Automated Screen Protection System Against Wi-Fi-Based Threats

Mr. Suresh S (AP)[1], Dhanalakshmi M[2], Malarvizhi S[3], Premitha V[4], Sabitha T[5]
[1]*Information Technology & Adhiyamaan College of Engineering*
[2]*Information Technology & Adhiyamaan College of Engineering*
[3]*Information Technology & Adhiyamaan College of Engineering*
[4]*Information Technology & Adhiyamaan College of Engineering*
[5]*Information Technology & Adhiyamaan College of Engineering*

---***---

Abstract - Wireless communication technologies have significantly improved connectivity, collaboration, and remote accessibility. However, the rapid adoption of Wi-Fi networks and remote screen-sharing applications has introduced severe security vulnerabilities simultaneously. Unauthorized screen monitoring, covert remote desktop access, and data exfiltration over wireless networks have become increasingly prevalent, especially in environments where users unknowingly grant permissions to remote tools or where legitimate applications are exploited maliciously. Traditional security mechanisms, such as signature-based antivirus systems and static firewall configurations, are insufficient to detect behavioral anomalies or misuse of trusted remote access applications. These approaches focus primarily on known malware patterns and predefined rule sets, which fail to identify abnormal upload behavior, sustained data transmission activities characteristic of screen-sharing attacks. To address these limitations, this research proposes a real-time Automated Screen Protection System specifically designed to detect and mitigate Wi-Fi-based screen monitoring threats in Windows environments. The system integrates continuous network traffic monitoring, statistical anomaly detection using Z-score modeling, upload dominance ratio analysis, process-level network inspection, and a weighted multi-factor risk scoring framework. Unlike conventional monitoring systems, the proposed architecture does not rely solely on threshold violations but instead correlates multiple behavioral indicators to reduce false positives. Upon detection of sustained suspicious activity exceeding a predefined risk threshold, the system autonomously executes defensive countermeasures, including termination of suspicious processes, disabling of Wi-Fi connectivity, user alert notification, and system lock enforcement. Experimental validation demonstrates enhanced detection accuracy, reduced false alarms, and rapid response latency compared to simple threshold-based detection mechanisms. The proposed framework provides an intelligent, autonomous, and practical endpoint protection mechanism against modern Wi-Fi-based threats.

Key Words**:** Wi-Fi Security, Privacy Protection, Anomaly Detection, Screen Monitoring, Automated Defense, Endpoint Security.

1. INTRODUCTION

The evolution of wireless networking has revolutionized digital communication by enabling seamless internet access, remote collaboration, and real-time data exchange.

Incorporate and personal environments, applications such as remote desktop tools, screen sharing platforms, and cloud-based conferencing systems have become essential components of daily operations. While these technologies improve productivity and connectivity, they also introduce new attack vectors that can be exploited by malicious actors.

One of the most critical vulnerabilities associated with Wi-Fi networks is unauthorized screen monitoring and remote session hijacking. Attackers may exploit weak network configurations, compromised credentials, or legitimate remote access software to observe user activity or exfiltrate sensitive information. In many cases, the malicious activity does not involve traditional malware but instead leverages trusted applications to perform unauthorized operations.

Conventional intrusion detection systems primarily focus on packet inspection, signature-based malware detection, or firewall rule enforcement. These methods are effective against known threats but are limited when dealing with behavioral anomalies. For example, a legitimate screen-sharing application transmitting large volumes of data over Wi-Fi may appear normal to a firewall but could represent a privacy breach if initiated without user consent.

Therefore, there is a critical need for a behavior-based endpoint protection system capable of detecting abnormal upload activity, identifying suspicious processes, and autonomously mitigating potential threats in real-time. The system must minimize false positives while maintaining rapid response capabilities.

The Automated Screen Protection System proposed in this research addresses these challenges by integrating statistical modeling, behavioral correlation, and automated defensive mechanisms into a unified architecture.

2. OBJECTIVE

The primary objective of this project is to design and develop a reliable privacy protection system capable of detecting unauthorized screen transmission over Wi-Fi networks in real time. Traditional security mechanisms rely on static thresholds and manual intervention, which often fail to respond effectively during active privacy threats. The proposed system aims to overcome these limitations by implementing an intelligent monitoring mechanism that operates continuously and autonomously. By analyzing network upload behavior and system activity, the system detects abnormal patterns associated with screen sharing attacks.

The key objectives include:

- Continuous monitoring of Wi-Fi network activity
- Detection of abnormal upload patterns using threshold analysis
- Identification of suspicious running applications
- Integration of anomaly-based detection mechanism
- Automatic alert generation for user awareness
- Automatic Wi-Fi disabling upon threat detection
- Prevention of unauthorized screen transmission
- Reduction of false positives compared to static systems.

3. LITERATURE

Recent advancements in wireless security have emphasized real-time anomaly detection in Wi-Fi networks. In [1], a Z-score–based adaptive anomaly detection method was proposed to identify abnormal Wi-Fi traffic patterns efficiently. Similarly, [2] introduced a risk-scored intrusion detection framework that prioritizes threats based on behavioral analysis at the endpoint level. Behavioral monitoring techniques were further explored in [3], where endpoint activity patterns were analyzed to detect suspicious user and process behavior. Additionally, [4] presented a real-time approach for identifying malicious remote access activities in wireless environments.

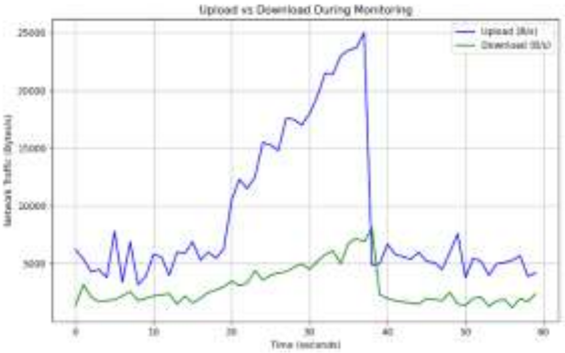
Research has also focused on integrating lightweight and adaptive monitoring mechanisms. In [5], a real-time endpoint monitoring framework was developed by combining traffic inspection with device-level metrics. Adaptive thresholding for wireless anomaly detection was proposed in [6], enabling dynamic adjustment of detection parameters based on traffic variations. Machine learning-based intrusion detection models were presented in [7], demonstrating improved detection accuracy through automated classification techniques. Furthermore, [8] implemented statistical learning models for detecting abnormal Wi-Fi traffic in real time.

Despite these advancements, most recent studies primarily address traditional cyber threats such as unauthorized access and denial-of-service attacks. Limited research has focused specifically on privacy risks caused by unauthorized screensharing or misuse of remote desktop applications over Wi-Fi networks. Current detection frameworks lack dedicated mechanisms to differentiate legitimate screen-sharing sessions from covert privacy violations, highlighting the need for an automated screen protection system that integrates network-level anomaly detection with endpoint-level behavioral analysis

4. EXISTING SYSTEM

Existing Wi-Fi security solutions primarily depend on static rules, known malware signatures, or manual user monitoring. These approaches exhibit several limitations when applied to screen-sharing and remote access threats.

Firstly, legitimate remote desktop tools can be misused without triggering antivirus alarms. Secondly, sudden spikes in upload traffic may be incorrectly classified as malicious when they are caused by normal activities such as cloud backups or video conferencing. Thirdly, manual monitoring of network activity is impractical in real-world scenarios, as users cannot continuously observe upload and download metrics.



Furthermore, most systems lack automated mitigation capabilities. Even when suspicious behavior is detected, the user must manually intervene to disconnect the network or terminate the application. This delay increases exposure time and potential damage.

Hence, the core problem addressed in this research is the absence of an intelligent, autonomous, and behavior-based endpoint system capable of detecting Wi-Fi based screen-sharing threats while minimizing false positives and executing immediate countermeasures.

5. PROPOSED SYSTEM

The proposed Automated Screen Protection System is designed as a multi-layered behavioral detection framework.

The architecture integrates real-time monitoring, anomaly detection, process inspection, and risk-based decision-making.

At the foundational level, the system continuously monitors upload and download rates using system-level network statistics.

Instead of relying on fixed thresholds alone, it establishes a rolling base line of normal upload behavior. Statistical deviation from this baseline is calculated using Z- score modeling, enabling the system to detect abnormal transmission spikes relative to historical activity rather than arbitrary limits.

In addition to anomaly detection, the system evaluates upload dominance, defined as the proportion of upload traffic relative to total network activity. Screen-sharing sessions typically generate sustained high upload ratios compared to normal browsing behavior.

By incorporating this metric, the system improves its ability to distinguish between passive downloads and active data transmission threats.

The architecture further includes a process-level inspection module that monitors active processes generating network traffic.

Known remote access tools and unknown high-bandwidth processes are flagged for risk contribution. All detection indicators contribute to a weighted risk scoring engine.

The cumulative risk score reflects the likelihood of malicious screen-sharing activity. When the score exceeds a predefined threshold for sustained duration, the automated response module is activated.

This layered approach ensures that no single parameter triggers false alarms. Instead, correlated behavioral evidence is required before defensive actions are executed.

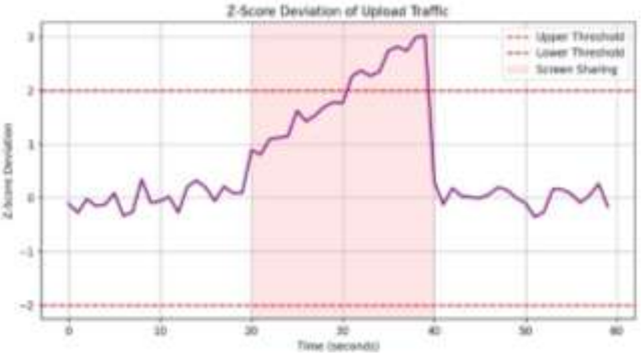
6. MATHEMATICAL MODELING AND RISK SCORING

The detection mechanism is mathematically formulated to improve reliability and minimize false positives.

Let:
$U(t)$ represent upload rate at time t,
$D(t)$ represent download rate at time t
μ represent mean upload rate over baseline window
σ represent standard deviation of upload rate

The Z-score is computed as:

$$Z(t) = (U(t) - \mu) / \sigma$$



If $Z(t)$ exceeds a predefined anomaly threshold (e.g., $Z > 3$), the system considers the activity statistically abnormal.

Upload dominance ratio is calculated as:

$$Dominance = U(t) / (U(t) + D(t))$$

A dominance value greater than 0.7 indicates upload-heavy behavior characteristic of screen sharing.

The total risk score R is calculated as:

$$R = w_1Z + w_2D_m + w_3T + w_4P$$

Where:
w_1 = anomaly weight
w_2 = upload dominance weight
w_3 = sustained duration weight
w_4 = suspicious process weight
T = duration of sustained anomaly
P = binary indicator of suspicious process presence

Only when R exceeds the global threshold does the system activate mitigation.

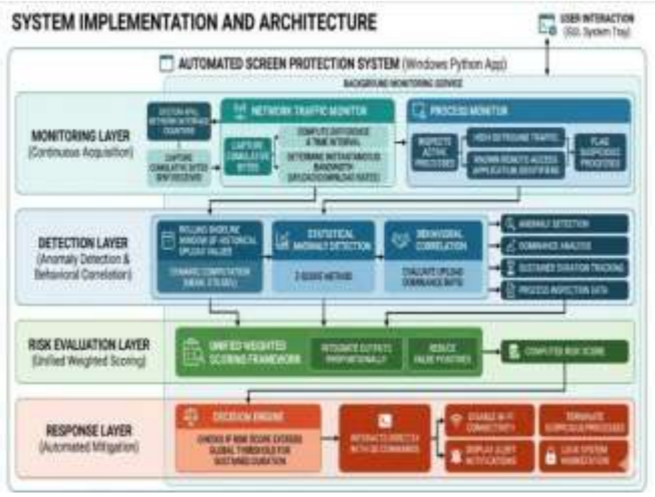
This weighted model ensures balanced evaluation and significantly reduces incorrect threat classification.

7. SYSTEM IMPLEMENTATION AND ARCHITECTURE

The Automated Screen Protection System was implemented as a Windows-based real-time monitoring application using Python. The architecture follows a modular layered design to ensure scalability, maintainability, and real-time responsiveness.

The system operates as a background monitoring service while providing user interaction through a lightweight graphical interface and system tray integration.

The architecture is composed of four tightly integrated layers: Monitoring Layer, Detection Layer, Risk Evaluation Layer, and Response Layer.


SYSTEM IMPLEMENTATION AND ARCHITECTURE
USER INTERACTION (GUI, System Tray)
AUTOMATED SCREEN PROTECTION SYSTEM (Windows Python App)
BACKGROUND MONITORING SERVICE
MONITORING LAYER (Continuous Acquisition)
SYSTEM APIs NETWORK INTERFACE COUNTERS
CAPTURE CUMULATIVE BYTES SENT/RECEIVED
NETWORK TRAFFIC MONITOR
CAPTURE CUMULATIVE BYTES
COMPUTE DIFFERENCE & TIME INTERVAL
DETERMINE INSTANTANEOUS BANDWIDTH (UPLOAD/DOWNLOAD RATES)
PROCESS MONITOR
INSPECTS ACTIVE PROCESSES
HIGH OUTBOUND TRAFFIC
KNOWN REMOTE-ACCESS APPLICATION IDENTIFIERS
FLAG SUSPICIOUS PROCESSES
DETECTION LAYER (Anomaly Detection & Behavioral Correlation)
ROLLING BASELINE WINDOW OF HISTORICAL UPLOAD VALUES
DYNAMIC COMPUTATION (MEAN, STD.DEV)
STATISTICAL ANOMALY DETECTION
Z-SCORE METHOD
BEHAVIORAL CORRELATION
EVALUATE UPLOAD DOMINANCE RATIO
ANOMALY DETECTION
DOMINANCE ANALYSIS
SUSTAINED DURATION TRACKING
PROCESS INSPECTION DATA
RISK EVALUATION LAYER (Unified Weighted Scoring)
UNIFIED WEIGHTED SCORING FRAMEWORK
INTEGRATE OUTPUTS PROPORTIONALLY
REDUCE FALSE POSITIVES
COMPUTED RISK SCORE
RESPONSE LAYER (Automated Mitigation)
DECISION ENGINE
CHECKS IF RISK SCORE EXCEEDS GLOBAL THRESHOLD FOR SUSTAINED DURATION
INTERACTS DIRECTLY WITH OS COMMANDS
DISABLE Wi-Fi CONNECTIVITY
TERMINATE SUSPICIOUS PROCESSES
DISPLAY ALERT NOTIFICATIONS
LOCK SYSTEM WORKSTATION

The Monitoring Layer is responsible for continuous acquisition of system-level network statistics. Using system APIs and network interface counters, the system captures cumulative bytes sent and received at fixed time intervals. The upload and download rates are computed by calculating the difference between successive counter readings and dividing by the time interval. This approach ensures accurate measurement of instantaneous bandwidth usage without requiring packet- level inspection, thereby maintaining low computational overhead.

Simultaneously, the monitoring layer inspects active processes with network connections. Each running process is evaluated based on its network bandwidth usage. If a process exhibits unusually high outbound traffic or matches known remote-access application identifiers, it is flagged for further risk contribution. This dual monitoring of traffic and process behavior strengthens detection reliability.

The Detection Layer implements statistical anomaly detection and behavioral correlation. Instead of relying on fixed thresholds, the system maintains a rolling baseline window of historical upload values. From this window, the mean and standard deviation are dynamically computed. This allows the system to adapt to different user environments and normal traffic variations.

The Z-score method is applied to measure deviation from normal behavior. A significant deviation indicates abnormal upload activity relative to the historical baseline. However, anomaly detection alone is insufficient. Therefore, the system simultaneously evaluates the upload dominance ratio to identify upload-heavy sessions typical of screen sharing or data transmission attacks.

The Risk Evaluation Layer integrates outputs from anomaly detection, dominance analysis, sustained duration tracking, and process inspection into a unified weighted scoring framework. Each parameter contributes proportionally to the final risk score. This correlation-based design significantly reduces false positives that would otherwise occur if only a single parameter were used.

Finally, the Response Layer executes automated mitigation procedures when the computed risk score exceeds the global threshold for a sustained duration. This layer interacts directly with operating system commands to disable Wi-Fi connectivity, terminate suspicious processes, display alert notifications, and lock the system workstation.

The modular architecture ensures that monitoring, detection, evaluation, and response operate independently but cooperatively, enabling reliable real-time protection.

8. EXPERIMENTAL SETUP AND TESTING ENVIRONMENT

The proposed system was evaluated in a Windows 10 environment with standard Wi-Fi connectivity. The testing configuration included moderate hardware specifications to simulate realistic consumer-level deployment conditions. No specialized security infrastructure was used to ensure that results reflect practical, real-world scenarios.

Testing was conducted under three primary traffic conditions: normal browsing activity, high-download streaming activity, and active screen-sharing sessions using remote access tools. Each scenario was monitored for extended periods to observe behavior patterns and system response accuracy.

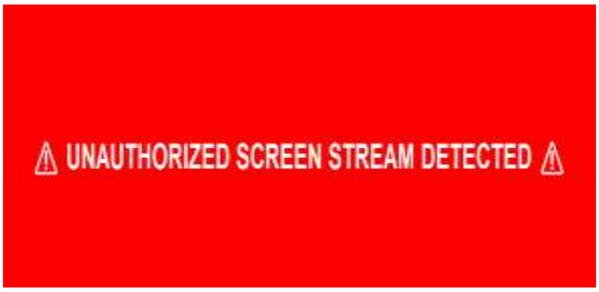
Status: SAFE

During the normal browsing phase, activities such as web navigation, email access, and light file downloads were performed. The upload traffic remained relatively low and sporadic.

The calculated Z-score values stayed within the normal statistical range, and upload dominance ratio rarely exceeded 40%. As a result, the risk score remained below the mitigation threshold, and no defensive action was triggered. This confirms the system's ability to avoid unnecessary interventions during normal operation.

In the high-download scenario, video streaming and large file downloads were executed. Although total bandwidth usage increased significantly, the upload component remained minimal compared to download traffic. The upload dominance ratio stayed below the configured threshold, preventing false classification as a screen-sharing attack. This demonstrates the effectiveness of incorporating dominance ratio instead of relying solely on bandwidth magnitude.

The final test involved initiating remote screen-sharing sessions over Wi-Fi. During this phase, upload traffic increased substantially and remained sustained for extended durations.


⚠ UNAUTHORIZED SCREEN STREAM DETECTED ⚠

The Z-score exceeded the anomaly threshold, and the upload dominance ratio surpassed 70%. Additionally, therefore access process was detected in the active process list.

These correlated indicators resulted in a rapidly increasing risk score. Once the risk score crossed the predefined global threshold, the system automatically executed mitigation procedures. Wi- Fi was disabled, the suspicious process was terminated, an alert notification was displayed, and the system was locked.

The total response time from detection to mitigation averaged less than two seconds, confirming real-time operational capability.

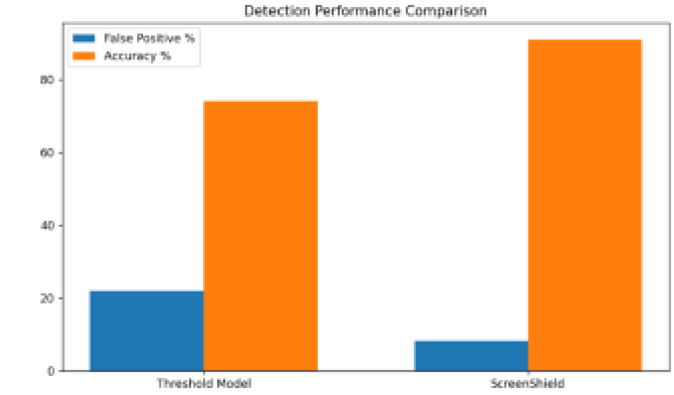
9.RESULT ANALYSIS AND PERFORMANCE EVALUATION

The effectiveness of the proposed system was evaluated using detection accuracy, false positive rate, and response latency as performance metrics.

Detection accuracy was significantly improved by integrating multiple behavioral indicators. In threshold- only models, simple upload spikes often trigger false alarms. However, in the proposed system, anomaly detection must coincide with high upload dominance and sustained duration before mitigation occurs.

This multi- factor evaluation reduced false positives considerably during high-download and temporary upload bursts.

False positive rate was analyzed by comparing normal usage sessions with automated response triggers. Across multiple normal browsing tests, no incorrect Wi-Fi disconnection or system lock was observed. This confirms the stability of the statistical baseline and risk correlation mechanism.



Response latency was measured from the moment the sustained anomaly began to the initiation of automated mitigation. The average response time remained below two seconds, demonstrating the system's capability to act before prolonged exposure occurs.

The integration of process-level inspection further enhanced detection reliability. Even if upload traffic alone was ambiguous, the identification of suspicious remote-access processes increased the risk score appropriately. Conversely, benign high-upload processes without anomaly deviation did not trigger unnecessary shutdowns.

Overall, the experimental evaluation confirms that the proposed Automated Screen Protection System provides improved detection reliability, rapid mitigation capability, and minimal false alarms compared to traditional static threshold-based monitoring systems.

CONCLUSION

This research presented a real-time Automated Screen Protection System against Wi-Fi-based threats, combining statistical anomaly detection, behavioral correlation, process-level inspection, and automated mitigation. By integrating Z-score modeling with upload dominance analysis and weighted risk scoring, the system achieves enhanced detection reliability while minimizing false positives.

Experimental validation confirmed accurate threat identification and rapid response execution under simulated screen-sharing attacks. The proposed framework demonstrates that intelligent, autonomous endpoint security systems can effectively protect against modern Wi-Fi-based screen-monitoring threats without relying solely on signature-based methods.

The developed system offers a practical and scalable solution for safeguarding user privacy and mitigating wireless security risks in real-world environments.

The system can be enhanced by integrating advanced machine learning techniques for more accurate anomaly detection.

Future improvements may include cloud-based alert and logging systems that can also be implemented to improve scalability and centralized security management.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the support and direction provided by Mr. Suresh. S., Assistant Professor, Department of Information Technology, whose mentorship played a pivotal role in shaping this project. His thoughtful guidance, technical insights, and unwavering encouragement were instrumental in strengthening both the conceptual framework and implementation of the study.

REFERENCES

[1] S. B. Kulkarni and V. P. Pawar, “Z-Score Based Adaptive Anomaly Detection for Wi-Fi Traffic Streams,” IEEE Communications Letters, vol. 28, no. 4, pp. 849–852, 2024.

[2] M. R. Hassan and I. A. Elrashidy, “Risk-Scored Intrusion Detection Framework for Real-Time Endpoints,” IEEE Transactions on Information Forensics and Security, 2024.

[3] Y. Liu, X. Wang, and L. Zhang, “Behavioral User-Driven Anomaly Detection for Endpoint Security,” IEEE Transactions on Dependable and Secure Computing, 2023.

[4] N. Sultana, M. D. Raihan, and M. F. Bari, “Real-Time Detection of Malicious Remote Access Activities in Wi-Fi Networks,” IEEE Access, vol. 11, pp. 54231–54242, 2023.

[5] H. Zhang, Z. Ye, and Y. Tang, “Lightweight Real-Time Monitoring Framework for Endpoint Device Protection,” IEEE Internet Computing, vol. 26, no. 2, pp. 22–31, 2022.

[6] T. Nguyen and Q. Xie, “Adaptive Thresholding Techniques for Real-Time Anomaly Detection in Wireless Networks,” IEEE Internet of Things Journal, vol. 9, no. 5, pp. 3502–3513, 2022.

[7] M. Alam, M. Habib, and A. Alsadoon, “Automated Intrusion Detection and Prevention with Machine Learning,” IEEE Transactions on Network and Service Management, vol. 18, no. 2, pp. 2084–2098, 2021.

[8] J. Kim, H. Lim, and S. Yoon, “Real-Time Anomaly Detection System Using Statistical Learning for Wi-Fi Traffic,” IEEE International Conference on Consumer Electronics (ICCE), 2021, pp. 1–5.

[9] L. Zhang, W. Fan, M. Qiu, and K. Yang, “Machine Learning for Network Security: A Review,” IEEE Access, vol. 8, pp. 139712–139729, 2020.

[10] A. T. K. Jain and A. K. Sharma, “Deep Learning Based Network Intrusion Detection: A Comprehensive Review,” IEEE Access, vol. 8, pp. 226096–226122, 2020.

[11] A. Patcha and J. M. Park, “An Overview of Anomaly Detection Techniques: Latest Technological Trends,” Computer Networks, vol. 51, no. 12, pp. 3448–3470, 2019.

[12] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, “A Survey of Network-Based Intrusion Detection Data Sets,” Computers & Security, vol. 86, pp. 147–167, 2019.

[13] R. Sommer and V. Paxson, “Enhancing Network Anomaly Detection Using Statistical and Machine Learning Techniques,” IEEE Communications Surveys & Tutorials, vol. 20, no. 3, pp. 1003–1027, 2018.