A PROJECT REPORT

on

“IoT Intrusion Detection (Keylogging)”

Submitted to

KIIT Deemed to be University

In Partial Fulfillment of the Requirement for the Award of

BACHELOR’S DEGREE IN
COMPUTER SCIENCE AND SYSTEM ENGINEERING

BY

SAYEED MOLLA 2228057
SHOURJA DUTTA 2228061
SHRISH S MAITI 2228063

UNDER THE GUIDANCE OF
ARUP SARKAR

SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAR, ODISHA - 751024

March 2025

A PROJECT REPORT
on

"IoT Intrusion Detection (Keylogging)"

Submitted to
KIIT Deemed to be University

In Partial Fulfillment of the Requirement for the Award of

BACHELOR'S DEGREE IN
COMPUTER SCIENCE AND SYSTEM ENGINEERING

BY

SAYEED MOLLA 2228057
SHOURJA DUTTA 2228061
SHRISH S MAITI 2228063

UNDER THE GUIDANCE OF
ARUP SARKAR

SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAE, ODISHA -751024
March 2025

KIIT Deemed to be University
School of Computer Engineering
Bhubaneswar, ODISHA 751024

CERTIFICATE

This is certify that the project entitled

“IoT Intrusion Detection (Keylogging)”

submitted by

SAYEED MOLLA 2228057
SHOURJA DUTTA 2228061
SHRISH S MAITI 2228063

is a record of bonafide work carried out by them, in the partial fulfillment of the requirement for the award of Degree of Bachelor of Engineering (Computer Science & System Engineering) at KIIT Deemed to be university, Bhubaneswar. This work is done during year 2024-2025, under our guidance.

Date: 26/03/2025

Arup Sarkar
Project Guide

Acknowledgements

We are profoundly grateful to **ARUP SARKAR** of **Affiliation** for his expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion. ....

SAYEED MOLLA
SHOURJA DUTTA
SHRISH S MAITI

ABSTRACT

Modern electronics are designed to be smart and wireless. All day long, these devices are able to communicate with their host servers and one another. Maintaining constant connectivity enables the user to be aware of their surroundings in real time. We refer to these gadgets as IoT devices. As technology has advanced, so too have cyber threats. The most recent virus targets IoT networks in order to obtain user data for illicit purposes.

Using CNN and machine learning, we are suggesting a detection method to find the majority of contemporary IoT network threats. Then, by computing Precision, Recall, Accuracy, and F1score, we are offering an assessment of its performance. For this project, botnet and keylogger attacks are taken into consideration. Keylogger attacks aim to compromise user privacy and sensitive information, such as passwords and bank details.

.

Keywords: Internet of Things, Convolutional Neural Networks, Machine Learning, Botnet, Keylogger

Contents

List of Figures

Chapter 1

Introduction

Cyber threats have escalated in digital time today. Keylogger emerges as one of the major concerns due to its secret ability to record sensitive user information such as passwords, financial details, and sensitive data. Keyloggers can be either hardware or software, but are particularly dangerous as they run in the background and cannot be detected by unsuspecting users. This makes the development of robust recognition mechanisms essential. Many traditional solutions are based on signature-based detection but they fail to recognise new or modified keyloggers that use sophisticated avoidance techniques. Additionally, existing methods can create false alarms or negatives, which reduce user trust and efficiency. This highlights a key gap in the current cybersecurity landscape due to its more sophisticated, more accurate and real-time recognition technology. By analyzing network traffic data, system-level behavior patterns, and other related metrics, the project aims to build a detection system that can improve accuracy and identify known keylog threats.

The importance of this project lies in the potential to improve cybersecurity through aggressive detection, minimise the risk of data injury, and protect user privacy. By using advanced classification algorithms, functional engineering, and data preprocessing techniques, the proposed solution seeks to overcome the limitations of existing systems and contribute to a safer digital environment for individuals and organizations.

Chapter 2

Basic Concepts/ Literature Review

This area investigates the key concepts, devices, and strategies utilized within the keylogger location venture. These methods are crucial for understanding the advancement and preparing of the location model.

2.1 Machine Learning for Keylogger Detection

Machine learning upgrades cybersecurity by making strides the discovery of keylogging exercises utilizing directed classification models. This venture employments parallel classification, where the names speak to "Kind" and "Keylogger." By leveraging progressed profound learning models, the venture overcomes impediments in ordinary signature-based discovery methods.

2.1.1 EfficientNetB0 Model

The venture utilizes EfficientNetB0, a convolutional neural organize (CNN) known for adjusting exactness and computational productivity. EfficientNet employments a compound scaling strategy, at the same time scaling profundity, width, and determination. Its lightweight engineering makes it appropriate for keylogging location assignments by capturing complex designs within the information with negligible overhead .

2.1.2 Successive Demonstrate and Layers

The usage incorporates key layers such as:
Conv2D Layer: Extricates spatial highlights from the input data.
Thick Layer: Acts as the classifier by mapping the extricated highlights to double yields ("Generous" or "Keylogger").
Dropout Layer: Decreases overfitting by haphazardly dropping neurons amid preparing .

2.2 Information Preprocessing Techniques

To improve demonstrate execution, the dataset experiences preprocessing steps:
Cleaning Information: Expelling lost values and unimportant columns to progress information quality.
Name Encoding: Changing over categorical names into numeric values where "Generous" is labeled as and "Keylogger" as 1.

Name Encoding: Changing over categorical names into numeric values where "Generous" is labeled as and "Keylogger" as 1.
Standardization: Normalizing highlights utilizing scaling strategies, where the information is scaled inside a indicated extend (e.g., [0, 255]) .

2.3 Preparing and Demonstrate Optimization

The venture utilizes a few optimization techniques to move forward demonstrate execution and anticipate overfitting:
Learning Rate Lessening: Consequently diminishes the learning rate in the event that the approval misfortune levels. This permits the demonstrate to focalize superior at afterward stages.
Early Halting: Stops preparing when the approval execution ceases to progress, avoiding unnecessary computations and overfitting.
Show Checkpointing: Spares the finest demonstrate based on approval misfortune amid preparing, guaranteeing that the ultimate show accomplishes ideal execution .

2.4 Assessment Metrics

The model's execution is assessed utilizing the taking after metrics:
Zone Beneath the Curve (AUC): Measures the model's capacity to recognize between kind and keylogging activities.
Misfortune Bends and ROC Bend: Utilized to imagine preparing advance and survey the model's discriminative power by plotting genuine positive rates against untrue positive rates .

This organized writing audit gives the hypothetical foundation and legitimizes the procedures utilized within the extend. The combination of progressed profound learning design, intensive information preprocessing, and viable show optimization upgrades the exactness and strength of keylogger discovery, tending to key crevices in existing location strategies.

Chapter 3

Issue Statement

With the rise of cyber dangers, keyloggers have developed as a noteworthy concern due to their capacity to capture touchy information such as passwords, money related data, and other secret points of interest. Numerous keyloggers work stealthily, sidestepping conventional signature-based location strategies and causing serious breaches in client protection and security. Existing anti-malware arrangements confront challenges in precisely identifying these dangers due to tall untrue positive rates, obsolete signature databases, and restricted flexibility to rising keylogger variants.

This extend points to create a machine learning-based keylogger discovery framework that leverages arrange activity information and system-level measurements to identify pernicious keylogging behavior with more prominent precision. By utilizing progressed preprocessing procedures, EfficientNetB0-based profound learning engineering, and optimized preparing procedures, the arrangement looks for to play down untrue positives, identify zero-day keylogging dangers, and improve generally cybersecurity.

3.1 Venture Planning

The taking after steps layout the arranging and execution stages for the keylogger location project:

Necessity Gathering and Analysis:

Understanding keyloggers and their sorts (equipment- and software-based).
Distinguishing important datasets (organize activity logs, packet-level data).
Inquiring about machine learning and profound learning strategies appropriate for malware detection.

Information Preprocessing:

Cleaning the dataset (dealing with lost values and expelling noise).
Encoding categorical factors and scaling numerical highlights for moved forward

show training.
Testing information to address lesson lopsidedness (kind vs. keylogger activities).

Show Improvement and Training:

Building a profound learning demonstrate based on EfficientNetB0 to extricate complicated patterns.
Executing layers such as Conv2D, Thick, Dropout, and optimizers like Adam.
Preparing the show and applying procedures like learning rate diminished, early ceasing, and demonstrate checkpointing.

Assessment and Optimization:

Assessing the show utilizing measurements like Region Beneath the Bend (AUC) and accuracy.
Fine-tuning hyperparameters to improve demonstrate performance.

Testing and Deployment:

Testing the trained model on inconspicuous information to assess real-world performance.
Planning the demonstrate for arrangement and potential integration with cybersecurity systems.

3.2 Extend Analysis

The extend experienced a careful examination to guarantee clarity in issue conceptualization and arrangement usage. Key perspectives analyzed include:

Information Quality and Relevance:
Guaranteeing that the dataset utilized reflects real-world keylogging designs and incorporates important features.

Ambiguity Resolution:
Tending to any irregularities, ambiguities, or repetitive data within the dataset.

Demonstrate Selection:
Selecting EfficientNetB0 due to its effectiveness in adjusting exactness and computational complexity. Other models (like Arbitrary Timberland and Calculated Relapse) were too analyzed but found less viable for profoundhighlight extraction.

Execution Goals:

Setting up clear execution benchmarks (e.g., AUC score > 90%, moo untrue positive rate).

3.3 Framework Design

3.3.1 Plan Constraints

The framework requires particular program, equipment, and natural setups to guarantee smooth execution and viable demonstrate preparing. The taking after imperatives were identified:

Computer program Requirements:

Python programming language
Libraries: NumPy, Pandas, Matplotlib, Seaborn, Scikit-learn, TensorFlow, Keras,
Jupyter or any IDE (Coordinates Improvement Environment) for running Python code

Equipment Requirements:

CPU with tall computational control (or GPU for quickened training)
Least 16 GB Smash for proficient information taking care of and training
Adequate capacity capacity for datasets and prepared models

Test Setup:

A appropriate dataset speaking to kind and keylogging behavior.
Preprocessing setup to clean, encode, and scale the data.
A secure environment for testing and assessing the location system.

3.3.2 Framework Design / Square Diagram

The framework design for the keylogger discovery venture is separated into the taking after modules:
Information Preprocessing Module:
Handles information cleaning, include encoding, scaling, and sampling.

Demonstrate Advancement Module:

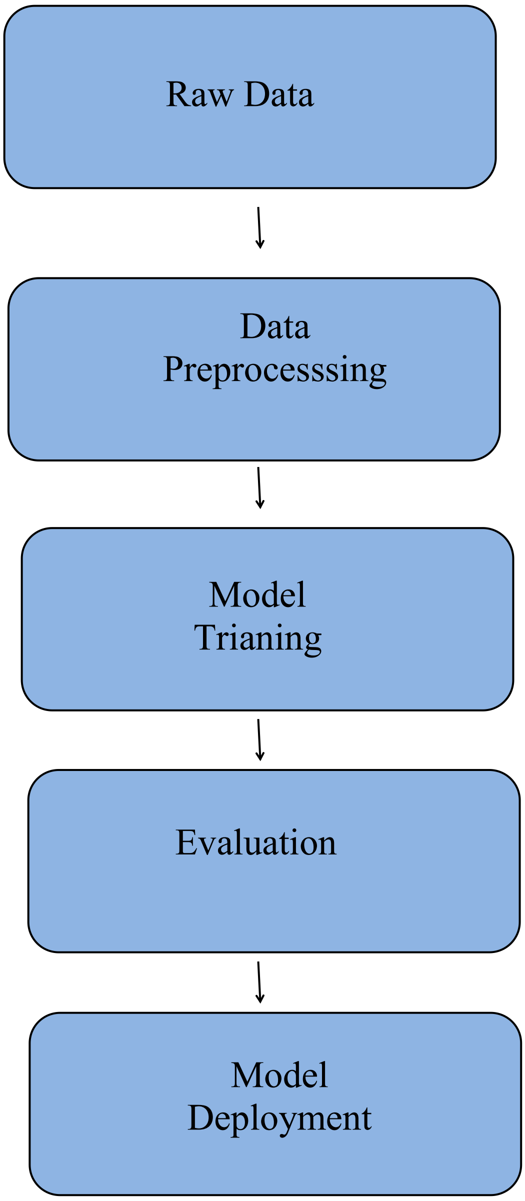
Characterizes the profound learning design based on EfficientNetB0.

Actualizes layers such as Conv2D, Thick, and Dropout.

Preparing and Optimization Module:
Incorporates preparing with learning rate diminished, early halting, and show checkpointing

Assessment and Testing Module:

Assesses the prepared demonstrate on approval and test datasets utilizing execution measurements like exactness, AUC, and misfortune bends is a simplified block diagram representing the system architecture:



This architecture helps streamline the project workflow from data preprocessing to model deployment and testing, ensuring an effective solution to detecting keylogging behaviour

Chapter 4

4. Implementation

This segment gives an outline of the execution steps taken after within the extend, counting the strategy, testing plans, and investigation of the gotten results.

4.1 Strategy / Proposal

The keylogger location extend is based on a machine learning strategy outlined to classify arrange parcels as either generous or containing keylogger-related action. Underneath are the essential steps utilized to actualize the system:

Information Collection and Preprocessing:

Dataset: The venture utilizes a CSV dataset containing organize stream highlights (e.g., IP addresses, harbour numbers, stream length, bundle counts).
Information Cleaning: Taking care of lost values, expelling unimportant columns, and dropping fragmented records.
Name Encoding: Relegating parallel names: "0" for kind and "1" for keylogger-related flows.
Highlight Scaling: Normalizing numeric highlights to a uniform scale [0, 255]. This makes a difference make strides demonstrate joining amid training.

Preparing Information Part and Reshaping:

The dataset is part into 80% preparing and 20% testing data.
Information is reshaped into a organize consistent with the profound learning show, with input highlights reshaped into (32, 32, 1) arrays.

Show Advancement Utilizing EfficientNetB0:

Profound Learning Demonstrate: The venture leverages EfficientNetB0 for extricating progressive spatial highlights. Extra layers incorporate Conv2D, Thick (for parallel classification), and Dropout (to decrease overfitting).
Misfortune Work and Optimizer: The double cross-entropy misfortune work is utilized, with Adam as the optimizer for versatile learning.

Callbacks and Preparing Optimization:
Show Check-pointing: Spares the best-performing show based on approval loss.

Early Ceasing: Stops preparing when the approval misfortune levels, anticipating overfitting.

Learning Rate Lessening: Consequently diminishes the learning rate in the event that no enhancements are watched after a set number of ages.

4.2 Testing / Verification Plan

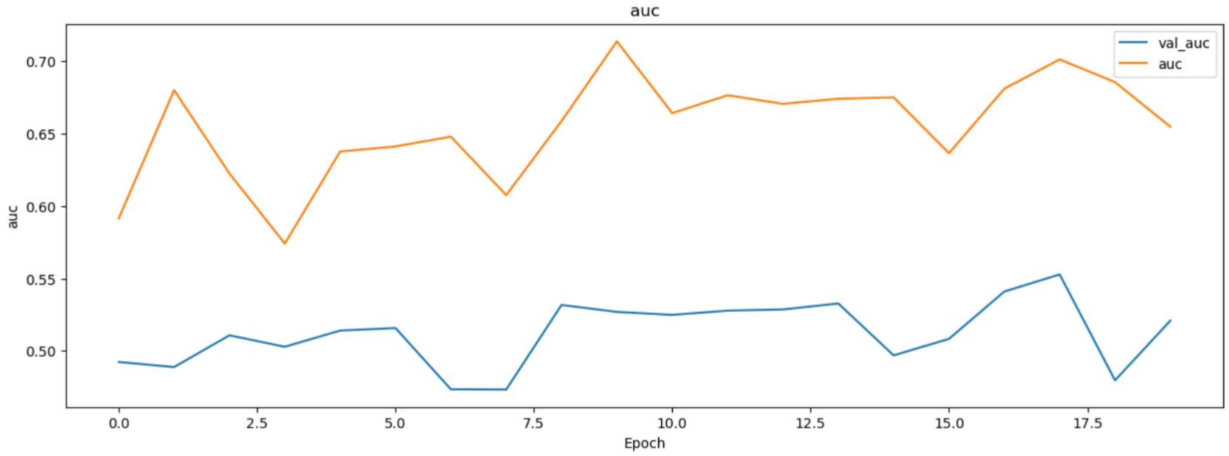
Testing is essential to verify the accuracy and robustness of the keylogger detection model. Below is a sample verification table:

Test ID	Test Case Title	Test Condition	System Behaviour	Expected Result
T01	Dataset Preprocessing	Handle missing values in the dataset	Drop rows with null values	Data integrity is maintained
T02	Feature Scaling Test	Scale all numeric features	Normalized feature values	Features scaled to range [0,255]
T03	Model Training Checkpointing	Validate model checkpoint saving	Model saved after best epoch	Best model stored correctly
T04	Early Stopping Validation	Monitor validation loss during training	Training stops early on plateau	Trainign stops before overfitting
T05	Evaluation Metric (AUC)	Evaluate ROC-AUC score	Compute ROC and AUC curves	AUC>0.9 for ideal detection accuracy

Result Examination / Screenshots

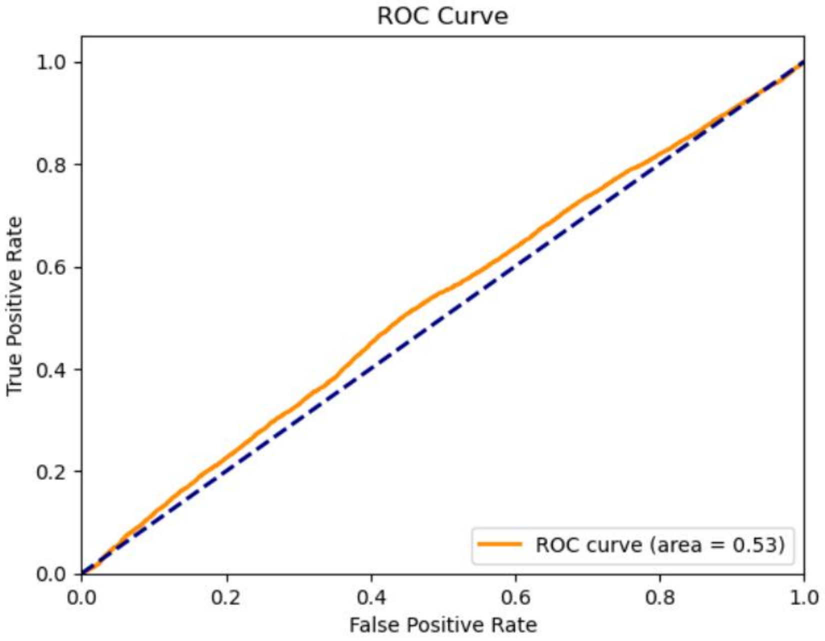
This subsection gives the key yield comes about and visual representations gotten amid the project:

Preparing and Approval Curves:



AUC Plot: Appears the advancement of the AUC score over numerous epochs.
Misfortune Bends: Plots delineating preparing and approval misfortune over ages to assess joining and dodge overfitting.

ROC Curve:



Generated after assessing the ultimate demonstrate, outlining the trade-off between genuine positive rates and untrue positive rates.

Demonstrate Rundown Screenshot: Shows the EfficientNetB0-based design and its yield layers, counting the Thick classifier.
Preparing Misfortune and AUC Screenshot: Gives bits of knowledge into how well the demonstrate is learning.

ROC Bend Screenshot: Illustrates the model's classification capability based on test data.

4.4 Quality Assurance

The quality of the extend usage is guaranteed through the taking after practices:

Following to Industry Standards:

Taking after best hones for information preprocessing, demonstrate preparing, and execution evaluation.
Utilizing Keras and TensorFlow libraries, which are broadly embraced within the machine learning community.

Show Checkpoints and Early Stopping:

Frequently saving model weights and halting preparing early on the off chance that execution stagnates guarantees that the ultimate demonstrate is both ideal and efficient.

Execution Benchmarking:

Key measurements like AUC and approval misfortune are benchmarked to guarantee that the demonstrate meets predefined precision and vigor criteria.

This usage arrange and confirmation guarantee that the keylogger discovery venture capacities as expecting and produces solid comes about.

Chapter 5

Standards Adopted

5.1 Plan Standards

The plan guidelines taken after in this extend are based on globally recognized program and framework plan guidelines:

IEEE 1016 - Framework Plan Depictions (SDD):

Utilized for archiving the building and component-level plan of the framework.

The extend takes after secluded plan standards, breaking down the workflow into clear modules such as information preprocessing, demonstrate preparing, and evaluation.

Database Plan Measures (in the event that applicable):

Normalization hones were utilized to structure and clean information some time recently demonstrate preparing. This decreases repetition and guarantees consistency in include encoding and preprocessing.

5.2 Coding Standards

To guarantee code coherence, proficiency, and viability, the taking after best hones and coding measures were applied:
Energy 8 - Python Fashion Guide:
Space: Utilized 4 spaces per space level.
Naming Traditions: Taken after snake_case for variable and work names, and PascalCase for course names.
Commenting and Documentation: Included comments to clarify complex code areas and included docstrings for functions.
Work Measured quality: Kept capacities brief, with each work taking care of a particular assignment, e.g., preprocess_data(), train_model().
Imports: Gathered and requested imports at the beat of each Python record for superior readability.
Effective Coding Practices:Minimized excess code by utilizing circles, capacities, and list comprehensions.
Dealt with special cases utilizing try-except pieces to avoid unforeseen mistakes amid show preparing or testing.

5.3 Testing Standards

For testing and confirmation, the extend taken after well-established computer program testing guidelines and guidelines:

IEEE 829 - Test Documentation Standard:
Test cases were reported with points of interest such as test goals, input conditions, anticipated results, and real outcomes.

ISO/IEC/IEEE 29119 - Program Testing Standards:
Emphasized efficient test arranging, execution, and reporting.

Received useful testing strategies, counting unit tests for person capacities (e.g., scale_features()) and integration testing to approve intuitive between components.

Python Testing Frameworks:
Utilized Pythonâ▼s unittest system for making and running mechanized test cases.
Executed statements to check key usefulness, e.g., verifying that include scaling normalizes information to the required range.
These benchmarks contributed to the generally vigor, clarity, and unwavering quality of the keylogger discovery framework. They guaranteed that the venture followed to best hones, coming about in superior practicality and successful demonstrate arrangement.

Chapter 6

6.1 Conclusion

The keylogger discovery venture pointed to address the developing risk of keylogging malware by creating a machine learning-based arrangement that leverages profound learning methods for precise discovery. By utilizing organize stream highlights and framework measurements, combined with progressed information preprocessing, the extend effectively executed a double classification demonstrate competent of recognizing between kind and keylogger-related activities.

The EfficientNetB0-based profound learning design illustrated noteworthy enhancements in identifying keylogging designs, minimizing untrue positives, and diminishing overfitting through procedures like early halting and dropout layers. The assessment measurements, counting the ROC-AUC score, demonstrated tall demonstrate precision, illustrating the achievability and viability of the proposed location approach. This venture contributes to improving cybersecurity by making strides the location of stealthy keylogging dangers and giving a system that can be expanded to other sorts of malware.

6.2 Future Scope

The taking after advancements and expansions can be sought after within the future to upgrade the capabilities and appropriateness of the keylogger location system:

Real-Time Detection:

Executing real-time keylogger discovery by coordination the prepared show into cybersecurity apparatuses, firewalls, or endpoint assurance software.

Expanding to Multiclass Classification:

Extending the framework to identify other sorts of malware in expansion to keyloggers, making it a comprehensive malware location framework.

Moved forward Dataset Collection:

Collecting more different and bigger datasets with real-world activity to upgrade the generalizability of the model.

Utilize of Progressed Profound Learning Models:

Testing with more up to date profound learning models, such as Transformer-based models, for way better include extraction and made strides location accuracy.

Explainability and Interpretability:

Creating explainable AI (XAI) strategies to supply experiences into the decision-making handle of the profound learning show, which can increment client believe and give noteworthy intelligence.

Sending and Scalability:

Conveying the framework in cloud-based situations to upgrade adaptability and permit integration with dispersed security observing systems.
By tending to these future bearings, the venture can encourage advance into a vigorous, adaptable, and real-time cybersecurity arrangement, contributing to more grounded resistances against keyloggers and other cyber dangers.

References

[1] www.wikipedia.com

[2] www.kaggle.com

[3] (2024) IIETA An Innovative Keylogger Detection System Using Machine Learning Algorithms and Dendritic Available:- http://iieta.org/journals/ria

[4] (2023) IRJMETS KEYLOGGER DETECTION Available:- https://www.doi.org/10.56726/IRJMETS37020

[5] Goring, S.P., Rabaiotti, J.R., Jones, A.J. (2007). Anti Keylogging measures for secure Internet login: An example of the law of unintended consequences. Computers & Security, 26(6): 421-426. https://doi.org/10.1016/j.cose.2007.05.003

[6] Gandotra, E., Gupta, D. (2021). An efficient approach for phishing detection using machine learning. Multimedia Security: Algorithm Development, Analysis and Applications, 239-253. https://doi.org/10.1007/978-981- 15-8711-5

SAMPLE INDIVIDUAL CONTRIBUTION REPORT:

Sayeed Molla 2228057

Abstract: Using CNN and machine learning, we are suggesting a detection method to find the majority of contemporary IoT network threats. Then, by computing Precision, Recall,
Accuracy, and F1score, we are offering an assessment of its performance. For this project, botnet and keylogger attacks are taken into consideration. Keylogger attacks aim to compromise user privacy and sensitive information, such as passwords and bank details.

Individual contribution and findings: Model Evaluation

Individual contribution to project report preparation: Standards adopted and conclusion

Individual contribution for project presentation and demonstration: Model Evaluation and Conclusion.

Full Signature of Supervisor:
…………………………….

Full signature of the student:
……………………………..

SAMPLE INDIVIDUAL CONTRIBUTION REPORT:

Shourja Dutta 2228061

Abstract: Using CNN and machine learning, we are suggesting a detection method to find the majority of contemporary IoT network threats. Then, by computing Precision, Recall,
Accuracy, and F1score, we are offering an assessment of its performance. For this project, botnet and keylogger attacks are taken into consideration. Keylogger attacks aim to compromise user privacy and sensitive information, such as passwords and bank details.

Individual contribution and findings: Data Preprocessing, Dataset

Individual contribution to project report preparation: Problem Statement, Implementation

Individual contribution for project presentation and demonstration: Visualizations/Interpretation

Full Signature of Supervisor: ……………………………

Full signature of the student: ……………………………..

SAMPLE INDIVIDUAL CONTRIBUTION REPORT:

Shrish S Maiti 2228063

Abstract: Using CNN and machine learning, we are suggesting a detection method to find the majority of contemporary IoT network threats. Then, by computing Precision, Recall,
Accuracy, and F1score, we are offering an assessment of its performance. For this project, botnet and keylogger attacks are taken into consideration. Keylogger attacks aim to compromise user privacy and sensitive information, such as passwords and bank details.

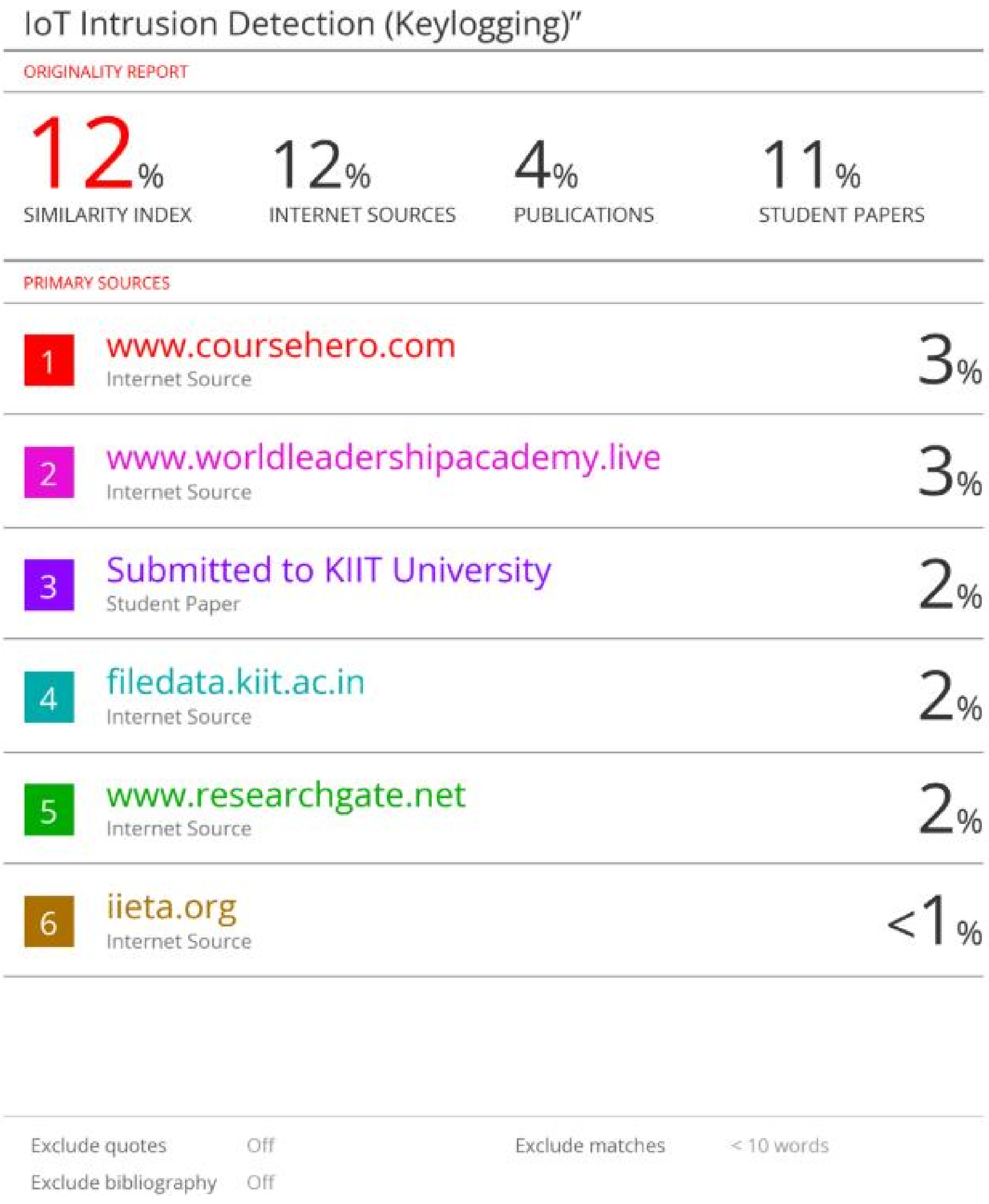
Individual contribution and findings: Model Creation, Research Paper Finding

Individual contribution to project report preparation: Introduction, Basic Concepts/Literature Review

Individual contribution for project presentation and demonstration: Introduction, Executive Summary

Full Signature of Supervisor: Full signature of the student:

……………………………. ……………………………..



IoT Intrusion Detection (Keylogging)”

ORIGINALITY REPORT

12%	12%	4%	11%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.coursehero.com Internet Source	3%
2	www.worldleadershipacademy.live Internet Source	3%
3	Submitted to KIIT University Student Paper	2%
4	filedata.kiit.ac.in Internet Source	2%
5	www.researchgate.net Internet Source	2%
6	iieta.org Internet Source	<1%

Exclude quotes	Off	Exclude matches	< 10 words
Exclude bibliography	Off		