International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 07, July 2026)

An Efficient Model for Detection and Prevention of Data Exfiltration Threats with Privacy Guarantees in Enterprise Network

I. M. Ossai[1], E. O. Bennett[2], D. Matthias[3], V. I. E. Anireh[4]
[1]*Information and Communication Technology Centre, Rivers State University, Port Harcourt*
[2,3]*Department of Computer Science, Rivers State University, Port-Harcourt, Nigeria*

***Abstract*—Enterprise networks and its critical operations for communication, data storage, and service delivery are increasingly targeted by sophisticated data exfiltration attack. This attack often compromises confidentiality, integrity, and regulatory compliance. The difficulty in preventing unauthorized access from trusted insiders and outsiders, complexity in the detection of convert channels, and lack of real-time classification of attack have caused enterprise networks to lose valuable resources. The thesis employed Random Forest Classifier for access control mechanism, Mamdani Fuzzy Inference System for detecting convert channels, and Support Vector Machine (SVM) for real-time classification of identified attacks. Implementation was achieved using Visual Basic programming language for interface design and coding with built-in Python machine learning libraries, and (NoSQL) as the system's database server. The model was trained, tested, and evaluated using CICIDS-2017 dataset achieving a high classification accuracy of 95.45%, which constitutes a statistically meaningful improvement over the ARKAIV baselines, reported accuracy of 93.45%. The system recorded 96.77% for precision, F1-Score, and True Positive Rate (TPR) respectively. Additionally, the system recorded an impressively low False Positive Rate (FPR) indicating that only a few percent of network traffic was misclassified. A comparative analysis was conducted and the developed system outperformed all other systems that were compared to it in terms of accuracy in detection and prevention of data exfiltration attack**

***Keywords*—Exfiltration, Support Vector Machine, Fuzzy logic, Insider Threat, Random Forest**

I. INTRODUCTION

In the contemporary enterprise network, data has emerged as one of the most valuable assets for organizations across various sectors. From Government to financial institutions and healthcare providers, the integrity, confidentiality, and availability of data are paramount giving the growing number of system vulnerabilities to cyber threats [1]. Enterprise networks are large-scale secure infrastructure, designed to support communication, data sharing, and access to resources within an organization.

It connects various devices, systems, and users across different locations, allowing for efficient and controlled information flow to meet organizational goals [2].

Traditionally, these networks are structured around Local Area Networks (LANs) for localized communication and Wide Area Networks (WANs) across distributed environments, whether on-premises, in the cloud, or in hybrid settings [3]. Technically, they are built to accommodate a wide range of applications, from file sharing to interconnecting remote sites. Enterprise networks are critical for enabling operations and email to cloud applications and Internet-of-Things (IoT) devices, supporting various business operations from administrative functions to client-facing services. Modern enterprise networks often incorporate Software-Defined Networking (SDN) and virtualization to improve flexibility, scalability, and manageability [4].

There have been more data breaches reported than one could care to count. A significant percentage of these breaches were experienced in critical national infrastructure [5]. However, as the volume of data generated and stored increases, so does the risk of unauthorized access and data exfiltration. Data exfiltration, also known as data extrusion or data theft, refers to the unauthorized transfer of sensitive information from a victim's system to an external destination controlled by an attacker [6]. It is one of the most significant threats in modern cloud computing environments. As organizations increasingly rely on digital infrastructure to store and process valuable data, the risk of data exfiltration has grown exponentially. Enterprise networks are increasingly vulnerable to various security threats due to the proliferation of interconnected devices with diverse functionalities and communication protocols. As organizations rely more heavily on digital systems and the protection of sensitive data becomes critical, the risk of unauthorized data transfers by malicious actors intensifies. Cyber attackers, often referred to as "cyber punks," are continually developing sophisticated techniques to exploit vulnerabilities in these interconnected environments.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 07, July 2026)

The current state of detection and prevention mechanisms for data exfiltration is inadequate. Existing systems struggle to detect covert channels, manage real-time detection, and maintain privacy across multiple platforms. This inadequacy leaves computer systems and infrastructures exposed to significant data breaches and potential financial losses.

II. Related Works

Surveys have explored various methods to detect data exfiltration, particularly in adversarial settings where attackers can modify their behavior to evade detection, emphasizing the challenges of detecting exfiltration in adversarial environments [7]. Another review systematically categorizes machine learning approaches for detecting data exfiltration, identifying gaps such as the need for high-quality datasets and resilience to adversarial attacks [8]. The difficulty in preventing unauthorized access lies in the dual nature of threats. Outsiders often exploit vulnerabilities or use social engineering, while insiders misuse legitimate access. For example, DNS exfiltration, as discussed [9], can be challenging to detect when attackers mimic normal behaviour. Similarly, machine learning approaches, while promising; require constant updates to address evolving threats.

Cyber security researchers have applied Convolutional Neural Network (CNN) to automate the detection of compressed and encrypted data exfiltration. The study addresses this challenge by leveraging deep learning, specifically a Three-Layer CNN, to analyze raw byte-level features of network traffic and files. The goal is to identify exfiltration attempts even when the data is compressed or encrypted [10]. The dataset includes normal network traffic, compressed data exfiltration traffic, and encrypted data exfiltration traffic [11]. Network packets and file data are converted into raw byte sequences, which are then transformed into grayscale images for input into the CNN [12]. The CNN has a three-layer architecture Input Layer that accepts grayscale images representing the byte sequences of network packets or files [13]. Then the Convolutional Layers that are used to extract hierarchical features from the input data. Each layer applies filters to detect patterns indicative of compression or encryption. The Pooling Layers, that is max pooling, is used to reduce dimensionality and highlight the most significant features. Fully Connected Layers classify the extracted features into categories (e.g., normal traffic, compressed exfiltration, encrypted exfiltration) and the output layer provides the final classification result [14].

Further, the model is trained on a labelled dataset of normal and exfiltration traffic Cross-validation used to ensure the model generalizes well to unseen data [15]. The Performance metrics used to evaluate this model are accuracy, precision, recall, and F1-score.

III. Methodology And System Design

Constructive Research Methodology (CRM) was employed in carrying out the research construct. Furthermore, the study adopted an Object-Oriented Design Approach (OODA) for the structural design of the various components in the proposed system. The study utilized the Canadian Institute for Cybersecurity Intrusion Detection (CICIDS2017) and the CERT Insider Threat dataset for training and testing of the proposed system.

Further, the proposed model for data exfiltration detection and prevention system is designed a hybrid model, combining the strengths of Random Forest Classier, Mamdani Fuzzy Inference System, and Support Vector Machines (SVM). Furthermore, the Random Forest Classifier is utilized for the development of the proposed system's access control mechanism. Additionally, the study incorporated Fuzzy Logic technology for the design of a multi-layer data exfiltration detection engine. This engine is responsible for identifying covert channels. Lastly, the Support Vector Machines are adopted to create an adaptive attack classification module that is tasked with categorizing the various types of data exfiltration attacks. The architecture of the proposed system is captured in Figure 1.

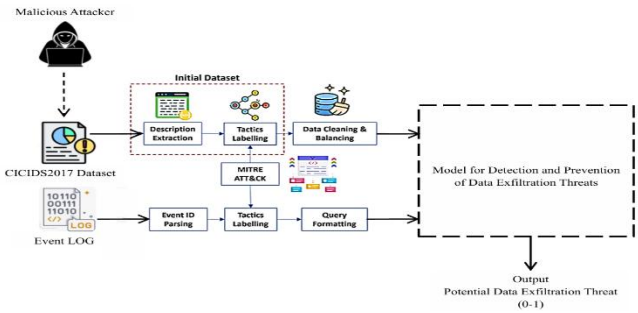

Figure 1: Architectural Design of the System

Seven network-based traffic data as utilized as input to the proposed system. These features are Total length of Fwd Packets, Flow Bytes, Flow Duration, Fwd IAT Mean (Forward Inter-Arrival Time Mean), Destination Port, Fwd Packets, and Flow Packet(s). Embedded in the proposed model for data exfiltration detection and prevention system is a robust access control mechanism, a data exfiltration detection engine, and an attack classification mechanism.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 07, July 2026)

Each of the embedded components serves distinct purposes and collectively contributes to the effective functioning of the entire system. Furthermore, these components will be deployed as an interactive entity to develop a distributed and robust model for detecting and preventing data exfiltration threats in enterprise network environments.

A. Design of Access Control Mechanism

The access control mechanism of the proposed system employs the Standard Random Forest Classifier (RFC) to establish robust access control based on various users and session attributes. This mechanism utilizes username, timestamp, file transfer vol., session duration, IP address, access location, resource type, and privilege level to determine the legitimacy of user access requests. Furthermore, the mechanism will use the standard random forest algorithm to build an ensemble of decision trees, where each tree is constructed from a bootstrap sample of the training data. Furthermore, the decision-making process will be governed by the following steps:

Step 1: Input Feature Vector: The input of the Random Forest-based Access Control mechanism is represented as a feature vector of x, using the stated input features.

Step 2: Building the Decision Trees: Each decision tree in the Random Forest is constructed using a random subset of the features. Further, the splitting criterion for each node is based on the Gini Impurity Index (GII) defined as follows:

$$GINI(D) = 1 - \sum_{j=1}^{C} p_j^2 \qquad (1)$$

Where p_j is the proportion of class j instances in dataset D, and C is the total number of classes.

Step 3: Number of Trees: The proposed access control mechanism is constructed using T decision trees, where T is empirically set to 127. This provides balanced performance and computational efficiency, ensuring robustness in predictions.

Step 4: Majority Voting: After constructing all 127 trees, the predictions for a new input vector x is made through majority voting among all the trees expressed as follows:

$$Predicted\ Class = argmax_{c \in C} \sum_{i=1}^{T} I(f_i(x) = c) \qquad (2)$$

Where f_i is the output of the $i-th$ tree and I is the indicator function that outputs 1 if the condition is true and 0 otherwise.

Step 5: Probability Estimation: The probability $P(Legitimate|x)$ of the output vector being legitimate will be estimated as follows:

$$P(Legitimate|x) = \frac{1}{T} \sum_{i=1}^{T} I(f_i(x) = Legitimate) \qquad (3)$$

Step 6: Threshold for Decision: The output of the access control mechanism is then subjected to a threshold θ to determine access legitimacy. If the predicted probability of an access request being legitimate exceeds, access is granted; otherwise, it is denied:

$$P(Legitimate|x) \geq \theta \Rightarrow Allow\ Access \qquad (4)$$

$$P(Legitimate|x) < \theta \Rightarrow Deny\ Access \qquad (5)$$

The threshold θ in this study is empirically set to 0.85, indicating that an access request must have at least an 85% likelihood of being legitimate to be granted.

B. Design of Fuzzy Exfiltration Detection Engine

To analytically justify the observed performance improvements, the Mamdani Fuzzy Inference System is employed to detect covert data exfiltration behaviours that are often subtle and ambiguous. Unlike traditional binary systems, fuzzy logic enable the proposed Fuzzy Exfiltration Detection Engine to assess traffic patterns based on varying degrees of abnormality or membership (MF) (e.g., "low", "medium", "high" risk). This is considered to be crucial in identifying exfiltration threats that may not immediately violate strict thresholds but still exhibit suspicious characteristics over time. Here, selected network traffic data serve as input to this module. These selected features are then fuzzified to "High", "Medium", or "Low" risk, depending on the degree of membership. However, to incorporate fuzzy logic into the proposed model, we will first define the following components:

Let $\underline{A}$ represent a fuzzy set defined on the feature space X, with a Membership Function (MF) $\mu_{\bar{A}}: X \rightarrow [0,1]$, mapping each feature vector $x \in X$ to a degree of MF in $\bar{A}$:

$$\mu\bar{A}: X \rightarrow [0,1], \quad x \mapsto \mu\bar{A}(x) \qquad (6)$$

The Fuzzy Exfiltration Detection Engine is designed using the Mamdani Fuzzy Inference System (MFIS). At the core of this system lies the Fuzzy Rule Base, which consists of a set of Fuzzy IF-THEN rules. These rules map combinations of input linguistic variables (derived from network traffic features) to output linguistic variables representing the risk level of a potential data exfiltration event. The Fuzzy "IF-THEN" rules are expressed as follows:

$$R_i: IF\ x\ is\ \bar{A}_i\ THEN\ y\ is\ \bar{B}_i,\ i = 1,2,\ldots..n \qquad (7)$$

Where $\bar{A}_i$ and $\bar{B}_i$ are fuzzy sets representing the antecedent and consequent respectively.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 07, July 2026)

$\overline{A}_i$ is defined on x, while $\overline{B}_i$ represents the output with respect to the predefined degree (The measure of membership [0-1] of an element in a fuzzy set) of MF to the data exfiltration threat. Furthermore, let $\mu_{\underline{A}_i}(x)$ denote the degree of membership function (MF) of x in $\underline{A}_i$, and $\mu_{\underline{B}_i}(y)$ denote the degree to MF to y in $\underline{B}_i$. The Mamdani Fuzzy Inference is utilized to evaluate the predefined rules and determine the degree of membership of an instance X, indicating the level of data exfiltration threats in enterprise network environments. The output MF $\mu_{\underline{B}_i}(x)$ of the proposed Fuzzy Exfiltration Detection Engine is computed as follows:

Fuzzification of Input Features: In this module, the input features is first be fuzzified to various degrees of MF $\mu_{\underline{A}_i}(x)$. Here, each antecedent fuzzy set $\underline{A}_i$ is computed using the corresponding MF. In the context of this study, the selected features (Total length of Fwd Packets, Flow Bytes, Flow Duration, Fwd IAT Mean, Destination port, Protocol, and Fwd Packets) are fuzzified to various degrees of linguistic term using the following equation:

$$\mu_A(x) = \begin{cases} 0 & If\ x < a \\ \frac{x-a}{b-a} & if\ a \le x < b \\ 1 & if\ b \le x < c \\ \frac{c-x}{c-b} & if\ b < x \le c \\ 0 & if\ x > c \end{cases} \quad (8)$$

Here, $\mu_A(x)$ is the degree of MF of x to the linguistic term in the fuzzy set A. Further, a, b, and c are the parameters defining the fuzzy set's shape (triangular MF), and x is the crisp input feature. This formula enables the system to determine how strongly a given input belongs to a fuzzy set based on the predefined threshold. The input variables of the proposed system together with their corresponding membership functions are captured in Table 1. The specific range boundaries defining the membership functions were determined empirically on the CICIDS-2017 training dataset.

Table I: Membership and Fuzzification of Input Variables

Feature	Low Range (0-0.3)	Medium Range (0.3-0.7)	High Range (0.7-1.0)
Total Length of Fwd Packets	0 ≤ x < 500	500 ≤ x < 1500	1500 ≤ x < 3000
Flow Bytes	0 ≤ x < 1000	1000 ≤ x < 5000	1000 ≤ x < 10000
Flow Duration	0 ≤ x < 10000	10000 ≤ x < 30000	30000 ≤ x< 60000
Fwd IAT Mean	0 ≤ x < 100	100 ≤ x < 300	300 ≤ x < 500
Destination Port	0 ≤ x < 1024	1024 ≤ x < 49152	49152≤ x < 65535
Fwd Packets	0 ≤ x < 50	50 ≤ x < 150	150 ≤ x < 300
Flow Packets	0 ≤ x < 50000	50000 ≤ x < 300000	30000 ≤ x < 60000

The fuzzy inference engine operates using the following systematic process which includes the Input Acquisition, Fuzzification, and Rule Evaluation. Each fuzzy rule is represented mathematically as follows:

$$R_k: IF\ (f_1\ is\ A_i) AND\ (f_2\ is\ A_2) THEN\ Risk\ Level\ is\ R_1 \quad (9)$$

Where f_1 and f_2 are the input features. A_1 and A_2 are the fuzzy sets. R represents the resulting risk level. The output MF μR_k is calculated using the following:

$$\mu R_k = min(\mu_{A_1}(f_1), \mu_{A_2}(f_2)) \quad (10)$$

Aggregation: The overall output MF for the risk level R is aggregated from the rules using the below formula.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 07, July 2026)

$$\mu_R = \bigvee_{k=1}^{N} \mu_{R_k} \quad (11)$$

Where $\bigvee$ represent the maximum operator.

Defuzzification: The centroid defuzzification method is employed to transform the fuzzy input variables back to crisp output R_{Crisp} value using the following mathematical expression.

$$R_{Crisp} = \frac{\int x \cdot \mu R(x) dx}{\int \mu R(x) dx} \quad (12)$$

Where R_{Crisp} represents the resulting crisp value that represent the final risk level after defuzzification. $\mu_R(x)$ is the MF of the fuzzy set representing the risk level. In this context, it is used to indicate the degree of MF of each possible risk value x. The integration numerator $\int x \cdot \mu R(x) dx$ is used as the integral calculations for the weighted sum of all possible risk value x, where each value is the weighted by its degree of MF $\mu R(x) x$. It is employed to effectively capture how much each risk value will contribute to the overall fuzzy set. Finally, the denominator $\int \mu R(x) dx$ is used for integral computation of the total area under the MF $\mu_R(x)$. It acts as a normalization factor, ensuring that the final crisp value reflects the relative contribution of all values in the fuzzy set. The study employed the following equation for classifying the defuzzied crisp numeric values for the Fuzzy Exfiltration Detection Engine to represent the data exfiltration score.

$$Output_{DataExfiltrationScore} = \begin{cases} Normal & 0 \leq y \leq 0.2 \\ Insider\ Threat & > 0.2 \leq y \leq 0.4 \\ External\ Threat & > 0.4 \leq y \leq 0.6 \\ Data\ Breach & > 0.6 \leq y \leq 0.8 \\ Botnet\ Activities & > 0.8 \leq y \leq 1.0 \end{cases} \quad (13)$$

The output of the Fuzzy Exfiltration Detection Engine together with their corresponding degree of membership values are captured in Table 2. The output risk score ranges defining the threat categories in Table 2 were established heuristically based on domain expertise and subsequent empirical tuning against the evaluation dataset. The scale of 0 to 1 was divided into five distinct risk levels, where each boundary was iteratively adjusted to align the defuzzified crisp output with the severity and classification of the known data exfiltration attack types.

Table II: Output Membership Function of the Fuzzy Engine

Output Variable	Membership Function (MF)	Range Value
Benign	Normal	0.0-0.2
Exfiltration Attack	Insider Threat	> 0.2-0.4
Exfiltration Attack	External Threat	> 0.4-0.6
Exfiltration Attack	Data Breach	> 0.6-0.8
Exfiltration Attack	Botnet Activities	> 0.8-1.0

C. *Design of the Classification Component*

The attack classification mechanism uses the techniques of SVM to find a hyperplane that maximally separates data points of different classes in a high-dimensional space. Here, given a dataset of n training data (x_i, y_i), where x_i represents the feature vector and y_i is the corresponding label (either +1 for benign or -1 for exfiltration), the goal of the proposed system is to find a hyperplane. This is defined as follows:

$$w \cdot x + b = 0 \quad (14)$$

Here, w is the weight vector perpendicular to the hyperplane, and b is the bias term. Further, the optimal hyperplane is found by maximizing the margin, which is the distance between the hyperplane and the nearest data points from each class. The margin γ is then expressed as follows:

$$\gamma = \frac{2}{\|w\|} \quad (15)$$

However, to maximize the margin of the proposed model, the study minimizes the following objective function which is subject to the constraints:

$$\frac{1}{2}\|w\|^2 \quad (16)$$

It is important to note that in real-world situations or scenarios, data may not be perfectly separable. Therefore, this study utilized the concept of a soft margin to allow misclassifications. This introduces what is called the slack variables ξ_i:

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i, \qquad \forall_i, \xi_i \geq 0\, \xi \quad (17)$$

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 07, July 2026)

This modifies the objective function to include a penalty term for any misclassifications using the following equation:

$$\min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \qquad (18)$$

Where C represents a regularization parameter that controls the trade-off between maximizing the margin and minimizing classification error. Technically, SVM can be extended to non-linear classification using the kernel functions. Therefore, a kernel function $K(x_i, x_j)$ is used to compute the dot product in a high-dimensional feature space without explicitly mapping the data points to the space. The study adopted the Linear Kernel function for the proposed system classification task. The adoption of the linear kernel is because of its simplicity in classification tasks. It enables the Attack Classification Mechanism to classify the input feature data that is linearly separable. The formula for the adopted Linear Kernel function is given as follows:

$$K(x_i, x_j) = x_i \cdot x_j \qquad (19)$$

Where x_i and x_j represents the input feature vectors, and . denotes the dot product between the two vectors. However, after the training of the Attack Classification Mechanism, its decision function for a new input x is expressed as follows:

$$f(x) = sign(w \cdot x + b) \qquad (20)$$

The system classifies the input data, and its output as either benign or exfiltration attack based on the sign of $f(x)$.

IV. Results And Discussion

The developed system was implemented using Visual Basic integrated with Python machine learning libraries, while NoSQL utilized as the backend database server for efficient storage and retrieval of network traffic data. The CICIDS-2017 dataset was used for training and evaluation of the system because of its comprehensive representation of modern enterprise network attack scenarios. Figures 2 and 3 capture the screenshot of the developed system interface for the Access Control Mechanism and that of the Fuzzy Detection Engine.

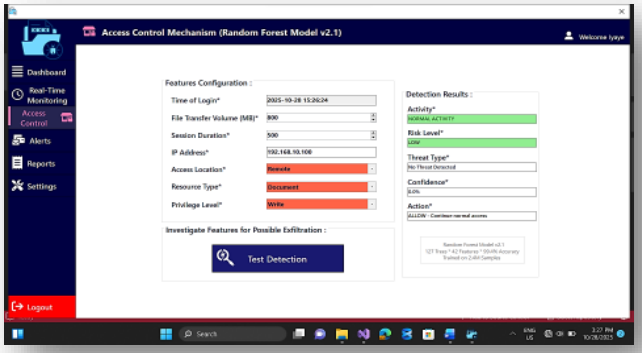

Figure 2: Screenshot Interface of the Access Control Mechanism

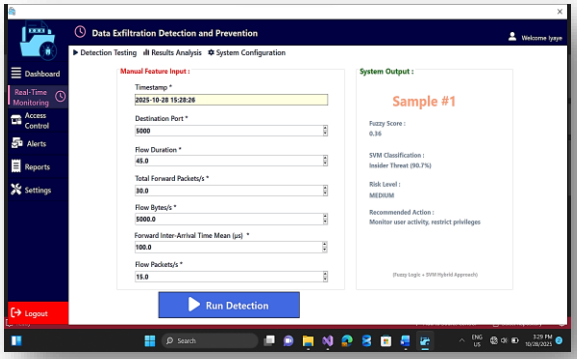

Figure 3: Screenshot Interface of the Fuzzy Detection Engine

The performance evaluation of the developed system focused on three major components, namely the Access Control Mechanism, the Fuzzy Exfiltration Detection Engine, and the Attack Classification and Prevention Mechanism. Standard cybersecurity performance metrics such as Accuracy, Precision, Recall, F1-Score, True Positive Rate (TPR), False Positive Rate (FPR), and Blocking Rate (BR) were used to assess the effectiveness of the proposed model.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 07, July 2026)

D. Performance of the Access Control Mechanism

The Access Control Mechanism achieved strong performance across all evaluated metrics. As presented in Table 3, the mechanism recorded an average precision of 97.5%, recall of 97.5%, F1-score of 97.5%, and overall accuracy of 97.8%. These results demonstrate the effectiveness of the Random Forest-based access control model in distinguishing between legitimate and unauthorized access attempts

Table III: Performance of the Access Control Mechanism

Metric	Legitimate Access	Unauthorized Access	Average (%)
Precision	97.87	97.64%	97.5%
Recall	97.49%	98.00%	97.5%
F1-Score	97.68%	97.82%	97.5%
Accuracy	97.49%	98.00%	97.8%

The high precision value indicates that the proposed mechanism effectively minimized the number of false access approvals, thereby reducing the risk of unauthorized users gaining access to enterprise resources. Similarly, the high recall value demonstrates the system's ability to correctly identify legitimate access requests without significantly disrupting normal enterprise operations.

E. Performance of the Fuzzy Detection Engine

The Fuzzy Exfiltration Detection Engine achieved a True Positive Rate of 96.77%, Precision of 96.77%, and F1-Score of 96.77%, while maintaining a relatively low False Positive Rate of 7.69% in attack classification. These results as tabulated in Table 4 indicate that the fuzzy-based detection engine was highly effective in identifying covert exfiltration activities within enterprise network traffic.

Table IV: Performance of the Fuzzy Detection Engine

Performance Metric	Computed Value
True Positive Rate	96.77
False Positive Rate	7.69%
Precision	96.77%
F1-Score	96.77%
False Positive Rate (FPR)	7.69%

The high True Positive Rate demonstrates the capability of the Mamdani Fuzzy Inference System to correctly detect malicious exfiltration behaviours, including subtle and ambiguous traffic patterns that may not clearly violate traditional rule-based thresholds. The relatively low False Positive Rate further indicates that the developed fuzzy detection engine effectively reduced the occurrence of false alarms, which is a major challenge in conventional intrusion detection systems. The improved performance of the fuzzy detection engine can be linked to the ability of fuzzy logic to model uncertainty and behavioural ambiguity within enterprise traffic flows. Unlike traditional binary classification systems, the fuzzy inference mechanism evaluates traffic behaviour using varying degrees of membership such as low, medium, and high risk. This flexible reasoning capability enabled the proposed system to detect suspicious activities more effectively, particularly in situations involving covert communication channels and gradual data leakage patterns. Presented in Figures 4 is the line graph representation of the system performance of the detection engine.

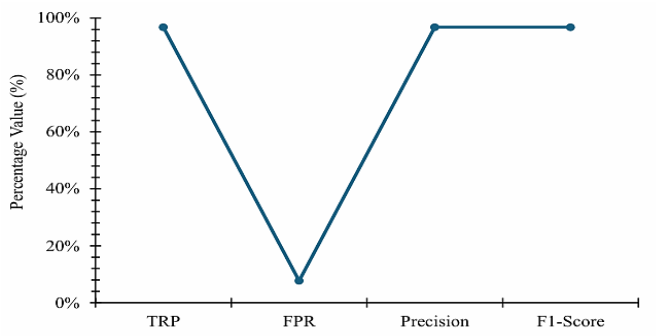

Figure 4: Performance of the Fuzzy Detection Engine

F. Performance in Attack Prevention

The results of system's performance in prevention is presented in Table V. The system achieved a Blocking Rate of 96.77% and a Fase Negative Prevention Rate of 3.23%, indicating that the proposed framework was highly effective in preventing malicious exfiltration attempts in real time.

Table V: Performance in Prevention

Performance Metric	Computed Value
Blocking Rate (BR)	96.77%
False Negative Prevention Rate (FNPR)	3.23%

The high blocking rate demonstrates the capability of the hybrid model to rapidly identify and prevent malicious traffic before sensitive organizational information could be transferred outside the enterprise network. Furthermore, the low False Negative Prevention Rate indicates that only a small percentage of malicious activities were not successfully blocked. This thereby improves the overall reliability and effectiveness of the developed system.

V. Conclusion

This study presented an efficient model for the detection and prevention of data exfiltration threats in enterprise network environments with integrated privacy guarantees. The system combined Random Forest Classifier for intelligent access control, Mamdani Fuzzy Inference System for covert channel detection, and Support Vector Machine (SVM) for adaptive real-time attack classification. The integration of these techniques enabled the system to address critical challenges associated with unauthorized access detection, covert data transfer identification, and intelligent classification of exfiltration threats within enterprise infrastructures. The developed model was implemented using Visual Basic with integrated Python machine learning libraries and evaluated using the CICIDS-2017 dataset. Experimental results demonstrated that the system achieved strong detection and prevention performance across all evaluated metrics. The Access Control Mechanism recorded an average accuracy of 97.8%, while the Fuzzy Detection Engine achieved a True Positive Rate, Precision, and F1-Score of 96.77% respectively, with a low False Positive Rate of 7.69%. Furthermore, the prevention module achieved a Blocking Rate of 96.77%, indicating the effectiveness of the framework in mitigating malicious exfiltration attempts in real time. The findings further revealed that the hybrid integration of machine learning and fuzzy inference techniques significantly improved the system's capability to detect subtle and ambiguous exfiltration patterns that are often missed by conventional intrusion detection systems and, also classify attack types real time.

In conclusion, the study established that combining intelligent access control, fuzzy-based behavioural analysis, and adaptive machine learning classification provides an effective framework for securing enterprise networks against modern data exfiltration threats. The developed system therefore contributes to the advancement of intelligent cybersecurity solutions capable of supporting real-time enterprise network protection while maintaining improved detection accuracy and operational efficiency.

REFERENCES

[1] T. Xu, H. Shi, Y. Shi, and J. You, 2024 "From data to data asset: conceptual evolution and strategic imperatives in the digital economy era," *Asia Pacific Journal of Innovation and Entrepreneurship,* vol. 18, pp. 2-20..

[2] O. Afolalu and M. S. Tsoeu, 2025 "Enterprise networking optimization: a review of challenges, solutions, and technological interventions," *Future internet,* vol. 17, p. 133.

[3] M. A. I. Mallick and R. Nath, 2024 "Navigating the cyber security landscape: A comprehensive review of cyber-attacks, emerging trends, and recent developments," *World Scientific News,* vol. 190, pp. 1-69.

[4] A. Yassin and F. Yalcin, 2019 "Enterprise transition to Software-defined networking in a Wide Area Network: Best practices for a smooth transition to SD-WAN," e,.

[5] H. Jamal, N. A. Algeelani, and N. Al-Sammarraie, 2024 "Safeguarding data privacy: strategies to counteract internal and external hacking threats," *Computer Science and Information Technologies,* vol. 5, pp. 46-54.

[6] Z. Tari, N. Sohrabi, Y. Samadi, and J. Suaboot, 2023 *Data Exfiltration threats and prevention techniques: Machine Learning and memory-based data security*: John Wiley & Sons.

[7] Ö. Aslan, S. S. Aktuğ, M. Ozkan-Okay, A. A. Yilmaz, and E. Akin, 2023 "A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions," *Electronics,* vol. 12, p. 1333.

[8] A. Yaseen, 2023 "The role of machine learning in network anomaly detection for cybersecurity," *Sage Science Review of Applied Machine Learning,* vol. 6, pp. 16-34.

[9] Y. Ozery, A. Nadler, and A. Shabtai, 2023 "Information-based heavy hitters for real-time dns data exfiltration detection and prevention," *arXiv preprint arXiv:2307.02614*.

[10] J. Vansiya, A. Chandi, and R. A. Khan, 2025 "AI-based intrusion detection & prevention models for smart home IoT systems: A literature review," *Journal of Computer Science and Technology Studies,* vol. 7, pp. 982-996.

[11] J. Ferdous, R. Islam, A. Mahboubi, and M. Z. Islam, 2024 "AI-based ransomware detection: A comprehensive review," *IEEE Access,* vol. 12, pp. 136666-136695,

[12] J. Zhao, Z. Cui, J. Fu, M. Shen, and Q. Li, 2025 "A Unified Framework for Robust Encrypted Malicious Traffic Detection in Adverse Environments via Graph Structure Learning," *IEEE Transactions on Network Science and Engineering*.

[13] D. S. Berman, A. L. Buczak, J. S. Chavis, and C. L. Corbett, 2019 "A survey of deep learning methods for cyber security," *Information,* vol. 10, p. 122.

[14] K. Kamatchi and E. Uma, 2025 "Insights into user behavioral-based insider threat detection: systematic review," *International Journal of Information Security,* vol. 24, p. 88.

[15] K. Saminathan, S. T. R. Mulka, S. Damodharan, R. Maheswar, and J. Lorincz, "An artificial neural network autoencoder for insider cyber security threat detection," *Future Internet,* vol. 15, p. 373, 2023.