not accounted for in the original CEFS. These might include more sophisticated security mechanisms or computational tasks.

(ii) Higher Upper Bound: The increase in the logarithmic term directly contributes to a higher upper bound on the expected number of packets. The presence of α in the E-CEFS formula reflects more complex or secure environmental factors affecting the network's operation.

C.3. Conclusion

The inclusion of α in the E-CEFS formula mathematically justifies the need for more packets due to increased complexities. Thus, E-CEFS offers a more robust approach in scenarios where enhanced security and reliability are critical, albeit at the cost of requiring more packet transmissions for effective attack path reconstruction.

APPENDIX D: CEFS PATH RECONSTRUCTION PROCEDURE FLOW DIAGRAM

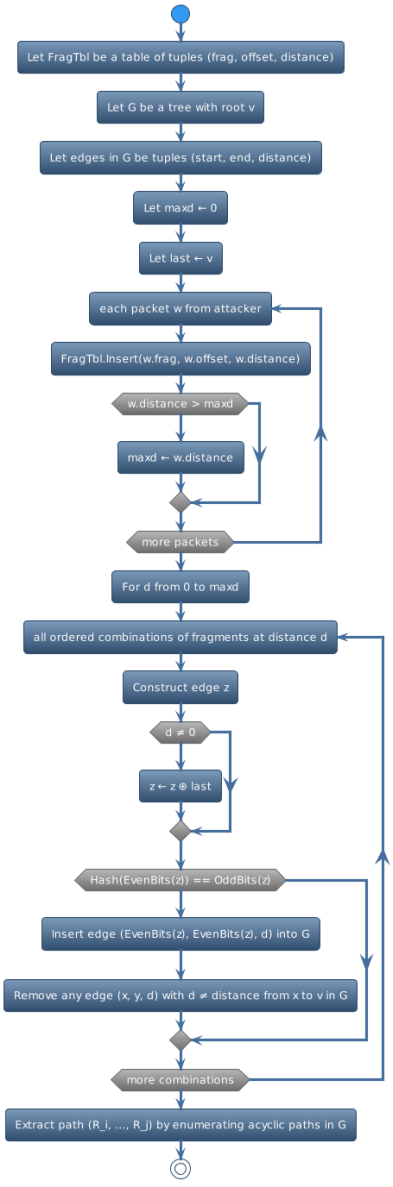

Figure D.1. CEFS Path Reconstruction Procedure Flow Diagram.

APPENDIX E: MODIFIED CEFS PATH MARKING RECONSTRUCTION PROCEDURE FLOW DIAGRAM

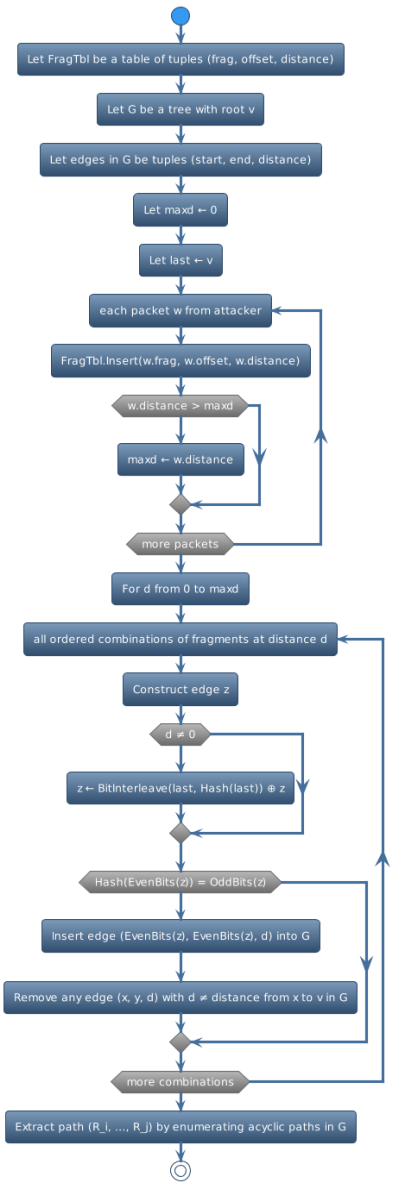

Figure E.1. Modified CEFS Path Marking Reconstruction Procedure Flow Diagram.

APPENDIX F: ENHANCED CEFS MARKING PROCEDURE FLOW DIAGRAM

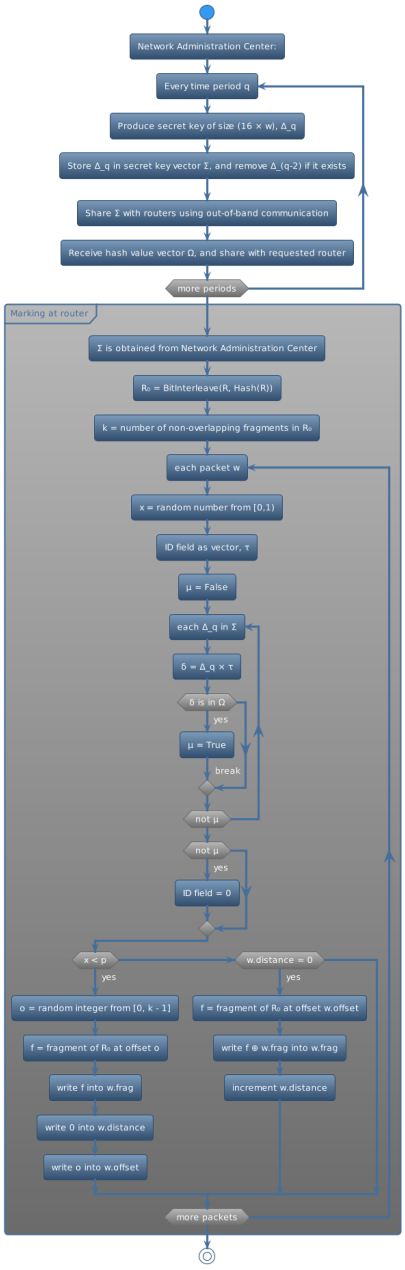

Figure F.1. Enhanced CEFS Marking Procedure Flow Diagram.

APPENDIX G: DISTANCE VS. PACKETS USING THE E-CEFS WITH DIFFERENT SECRET KEY MATRIX SIZE

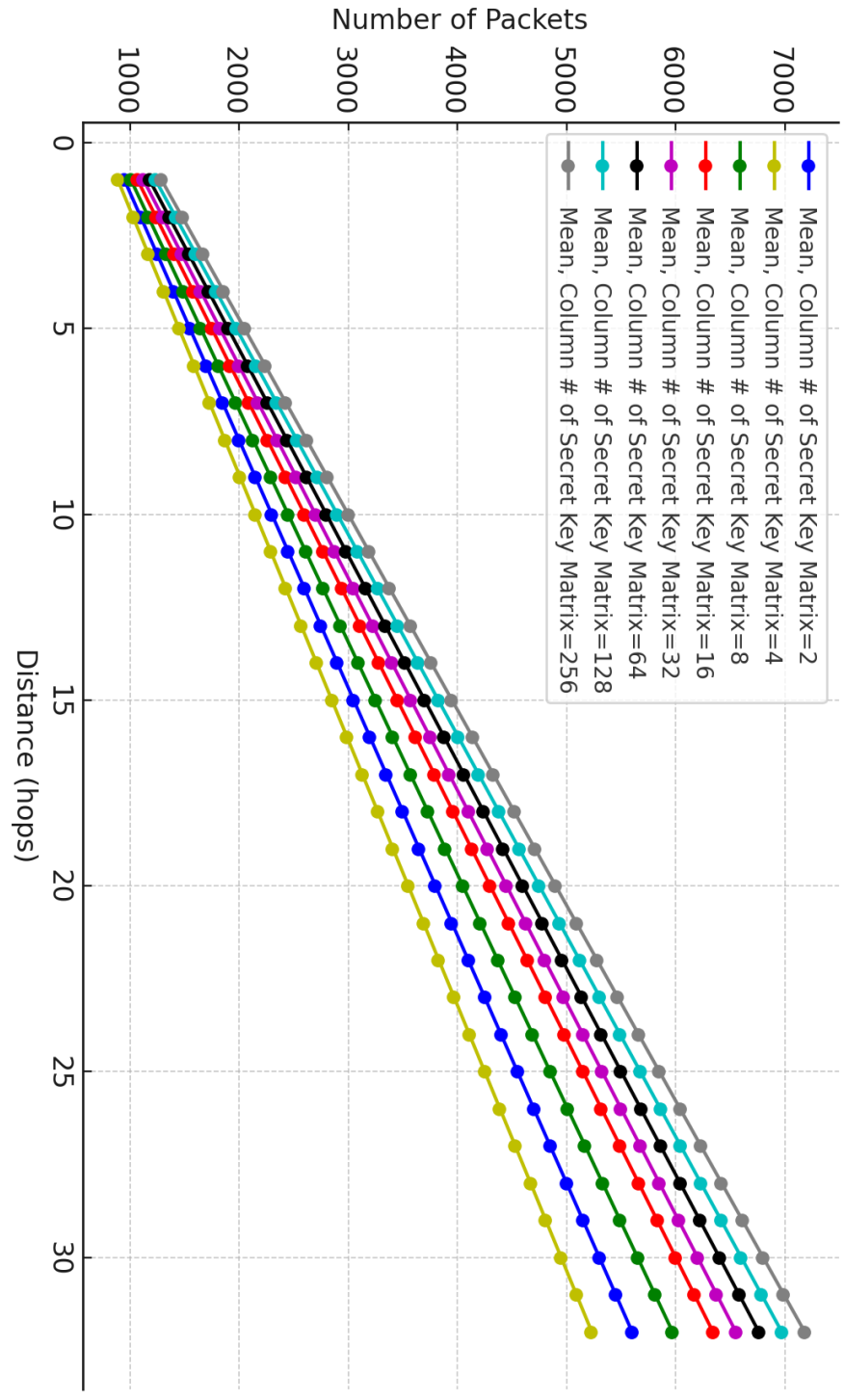

Figure G.1. Distance vs. Packets using the E-CEFS with different secret key matrix size, p = $\frac{1}{25}$, k = 8, FPR = 0.95(based on 100 simulations).