iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** V **Month of publication:** May 2024

DOI: https://doi.org/10.22214/ijraset.2024.61393

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue V May 2024- Available at www.ijraset.com

Combating Evolving Threats: A Signature-Anomaly Based Hybrid Intrusion Detection System for Smart Homes with False Positive Mitigation

Hasan MD Mehedy[1], Tan Yubo[2]
[1]*Research Scholar, School of Information Science and Engineering, Henan University of Technology, China*
[2]*Associate Professor, School of Information Science and Engineering, Henan University of Technology, China*

Abstract: ***As people are looking for a more comfortable life, IoT applications are coming to play. Smart home system is one of the most popular IoT applications in the last decade. A smart home network is crucial to function smart home system properly. Cyber attacks on a smart home network can damage a lot. Network intrusion detection and prevention system (NIDPS) is a good solution to protect against Cyber threat in smart home network. This research will implement hybrid NIDPS in smart home network by combining signature based and anomaly-detection based NIDPS. This hybrid NIDPS will prevent known known attack from public internet, local internet and zero-day attack. Also, this system will be able to reduce false positive result and improve signature based NIDPS rules accurately by manual inspection.***

Keywords: Hybrid Intrusion Detection and Prevention System (HIDPS), Cyber Threats, Smart Home Security, Network Security, Threat Detection, False Positive Reduction

I. INTRODUCTION

Smart home technology allows users to remotely control various features such as temperature, lighting, and security through internet-connected devices, providing convenience and flexibility. However, the increasing popularity of Smart Home Internet of Things (IoT) devices also brings the risk of cyberattacks [1]. These attacks can result in the theft of sensitive information, financial loss, and even jeopardize user safety. Traditional methods of protection, such as firewalls, are often ineffective against insider threats, requiring alternative solutions. Intrusion Detection and Prevention Systems (IDPS) are essential in safeguarding smart home networks from cyber threats [4, 5]. These systems continuously monitor network traffic for any suspicious activity, notifying administrators and potentially preventing attacks. By examining data packets, IDPS can identify and block malicious traffic. There are two main types of IDPS: host-based and network-based (NIDPS). NIDPS specifically focuses on network traffic and comes in two primary forms: Signature-based NIDPS (SNIDPS) and Anomaly Detection-based NIDPS (ADNIDPS). This study proposes a hybrid approach to NIDPS that combines both SNIDPS and ADNIDPS for improved security. The use of Proofpoint Emerging Threats Rules will ensure up-to-date signature files for SNIDPS, while Artificial Intelligence will be utilized for anomaly detection in ADNIDPS. After reviewing existing literature, this research will outline the proposed system's strategy, implementation process, and performance evaluation.

A. Background of Study

The internet's widespread presence has significantly changed modern life. Whether we are at home, in a busy office, or in an educational institution, we heavily rely on internet connectivity to simplify our daily tasks. This trend is exemplified by smart homes, which allow for remote control of appliances, cameras, sensors, and other devices through internet-connected devices such as mobiles, tablets, computers, and even voice or gesture commands. However, this interconnectedness also presents a potential threat. Cybercriminals can exploit these vulnerabilities to gain unauthorized access to sensitive information, resulting in financial loss, data breaches, and even compromising personal safety [6-8]. To protect local networks from cyberattacks, it is crucial to implement robust security measures [9, 10]. This threat extends beyond external attacks, as malicious actors within your home's vicinity can also attempt to infiltrate your local network by exploiting your Wi-Fi. While traditional firewalls are essential for defending against threats from the public internet, they do have limitations. They struggle to detect new attack methods and offer minimal protection against internal network threats within smart homes [11, 12]. As a solution, network-based intrusion detection and prevention systems (NIDPS) have emerged, capable of identifying and preventing attacks from both the public internet and local smart home networks.

©IJRASET: All Rights are Reserved

B. Literature Review

NIDPS (Network Intrusion Detection and Prevention Systems) are crucial for protecting networks against cyberattacks. These systems continuously monitor network traffic and analyze it for any suspicious activity that may indicate a potential threat or intruder [4, 5]. NIDPS use two main methods for detecting attacks: signature-based and anomaly-based. Signature-based NIDPS (SNIDPS) works like a digital fingerprint scanner for network traffic. It maintains a database of known attack signatures, which are unique patterns associated with specific cyberattacks. When a data packet arrives, the SNIDPS compares its signature to the database and triggers an alert and potentially blocks the packet if there is a match, thus preventing the attack from occurring [13, 14]. SNIDPS has a high accuracy rate in detecting previously encountered attacks, but it struggles with identifying new or zero-day attacks that have not been documented yet. On the other hand, anomaly-based NIDPS (ADNIDPS) uses artificial intelligence (AI) to establish a baseline for normal network traffic patterns. By analyzing historical data, ADNIDPS trains a machine learning model to recognize deviations from the established baseline. Significant deviations are flagged as potential anomalies, which could indicate an ongoing cyberattack [15-17]. ADNIDPS overcomes the limitation of SNIDPS by detecting zero-day attacks, providing a valuable layer of defense against evolving threats. However, a common challenge with ADNIDPS is generating false positives, which occur when legitimate traffic is wrongly identified as malicious, leading to unnecessary alerts and potential disruptions in network operations. To address the limitations of both SNIDPS and ADNIDPS, researchers have explored combining these techniques to create a more robust defense system, known as Hybrid Network Intrusion Detection and Prevention Systems (HNIDPS). HNIDPS typically employ a layered approach, with the first layer using SNIDPS to efficiently detect and block known attacks, while the second layer uses ADNIDPS to identify anomalies indicative of new attacks. Some HNIDPS also incorporate a central database to store attack signatures, which can be updated dynamically with new threats identified by the ADNIDPS, improving threat coverage without manual intervention [18]. However, this approach also introduces the risk of perpetuating false positives, as the SNIDPS may block safe traffic if the ADNIDPS wrongly identifies it as malicious and adds it to the database. Even if the AI model in ADNIDPS improves in the future, the SNIDPS may still trigger alerts or block such packets based on outdated information in the database. Additionally, current hybrid NIDPS designs do not have a method to identify and address these false positives, hindering the learning process of the system. Some researchers have proposed alternative hybrid NIDPS designs that eliminate the dependency on a central database [19]. This research expands on these ideas by introducing a new hybrid NIDPS architecture with two key functionalities: Manual Threat Inspection and Known False Positive Database. Manual Threat Inspection allows for manual analysis of potential threats identified by ADNIDPS, and confirmed threats can be added to the SNIDPS signature database for efficient future detection. The Known False Positive Database is a dedicated database that stores verified false positives identified by the system, serving as a reference for the SNIDPS to prevent it from mistakenly blocking legitimate traffic based on outdated information. This enhanced hybrid NIDPS design aims to address the limitations of existing systems by reducing false positives, improving overall accuracy, and facilitating a more efficient learning process for the AI model used in ADNIDPS.

II. RESEARCH METHODOLOGY

This study presents a unique Hybrid Network Intrusion Detection and Prevention System (HNIDPS) specially designed to secure smart home networks. The system utilizes a combination of signature-based and anomaly-based detection techniques to provide comprehensive protection against both known and unknown cyberattacks.

A. Architecture

The HNIDPS architecture is illustrated in Fig. 1. All incoming and outgoing traffic within the smart home network is directed to the NIDPS for inspection. The MitmProxy tool serves as a central traffic inspection point, enabling efficient analysis of data packets.

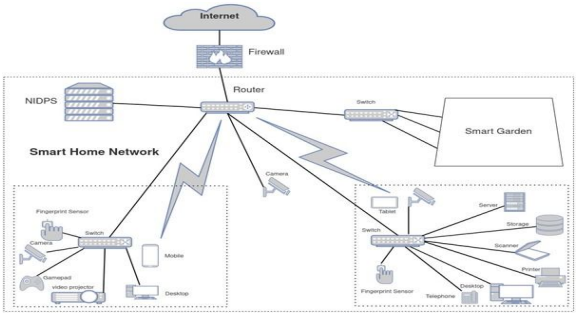


Fig. 1 Smart home network structure

B. Workflow for Data Packet Inspection

The workflow for data packet inspection utilized by the HNIDPS is explained below:

1) *Verification of Trusted/Blacklisted IP/Network*: The MitmProxy addon initiates the inspection process by checking if the source IP address or network associated with the data packet is on the trusted or blacklisted list. If the packet is found to be on the trusted list, it is directly forwarded to its destination without further inspection. Conversely, packets from blacklisted sources are immediately dropped.
2) *Signature-Based Detection:* The HNIDPS offers configurable signature-based detection, allowing users to enable or disable this function through the control panel UI. When activated, the MitmProxy addon carefully compares each data packet against a comprehensive database of known attack signatures. If a match is found, the system triggers a pre-defined action (alert and/or drop) based on the control panel settings and the specific signature matched. This approach ensures efficient detection of known threats.
3) *Anomaly-Based Detection:* Similar to signature-based detection, the HNIDPS also provides an optional anomaly-based detection feature. Users can enable or disable this function through the control panel UI. When activated, the MitmProxy addon uses an AI model to analyze the intricacies of each data packet. The AI model is trained on historical network traffic data to establish a baseline for "normal" behavior. Any significant deviations from this baseline are flagged as anomalies, potentially indicating a cyberattack in progress. If the AI model identifies a packet as anomalous, the system takes a pre-defined action (alert and/or drop) based on the control panel settings.
4) *Forwarding of Benign Traffic:* If neither signature-based nor anomaly-based detection raises any red flags, the MitmProxy addon classifies the data packet as benign (safe) and transmits it to its intended destination.

C. Inspection of New Threats/Intruders by ADNIDPS

The process followed by the ADNIDPS component to inspect and incorporate newly discovered threat's workflow can be summarized as follows:

1) *Identification of New Anomalies*: The system constantly monitors the user interface (UI) data flow for any anomalies flagged by the AI model that have not been previously inspected.
2) *Manual Inspection of Signatures*: Upon identification of a new anomaly, a system administrator or security expert can manually examine the specific signature from the UI data flow.
3) *Verification of Threats and Update of Signature Database*: If the manual inspection confirms the signature as a genuine threat, the corresponding signature and mitigation strategy are incorporated into the SNIDPS database. This ensures that the SNIDPS can effectively block similar attacks in the future.
4) *Identification and Logging of False Positives*: Alternatively, if the manual inspection determines the signature to be a false positive, it is logged in a dedicated "Known False Positive" database. This database serves as a reference for the MitmProxy addon, preventing it from erroneously dropping legitimate traffic in the future.

III.SYSTEM DESIGN

The effectiveness of a Hybrid Network Intrusion Detection and Prevention System (HNIDPS) is dependent on its ability to accurately identify malicious activity within smart home network traffic. This chapter provides an in-depth look at the technical design and architecture of the HNIDPS, outlining the various components that work together to achieve this goal. The discussion begins with the data acquisition and preprocessing stage, where network traffic data from publicly available datasets is carefully prepared for use in the system's AI model. This process involves merging datasets, addressing inconsistencies, and selecting the most informative features for anomaly detection. Next, the rationale behind choosing a Convolutional Neural Network (CNN) as the core AI model for anomaly detection is explained. The chapter then goes into detail about the implementation of the CNN architecture, which includes using a sequential model with Conv1d layers for efficiency. The process of hyperparameter tuning using Optuna, a technique used to optimize the model's performance, is also explored. After the model selection and training, the chapter discusses the design of the PostgreSQL database used by the HNIDPS. This database serves as a central repository for storing various system information, including network traffic details, attack signatures, and user credentials. The development of a RESTful API server using FastAPI is then described. This server acts as a communication bridge, facilitating interaction between the user interface and the database. The functionalities of the API server, such as model creation, route establishment, and secure authentication mechanisms, are also explained. Finally, the chapter discusses the integration of MitmProxy addons to serve as a proxy server within the HNIDPS architecture.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

The use of MitmProxy for implementing both signature-based and anomaly-based intrusion detection functionalities is explored. Additionally, the development of the user interface (UI) for interacting with the HNIDPS is discussed. The UI framework and its functionalities are explained, highlighting how users can use the interface to manage the system and monitor network traffic activity within their smart home environment.

A. Data Acquisition and Preprocessing

Effective anomaly detection within the HNIDPS relies on high-quality training data. This subchapter delves into the data acquisition and preprocessing stage, a crucial step in preparing network traffic data for use in the system's AI model. The selection of publicly available datasets, such as CIC-IDS2017 and CIC-DDoS2019, is discussed, highlighting their suitability for training the anomaly detection model. Furthermore, the intricacies of preprocessing this raw data, including merging datasets, addressing inconsistencies, and selecting the most informative features for the model, are explored. This meticulous process ensures that the AI model learns from a comprehensive and well-prepared foundation, ultimately enhancing its ability to identify anomalies within smart home network traffic.

1) *Dataset Collection*: The selection of training data plays a critical role in the effectiveness of the HNIDPS's anomaly detection model. Publicly available datasets, such as CIC-IDS2017, CIC-IDS2018, and CIC-DDoS2019, were chosen for their suitability in this context. These datasets offer several advantages. Firstly, they encompass real-world network traffic data captured from controlled smart home network environments, providing a realistic foundation for training the model. Secondly, the inclusion of multiple datasets (2017, 2018, 2019) ensures a broader spectrum of network traffic patterns is captured, reflecting potential advancements in cyberattacks over time. Finally, the public availability of these datasets promotes research reproducibility and enables comparisons with other intrusion detection systems, promoting transparency and advancement in the field. By leveraging these well-suited datasets, the HNIDPS lays the groundwork for a robust and generalizable anomaly detection model.
2) *Dataset Preprocessing*: Before training the anomaly detection model with CIC-IDS2017, CIC-IDS2018, and CIC-DDoS2019 datasets, an essential step called data preprocessing is carried out. This thorough process ensures that the data is uniform, relevant, and suitable for the model's learning process. The first step involves merging the datasets from different years, carefully maintaining data integrity and alignment to create a complete training dataset. However, inconsistencies may arise due to variations in data collection methods over the years. These inconsistencies can take the form of differences in column names, data types (such as integer versus string representation for the same feature), or missing values. To address these issues, techniques like data type conversion, column name mapping, and imputation of missing values are used to achieve consistency across the combined dataset. Network traffic data contains a vast number of features, not all of which are equally important for anomaly detection. Thus, feature selection is a crucial step that involves identifying the most informative and relevant features for the model. Feature importance analysis techniques are used to determine which features are significant in distinguishing between normal and anomalous traffic patterns. Additionally, compatibility with MitmProxy addons is considered during feature selection. Features that cannot be effectively captured or analyzed by MitmProxy due to limitations (such as flags) are excluded, ensuring that the model focuses on features that can be used directly within the system's architecture. Finally, the preprocessing stage also deals with data errors or inconsistencies within the CIC-IDS datasets that may hinder the training process. These errors can include inconsistencies within a single file or corrupt data points. By carefully identifying and removing such data, the preprocessing stage ensures that the model learns from clean and reliable information. This emphasis on high-quality data sets the foundation for a more accurate and robust anomaly detection model in the HNIDPS.
3) *Model Selection and Training*: The effectiveness of anomaly detection depends on selecting an appropriate AI model capable of learning and identifying unusual patterns in network traffic data. Convolutional Neural Networks (CNNs) were chosen as the main model for several reasons [20-23]. Firstly, CNNs excel at recognizing complex patterns in data, which is crucial for anomaly detection. Network traffic data inherently displays sequential patterns across its features, and CNNs can effectively extract these patterns to differentiate between normal and anomalous traffic flows. The convolutional layers within a CNN automatically learn these patterns, eliminating the need for manual feature engineering, a time-consuming and error-prone process. Secondly, CNNs, especially those using 1D convolutional layers (Conv1d), are well-suited for processing time series data like network traffic. Conv1d layers can effectively capture temporal dependencies within the network traffic features, allowing the model to learn how features evolve over time and identify deviations from established normal patterns that may indicate anomalies. Finally, CNNs have inherent feature extraction capabilities. Through the convolutional layers, the model automatically learns valuable features from the raw network traffic data, reducing the need for extensive preprocessing by human experts. Additionally, CNNs can perform dimensionality reduction, compressing the input data into a more manageable

representation while preserving essential information for anomaly detection. To further optimize the CNN model's performance for the specific network traffic data used for training, hyperparameter tuning was carried out using Optuna, a hyperparameter optimization library [24, 25]. This ensures that the chosen CNN configuration is finely tuned to maximize its anomaly detection capabilities. The training process involves creating and training two distinct CNN models: a full model that uses all available features and a forward-traffic-only model that focuses on features that are readily available from the initial data packet inspection. This two-model approach aims to achieve comprehensive anomaly detection while minimizing potential delays caused by missing response data.

B. Database Design

The HNIDPS relies on a robust database to store and manage various system information. This section will discuss the PostgreSQL database's design, outlining the structure and purpose of each data table. These tables include network traffic data, security alerts, attack signatures, and user credentials. The well-designed database is the central information hub of the HNIDPS, making data storage, retrieval, and management efficient and contributing to the system's overall functionality.

1) *Dataflow Table*: The Dataflow table is crucial to the HNIDPS, storing detailed information about the network traffic analyzed by the system. This table acts as a comprehensive repository for features extracted from the network traffic data, providing insights into network activity. Each entry in the Dataflow table captures critical details, such as the source and destination of the traffic, direction (incoming or outgoing), and various feature values representing different aspects of network communication. These features may include protocol information, packet size, and other relevant attributes. The Dataflow table's structure is meticulously designed to facilitate efficient storage, retrieval, and analysis of network traffic data. Storing this detailed information allows the HNIDPS to learn from historical traffic patterns and identify anomalies that deviate from established norms. This data collection forms the foundation for the anomaly detection model, distinguishing legitimate network activity from potential cyberattacks within the smart home environment.
2) *Warnings Table*: The Warnings table serves as the HNIDPS's alert system, recording security incidents and dropped packet information. This table plays a critical role in notifying administrators and users of potential threats within the smart home network. Each entry in the Warnings table captures details about a security event, including timestamps, the nature of the detected anomaly (if any), and the source or destination IP addresses involved. Additionally, the Warnings table may store information about dropped packets, which could indicate attempts to evade detection or disrupt network communication. The table's meticulous logging provides a valuable audit trail for security analysis and incident response, allowing administrators to investigate suspicious activity, take appropriate actions, and maintain the smart home network's security posture.
3) *Signature Table*: The Signature Table is the cornerstone of the HNIDPS's signature-based intrusion detection system (SIDS) functionality. This table serves as a comprehensive repository for attack signatures, including pre-defined signatures and those imported from external sources like Proofpoint. These signatures act as predetermined patterns of malicious network activity that the SIDS can use to identify potential cyberattacks. Each entry in the Signature Table includes details such as a unique signature identifier (SID), a description of the signature, and the specific pattern it represents. This pattern may include a sequence of bytes within a packet, a specific network protocol usage, or a combination of various network traffic characteristics. The SIDS can efficiently identify known attack attempts by meticulously matching incoming network traffic against the pre-defined signatures stored in this table, raising alerts to mitigate potential security risks within the smart home network. The Signature Table may also include information about the attack's severity, allowing for prioritized response actions during security incidents.
4) *Signature Activate Table*: The Signature Activate Table complements the Signature Table by managing the activation status of individual signatures during automated imports from external sources such as Proofpoint. This table ensures that only relevant signatures are actively monitored by the SIDS. When a new batch of signatures is imported, the system checks the Signature Activate Table. If a signature's ID (SID) matches an entry in this table, the system uses the activation status defined there (enabled/disabled) instead of relying solely on rules from the external source. This mechanism allows administrators to customize and control the SIDS, prioritizing specific signatures based on the smart home network's security posture and potential threats.
5) *False Positive Table*: The False Positive Table plays a vital role in continuously improving the effectiveness of the ADIDS (anomaly-based intrusion detection system) of HNIDPS. This table acts as a storage for signatures that were previously identified as false positives by the ADIDS. False positives occur when legitimate network traffic is mistakenly flagged as anomalous. By meticulously logging these false positives in the False Positive Table, valuable insights can be gained about the

limitations of the current anomaly detection model. Each entry typically includes details about the specific signature, the network traffic flagged as anomalous, and timestamps associated with the event. This information empowers system administrators to fine-tune the ADIDS model by refining the detection logic or excluding specific features that frequently trigger false positives. Over time, the False Positive Table serves as a historical record of the learning process of the ADIDS. By continuously analyzing and addressing false positives, the system can be iteratively improved, resulting in a more accurate and robust anomaly detection capability for the HNIDPS. This ongoing process helps to reduce false alarms and ensure that the system effectively identifies genuine threats within the smart home network environment.

6) *IDS Options Table*: The IDS Options Table serves as the central configuration center for the HNIDPS, allowing administrators to customize the behavior of the intrusion detection system (IDS) according to the specific needs of the smart home network environment. This table stores various critical settings that govern how the IDS identifies and responds to potential threats. One important aspect of the IDS Options Table is the definition of trusted sources and destinations, which can include trusted IP addresses, network ranges, or domain names. Traffic from or to these trusted entities is usually exempt from stricter scrutiny, allowing authorized devices and services to operate smoothly within the network. The table also allows administrators to decide whether this trusted traffic information should be stored in the Dataflow table for further analysis. On the other hand, the IDS Options Table also facilitates the creation of blacklisted sources and destinations. These blacklisted entities represent known malicious actors or suspicious network locations. Network traffic from or to blacklisted entries can be subjected to more rigorous inspection by the IDS. Furthermore, the IDS Options Table provides granular control over the response mechanisms of the IDS. Administrators can specify whether the system should intercept or raise warnings for packets identified as anomalies by the ADIDS or the SIDS (Signature-Based Intrusion Detection System). This allows for a personalized response strategy based on the severity of the potential threat. Additionally, the IDS Options Table stores a critical threshold value used by the ADIDS model to classify a packet as an intrusion. This threshold determines the level of confidence required in the model's prediction before triggering an interception or warning. Finally, the table also stores timestamps for the last updates to the intrusion signature list and the false positive list. This information helps administrators stay updated on the current threat intelligence and anomaly detection capabilities of the system. By providing a comprehensive set of configuration options in the IDS Options Table, the HNIDPS enables administrators to establish a balanced and effective intrusion detection strategy for their smart home network environment.

7) *Users Table*: The Users Table forms the foundation of the user authentication system of HNIDPS. This table stores critical information about each registered user, including their full name, email address, and a securely hashed password. The password hashing mechanism ensures that user credentials are stored in a non-reversible format, enhancing the security of the system. Additionally, the Users Table includes two crucial user access control flags: "isSuperuser" and "isDeactive". The "isSuperuser" flag distinguishes between standard users and administrators with elevated privileges. Administrators typically have broader access rights within the HNIDPS, allowing them to manage other users. The "isDeactive" flag allows administrators to disable user accounts temporarily or permanently, further enhancing the security of the system by preventing unauthorized access.

C. RESTful API Server

The HNIDPS utilizes a RESTful API server to facilitate smooth communication between the user interface and PostgreSQL database. In this section, we will delve into the development of this server using the FastAPI framework. We will also explore its functionalities, including model creation, route establishment, and strong authentication mechanisms. The RESTful API server plays a vital role in enabling user interaction and system management within the HNIDPS by providing a secure and well-defined communication channel.

1) *Server Implementation*: The foundation for the HNIDPS's RESTful API server is the FastAPI framework [26]. Acting as a crucial intermediary, this server enables communication between the user interface and the PostgreSQL database. The server's implementation relies on various external Python libraries to achieve its functionalities. The "jose" library allows the server to securely handle JSON Web Tokens (JWT) for user authentication, providing a reliable mechanism to verify user identity during API requests. Additionally, the "passlib" library is utilized for password hashing, ensuring secure storage of user credentials in the database. The "psycopg" library bridges the gap between the API server and the PostgreSQL database, allowing for tasks such as database connection establishment and table creation (if required). To ensure data integrity and clarity in the API's communication, custom PydanticBaseModel classes are defined for each database table. These models serve as data validation schemas, ensuring that only expected data structures and formats are processed by the API server. This approach protects

against potential data manipulation attempts and guarantees that the server transmits information consistently and in a well-defined manner.

2) *Server Functionality:* The RESTful API server goes beyond basic communication and offers a comprehensive range of features for user interaction with the HNIDPS. A cornerstone of its functionality is the implementation of a robust OAuth 2.0 authentication system. This industry-standard protocol ensures secure user access by verifying user identity through access tokens. All routes within the API, except those providing server specifications, require OAuth authentication, safeguarding sensitive system information and functionalities from unauthorized access. To enhance the developer experience and facilitate seamless integration with client-side applications, the server automatically generates OpenAPI documentation (formerly known as Swagger API). This documentation is stored in the OpenAPI Specification JSON file, providing a clear and comprehensive overview of available API endpoints, request parameters, and expected data formats. Developers can utilize this documentation to efficiently integrate the HNIDPS's functionalities into their user interface applications. Furthermore, the server enables users to perform CRUD (Create, Read, Update, Delete) operations for each table in the PostgreSQL database, allowing authorized users to interact with the system's data through the API. Additionally, the server offers filtering capabilities based on specific fields, pagination options with limit and offset parameters, and other features that provide granular control over data retrieval and manipulation. This comprehensive functionality makes the RESTful API server a powerful tool for user interaction and system management within the HNIDPS.

D. Proxy Server

A proxy server serves as an intermediary between the smart home network and the internet in the HNIDPS. This section will explore the implementation of this proxy server and its role in the system's architecture. We will discuss how the proxy server integrates with MitmProxy addons to facilitate traffic inspection and manipulation for intrusion detection purposes. Additionally, we will cover the available configuration options and its role in connecting to the PostgreSQL database for threat signature management.

1) *MitmProxy Integration*: The MitmProxy addons play a crucial role in the proxy server's functionality within the HNIDPS. These addons transform the proxy server into a powerful Man-in-the-Middle (MitM) attack simulation tool, but for a legitimate purpose: network traffic inspection and intrusion detection. By integrating with MitmProxy, the proxy server gains the ability to intercept and analyze all network traffic flowing through the smart home network. This intercepted traffic can then be scrutinized for anomalies or suspicious patterns that may indicate potential cyberattacks.

2) *Proxy Server Configuration*: Configuration of Proxy Server The process of configuring the Proxy Server is crucial in customizing the intrusion detection behavior of HNIDPS to suit the specific requirements of the smart home network environment. This configuration utilizes the settings defined in the IDS Options Table. It enables administrators to establish a connection to the PostgreSQL database, which grants the proxy server access to crucial information for intrusion detection purposes. For example, the server can retrieve the most recent intrusion signatures stored in the Signature Table to identify potential threats based on predefined patterns of malicious network traffic. Additionally, the configuration process can integrate settings from the IDS Options Table, allowing the proxy server to adapt its behavior dynamically based on the preferences of the administrator. For instance, the configuration may instruct the proxy server to either intercept or simply raise warnings for packets flagged as anomalous by the anomaly detection system. By linking configuration options with database information, the proxy server configuration process empowers administrators to establish a comprehensive and adaptable intrusion detection strategy within HNIDPS.

3) *Signature-Based Intrusion Detection System (SIDS)*: The Signature-Based Intrusion Detection System (SIDS) in HNIDPS utilizes MitmProxy addons to efficiently identify known patterns of malicious network traffic. These addons enable the proxy server to act as a layer of packet inspection, carefully scrutinizing intercepted traffic against predefined intrusion signatures stored in the Signature Table. The implementation relies on the development of customized MitmProxy scripts that interact with the PostgreSQL database through the established connection configured during proxy server setup. When a network packet is intercepted, the script retrieves the latest intrusion signatures from the database, ensuring that the SIDS operates with the most up-to-date threat intelligence. The script then compares the features extracted from the intercepted packet with the retrieved intrusion signatures, meticulously matching byte sequences within the packet data, specific network protocol usage, or a combination of different traffic characteristics defined within the signatures. If a match is found between the intercepted packet and a known intrusion signature, the SIDS raises an alert within HNIDPS. This alert typically includes details about the detected threat, the associated signature, and the source of the suspicious traffic. By utilizing MitmProxy addons for packet interception and signature matching, the SIDS forms a critical layer of defense in HNIDPS's intrusion detection capabilities.

4) *Anomaly-Based Intrusion Detection System (ADIDS)*: The Anomaly-Based Intrusion Detection System (ADIDS) in HNIDPS operates within the MitmProxy addon framework to identify previously unseen or unknown malicious activities. Unlike SIDS, ADIDS focuses on analyzing network traffic patterns for deviations from established baselines. The MitmProxy addons for ADIDS utilize the pre-trained Convolutional Neural Network (CNN) model discussed in Section 2.1.3. When the proxy server intercepts a network packet, ADIDS extracts relevant features from the packet data, which may include similar elements analyzed by SIDS, such as protocol information, packet size, and other network traffic characteristics. These features are then fed into the CNN model for analysis. The model, trained on historical network traffic data, has learned to distinguish between normal and anomalous patterns. Based on the model's prediction, ADIDS determines whether the intercepted packet exhibits characteristics of an anomaly. However, to minimize false positives, an additional layer of verification is implemented. ADIDS checks the packet's attributes against the False Positive Table, which stores signatures of previously identified false positives - network traffic patterns that were mistakenly flagged as anomalous. If the packet's attributes match an existing false positive signature, ADIDS reclassifies the packet as legitimate traffic. On the other hand, if there is no match in the False Positive Table, ADIDS raises an alert, indicating a potential intrusion in the smart home network. This two-step verification process ensures the accuracy and reliability of ADIDS's anomaly detection capabilities.

E. User Interface (UI)

The User Interface (UI) serves as the central hub for user interaction with the HNIDPS, providing various design and functionality features. These components allow users to easily monitor network traffic, view security alerts, manage user accounts, and configure system settings. The intuitive design and informative visualizations of the UI help users understand the HNIDPS's operational status and effectively manage network security in a smart home environment.

1) *UI Framework Selection*: The decision to use Flutter as the UI framework for HNIDPS was based on several factors. The open-source nature of Flutter ensures transparency and community support, providing a wealth of resources and continuous development. Its popularity also means there is a large community of developers and readily available learning materials, making UI development and troubleshooting efficient. Additionally, Flutter is excellent for cross-platform development, allowing for a seamless UI experience across desktop, web, and mobile platforms. This versatility enhances accessibility and user convenience within the smart home environment.
2) *UI Functionality*: The HNIDPS UI offers various functionalities for effective network security management. A key feature is the implementation of CRUD operations for each table in the PostgreSQL database, allowing authorized users to manage system aspects like adding trusted devices, editing user accounts, and deleting obsolete security alerts. The UI also includes robust filtering options for data tables, enabling users to refine their view of system information by specific protocols, IP addresses, or timestamps. Pagination options with limit and offset parameters make navigating large datasets efficient. The comprehensive dashboard of the UI provides real-time visualizations of network traffic volume, security alerts, and system resource utilization, giving users a concise overview of the network's security posture. To ensure security and privacy, the UI includes a secure login page, where users can authenticate themselves using credentials stored in the Users Table. This mechanism ensures only authorized individuals can access and manage the HNIDPS. By offering CRUD operations, data filtering, informative dashboards, and secure authentication, the HNIDPS UI empowers users to effectively manage network security in a smart home environment.

IV. CONCLUSIONS

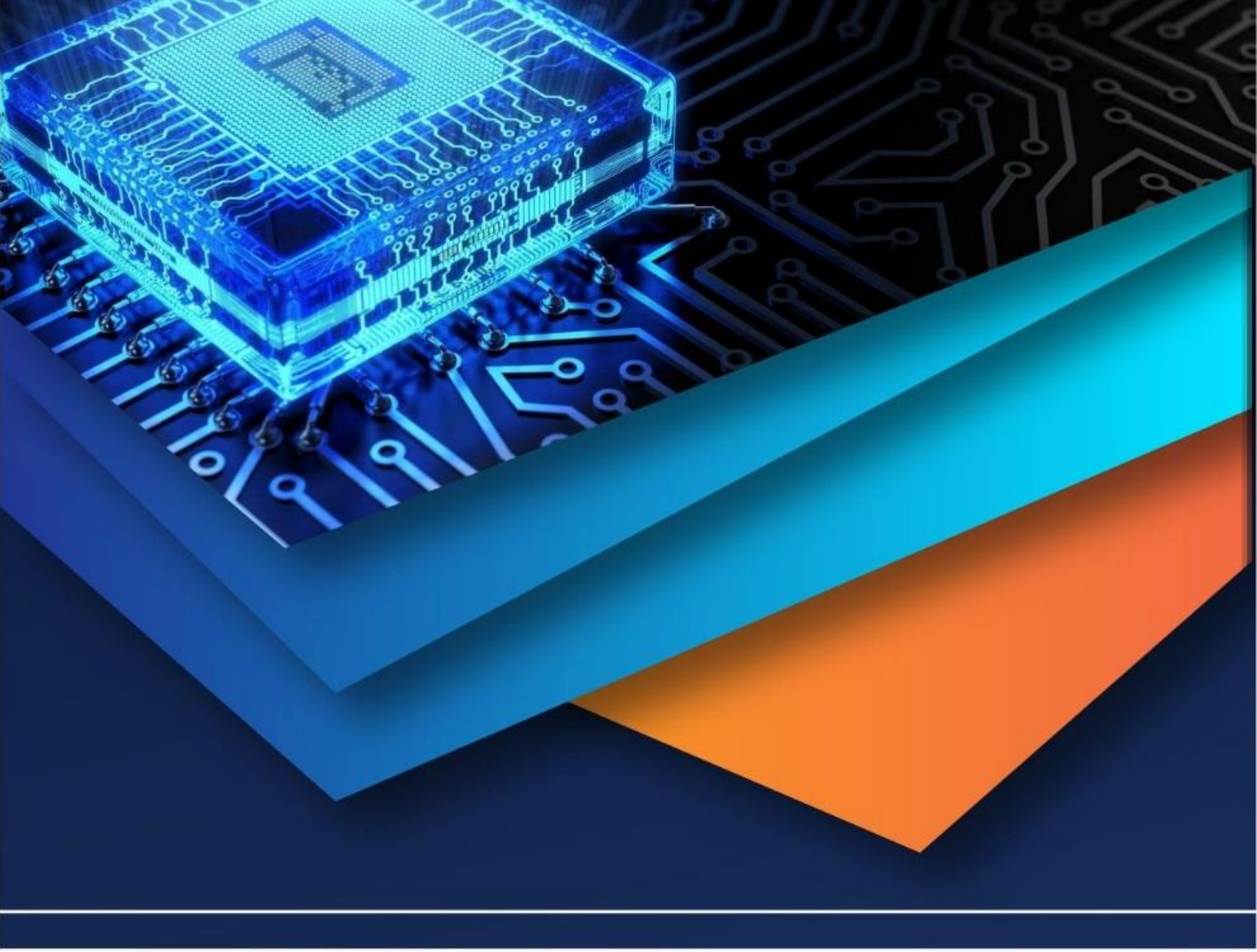
In this paper, a Home Network Intrusion Detection and Prevention System (HNIDPS) is introduced as a comprehensive solution for securing smart homes. The system implements a multi-layered architecture that combines signature-based and anomaly-based intrusion detection methods to effectively identify threats. Real-time traffic inspection is enabled through the use of a MitmProxy addon in the proxy server, while a Convolutional Neural Network model is utilized to detect new malicious activities. The secure RESTful API server ensures smooth communication between the user interface and the PostgreSQL database, allowing for efficient management of the system. The user-friendly interface allows users to monitor network traffic, access security alerts, and manage user accounts. By integrating these features, the HNIDPS has the potential to greatly improve the security of smart home networks. Future work could involve extensive testing of the system to validate its effectiveness in real-world situations. Additionally, further development could explore machine learning techniques for more advanced anomaly detection and integration with a wider variety of smart home devices.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

REFERENCES

[1] S. Zheng, N. Apthorpe, M. Chetty, and N. Feamster, "User perceptions of smart home IoT privacy," Proceedings of the ACM on human-computer interaction, vol. 2, no. CSCW, pp. 1–20, 2018.
[2] A. Lamba, S. Singh, N. Dutta, and S. Rela, "Uses of different cyber security service to prevent attack on smart home infrastructure," International Journal For Technological Research In Engineering, vol. 1, no. 11, 2014.
[3] T. A. Abdullah, W. Ali, S. Malebary, and A. A. Ahmed, "A review of cyber security challenges attacks and solutions for Internet of Things based smart home," Int. J. Comput. Sci. Netw. Secur, vol. 19, no. 9, p. 139, 2019.
[4] A. Patel, Q. Qassim, and C. Wills, "A survey of intrusion detection and prevention systems," Information Management & Computer Security, vol. 18, no. 4, pp. 277–290, 2010.
[5] K. Scarfone, P. Mell, and others, "Guide to intrusion detection and prevention systems (idps)," NIST special publication, vol. 800, no. 2007, p. 94, 2007.
[6] M. Uma and G. Padmavathi, "A survey on various cyber attacks and their classification.," Int. J. Netw. Secur., vol. 15, no. 5, pp. 390–396, 2013.
[7] I. Agrafiotis, J. R. Nurse, M. Goldsmith, S. Creese, and D. Upton, "A taxonomy of cyber-harms: Defining the impacts of cyber-attacks and understanding how they propagate," Journal of Cybersecurity, vol. 4, no. 1, p. tyy006, 2018.
[8] H. S. Lallie et al., "Cyber security in the age of COVID-19: A timeline and analysis of cyber-crime and cyber-attacks during the pandemic," Computers & security, vol. 105, p. 102248, 2021.
[9] A. Bendovschi, "Cyber-attacks–trends, patterns and security countermeasures," Procedia Economics and Finance, vol. 28, pp. 24–31, 2015.
[10] X. Li, X. Liang, R. Lu, X. Shen, X. Lin, and H. Zhu, "Securing smart grid: cyber attacks, countermeasures, and challenges," IEEE Communications Magazine, vol. 50, no. 8, pp. 38–45, 2012.
[11] T. Javid, T. Riaz, and A. Rasheed, "A layer2 firewall for software defined network," in 2014 Conference on Information Assurance and Cyber Security (CIACS), 2014, pp. 39–42.
[12] O. Rysavy, J. Rab, and M. Sveda, "Improving security in SCADA systems through firewall policy analysis," in 2013 Federated Conference on Computer Science and Information Systems, 2013, pp. 1435–1440.
[13] U. A. Sandhu, S. Haider, S. Naseer, and O. U. Ateeb, "A survey of intrusion detection & prevention techniques," in 2011 International Conference on Information Communication and Management, IPCSIT, 2011, vol. 16, pp. 66–71.
[14] V. Kumar and O. P. Sangwan, "Signature based intrusion detection system using SNORT," International Journal of Computer Applications & Information Technology, vol. 1, no. 3, pp. 35–41, 2012.
[15] Y. Sani, A. Mohamedou, K. Ali, A. Farjamfar, M. Azman, and S. Shamsuddin, "An overview of neural networks use in anomaly intrusion detection systems," in 2009 IEEE Student Conference on Research and Development (SCOReD), 2009, pp. 89–92.
[16] S. T. F. Al-Janabi and H. A. Saeed, "A neural network based anomaly intrusion detection system," in 2011 Developments in E-systems Engineering, 2011, pp. 221–226.
[17] U. Ravale, N. Marathe, and P. Padiya, "Feature selection based hybrid anomaly intrusion detection system using K means and RBF kernel function," Procedia Computer Science, vol. 45, pp. 428–435, 2015.
[18] Z. Chiba, N. Abghour, K. Moussaid, A. E. Omri, and M. Rida, "Newest collaborative and hybrid network intrusion detection framework based on suricata and isolation forest algorithm," in Proceedings of the 4th international conference on smart city applications, 2019, pp. 1–11.
[19] S. Ouiazzane, M. Addou, and F. Barramou, "A Suricata and Machine Learning Based Hybrid Network Intrusion Detection System," in Advances in Information, Communication and Cybersecurity: Proceedings of ICI2C'21, 2022, pp. 474–485.
[20] L. Mohammadpour, T. C. Ling, C. S. Liew, and A. Aryanfar, "A survey of CNN-based network intrusion detection," Applied Sciences, vol. 12, no. 16, p. 8162, 2022.
[21] J. Kim, J. Kim, H. Kim, M. Shim, and E. Choi, "CNN-based network intrusion detection against denial-of-service attacks," Electronics, vol. 9, no. 6, p. 916, 2020.
[22] R. Vinayakumar, K. Soman, and P. Poornachandran, "Applying convolutional neural network for network intrusion detection," in 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2017, pp. 1222–1228.
[23] B. Riyaz and S. Ganapathy, "A deep learning approach for effective intrusion detection in wireless networks using CNN," Soft Computing, vol. 24, no. 22, pp. 17265–17278, 2020.
[24] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, 2019, pp. 2623–2631.
[25] S. Shekhar, A. Bansode, and A. Salim, "A comparative study of hyper-parameter optimization tools," in 2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), 2021, pp. 1–6.
[26] M. Lathkar, High-Performance Web Apps with FastAPI. Springer, 2023.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)