ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

An Adaptive Federated Learning Model for Decentralized Intrusion Detection

Kennedy Omondi Okeyo[1]

[1]GMIA/ON/2048/09/16

[1]A Research Thesis Submitted to the Institute of Postgraduate Studies of Kabarak University in Partial Fulfillment of the Requirements for the Award of the degree of Master of Science in IT Security & Audit of Kabarak University

Publication Date: 2026/11/14

How to Cite: Kennedy Omondi Okeyo (2026) An Adaptive Federated Learning Model for Decentralized Intrusion Detection. *International Journal of Innovative Science and Research Technology*, 11(11), 193-336. https://doi.org/10.38124/ijisrt/26nov017

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

DECLARATION

➢ *I Do Hereby Declare that:*

- This thesis is my work, and as far as I know, it has not been presented for the award of a degree in any university or college.
- The work has not incorporated material from other works or a paraphrase of such material without due and appropriate acknowledgment.
- The work has been subjected to processes of anti-plagiarism and has met Kabarak University's 15% similarity index threshold.

I do understand that issues of academic integrity are paramount, and therefore, I may be suspended or expelled from the University or my degree may be recalled for academic dishonesty or any other related academic malpractices.

Signed:_  _ Date:02/04/2026

Name of Student: Kennedy Okeyo

Admission Number: GMIA/ON/2048/09/16

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

RECOMMENDATION

To the Institute of Postgraduate Studies:

The research thesis entitled “Development of Decentralized Intrusion Detection Model using Federated Learning,” written by Kennedy Okeyo, is presented to the Institute of Postgraduate Studies of Kabarak University. We have reviewed the research thesis and recommend it be accepted in partial fulfillment of the requirement for the award of the degree of Master of Science in Models security and Audit project.

Signed:_  Date:02/04/2026___Dr. Ruth Oginga

Department of Computer Science and Informatics Kabarak University

Signed:  _Date:02/04/2026___ Dr.Andrew Kipkebut

Department of Computer Science and Informatics Kabarak University

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

COPYRIGHT

© 2025

Kennedy Okeyo

All rights reserved. No part of this Thesis may be reproduced or transmitted in any form using either mechanical, including photocopying, recording, or any other information storage or retrieval Model without permission in writing from the author or Kabarak University.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to all individuals who contributed to the successful development of this research thesis.

First and foremost, my profound appreciation goes to my supervisor, Dr. Ruth Oginga and Dr. Andrew Kipkebut, for their invaluable guidance, unwavering support, constructive criticism, and insightful feedback throughout the conceptualization and drafting phases of this thesis. Their expertise and encouragement were instrumental in shaping this work.

My sincere thanks extend to Kabarak University for providing the academic environment and resources necessary for this research.

Finally, I wish to acknowledge my family and friends for their constant encouragement, understanding, and patience. Their belief in me was a significant source of motivation.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

DEDICATION

This research thesis is dedicated to:

All who strive to make the digital world a safer place through innovation and relentless

My beloved family, whose unwavering support, sacrifices, and belief in my academic pursuits have been my greatest strength.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

ABSTRACT

Centralized intrusion detection models face challenges relating to data privacy, scalability, and collaborative threat detection across distributed network environments. This study aimed to develop and evaluate a Decentralized Intrusion Detection Model (DIDM) based on Federated Learning to improve intrusion detection performance while maintaining the locality of raw network data. The study adopted the Design Science Research Methodology (DSRM). A centralized Artificial Neural Network (ANN)-based Intrusion Detection Model (CIDM) was established as the experimental baseline. Subsequently, a federated DIDM was designed and implemented using Federated Averaging (FedAvg), enabling distributed clients to train local ANN models and collaboratively update a global model without centralizing their raw training data. CICIDS2017 was used for the principal empirical evaluation. The performance of DIDM was compared with that of CIDM using an independent test dataset comprising 504,151 records. The DIDM achieved an accuracy of 98.27%, weighted precision of 98.32%, weighted recall of 98.27%, and weighted F1-score of 98.25%. The CIDM achieved an accuracy of 96.74%, weighted precision of 98.36%, weighted recall of 96.74%, and weighted F1-score of 97.39%. The DIDM recorded an accuracy improvement of 1.5317 percentage points over the CIDM. The exact McNemar test confirmed a statistically significant difference in paired classification outcomes ($p < 0.001$). However, lower detection performance in some minority attack classes highlighted the continuing challenge of class imbalance. The findings demonstrate that Federated Learning can support decentralized intrusion detection by enabling collaborative model training while retaining raw training data at distributed clients. Nevertheless, further evaluation under real-world network conditions is necessary to establish the model's scalability and resilience to adversarial threats.

Keywords: *Decentralized Intrusion Detection Model (DIDM), Federated Learning, Federated Averaging (FedAvg), Centralized Intrusion Detection Model (CIDM), Network Security.*

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

TABLE OF CONTENTS

DECLARATION 194
RECOMMENDATION 195
COPYRIGHT 196
ACKNOWLEDGEMENTS 197
DEDICATION 198
LIST OF FIGURES 204
LIST OF ABBREVIATIONS 205
CHAPTER ONE 206
INTRODUCTION 206
Background of the Study 206
Statement of the Problem 207
Justification of the Study 209
Significance of the Study 209
Significance to Cybersecurity Research 209
Significance to Cybersecurity Practitioners 210
Significance to Organizations 210
Significance to Policymakers and Information Governance Stakeholders 210
Significance to Future Researchers 210
Scope of the Study 210
Assumptions of the Study 211
Limitations of the Study and Mitigation Strategies 212
CHAPTER TWO 214
LITERATURE REVIEW 214
Introduction 214
Theoretical Review 214
Distributed Artificial Intelligence (DAI) Theory 214
Federated Learning Theory 215
Empirical Literature Review 215
Limitations of Traditional Intrusion Detection Models 215
Federated Learning-Based Intrusion Detection Models 216
Local Anomaly Detection Agents in Intrusion Detection Models 217
Evaluation of Intrusion Detection Models 218
Conceptual Framework 218
Chapter Summary 221
Research Gaps 222
CHAPTER THREE 223
RESEARCH METHODOLOGY 223
Introduction 223
Research Design 224
Study Environment 225
Population of the Study 227
Sampling Procedure and Sample Size 228
Data Partitioning 228
Expert Sampling 228
Relationship Between Original Dataset and Processed Distributions 229
Tools and Technologies 230
Data Collection and Preprocessing 232
Data Collection 232
Data Distribution Strategy 233

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Data Preprocessing Pipeline 233
Federated Data Handling Considerations 234
Summary of Data Preparation Approach 234
Model Development and Evaluation Framework 235
Centralized Intrusion Detection Model (CIDM) 235
Decentralized Intrusion Detection Model (DIDM) 236
Local Anomaly Detection Models 237
Ensemble-Based Detection Framework 237
Model Evaluation 238
Expert Validation 239
Data Analysis Methods 240
DIDM Model Architecture and Representation 240
Summary of the Model Architecture 243
Chapter Summary 244
Ethical Considerations 244
Confidentiality and Data Protection 245
Research Integrity and Responsible Conduct 246
CHAPTER FOUR 247
DATA ANALYSIS, INTERPRETATION, AND PRESENTATION 247
Introduction 247
Dataset Description and Preparation 247
Federated Learning Implementation Results 249
Analysis of Experimental Results 251
Comparative Performance Evaluation of the CIDM and DIDM 251
Classification Performance Analysis Using the Confusion Matrix 253
Precision–Recall Curve Analysis of the Developed DIDM 257
Model Implementation and Runtime Results 259
Real-Time Operation of the DIDM/DIDS Model 259
Model Initialization and Startup Process 261
Real-Time Alert Generation 262
SIEM Event Correlation 264
SOAR Automated Response Execution 265
Model Structural Organization 267
End-to-End Execution Flow 268
Federated Learning Workflow 270
High-Level Architecture 274
Intrusion Detection Workflow 275
SIEM Processing Workflow 276
SOAR Incident Response Workflow 277
Integrated End-to-End Workflow 278
Performance Evaluation and Operational Demonstration of the DIDM Model 279
Detection Accuracy Trend 280
Latency Performance 281
False Positive Rate (FPR) Performance 282
Overall Performance Summary 284
Expert Validation Analysis 287
Identified Weaknesses of Existing IDM 287
CHAPTER FIVE 303
SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS 303
Introduction 303

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Recommendations 310
Recommendations for Organizations 310
Recommendations for Cybersecurity Practitioners 310
Recommendations for Policymakers and Regulatory Agencies 311
Recommendations for Academic and Training Institutions 311
Recommendations for Future Researchers 311
General Recommendation 312
Limitations of the Study 312
Experimental Environment 312
Dataset Limitations 312
Federated Learning Implementation 313
Computational and Communication Resources 313
Scope of Performance Evaluation 313
Security Threat Coverage 313
Operational Demonstration and Integration 314
Overall Limitation 314
Suggestions for Further Research 314
Large-Scale Federated Deployment 314
Application to IoT, Edge, and Cloud Environments 315
Explainable Artificial Intelligence for Intrusion Detection 315
Privacy-Preserving Federated Learning 315
Robustness Against Adversarial Attacks 315
Communication-Efficient Federated Learning 315
Adaptive Learning for Evolving Threats 316
Blockchain-Enabled Federated Learning 316
Evaluation Against a Broader Range of Threats 316
Governance, Compliance, and Accountability 316
Cost-Benefit and Operational Feasibility 316
Comparative Evaluation of Emerging AI Models 317
Chapter Summary 317
REFERENCES 319
APPENDICES 321
APPENDIX II 324
NACOSTI RESEARCH LICENSE 324
APPENDIX III 325
QUESTIONNAIRE FOR CYBERSECURITY EXPERTS 325
APPENDIX IV 329
INFORMED CONSENT FORM 329
APPENDIX V 333
ETHICAL REVIEW LOW RISK APPLICATION FORM 333

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

LIST OF TABLES

Table 1: Research Workplan……...222
Table 2: Dataset Summary……..247
Table 3 Distribution of Network Traffic Classes……248
Table 4 Federated Learning Training Performance Across Communication Rounds……249
Table 5 Presents the Comparative Performance of the Two Models……..252
Table 6 Receiver Operating Characteristic (ROC) Performance of the Developed DIDM……256
Table 7 Statistical Validation of DIDM Performance Compared with CIDM……..258
Table 8 Expert-Identified Limitations of Existing IDM……..288
Table 9 Expert-Identified Design Requirements for Next-Generation Intrusion Detection Systems…….291
Table 10 Priority Ranking of Cybersecurity Attack Types Based on Expert Assessment……294
Table 11 Alignment Between Expert Expectations and DIDM Model Performance Outcomes……296

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

LIST OF FIGURES

Figure 1 Conceptual Framework of the Decentralized Intrusion Detection Model……220
Figure 2 Input–Process–Guidance–Model Output (IPGS) Conceptual Framework……220
Figure 3 Conceptual and Methodological Flow of the DIDM Study. This diagram illustrates how insights from the literature review inform the study objectives, research methodology, and expected outcomes……222
Figure 4 Python Libraries and Environment Used for DIDM Implementation……232
Figure 5 Memory-Efficient Data Loading and Preprocessing for CICIDM2017……235
Figure 6 Global ANN Model……238
Figure 7 Input–Process–Guidance–Model Output (IPGS) Conceptual Framework……241
Figure 8 DIDM Model Architecture……241
Figure 9 Federated Learning Model Workflow (FedAvg-Based Training Process)……241
Figure 10 AI Agent Framework in the DIDM Model……242
Figure 11 Ensemble-Based Intrusion Detection Model Pipeline……242
Figure 12 Federated Learning Model Testbed (Docker + Flower Environment)……243
Figure 13 Distribution of Network Traffic Classes in the CICIDS2017 Dataset……248
Figure14 (a): Federated Training Accuracy Across Communication Rounds……250
Figure14 (b): Federated Training Loss Across Communication Rounds……250
Figure15 (a): Comparative Performance of the CIDM and DIDM……252
Figure15 (b): Confusion Matrix for the Centralized Intrusion Detection Model (CIDM)……253
Figure 15 (c): Confusion Matrix for the Developed Decentralized Intrusion Detection Model (DIDM)……254
Figure 15 (d) Receiver Operating Characteristic (ROC) Curves of the Developed Decentralized Intrusion Detection Model (DIDM)……255
Figure 15 (e): Precision–Recall Curves for the Developed Decentralized Intrusion Detection Model (DIDM)……257
Figure 16 Overall Operational View of the DIDM/DIDS SOC Model……260
Figure 17 Initialization and Startup Logs of the DIDM SOC Model……261
Figure 18 Real-Time Alert Output Generated by the DIDM SOC Model……263
Figure 19 SIEM Event Correlation Output of the DIDM SOC Model……264
Figure 20 SOAR Automated Response Execution of the DIDM SOC Model……265
Figure 21 Structural Organization of the DIDM SOC Model……267
Figure 22 End-to-End Execution Flow of the DIDM SOC Model……268
Figure 23 Federated Learning Work Flow of the DIDM SOC Model……270
Figure 24 Real-Time Security Operations Center (SOC) Dashboard……273
Figure 25 High-Level Architecture of the DIDM SOC Model……274
Figure 26 Intrusion Detection Workflow of the DIDM Model……275
Figure 27 SIEM Processing Workflow of the DIDM Model……276
Figure 28 SOAR Incident Response Workflow of the DIDM Model……277
Figure 29 Integrated End-to-End Workflow of the DIDM Model……277
Figure 30 Detection Accuracy Trend of the DIDM Model……280
Figure 31 Latency Reduction Trend of the DIDM Model……281
Figure 32 False Positive Rate (FPR) Performance Comparison……283
Figure 33 Overall Performance Summary of the DIDM Model……284
Figure 34 Confusion Matrix Comparison Between CIDS and DIDM……285
Figure 35 ROC Curve Comparison Between CIDS and DIDM……285
Figure 36 Precision–Recall Curve Comparison Between CIDM and DIDM……286
Figure 37 (a) Frequency of Expert-Identified IDM Limitations……288
Figure 37(b): Polar Representation of Expert-Identified IDM Limitations……289
Figure 37(c): Heatmap Representation of Expert-Identified IDM Limitations……289
Figure 38(a): Frequency Distribution of Expert-Identified Design Requirements……291
Figure 38(b): Ranked Representation of Expert-Identified Design Requirements……292
Figure 38(c): Heatmap Representation of Expert-Identified Design Requirements……292
Figure 39(a): Frequency Distribution of Expert-Identified Priority Attack Types……294
Figure 39(b): Ranked Representation of Expert-Identified Priority Attack Types……295
Figure 39(c): Heatmap Representation of Expert-Identified Priority Attack Types……295
Figure 40(a): DIDM Model Alignment with Quantitative Performance Expectations……298
Figure 40(b): Comparative Representation of DIDM Quantitative Performance Measures……299
Figure 40(c): DIDM Quantitative Performance Alignment Heatmap……299

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

LIST OF ABBREVIATIONS

AI – Artificial Intelligence

AUC – Area Under the Curve

CICIDM – Canadian Institute for Cybersecurity Intrusion Detection Model Dataset

DIDM/DIDS – Decentralized Intrusion Detection Model

DSRM – Design Science Research Methodology

EWMA – Exponentially Weighted Moving Average

FL – Federated Learning

FL Server – Federated Learning Server

GDPR – General Data Protection Regulation

IDM – Intrusion Detection Model

IoT – Internet of Things

NSL-KDD – Network Security Laboratory – Knowledge Discovery in Databases Dataset

OC-SVM – One-Class Support Vector Machine

ROC – Receiver Operating Characteristic

SVM – Support Vector Machine

UNSW-NB15 – University of New South Wales – New Benchmark 2015 Dataset

➢ *Model & Simulation Components*

Containernet – Container-based network emulation platform extending Mininet

Edge Node – A decentralized network node performing local intrusion detection and model training

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

CHAPTER ONE
INTRODUCTION

➢ *Background of the Study*

The rapid advancement of digital technologies has transformed the manner in which organizations store, process, and exchange information. The widespread adoption of cloud computing, the Internet of Things (IoT), edge computing, and distributed enterprise systems has increased connectivity and operational efficiency while simultaneously expanding the cybersecurity threat landscape (Musa et al., 2020; Al-Garadi et al., 2020). As organizational networks become increasingly interconnected and geographically distributed, the volume, diversity, and complexity of network traffic generated by users, devices, applications, and services continue to increase. This development has made the protection of digital infrastructure and information assets an increasingly important cybersecurity requirement.

Intrusion Detection Models (IDMs) constitute an important component of modern cybersecurity because they continuously monitor network activities and identify patterns that may indicate malicious or abnormal behaviour. Conventional intrusion detection approaches have predominantly relied on centralized architectures, in which network traffic, logs, and security events from multiple endpoints are transmitted to a central location for analysis (Ayodeji et al., 2020). Although centralized approaches have demonstrated effectiveness in controlled environments, their suitability becomes increasingly constrained as network infrastructures expand and become more distributed.

Centralized intrusion detection architectures present several operational and security challenges. The aggregation of large volumes of network data at a central processing location can create computational and communication bottlenecks, particularly as the number of connected devices increases. The dependence on centralized data collection may also increase exposure of sensitive network information during transmission and storage. In addition, the concentration of detection and analysis functions within a central infrastructure creates a potential single point of failure, which may affect the availability and resilience of the detection process if the central component is compromised or becomes unavailable. These challenges are particularly significant in distributed environments where organizations operate across multiple locations and generate heterogeneous network traffic.

Decentralized intrusion detection approaches have emerged as an alternative to centralized architectures by distributing detection and analysis capabilities across multiple network nodes. Under such architectures, individual nodes can perform local monitoring and analysis, thereby reducing dependence on a single processing location and potentially improving responsiveness. However, decentralized approaches that operate independently may have limited mechanisms for sharing knowledge among participating nodes. Consequently, models trained using localized data may exhibit inconsistent detection performance because each node learns from its own traffic patterns and attack characteristics without benefiting from information acquired by other nodes (Rezaeifar et al., 2020; Kasongo & Sun, 2020).

Federated Learning (FL) provides a promising approach for enabling collaborative machine learning across distributed environments. Unlike conventional centralized learning, Federated Learning allows participating nodes to train models using locally available data while sharing model updates rather than transferring raw data to a central repository. A central aggregation mechanism can then combine the locally trained model parameters to produce an improved global model that is redistributed to participating nodes for subsequent training rounds (Wang & Liu, 2020). This approach supports collaborative learning while maintaining data locality and reducing the need to centralize raw network traffic.

The application of Federated Learning to intrusion detection therefore provides an opportunity to address some of the limitations associated with centralized and independently operating decentralized models.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Through collaborative model training, distributed nodes can benefit from knowledge learned across different network environments without directly exchanging their raw network traffic. When combined with artificial intelligence-based detection techniques, Federated Learning can provide a framework for developing intrusion detection models that are suitable for distributed environments and capable of learning from heterogeneous network conditions.

Artificial intelligence techniques have increasingly been applied to intrusion detection because of their ability to identify complex patterns within large volumes of network traffic. Anomaly detection methods such as Autoencoders, Isolation Forest, and One-Class Support Vector Machines (One-Class SVM) provide different mechanisms for identifying deviations from learned patterns of normal network behaviour. The combination of multiple detection techniques can potentially improve detection reliability because different algorithms may capture different characteristics of anomalous traffic.

Against this background, this study developed and evaluated a Decentralized Intrusion Detection Model (DIDM) based on Federated Learning. The model incorporated the Federated Averaging (FedAvg) algorithm to support collaborative model training across distributed nodes while maintaining data locality. The DIDM further incorporated artificial intelligence-based anomaly detection techniques, including Autoencoders, Isolation Forest, and One-Class Support Vector Machines, to support local anomaly identification. The model was implemented and evaluated within a controlled distributed computing environment using established benchmark intrusion detection datasets. Its performance was compared with that of a Centralized Intrusion Detection Model (CIDM) using appropriate detection and operational performance metrics. The study therefore sought to establish whether a Federated Learning-based decentralized architecture could provide improved intrusion detection performance while addressing selected limitations associated with centralized intrusion detection architectures.

➢ *Statement of the Problem*

The rapid growth of distributed computing environments, cloud services, Internet of Things (IoT) devices, and interconnected enterprise networks has significantly increased the volume and complexity of network traffic that organizations must continuously monitor. This expansion has also increased the opportunities available to cyber attackers and created a growing need for effective intrusion detection models capable of identifying malicious activities in complex and distributed network environments.

Conventional centralized intrusion detection architectures typically depend on collecting network traffic and security information from multiple sources and transferring the information to a central processing location. Although centralized architectures simplify the management and analysis of security information, their dependence on a central processing infrastructure presents several challenges. The increasing volume of network traffic can create processing and communication bottlenecks, while centralized collection of security data may increase the exposure of sensitive information. Furthermore, the dependence on a central infrastructure creates a potential single point of failure that may affect the availability and resilience of intrusion detection operations.

Decentralized intrusion detection approaches have attempted to address some of these limitations by distributing detection activities across multiple network nodes. However, independently operating decentralized models may not effectively share knowledge acquired from different network environments. As a result, individual nodes may learn from limited and heterogeneous data distributions, leading to variations in detection performance across participating nodes. The absence of effective collaborative learning mechanisms may therefore limit the ability of decentralized models to generalize across diverse network environments.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Federated Learning has emerged as a promising approach for enabling collaborative model development across distributed data sources without requiring the direct transfer of raw data. However, the application of Federated Learning within decentralized intrusion detection remains an area requiring further empirical investigation. In particular, there is a need for models that integrate federated collaborative learning with complementary anomaly detection techniques and that are evaluated systematically against centralized approaches using multiple benchmark datasets and performance indicators.

The existing research gap is therefore concerned with the need for a unified decentralized intrusion detection framework that can combine collaborative Federated Learning with local artificial intelligence-based anomaly detection while maintaining data locality. In addition, there is a need for empirical evaluation that considers not only conventional classification performance but also operational characteristics relevant to distributed environments, including false positive rates, detection latency, scalability, and the effectiveness of collaborative learning.

This study addressed this gap by designing, implementing, and evaluating a Decentralized Intrusion Detection Model (DIDM) based on Federated Learning. The developed model combined collaborative learning through the Federated Averaging algorithm with local anomaly detection techniques, including Autoencoders, Isolation Forest, and One-Class Support Vector Machines. The model was evaluated using established intrusion detection benchmark datasets and compared with a Centralized Intrusion Detection Model (CIDM). The evaluation considered multiple performance measures to determine the effectiveness and operational suitability of the DIDM within a simulated distributed network environment.

➢ *Objectives of the Study*

- *General Objective*

The general objective of this study was to develop and evaluate a Federated Learning-based Decentralized Intrusion Detection Model (DIDM) for improving intrusion detection performance in distributed network environments.

- *Specific Objectives*

The study was guided by the following specific objectives:

✓ To establish a centralized intrusion detection performance baseline using an Artificial Neural Network (ANN) in a controlled experimental environment.
✓ To design a Federated Learning-based Decentralized Intrusion Detection Model (DIDM) that enables collaborative neural-network training across distributed clients while retaining raw training data locally.
✓ To implement the developed DIDM using the Federated Averaging (FedAvg) algorithm across simulated distributed computational clients.
✓ To evaluate and compare the intrusion detection performance of the developed DIDM and the Centralized Intrusion Detection Model (CIDM) using established classification performance metrics.

- *Research Questions*

The study sought to answer the following research questions:

✓ What intrusion detection performance does the centralized Artificial Neural Network (ANN) achieve in the controlled experimental environment?
✓ How can a Federated Learning-based Decentralized Intrusion Detection Model (DIDM) be designed to support collaborative neural-network training while retaining raw training data locally?
✓ How can the developed DIDM be implemented using the Federated Averaging (FedAvg) algorithm across simulated distributed computational clients?

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

✓ How does the intrusion detection performance of the developed DIDM compare with that of the Centralized Intrusion Detection Model (CIDM) based on established classification performance metrics?

➢ *Justification of the Study*

The increasing frequency and complexity of cyber threats have highlighted the need for intrusion detection approaches that can operate effectively within increasingly distributed digital environments. Organizations now rely on interconnected infrastructures involving cloud services, distributed applications, IoT devices, and geographically dispersed computing resources. These environments generate large and heterogeneous volumes of network traffic that require continuous monitoring and analysis.

Although centralized intrusion detection models have provided important cybersecurity capabilities, their dependence on centralized data collection and processing may create challenges related to scalability, communication overhead, data management, and system resilience. As network environments expand, the concentration of security data and analysis functions at a central location may limit the ability of the detection architecture to respond efficiently to increasingly distributed network activities.

Decentralized intrusion detection provides an alternative approach by distributing monitoring and analysis capabilities across multiple nodes. However, independently operating detection models may lack effective mechanisms for sharing knowledge among participating nodes. This limitation may result in inconsistent detection capabilities, particularly when network nodes observe different traffic patterns and attack distributions.

Federated Learning provides an alternative collaborative learning paradigm that enables multiple nodes to participate in the development of a shared model while retaining their local training data. This characteristic makes Federated Learning relevant to distributed intrusion detection because network traffic can remain within local environments while model knowledge is collaboratively aggregated.

The study was therefore justified by the need to investigate whether the integration of Federated Learning and artificial intelligence-based anomaly detection could provide a more effective approach to intrusion detection in distributed environments. The DIDM developed in this study was designed to combine the advantages of decentralized data processing with collaborative model learning.

The study also provided an opportunity to empirically compare decentralized and centralized intrusion detection approaches using standardized benchmark datasets and multiple evaluation measures. Such comparison was important for determining whether the proposed architectural approach produced measurable improvements in detection performance and selected operational characteristics.

The findings of the study therefore contribute evidence on the practical application of Federated Learning within decentralized intrusion detection. The research also provides a foundation for future investigations into privacy-aware collaborative learning, distributed cybersecurity architectures, and scalable intrusion detection approaches for modern network environments.

➢ *Significance of the Study*

The study contributes to the field of cybersecurity by providing empirical and practical insights into the development and evaluation of a Federated Learning-based Decentralized Intrusion Detection Model (DIDM).

- *Significance to Cybersecurity Research*

From a research perspective, the study contributes to existing knowledge on decentralized intrusion detection by examining the integration of Federated Learning with artificial intelligence-based anomaly

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

detection techniques. The research provides empirical evidence on the performance of a decentralized collaborative learning architecture compared with a centralized baseline.

The study further contributes to research on distributed artificial intelligence by demonstrating how collaborative model training can be implemented while maintaining the locality of network traffic data. The resulting findings may provide a basis for further research on Federated Learning applications within cybersecurity and distributed machine learning.

- *Significance to Cybersecurity Practitioners*

For cybersecurity practitioners, the developed DIDM provides a conceptual and experimental framework for implementing collaborative intrusion detection across distributed network nodes. The architecture demonstrates how local detection capabilities can be combined with a globally aggregated model to support collaborative threat identification.

The findings may assist cybersecurity professionals in understanding the potential benefits and limitations of decentralized learning architectures when compared with conventional centralized intrusion detection approaches.

- *Significance to Organizations*

Organizations operating distributed digital infrastructures may benefit from the findings of the study by gaining insights into alternative approaches for deploying intrusion detection capabilities across multiple network environments. The decentralized architecture investigated in this research provides an approach in which participating nodes retain their local network data while contributing model updates to a collaborative learning process.

The study may therefore inform future organizational decisions concerning the design of distributed intrusion detection architectures, particularly where network data locality and distributed processing are important considerations.

- *Significance to Policymakers and Information Governance Stakeholders*

The study is also relevant to policymakers and information governance stakeholders concerned with cybersecurity and responsible data management. Federated Learning provides a framework in which collaborative model development can occur without requiring direct centralization of raw training data. Although Federated Learning does not by itself guarantee complete privacy protection, its data-locality principle may support privacy-aware system architectures when combined with appropriate security controls.

The findings may therefore contribute to discussions concerning the development of cybersecurity policies, data governance practices, and security architectures for distributed computing environments.

➢ *Significance to Future Researchers*

The study provides a foundation for further research on Federated Learning-based intrusion detection. Future researchers may build upon the developed DIDM architecture by investigating larger numbers of participating clients, more complex non-IID data distributions, advanced privacy-preserving mechanisms, adversarial attacks against federated systems, secure aggregation, and real-world enterprise deployment.

➢ *Scope of the Study*

The study focused on the design, implementation, and evaluation of a Federated Learning-based Decentralized Intrusion Detection Model (DIDM) for distributed network environments.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The study incorporated the Federated Averaging (FedAvg) algorithm to support collaborative model training among distributed nodes. The DIDM incorporated artificial intelligence-based detection techniques, including Autoencoders, Isolation Forest, and One-Class Support Vector Machines, to support anomaly detection at the local level. The model was implemented using Python-based machine learning and deep learning technologies within a controlled distributed computing environment.

The experimental evaluation utilized the CICIDS2017, NSL-KDD, and UNSW-NB15 benchmark intrusion detection datasets. The datasets provided network traffic records representing normal and malicious activities and were used for model development and evaluation.

The performance of the DIDM was evaluated using appropriate classification and operational metrics. These included accuracy, precision, recall, F1-score, false positive rate, detection latency, Receiver Operating Characteristic Area Under the Curve (ROC-AUC), and Average Precision (AP), where applicable to the evaluation design. The DIDM was compared with a Centralized Intrusion Detection Model (CIDM) to establish the relative performance of decentralized and centralized learning approaches.

The study was limited to a simulated distributed network environment and publicly available benchmark datasets. It did not involve deployment of the developed model within a live operational enterprise network. The study also did not provide comprehensive evaluation of all possible privacy-preserving mechanisms, adversarial attacks, automated incident response, digital forensics, vulnerability management, or full Security Orchestration, Automation and Response (SOAR) functionality.

Similarly, although the anomaly detection components were intended to improve the detection of previously unobserved anomalous patterns, the study did not claim comprehensive detection of genuine zero-day attacks beyond the attack patterns represented within the evaluation framework. Such claims would require dedicated zero-day or open-set experimental validation.

- *Assumptions of the Study*

The study was based on several assumptions.

First, it was assumed that the selected benchmark datasets, namely CICIDS2017, NSL-KDD, and UNSW-NB15, provided sufficiently diverse representations of normal and malicious network activities to support the development and evaluation of the DIDM. The datasets were assumed to contain adequate variation in traffic characteristics and attack categories for the purposes of the experimental investigation.

Second, it was assumed that the simulated distributed network environment provided an appropriate controlled representation of the distributed conditions required to evaluate the proposed architecture. Although the simulation could not reproduce every characteristic of a live enterprise network, it was assumed to provide sufficient experimental conditions for assessing the comparative performance of the DIDM and CIDM.

Third, it was assumed that the distributed clients participating in the Federated Learning process operated according to the defined training protocol and successfully completed the required local training and model-update exchange processes.

Fourth, it was assumed that the Federated Averaging algorithm correctly aggregated locally trained model parameters and that the resulting global model could be redistributed to participating clients during successive communication rounds.

Fifth, it was assumed that the selected performance evaluation metrics provided an appropriate basis for assessing the effectiveness of the developed DIDM. These metrics included accuracy, precision, recall, F1-

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

score, false positive rate, detection latency, ROC-AUC, and Average Precision, together with relevant scalability and communication measures where applicable.

Finally, it was assumed that the experimental procedures, including dataset partitioning, preprocessing, model training, and evaluation, were applied consistently to the centralized and decentralized approaches to support a fair comparative assessment.

➢ *Limitations of the Study and Mitigation Strategies*

The study was subject to several limitations that were considered during the research process.

The first limitation was the use of a simulated distributed network environment rather than a live enterprise network. Although simulation provided a controlled and reproducible environment for experimentation, it could not fully reproduce the scale, complexity, unpredictability, and operational constraints of real-world cybersecurity environments. This limitation was mitigated by using distributed computational nodes and controlled experimental configurations designed to represent the essential characteristics of decentralized network environments.

The second limitation concerned the use of publicly available benchmark datasets. CICIDS2017, NSL-KDD, and UNSW-NB15 are widely used in intrusion detection research and contain diverse attack scenarios; however, they may not fully represent current organizational traffic patterns, emerging threats, or previously unknown attack techniques. The use of multiple benchmark datasets was therefore adopted to increase the diversity of the evaluation and reduce dependence on a single dataset.

The third limitation related to the heterogeneous nature of distributed data. Federated Learning environments may involve non-independent and non-identically distributed (non-IID) data, where different clients observe different traffic patterns and attack distributions. This challenge was addressed by incorporating distributed data partitioning strategies and evaluating the model under conditions intended to reflect data heterogeneity.

The fourth limitation was computational resource availability. Federated Learning requires repeated local model training and communication of model updates across participating nodes. Conducting large-scale experiments involving many clients and communication rounds can therefore be computationally demanding. The study addressed this limitation by defining an experimentally manageable number of participating clients and federated communication rounds while maintaining sufficient conditions for evaluating the proposed architecture.

The fifth limitation concerned the scope of privacy evaluation. Although the DIDM was designed according to a data-locality principle in which raw network traffic remained at participating nodes, the study did not undertake a comprehensive evaluation of advanced privacy threats such as model inversion, membership inference, or gradient leakage. Therefore, the study did not interpret Federated Learning alone as providing absolute privacy protection. Rather, the research evaluated the architectural benefit of avoiding direct centralization of raw network traffic.

The sixth limitation concerned adversarial robustness. The study focused primarily on intrusion detection performance and distributed learning characteristics. It did not provide a comprehensive experimental assessment against all forms of adversarial machine learning attacks, poisoning attacks, or malicious federated clients. Such investigations were beyond the defined scope of the research and were identified as areas for future research.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Finally, the study focused on intrusion detection and did not extend to comprehensive automated incident response, digital forensics, vulnerability management, or full security orchestration. The model architecture may provide a foundation for future integration with response and orchestration mechanisms, but these functions were outside the scope of the present study.

Despite these limitations, the use of multiple benchmark datasets, controlled distributed experimentation, comparative evaluation, and expert-informed assessment provided a structured basis for investigating the effectiveness of the DIDM. The limitations were explicitly recognized to ensure that the findings were interpreted within the appropriate boundaries of the research design and experimental evidence.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

CHAPTER TWO
LITERATURE REVIEW

➢ *Introduction*

This chapter examined the body of literature relevant to decentralized intrusion detection, Federated Learning (FL), distributed artificial intelligence, and AI-driven anomaly detection techniques used in cybersecurity. The review brought together theoretical perspectives and empirical evidence to explain current developments in these fields, evaluate existing approaches, and identify the knowledge gaps that guided the design and development of the Federated Learning-based Decentralized Intrusion Detection Model (DIDM).

Evidence from previous studies suggested that considerable advances had been made in applying machine learning to intrusion detection. Despite these developments, many existing models continued to rely on centralized architectures that introduced challenges related to scalability, privacy protection, communication overhead, and the timely detection of increasingly sophisticated cyberattacks. These shortcomings highlighted the need for alternative approaches capable of supporting collaborative learning while preserving the confidentiality of locally generated network data.

The review therefore provided the theoretical and empirical foundation upon which the developed DIDM was built. It also established the research context by examining the strengths and limitations of existing intrusion detection approaches, thereby demonstrating the need for a decentralized, privacy-preserving, and scalable framework capable of addressing the security demands of modern distributed network environments.

➢ *Theoretical Review*

- *Distributed Artificial Intelligence (DAI) Theory*

Distributed Artificial Intelligence (DAI) theory explained intelligence as a collaborative process in which multiple autonomous agents interacted to solve computational problems that could not be efficiently addressed by a single centralized entity (Appiah-Kubi & Liu, 2020). Within cybersecurity, the theory provided a foundation for designing distributed intrusion detection systems where individual network nodes monitored local activities, analysed security events independently, and cooperated to strengthen overall network protection (Kasongo & Sun, 2020).

A key strength of DAI lies in its ability to distribute computational tasks among multiple agents instead of relying on a central processing point. Previous studies reported that this distributed architecture improved scalability, reduced processing bottlenecks, and increased operational resilience because individual nodes continued functioning even when other parts of the network experienced failures (Musa et al., 2020). Localized decision-making also enabled faster identification of suspicious activities while reducing dependence on centralized monitoring infrastructures.

Despite these advantages, evidence from the literature showed that DAI-based intrusion detection models continued to face important challenges. Achieving consistent coordination among autonomous agents remained difficult, particularly in large and heterogeneous network environments where participating nodes differed in computational capacity, network conditions, and local data characteristics. Several studies also observed that many DAI implementations struggled to adapt effectively to rapidly evolving cyber threats and changing attack patterns, limiting their long-term detection performance (Al-Garadi et al., 2020).

These observations suggested that, although DAI provided an effective foundation for distributed intrusion detection, it did not fully address the need for continuous collaborative learning across network nodes. Incorporating machine learning techniques capable of sharing knowledge while preserving local data privacy therefore emerged as a logical progression toward more adaptive and intelligent intrusion detection

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

models. This theoretical perspective informed the adoption of Federated Learning as the collaborative framework underpinning the developed Decentralized Intrusion Detection Model (DIDM).

- *Federated Learning Theory*

Federated Learning (FL), first proposed by McMahan et al. (2017), introduced a decentralized machine learning framework that enabled multiple clients to jointly train a shared model without transferring their raw data to a central repository. Under this approach, each participating client performed local model training and transmitted only the updated model parameters to a central server, where the Federated Averaging (FedAvg) algorithm combined the individual updates into an improved global model.

The decentralized nature of Federated Learning attracted considerable attention because it addressed one of the major limitations of conventional machine learning approaches: the need to centralize sensitive data. Studies by Yang et al. (2019) reported that retaining data within local devices strengthened privacy protection while supporting collaborative model development across geographically distributed environments. Within cybersecurity, this capability proved particularly valuable because organizations could improve intrusion detection models collectively without exposing confidential network traffic. Researchers also observed that Federated Learning facilitated knowledge sharing across different administrative domains, enhancing the detection of emerging cyber threats while reducing the risks associated with centralized data storage (Li et al., 2020).

Although Federated Learning addressed several privacy and scalability concerns, researchers continued to report technical challenges that affected its performance in practical deployments. Frequent communication between participating clients and the aggregation server increased network overhead, especially in environments involving numerous distributed nodes. Model convergence also became more difficult when local datasets were non-independent and identically distributed (non-IID), resulting in inconsistent learning outcomes across participating clients. In addition, Kairouz et al. (2021) and Zhou et al. (2021) identified security vulnerabilities associated with model poisoning, inference attacks, and other adversarial techniques capable of compromising the integrity of collaborative learning.

Another limitation identified in the literature related to the design of local detection models. Many Federated Learning-based intrusion detection frameworks concentrated primarily on collaborative model aggregation while giving comparatively little attention to intelligent anomaly detection at individual network nodes. This reduced their ability to identify previously unseen attacks and zero-day threats before the locally generated knowledge was incorporated into the global model.

Taken together, these findings suggested that Federated Learning provided an effective mechanism for collaborative and privacy-preserving model training, but it did not, by itself, guarantee robust intrusion detection in highly dynamic distributed environments. Achieving stronger detection performance required complementary local intelligence capable of identifying anomalous behaviour before collaborative model updates were aggregated. This observation provided the theoretical basis for integrating Federated Learning with an ensemble of AI-based anomaly detection techniques in the developed Decentralized Intrusion Detection Model (DIDM).

➢ *Empirical Literature Review*

- *Limitations of Traditional Intrusion Detection Models*

Early intrusion detection research largely focused on centralized architectures in which network traffic from multiple endpoints was collected and analysed at a single processing location (Wang & Liu, 2020). This design simplified system administration and enabled security events to be monitored from a unified platform.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Nevertheless, evidence from empirical studies suggested that the effectiveness of centralized intrusion detection declined as network infrastructures became larger, more distributed, and increasingly dynamic.

One of the principal weaknesses associated with centralized intrusion detection models was their dependence on a central processing point. Concentrating large volumes of network traffic within a single analytical platform increased the risk of service disruption should that platform become unavailable or compromised. As organizational networks expanded, the continuous growth in traffic volumes also imposed substantial computational demands, leading to processing delays, scalability challenges, and reduced detection performance in high-throughput environments (Ayodeji et al., 2020).

Researchers subsequently introduced anomaly-based intrusion detection techniques to improve the identification of previously unseen attacks. Unlike signature-based approaches, anomaly detection models compared observed network behaviour with established normal patterns, allowing them to recognize novel or evolving threats. Despite this advantage, Motwakel et al. (2023) reported that these models frequently generated high false-positive rates because legitimate network behaviour often exhibited considerable variability. Their performance also depended heavily on the quality of the training data, making adaptation to polymorphic malware, zero-day exploits, and other rapidly evolving attack techniques particularly challenging.

Distributed intrusion detection architectures emerged as an alternative to centralized systems by allocating detection responsibilities across multiple network nodes. Although this approach reduced reliance on a single processing location, many early implementations operated independently with limited coordination among participating nodes. Rezaeifar et al. (2020) observed that the absence of collaborative learning mechanisms often resulted in inconsistent detection decisions, fragmented threat intelligence, and limited situational awareness across distributed environments.

Collectively, the empirical evidence demonstrated that neither centralized intrusion detection models nor their early decentralized counterparts adequately addressed the operational requirements of modern cybersecurity environments. These shortcomings underscored the need for intrusion detection frameworks capable of combining distributed intelligence, collaborative learning, privacy preservation, and adaptive threat detection within a unified architecture. This gap provided further justification for the development of the Federated Learning-based Decentralized Intrusion Detection Model (DIDM).

- *Federated Learning-Based Intrusion Detection Models*

The introduction of Federated Learning (FL) marked an important shift in the development of intrusion detection models for distributed computing environments. Unlike conventional centralized approaches, FL enabled participating nodes to train machine learning models locally while exchanging only model parameters for global aggregation. This decentralized learning strategy reduced the need to transfer sensitive network data and addressed growing concerns related to privacy, data ownership, and regulatory compliance (Yang et al., 2019).

Empirical investigations reported encouraging results following the adoption of Federated Learning in intrusion detection. Li et al. (2020) found that collaborative model training improved the ability of distributed systems to recognize malicious network activities while benefiting from knowledge acquired across multiple participating nodes. The approach also supported continuous model improvement without requiring organizations to disclose confidential traffic data, making it particularly suitable for geographically distributed cybersecurity environments.

Although these findings demonstrated the practical value of Federated Learning, several implementation challenges remained evident. Differences in local data distributions frequently affected model convergence,

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

especially when participating clients generated non-independent and identically distributed (non-IID) datasets that reflected diverse network conditions. Such variations often reduced the consistency and overall performance of the aggregated global model under realistic deployment scenarios.

Scalability presented an additional challenge. Kairouz et al. (2021) observed that repeated communication between distributed clients and the aggregation server increased network overhead and prolonged training time as the number of participating nodes expanded. Researchers also identified security concerns associated with collaborative learning, noting that adversarial techniques such as model poisoning and inference attacks could compromise the integrity of aggregated models if appropriate protection mechanisms were not implemented (Zhou et al., 2021).

Despite these limitations, the empirical evidence consistently supported Federated Learning as an effective foundation for decentralized intrusion detection. Its ability to facilitate collaborative model development while retaining sensitive data within local environments represented a significant advancement over traditional centralized architectures. The remaining challenges highlighted in the literature, however, suggested that stronger local intelligence and more resilient anomaly detection techniques were needed to complement the collaborative learning process. This observation reinforced the rationale for integrating Federated Learning with an ensemble of AI-based anomaly detection models in the developed Decentralized Intrusion Detection Model (DIDM).

- *Local Anomaly Detection Agents in Intrusion Detection Models*

Local anomaly detection agents have increasingly been incorporated into distributed intrusion detection architectures to strengthen security monitoring at the network edge. Rather than forwarding all traffic to a centralized processing unit, these agents analysed locally generated network activities and responded to suspicious behaviour in near real time. This decentralized approach reduced communication delays and improved the responsiveness of intrusion detection systems operating in distributed environments (Appiah-Kubi & Liu, 2020).

Researchers investigated a wide range of machine learning algorithms to improve the effectiveness of local anomaly detection. Techniques including Support Vector Machines (SVM), Random Forests, Isolation Forests, One-Class Support Vector Machines (One-Class SVM), and Autoencoders demonstrated the ability to distinguish malicious activities from legitimate network behaviour by learning complex traffic patterns (Kasongo & Sun, 2020). Their capacity to detect both previously known attacks and novel anomalies made them suitable for dynamic cybersecurity environments where threat patterns continually evolved.

Performance, however, varied considerably across individual algorithms. Models that produced strong results under one network condition often experienced reduced accuracy when exposed to different traffic characteristics or emerging attack techniques. To overcome this limitation, researchers increasingly adopted ensemble learning strategies that combined multiple anomaly detection algorithms. Al-Garadi et al. (2020) reported that integrating complementary models improved detection stability, enhanced generalization, and reduced dependence on the weaknesses of individual classifiers.

Although local anomaly detection continued to advance, empirical evidence suggested that relatively few studies combined these intelligent edge-level detection mechanisms with Federated Learning-based collaborative training frameworks. In many implementations, local models generated intrusion alerts independently without contributing effectively to collective model improvement across distributed nodes. This separation limited the potential benefits of combining localized intelligence with collaborative knowledge sharing.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The reviewed literature therefore, indicated that effective decentralized intrusion detection required more than accurate local anomaly detection. Achieving robust performance across distributed environments also depended on mechanisms capable of integrating local intelligence with collaborative model learning while preserving data privacy. This gap provided further support for the development of the Decentralized Intrusion Detection Model (DIDM), which combined Federated Learning with an ensemble of AI-based anomaly detection techniques to strengthen both local and global intrusion detection capabilities.

➢ *Evaluation of Intrusion Detection Models*

Assessing the effectiveness of intrusion detection models has remained a central focus of cybersecurity research. Most empirical studies relied on well-established benchmark datasets, including CICIDS2017 (Sharafaldin et al., 2018), NSL-KDD (Musa et al., 2020), and UNSW-NB15 (Moustafa & Slay, 2015), to evaluate model performance under standardized attack scenarios. The widespread use of these datasets enabled meaningful comparisons between different intrusion detection approaches by providing a consistent basis for performance assessment across diverse categories of cyberattacks.

Researchers generally measured model effectiveness using statistical indicators such as accuracy, precision, recall, F1-score, and false-positive rate. These metrics provided valuable insights into classification performance and detection capability. Nevertheless, Mohamed and Khalifah (2024) observed that relatively few studies extended their evaluation to operational characteristics such as communication overhead, computational efficiency, model convergence, and scalability within distributed network environments. Consequently, important factors affecting practical deployment often received less attention than classification accuracy alone.

Simulation platforms, including NS-3, Mininet, and GNS3, were frequently adopted to reproduce network conditions in a controlled environment. These tools enabled researchers to evaluate intrusion detection models without exposing operational systems to unnecessary security risks. Although simulation-based experimentation supported repeatable performance assessment, many studies relied on simplified attack scenarios that did not adequately represent the complexity of contemporary cyber threats. Limited incorporation of expert-informed threat modelling further reduced the realism of some evaluation frameworks.

The reviewed literature demonstrated that a comprehensive assessment of intrusion detection models required more than conventional classification metrics. Reliable evaluation also depended on examining operational performance within realistic distributed environments, including the efficiency of collaborative learning, scalability across participating nodes, communication costs, and resilience under diverse attack conditions. These observations informed the evaluation framework adopted in the present study, which combined statistical performance measures with system level analysis to provide a more comprehensive assessment of the developed Decentralized Intrusion Detection Model (DIDM).

➢ *Conceptual Framework*

The conceptual framework developed for this study illustrated the interaction between Federated Learning and intelligent local anomaly detection agents within the proposed Decentralized Intrusion Detection Model (DIDM). The framework explained how collaborative model training and localized threat detection complemented one another to strengthen intrusion detection while maintaining the privacy of network data.

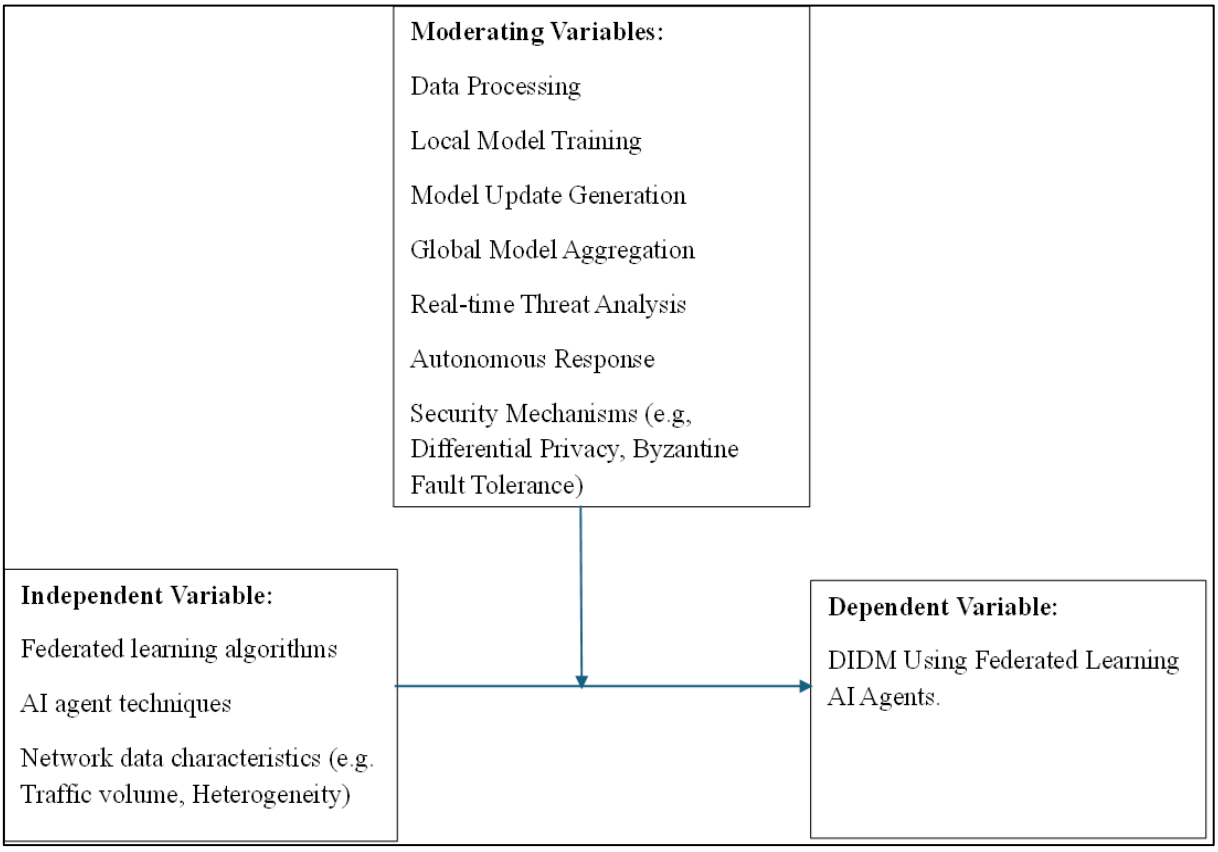
Intrusion detection activities began at the individual network node, where locally generated traffic was analysed to identify anomalous behaviour. Processing data at the point of generation reduced dependence on centralized infrastructures and supported timely detection of suspicious network activities. Each participating node independently trained its local detection model using the available traffic data while retaining all sensitive information within the local environment.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Rather than exchanging raw network traffic, participating nodes transmitted only the updated model parameters to a central aggregation server. These parameters were combined using the Federated Averaging (FedAvg) algorithm to generate an improved global intrusion detection model. The updated global model was then redistributed to all participating nodes, allowing each client to benefit from knowledge acquired across the distributed network while continuing subsequent rounds of local model training.

Local anomaly detection relied on complementary machine learning techniques, including Isolation Forest and One-Class Support Vector Machines (One-Class SVM), to distinguish normal network behaviour from potential intrusions. Combining the outputs generated by these local detection models with the collaboratively trained global model strengthened overall detection capability by improving robustness against diverse attack patterns and reducing dependence on any single detection technique.

The interaction between localized anomaly detection and Federated Learning formed the basis of the proposed DIDM. This integrated architecture supported collaborative knowledge sharing without exposing raw network traffic, improved scalability across distributed environments, and enhanced the model's capacity to respond to evolving cyber threats. The conceptual framework therefore provided the theoretical foundation for the system architecture, implementation, and experimental evaluation presented in the subsequent chapters.



ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

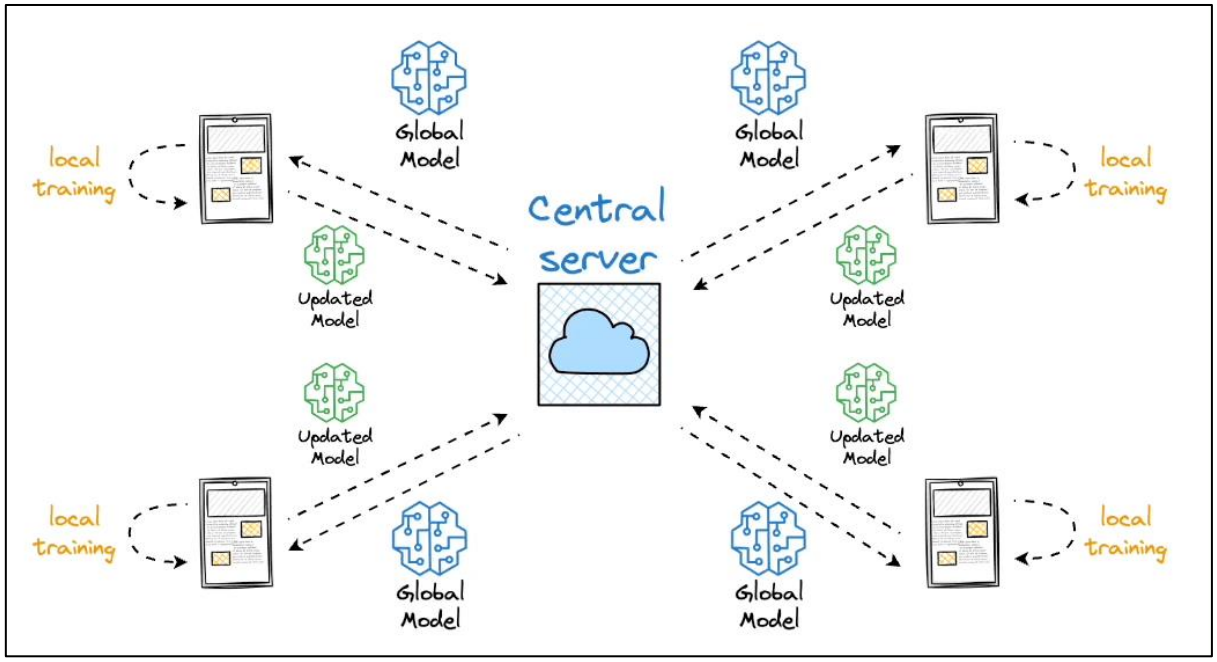


Fig 1 Conceptual Framework of the Decentralized Intrusion Detection Model

The conceptual framework served as the basis for designing and developing the Decentralized Intrusion Detection Model (DIDM). It adopted the Input–Process–Guidance–Model Output (IPGS) approach to illustrate how the major study variables interacted throughout the intrusion detection process. The framework also demonstrated how network traffic data progressed through successive analytical stages before being translated into intrusion detection decisions using Federated Learning and artificial intelligence techniques.

Its design drew from the theoretical perspectives and empirical evidence reviewed in this chapter. Concepts derived from Distributed Artificial Intelligence (DAI) and Federated Learning (FL) Theory were combined to explain the relationships among decentralized learning, collaborative model training, and intelligent anomaly detection. Integrating these theoretical perspectives addressed key shortcomings reported in conventional centralized intrusion detection models, particularly those associated with data privacy, scalability, and the ability to respond to evolving cyber threats.

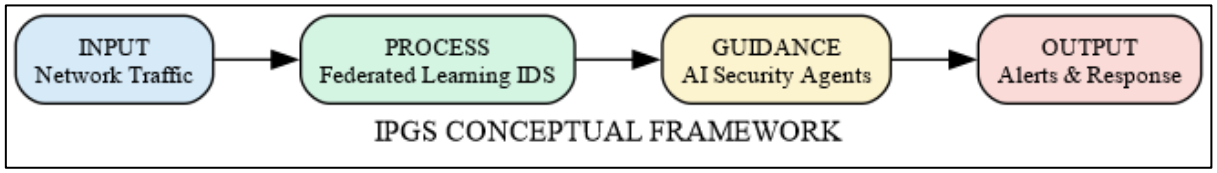


Fig 2 Input–Process–Guidance–Model Output (IPGS) Conceptual Framework

Following Figure 1, the interaction among the components of the conceptual framework may be described as follows.

The Input represented the network traffic data used throughout the study. These data were obtained from the CICIDS2017, NSL-KDD, and UNSW-NB15 benchmark datasets, each of which contained both legitimate network activities and diverse categories of malicious traffic. The combination of these datasets provided a broad representation of contemporary intrusion scenarios for model development and evaluation.

Within the Process stage, the collected data underwent preprocessing and feature engineering before local anomaly detection models were trained. Federated Learning was then applied to coordinate collaborative model training across participating nodes. Local model updates were aggregated using the Federated

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Averaging (FedAvg) algorithm, after which ensemble decision fusion combined the outputs generated by the individual detection models. This sequence allowed the model to identify intrusion patterns while retaining sensitive network data within the local environment.

The Guidance element incorporated Artificial Intelligence agents that supported monitoring, anomaly detection, adaptive learning, and coordination of model updates. Through continuous analysis of network behaviour, these agents refined local decision-making, improved model optimization, and strengthened the system's ability to recognize emerging attack patterns across distributed environments.

The Model Output reflected the expected outcomes generated by the developed Decentralized Intrusion Detection Model (DIDM). These included intrusion alerts, anomaly classifications, attack detection reports, improved detection accuracy, stronger privacy protection, lower false-positive rates, and enhanced scalability when deployed across distributed network infrastructures.

Taken together, the four components illustrated the progression of network traffic data from acquisition to intelligent decision-making. The framework also explained how Federated Learning and AI-driven anomaly detection interacted to produce accurate, privacy-preserving, and scalable intrusion detection within distributed computing environments.

➢ *Chapter Summary*

The literature reviewed in this chapter provided a comprehensive understanding of the theoretical concepts and empirical evidence underpinning decentralized intrusion detection. The discussion examined Distributed Artificial Intelligence, Federated Learning, and AI-based anomaly detection techniques while evaluating their application within contemporary cybersecurity environments. Collectively, these studies highlighted the evolution of intrusion detection from conventional centralized architectures toward more collaborative and privacy-aware approaches.

Evidence from the reviewed studies showed that conventional intrusion detection models continued to experience limitations associated with centralized processing, restricted scalability, privacy concerns, and limited adaptability to increasingly sophisticated attack techniques. Although Federated Learning addressed several of these challenges by supporting decentralized and privacy-preserving collaborative learning, practical implementation remained constrained by communication overhead, non-independent and identically distributed (non-IID) data, and security risks associated with adversarial model manipulation. Similarly, local anomaly detection models demonstrated strong capabilities in identifying malicious activities at the network edge but were seldom integrated with collaborative learning frameworks capable of improving detection performance across distributed environments.

A critical analysis of the reviewed literature identified a clear gap in the development of intrusion detection models that effectively combined Federated Learning with intelligent local anomaly detection within a unified decentralized architecture. Existing approaches generally addressed these technologies independently, leaving limited evidence on their combined effectiveness in improving detection accuracy, preserving data privacy, and supporting scalable deployment across distributed networks.

The findings of this review provided the conceptual and empirical basis for the design of the Decentralized Intrusion Detection Model (DIDM) developed in this study. Building on the identified research gap, the next chapter describes the model architecture, implementation procedures, and experimental framework adopted to evaluate its performance.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

➢ *Research Gaps*

The comprehensive review of existing literature reveals several critical gaps that the study seeks to address, as summarized below:

Table 1 Research Gaps

Existing Approach/Technology	Description and Shortcomings	Research Gap Addressed by This Study	Type of Research Gap
Traditional Centralized IDM	Centralized IDM architectures struggle with scalability, responsiveness, and privacy due to reliance on centralized data aggregation.	This study addresses the need for scalable and privacy-preserving IDM for distributed environments.	Knowledge Gap
Existing Decentralized IDM Models	Most models lack full integration of Federated Learning for secure collaborative training and efficient model aggregation.	This study introduces a FL-based decentralized IDM ensuring privacy-preserving learning across distributed nodes.	Knowledge Gap
Multi-Agent Cybersecurity Models	Many Models rely on static rule-based detection that cannot adapt to evolving threats.	This study develops adaptive intelligent agents capable of real-time autonomous threat detection.	Knowledge Gap
Performance Evaluation of IDM Models	Lack of standardized evaluation frameworks for scalability, robustness, and adversarial testing.	This study provides comprehensive empirical evaluation using multiple metrics and statistical validation.	Methodological Gap

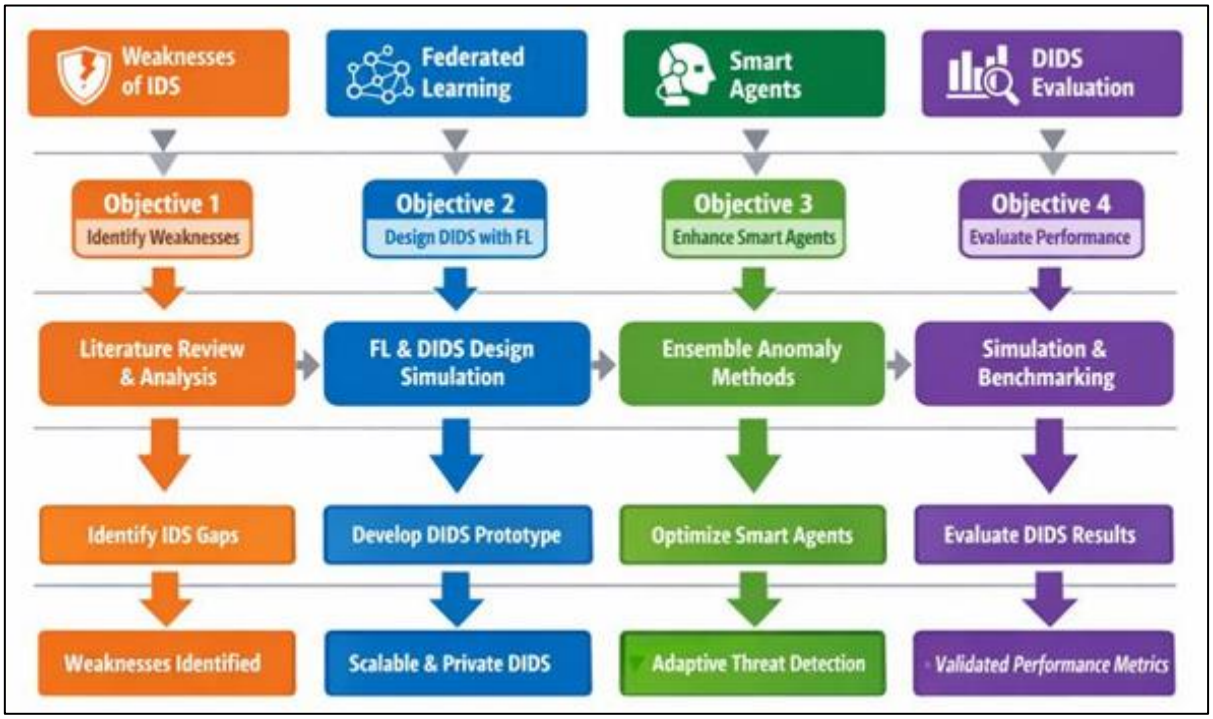


Fig 3 Conceptual and Methodological Flow of the DIDM Study. This diagram illustrates how insights from the literature review inform the study objectives, research methodology, and expected outcomes.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

CHAPTER THREE
RESEARCH METHODOLOGY

➢ *Introduction*

This chapter presents the methodology adopted in the design, implementation, and evaluation of the Federated Learning-based Decentralized Intrusion Detection Model (DIDM)**.** It describes the research design, study environment, population of the study, sampling procedures, sample size, tools and technologies, data collection and preprocessing procedures, model development process, evaluation framework, model architecture, data analysis methods, and ethical considerations that guided the study.

The methodology was developed to address the objectives of the study, which included examining the limitations of existing intrusion detection approaches, designing a Federated Learning-based Decentralized Intrusion Detection Model, implementing the developed model within a distributed computing environment, and evaluating its performance against a centralized baseline model.

The study adopted the Design Science Research Methodology (DSRM) as the overarching methodological framework. DSRM was appropriate because the primary outcome of the research was the development and evaluation of a technological artefact, namely the Decentralized Intrusion Detection Model (DIDM). The research process therefore involved problem identification, definition of research objectives, design and development of the artefact, implementation, demonstration, evaluation, and communication of the research findings.

The experimental component of the study involved the use of benchmark intrusion detection datasets within a controlled distributed computing environment. The developed DIDM employed Federated Learning to facilitate collaborative model training across distributed clients without requiring the direct exchange of raw training data. The Federated Averaging (FedAvg) algorithm was used to aggregate locally trained model parameters into a shared global model.

The study also incorporated artificial intelligence-based detection mechanisms to support local anomaly identification. Autoencoders, Isolation Forest, and One-Class Support Vector Machines (One-Class SVM) were incorporated as complementary detection techniques within the DIDM framework. Their outputs were integrated within the overall detection architecture to strengthen the identification of abnormal network behaviour.

To establish a comparative basis for evaluating the developed model, a Centralized Intrusion Detection Model (CIDM) was implemented as a baseline. The centralized and decentralized approaches were evaluated using consistent experimental procedures and appropriate performance metrics. The evaluation considered classification performance and selected operational characteristics relevant to distributed intrusion detection environments.

In addition to the experimental evaluation, expert validation was incorporated to obtain informed assessments of the practical relevance and applicability of the developed model. The expert evaluation provided complementary evidence regarding the suitability of the model architecture, threat scenarios, evaluation criteria, and potential operational application.

The methodology presented in this chapter therefore provided a structured and reproducible approach for developing, implementing, and evaluating the DIDM while maintaining methodological alignment with the objectives of the study.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

➢ *Research Design*

The study adopted a mixed-methods research design combining quantitative experimental evaluation and qualitative expert validation to support the development and assessment of the Federated Learning-based Decentralized Intrusion Detection Model (DIDM). The adoption of this design was appropriate because the study involved both the development of a technological artefact and the evaluation of its effectiveness within a cybersecurity context. The quantitative component provided empirical evidence regarding the detection performance of the developed DIDM in comparison with a Centralized Intrusion Detection Model (CIDM), while the qualitative component provided expert-based assessment of the model's practical relevance and applicability within distributed cybersecurity environments.

The quantitative component focused on the design, implementation, demonstration, and experimental evaluation of the developed intrusion detection models. The performance of the CIDM and DIDM was assessed using objective evaluation measures including accuracy, precision, recall, F1-score, false positive rate, detection latency, Receiver Operating Characteristic Area Under the Curve (ROC-AUC), and Average Precision (AP), where applicable. These measures provided a comprehensive assessment of classification effectiveness by examining the ability of the models to correctly identify malicious and legitimate network traffic, minimize incorrect detections, and maintain reliable discrimination across different classification thresholds.

The experimental evaluation followed a controlled and reproducible procedure to ensure reliable comparison between the centralized and decentralized approaches. The evaluation process involved dataset assessment, class-aware sampling, stratified separation of training and testing data, feature preprocessing based exclusively on the training partition, and application of Synthetic Minority Over-sampling Technique (SMOTE) only to the training data. The held-out testing dataset remained independent from the training process and was used to obtain unbiased performance estimates. This approach reduced the likelihood of information leakage and strengthened the validity of the comparative evaluation.

The quantitative evaluation consisted of two complementary assessment perspectives. The first involved formal empirical benchmarking of the CIDM and DIDM under predefined experimental conditions. This assessment represented the primary source of evidence for determining comparative model performance. The second involved operational demonstration of the implemented DIDM within an integrated cybersecurity monitoring environment using a Streamlit-based dashboard. The operational demonstration illustrated the practical behaviour of the developed framework by presenting model outputs, detection decisions, latency measurements, confusion matrix results, ROC analysis, and Precision–Recall analysis.

The operational demonstration served a different purpose from the formal empirical evaluation. While the formal evaluation established comparative model performance using controlled experimental procedures, the dashboard-based demonstration illustrated the practical execution of the developed DIDM within a cybersecurity monitoring environment. Therefore, operational outputs were interpreted as evidence of system functionality and deployment behaviour rather than direct replacements for benchmark evaluation results. This distinction ensured that scientific model assessment and practical system demonstration were evaluated independently.

The qualitative component involved expert validation by cybersecurity professionals with relevant experience in network security, intrusion detection, and cybersecurity operations. Expert assessment was conducted to evaluate the practical relevance of the developed model, the suitability of the considered threat scenarios, the appropriateness of the evaluation criteria, and the potential applicability of the architecture within distributed network environments. The inclusion of expert validation complemented the quantitative findings by providing professional perspectives on the usefulness and practicality of the developed DIDM.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The overall research design was guided by the Design Science Research Methodology (DSRM), which provided a structured framework for developing and evaluating the DIDM artefact. DSRM was considered appropriate because the study focused on creating and validating a technological solution aimed at addressing identified limitations in existing intrusion detection approaches. The methodology supported systematic progression through problem identification, objective definition, artefact design and development, demonstration, evaluation, and communication of research outcomes.

The study design incorporated Federated Learning as the primary collaborative learning mechanism within the developed DIDM. Through Federated Learning, participating clients trained models using locally available network traffic data while sharing only model updates during the aggregation process. This approach reduced the requirement for centralizing sensitive network information while enabling collaborative improvement of intrusion detection capability across distributed environments.

The developed DIDM incorporated a federated Artificial Neural Network (ANN) as the primary multiclass classification model, supported by additional anomaly detection mechanisms including Isolation Forest and One-Class Support Vector Machine (One-Class SVM). The integration of these components enabled the model to combine collaborative learning with local anomaly analysis, thereby improving its ability to identify both known and potentially unknown malicious behaviours within distributed network environments.

The mixed-methods design was therefore appropriate for achieving the objectives of the study because it integrated experimental evidence, system-level demonstration, and expert judgement. The quantitative evaluation established measurable performance differences between centralized and decentralized intrusion detection approaches, the operational demonstration illustrated practical system functionality, and expert validation provided additional insight into real-world applicability. The combination of these approaches provided a comprehensive evaluation framework for determining the effectiveness and suitability of the developed Decentralized Intrusion Detection Model.

- *Study Environment*

The study was conducted within a controlled and simulated distributed computing environment developed to support the design, implementation, demonstration, and evaluation of the Federated Learning-based Decentralized Intrusion Detection Model (DIDM). The environment was configured to represent the operational characteristics of distributed network systems in which multiple computational nodes independently analyse network traffic while participating in collaborative machine learning processes.

A controlled experimental environment was selected to enable systematic, repeatable, and secure evaluation of the developed intrusion detection framework. Deploying an experimental intrusion detection model directly within a live production network would introduce operational risks, including disruption of network services, exposure of sensitive traffic information, and difficulties in maintaining consistent evaluation conditions. The simulated environment therefore provided an appropriate setting for investigating the performance and behaviour of the DIDM under controlled experimental conditions.

The distributed computing environment consisted of multiple simulated client nodes representing independent network entities participating in the Federated Learning process. The DIDM implementation incorporated five federated clients, where each client performed local model training before contributing model updates to the federated aggregation process. The Federated Averaging (FedAvg) algorithm was applied to combine locally trained model parameters and generate an updated global model. This architecture enabled collaborative learning among distributed clients while maintaining the locality of raw network traffic data.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

To support reproducibility and isolation of the distributed components, containerized computational environments were incorporated into the experimental setup. Docker-based environments and the Containernet framework were used to represent distributed network nodes and facilitate controlled interaction between computational components. The containerized approach enabled consistent configuration of participating clients, simplified experimental replication, and provided a practical foundation for examining decentralized intrusion detection behaviour.

The study environment supported two complementary evaluation perspectives. The first involved formal empirical evaluation of the Centralized Intrusion Detection Model (CIDM) and the developed DIDM using benchmark intrusion detection datasets under controlled experimental conditions. This evaluation environment provided the primary basis for comparing model performance using predefined evaluation metrics. The second involved operational demonstration of the implemented DIDM through an integrated Streamlit-based cybersecurity monitoring dashboard. This environment enabled visualization of model outputs and illustrated how the developed framework could support practical intrusion detection monitoring.

The formal experimental environment incorporated benchmark intrusion detection datasets to support model development and evaluation. The study utilized established datasets, including CICIDS2017, NSL-KDD, and UNSW-NB15, during the broader research process. These datasets provided representative examples of legitimate network behaviour and diverse categories of malicious activities, enabling the investigation of machine learning-based intrusion detection approaches under standardized conditions. The principal empirical evaluation presented in the study relied on the CICIDS2017 dataset due to its comprehensive representation of contemporary network attacks and normal traffic patterns.

Within the study environment, the developed DIDM integrated Federated Learning with complementary artificial intelligence-based anomaly detection mechanisms. The federated Artificial Neural Network (ANN) functioned as the primary classification component, while additional anomaly detection approaches, including Isolation Forest and One-Class Support Vector Machine (One-Class SVM), provided supplementary detection evidence. This integration enabled investigation of how collaborative learning and local anomaly analysis could operate together within a decentralized intrusion detection architecture.

The Streamlit-based cybersecurity dashboard was incorporated as an operational demonstration component of the study environment. The dashboard provided a user interface for presenting selected system outputs, including classification results, detection latency, false-positive measurements, confusion matrices, Receiver Operating Characteristic (ROC) analysis, and Precision–Recall analysis. The dashboard was used to demonstrate the practical operation of the developed framework and to provide an accessible representation of model behaviour.

A distinction was maintained between the formal empirical evaluation environment and the operational demonstration environment. The formal evaluation provided the main evidence for comparing CIDM and DIDM performance using controlled experimental procedures and independent test data. Conversely, the dashboard environment demonstrated system functionality and operational behaviour when the implemented DIDM was executed within an integrated monitoring interface. Therefore, demonstration outputs were interpreted as evidence of practical system operation rather than direct substitutes for controlled benchmark evaluation results.

The controlled study environment provided a suitable foundation for investigating decentralized intrusion detection by enabling repeatable experimentation, controlled comparison, and practical demonstration of the developed model. It supported assessment of detection effectiveness, collaborative learning behaviour, and selected operational characteristics associated with Federated Learning-based intrusion detection.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

However, the limitations of the simulated environment were acknowledged. The use of benchmark datasets and simulated federated clients could not completely reproduce the complexity of operational enterprise networks, including dynamic traffic behaviour, heterogeneous infrastructure, real-time communication constraints, and evolving threat conditions. Consequently, the findings of the study were interpreted within the context of the selected datasets, experimental configuration, and simulated distributed environment. The operational dashboard demonstration was similarly considered evidence of system functionality and not a replacement for deployment evaluation within a live production network.

➢ *Population of the Study*

The population of the study consisted of three complementary components that supported the development, evaluation, and validation of the Federated Learning-based Decentralized Intrusion Detection Model (DIDM). These components included benchmark intrusion detection datasets, simulated federated computing clients, and cybersecurity professionals involved in expert validation. The inclusion of these components enabled the study to evaluate the developed model from empirical, computational, and practical perspectives.

The first component of the study population comprised benchmark cybersecurity datasets used for model development and empirical evaluation. The principal dataset used for the formal experimental evaluation was the CICIDS2017 dataset, which contains network traffic records representing both benign activities and multiple categories of malicious behaviour. The dataset provided the empirical foundation for training and evaluating the Centralized Intrusion Detection Model (CIDM) and the Federated Learning-based Decentralized Intrusion Detection Model (DIDM). Additional benchmark datasets, including NSL-KDD and UNSW-NB15, were considered during the broader model development process to provide contextual understanding of existing intrusion detection evaluation practices.

The second component comprised simulated distributed computational clients participating in the Federated Learning process. The developed DIDM incorporated five federated clients representing independent computational nodes within a distributed network environment. Each client performed local model training using its assigned data partition while retaining the original training observations within the local environment. The Federated Averaging (FedAvg) algorithm was then applied to combine locally generated model updates into a global model. This population component enabled investigation of collaborative learning behaviour without requiring centralization of raw network traffic data.

The third component consisted of cybersecurity professionals who participated in the expert validation process. The experts were selected based on their knowledge and professional experience in cybersecurity-related domains, including network security, intrusion detection, information security, and cybersecurity operations. Their participation provided an additional perspective for assessing the practical relevance, technical appropriateness, and potential applicability of the developed DIDM within distributed cybersecurity environments.

The integration of these three population components supported a comprehensive evaluation of the developed intrusion detection framework. The benchmark datasets provided the foundation for measuring model performance, the simulated federated clients enabled assessment of decentralized collaborative learning, and the cybersecurity professionals contributed practical validation of the developed artefact.

The study population was therefore aligned with the objectives of the Design Science Research Methodology adopted in this research. Rather than focusing solely on human participants, the population represented the primary resources and stakeholders required for designing, implementing, demonstrating, and evaluating the DIDM artefact.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

A distinction was maintained between the empirical evaluation population and the operational demonstration environment. The Streamlit-based cybersecurity dashboard was used to demonstrate the execution of the implemented DIDM and visualize selected model outputs. The dashboard environment was therefore considered part of the system demonstration component and was not treated as an independent population from which the principal empirical performance results were obtained.

➢ *Sampling Procedure and Sample Size*

- *Data Partitioning*

The datasets used in the study were partitioned into training and testing subsets to support model development and objective performance evaluation. For the formal CICIDS2017 experimental evaluation, the preprocessed dataset was divided using an approximately 80:20 training-to-testing ratio. The training subset comprised 2,016,600 samples, while the independent testing subset contained 504,151 samples, giving a total of 2,520,751 samples. Each sample consisted of 52 features representing network traffic characteristics and was assigned to one of seven traffic classes: Bots, Brute Force, DDoS, DoS, Normal Traffic, Port Scanning, and Web Attacks.

Stratified partitioning was used to maintain the representation of the different traffic classes across the training and testing subsets. This was important because the CICIDS2017 dataset exhibits class imbalance, with some traffic categories having substantially more samples than others. Maintaining class representation in both subsets helped ensure that model training and subsequent evaluation considered the different categories of network traffic.

The training subset was used for model development and learning. For the Centralized Intrusion Detection Model (CIDM), the training data were used to train the neural network centrally. For the Decentralized Intrusion Detection Model (DIDM), the training data were distributed among five simulated federated clients for local model training and subsequent aggregation using the Federated Averaging (FedAvg) algorithm. The testing subset was kept independent of the training and federated aggregation processes.

Both CIDM and DIDM were evaluated using the same independent testing subset of 504,151 samples. This ensured that the two models were assessed against identical evaluation data, providing a consistent basis for comparing their classification performance. The test data were not used for model fitting or federated aggregation, thereby reducing the risk of data leakage into the evaluation process.

- *Expert Sampling*

A purposive sampling technique was employed to select cybersecurity professionals with relevant domain expertise in network security, intrusion detection, and cybersecurity operations. This non-probability sampling approach was considered appropriate due to the need for targeted selection of individuals with specialized knowledge and practical experience in evaluating intrusion detection models.

A sample size of 10–20 experts was deemed adequate based on established qualitative research guidelines for expert judgment validation in technical Model evaluation studies. This range was considered sufficient to obtain diverse yet focused expert feedback while maintaining depth and quality of evaluation.

The rationale for purposive sampling was grounded in the need to obtain informed, experience-based evaluations rather than statistically generalizable responses. The selected experts were drawn from Security Operations Centres (SOCs), financial institutions, cybersecurity consultancy firms, and Managed Security Service Providers (MSSPs). This ensured that the feedback obtained was anchored in real-world operational cybersecurity environments.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The inclusion of domain experts enhanced the credibility, relevance, and practical validity of the evaluation process for the Federated Learning-based Decentralized Intrusion Detection Model (DIDM). Their insights contributed to assessing the model's architectural suitability, detection effectiveness, and applicability within contemporary cybersecurity operational settings.

- *Relationship Between Original Dataset and Processed Distributions*

The class distributions presented in Chapter Four represent the different stages of dataset preparation and model evaluation. The distributions should therefore be interpreted according to the specific purpose of each stage, including the original dataset, the processed experimental dataset, and the resulting training and testing partitions.

The CICIDS2017 dataset served as the primary source of network traffic observations used in the formal empirical evaluation. It contains records representing legitimate network activities and different categories of malicious traffic. The dataset was examined to establish the representation of the identified traffic categories and to understand the class distribution before model development and evaluation.

Following dataset preparation, the processed CICIDS2017 dataset comprised 2,520,751 observations, 52 features, and seven traffic classes: Bots, Brute Force, DDoS, DoS, Normal Traffic, Port Scanning, and Web Attacks. These classes represented the categories used in the comparative evaluation of the Centralized Intrusion Detection Model (CIDM) and the Decentralized Intrusion Detection Model (DIDM).

The processed dataset was partitioned into training and testing subsets using stratified sampling at an approximately 80:20 ratio. The training subset contained 2,016,600 observations, while the independent testing subset contained 504,151 observations. Stratification maintained the representation of the identified traffic categories across the two partitions, providing a basis for model training and subsequent performance assessment.

Feature preprocessing was performed using a StandardScaler. The scaling procedure was used to standardize the input features for neural network training. The resulting training and testing datasets retained the same feature structure, comprising 52 input features, to ensure consistency during model development and evaluation.

The training subset was used to develop the two comparative models. The CIDM was trained centrally using the prepared training data, whereas the DIDM employed a federated learning approach in which training was distributed among five simulated clients. The local model updates were subsequently aggregated using the Federated Averaging (FedAvg) algorithm. This enabled the evaluation of decentralized collaborative model training against the centralized learning approach.

The testing subset was maintained as an independent evaluation dataset. Both CIDM and DIDM were assessed using the same 504,151 testing observations and their corresponding ground-truth labels. This ensured that the comparative performance results were obtained under consistent evaluation conditions. The independent test dataset was not used for model training or federated aggregation.

The relationship among the dataset stages can therefore be understood as a sequence beginning with the original CICIDS2017 source data, followed by dataset preparation and feature preprocessing, stratified partitioning, model development, and independent performance evaluation. The resulting training and testing distributions provided the empirical basis for assessing the classification performance of the centralized and decentralized models.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

A distinction was also maintained between the formal empirical evaluation and the operational demonstration presented later in Chapter Four. The formal evaluation involved the controlled comparison of CIDM and DIDM using the prepared CICIDS2017 dataset and a common independent test set. In contrast, the Streamlit-based cybersecurity dashboard demonstrated the presentation and monitoring capabilities of the implemented system within an integrated operational environment. These two components served different purposes and their results should therefore be interpreted within their respective experimental and demonstration contexts.

Maintaining this distinction between dataset preparation, model training, independent testing, and operational demonstration provided a clear methodological basis for interpreting the results presented in Chapter Four.

➢ *Tools and Technologies*

The development, implementation, demonstration, and evaluation of the Decentralized Intrusion Detection Model (DIDM) were supported by a combination of programming environments, machine learning frameworks, data processing libraries, federated learning platforms, distributed computing technologies, and visualization tools. The selected technologies provided the computational foundation required for preparing intrusion detection datasets, developing machine learning models, implementing federated learning processes, simulating distributed environments, and demonstrating the operation of the developed framework.

Python 3.10 was selected as the primary programming language for implementing the DIDM. The choice of Python was based on its extensive support for artificial intelligence, machine learning, deep learning, data analysis, and cybersecurity research applications. The language provided a flexible environment for integrating data processing components, machine learning algorithms, federated learning mechanisms, evaluation procedures, and system demonstration components within a unified development workflow.

The data preparation and numerical processing activities were supported by Pandas and NumPy libraries. Pandas was used for data loading, manipulation, cleaning, transformation, and exploratory analysis of the benchmark intrusion detection datasets. NumPy supported numerical computation and efficient handling of large-scale data operations required during preprocessing and model preparation. These libraries were particularly useful in managing the computational requirements associated with the CICIDS2017 dataset.

Scikit-learn was used for implementing machine learning algorithms, preprocessing procedures, anomaly detection techniques, and evaluation functions. The framework supported data partitioning, feature scaling, statistical performance measurement, and implementation of anomaly detection algorithms, including Isolation Forest and One-Class Support Vector Machine (One-Class SVM). The library also provided evaluation utilities for generating classification metrics used in assessing the performance of the Centralized Intrusion Detection Model (CIDM) and the DIDM.

The imbalanced-learn library was incorporated to address class imbalance within the training dataset through the Synthetic Minority Oversampling Technique (SMOTE). The library enabled controlled oversampling of minority classes during model development while maintaining separation between training and testing datasets. The application of SMOTE was restricted to the training partition to support independent evaluation of the developed models.

TensorFlow and Keras were used for the development and training of deep learning components within the DIDM. These frameworks supported the construction, optimization, and training of Artificial Neural Network (ANN) and Autoencoder architectures. Within the developed framework, the federated ANN served as the primary multiclass classification model, while the Autoencoder contributed additional anomaly detection capability.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Flower Federated Learning Framework was used to implement and coordinate the federated learning process. The framework enabled communication between participating clients, management of federated training rounds, and aggregation of locally trained model parameters. The Federated Averaging (FedAvg) algorithm was adopted as the aggregation mechanism to combine local model updates into a global model while ensuring that raw client data remained within individual client environments.

Docker and Containernet were used to establish the simulated distributed computing environment. Docker provided isolated container-based environments for representing federated clients, while Containernet enabled the creation of networked container environments that approximated distributed network conditions. These technologies supported reproducible experimentation and enabled investigation of decentralized intrusion detection behaviour within a controlled environment.

Streamlit was used to develop the cybersecurity monitoring dashboard for operational demonstration of the implemented DIDM. The dashboard provided an interactive interface for displaying model outputs, intrusion detection decisions, performance indicators, and selected analytical visualizations. The Streamlit environment supported demonstration of system functionality and visualization of results and was therefore considered complementary to the formal empirical evaluation process.

Matplotlib was used for generating graphical representations of experimental findings. The visualization framework supported the creation of performance comparison graphs, confusion matrices, Receiver Operating Characteristic (ROC) curves, Precision–Recall curves, and other figures used to interpret and communicate the evaluation results. These visualizations supported both quantitative analysis and presentation of the operational behaviour of the developed system.

The study utilized benchmark intrusion detection datasets, including CICIDS2017, NSL-KDD, and UNSW-NB15, to support model development and evaluation. These datasets provided examples of normal network behaviour and diverse categories of malicious activities, enabling investigation of machine learning-based intrusion detection approaches under standardized conditions. The CICIDS2017 dataset served as the primary dataset for the formal empirical evaluation.

The developed DIDM integrated multiple artificial intelligence components, including a federated Artificial Neural Network, Autoencoder, Isolation Forest, and One-Class Support Vector Machine. These technologies were combined to support collaborative learning and intelligent anomaly detection within the decentralized intrusion detection framework. The integration of these components enabled the investigation of how federated learning and local anomaly detection mechanisms could complement each other in improving intrusion detection capability.

Overall, the selected tools and technologies provided a comprehensive computational environment for implementing and evaluating the DIDM. The integration of programming libraries, machine learning frameworks, federated learning technologies, distributed simulation tools, and visualization platforms supported the systematic development of the intrusion detection framework and enabled comparison with the Centralized Intrusion Detection Model under controlled experimental conditions.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

```
# Core data processing libraries
import pandas as pd
import numpy as np

# Visualization libraries
import matplotlib.pyplot as plt
import seaborn as sns

# Machine learning preprocesing and evaluation
from sklearn.preprocesing import StandardScaler, LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix

# Anomaly detection models
from sklearn.ensemble import IsolationForest
from sklearn.svm import OneClassSVM

# Handling imbalanced datasets
from imblearn.over_samling import RandomOverSampler

# Deep learning framework
import tensorflow as tf
from tensorflow.keras.models impot Sequential
from tensorflow.keras.layers import Dense,|  Input
```

Fig 4 Python Libraries and Environment Used for DIDM Implementation.

➢ *Data Collection and Preprocessing*

The data collection and preprocessing procedures constituted an essential stage in the development of the Decentralized Intrusion Detection Model (DIDM). These procedures ensured that the network traffic data used for model training and evaluation were appropriately prepared for machine learning and Federated Learning processes. The adopted preprocessing framework focused on improving data quality, ensuring consistency between centralized and decentralized experiments, addressing class imbalance, and maintaining the independence of evaluation data.

- *Data Collection*

The study utilized three benchmark intrusion detection datasets, namely CICIDS2017, NSL-KDD, and UNSW-NB15. These datasets were selected because they contain labelled network traffic observations representing both legitimate activities and different categories of cyberattacks. Their extensive use in intrusion detection research provided a suitable basis for developing and evaluating machine learning-based cybersecurity models.

The selected datasets provided diverse representations of network behaviour, including normal traffic patterns and malicious activities such as denial-of-service attacks, scanning activities, brute-force attempts, and other intrusion categories. Each dataset contained network traffic features describing communication characteristics together with corresponding class labels indicating whether observations represented normal or malicious behaviour.

The datasets were acquired and examined before model development to identify data quality issues that could affect subsequent analysis. The initial assessment involved checking for missing values, duplicate records, inconsistent entries, and other conditions that could negatively influence model performance. The resulting cleaned datasets provided the foundation for subsequent preprocessing, model training, and evaluation procedures.

Although CICIDS2017, NSL-KDD, and UNSW-NB15 were incorporated during the broader research process, CICIDS2017 served as the primary dataset for the formal empirical evaluation of the CIDM and

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

DIDM. This dataset was selected because it contains a comprehensive representation of contemporary network attacks and normal traffic patterns suitable for evaluating decentralized intrusion detection performance.

- *Data Distribution Strategy*

To represent the decentralized nature of the DIDM, the processed training data were distributed among multiple simulated federated clients. Each client represented an independent network node participating in the Federated Learning process while maintaining access to its own local training data.

The distribution strategy was designed to reflect real-world distributed network environments where different organizations, departments, or network locations may generate and process their own traffic observations. Rather than transferring raw network traffic records to a centralized repository, each client retained its assigned data locally and performed model training within its own computational environment.

The participating clients independently trained local models using their allocated data partitions. The resulting model updates were then communicated to the federated aggregation mechanism, where they were combined using the Federated Averaging (FedAvg) algorithm to generate an improved global model. This approach enabled collaborative learning while preserving the locality of the original network observations.

The distributed arrangement also allowed the study to examine the ability of Federated Learning to integrate knowledge from multiple clients operating within a decentralized environment. This was particularly relevant because network environments often contain heterogeneous data distributions resulting from differences in user behaviour, infrastructure, and security conditions.

- *Data Preprocessing Pipeline*

A structured preprocessing pipeline was implemented to transform the network traffic datasets into suitable formats for machine learning and deep learning algorithms. The preprocessing procedures focused on data quality assessment, feature preparation, feature scaling, and computational optimization to support the development and evaluation of the Decentralized Intrusion Detection Model (DIDM).

The initial preprocessing stage involved examining the datasets for data quality issues, including duplicate observations, missing values, inconsistent records, and unsuitable entries. Where necessary, these issues were addressed to improve the consistency and reliability of the data used during model development and evaluation.

The network traffic attributes were subsequently prepared for machine learning processing. Features requiring numerical representation were converted into suitable numerical formats to ensure compatibility with the machine learning algorithms and neural network architectures incorporated into the DIDM framework. The resulting feature representation provided the input structure required for model training and classification.

Feature scaling was performed to standardize numerical attributes and reduce the influence of differences in feature magnitudes during model training. For the formal CICIDS2017 experimental evaluation, the preprocessing pipeline employed the StandardScaler technique. The scaling parameters were derived from the training data and applied consistently to the corresponding testing data. This ensured that both datasets maintained a consistent feature representation without using information from the independent testing subset to determine the scaling parameters.

Following preprocessing, the CICIDS2017 dataset was partitioned into training and testing subsets using stratified sampling at an approximately 80:20 ratio. The training subset contained 2,016,600 observations,

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

while the independent testing subset comprised 504,151 observations. Both subsets retained the same 52 input features and represented the seven traffic classes used in the formal comparative evaluation.

The training partition was used for model development, while the testing partition was reserved for independent performance evaluation. The same testing observations were used to assess both the Centralized Intrusion Detection Model (CIDM) and the Decentralized Intrusion Detection Model (DIDM). This ensured consistency in the evaluation process and provided a common basis for comparing the classification performance of the two models.

Considering the computational demands associated with large-scale network traffic datasets, memory and processing optimization techniques were also considered during data preparation and model development. These included efficient data handling, appropriate data type management, and processing operations suited to the available computational resources. Such measures supported the handling of large volumes of network traffic observations while maintaining the required feature structure for model development and evaluation.

- *Federated Data Handling Considerations*

A key design consideration within the DIDM was preservation of data locality during the Federated Learning process. Unlike centralized intrusion detection approaches where raw network traffic data are transferred to a central repository, the developed framework retained training observations within individual client environments.

Each federated client performed local model training using its assigned dataset partition. Following local training, only model parameters and updates were communicated for aggregation. The raw network traffic observations remained within their respective client environments throughout the federated training process.

The aggregated global model generated through the FedAvg algorithm was subsequently distributed back to participating clients for subsequent training rounds. This approach supported collaborative model improvement while reducing dependence on centralized data storage and supporting the privacy-preserving objectives of the DIDM.

- *Summary of Data Preparation Approach*

The data collection and preprocessing procedures established a structured foundation for the development and evaluation of the DIDM. The adopted pipeline ensured that benchmark intrusion detection datasets were cleaned, transformed, scaled, and prepared appropriately for machine learning and Federated Learning applications.

The controlled application of SMOTE exclusively to training data, combined with preservation of an independent held-out testing dataset, strengthened the validity of the comparative evaluation between the Centralized Intrusion Detection Model (CIDM) and the Decentralized Intrusion Detection Model (DIDM).

Furthermore, the adoption of federated data handling procedures ensured that raw network traffic observations remained local to participating clients while allowing collaborative model development through parameter aggregation. These procedures collectively supported the objectives of developing a scalable, privacy-preserving, and effective decentralized intrusion detection framework.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

```
# Load & preprocess CICIDS2017 dataset in chunks
file_path = "CICIDS2017.csv"
chunksize = 200_000
unique_labels = set()
for chunk in pd.read_csv(file_path, chunksize=chunksize):
    unique_labels.update(chunk['Attack Type'].unique())
le = LabelEncoder()
le.fit(sorted(list(unique_labels)))

chunks = []
for chunk in pd.read_csv(file_path, chunksize=chunksize):
    for col in chunk.select_dtypes("float64").columns:
        chunk[col] = chunk[col].astype("float32")
    chunk['Attack Type'] = le.transform(chunk['Attack Type']).astype("int8")
    chunks.append(chunk)
df = pd.concat(chunks, ignore_index=True)

X = df.drop(columns=['Attack Type']).astype(np.float32)
y = df['Attack Type'].astype(np.int8)
```

Fig 5 Memory-Efficient Data Loading and Preprocessing for CICIDM2017.

➢ *Model Development and Evaluation Framework*

The development and evaluation of the Decentralized Intrusion Detection Model (DIDM) followed a structured experimental framework designed to assess the effectiveness of decentralized intrusion detection compared with conventional centralized approaches. The framework consisted of baseline model development, Federated Learning-based model development, local anomaly detection integration, ensemble decision formation, and comparative performance evaluation.

A Centralized Intrusion Detection Model (CIDM) was first developed to establish a performance baseline. The CIDM represented a conventional intrusion detection approach in which network traffic data were consolidated within a single computational environment for model training and classification. The developed DIDM was subsequently evaluated against this baseline to determine whether the integration of Federated Learning and distributed anomaly detection mechanisms provided measurable improvements in intrusion detection capability.

Both models were developed using the same prepared dataset, feature representation, training procedures, and independent evaluation dataset to ensure a fair and consistent comparison. The use of identical experimental conditions enabled observed performance differences to be attributed to the architectural differences between centralized and decentralized learning approaches rather than variations in data preparation or evaluation procedures.

- *Centralized Intrusion Detection Model (CIDM)*

The Centralized Intrusion Detection Model (CIDM) was developed as the baseline classification framework for comparison with the Federated Learning-based Decentralized Intrusion Detection Model (DIDM). It represented the conventional intrusion detection approach in which network traffic observations are consolidated and analysed within a centralized computational environment.

In the CIDM configuration, the prepared training dataset was used within a single training environment for model development. The dataset underwent the preprocessing procedures described in the preceding sections, including data preparation, feature transformation, and feature scaling. For the formal CICIDS2017 experiment, the training dataset comprised 2,016,600 observations with 52 input features representing seven traffic classes. The independent testing dataset, containing 504,151 observations, was reserved for model evaluation.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

A feedforward Artificial Neural Network (ANN) was implemented as the primary classification model within the CIDM framework. The ANN was trained centrally using the prepared training dataset to learn the relationships between network traffic characteristics and their corresponding traffic categories. During classification, the trained network generated predictions representing the identified traffic classes based on patterns learned from the training observations.

Following training, the CIDM was evaluated using the independent held-out testing dataset. The evaluation employed the same testing observations and assessment procedures used for the DIDM to ensure a consistent comparative framework. Performance was assessed using classification metrics, including accuracy, precision, recall, F1-score, and false positive rate. Additional threshold-based evaluation measures, including Receiver Operating Characteristic (ROC) and Precision-Recall (PR) analysis, were considered where applicable.

The CIDM provided the centralized baseline against which the Federated Learning-based DIDM was evaluated. By maintaining a common dataset, consistent preprocessing procedures, an identical independent testing set, and comparable evaluation criteria, the experiment enabled an assessment of the differences in classification performance between centralized ANN training and decentralized federated ANN training.

- *Decentralized Intrusion Detection Model (DIDM)*

The Decentralized Intrusion Detection Model (DIDM) was developed using a Federated Learning (FL) architecture to enable collaborative training of an intrusion detection model across multiple simulated clients without transferring raw network traffic records to a central aggregation server. Unlike the centralized configuration, in which training observations are processed within a single computational environment, the DIDM maintained the training data locally at the participating clients.

The DIDM implementation consisted of five simulated federated clients, each representing an independent computational node within the experimental environment. Each client was allocated a portion of the training dataset and independently trained a local Artificial Neural Network (ANN). The local training process enabled each client to learn patterns associated with the different network traffic categories using its assigned observations.

Following local training, the participating clients transmitted their model parameters to the central federated aggregation server rather than exchanging their raw training records. The server combined the received model parameters using the Federated Averaging (FedAvg) algorithm to produce an updated global model. The aggregated global model was then redistributed to the participating clients for subsequent local training.

The federated training process was conducted over five communication rounds. Each round involved distributing the current global model, performing local ANN training at the participating clients, transmitting the resulting model parameters to the aggregation server, and applying FedAvg to update the global model. This iterative process enabled the participating clients to contribute to collaborative model training while retaining their respective training observations locally.

The DIDM therefore combined decentralized local model training with centralized aggregation of model parameters. By retaining raw training observations at the simulated client level, the architecture reduced the need to consolidate the underlying training records within a single repository. However, the experiment evaluated data locality through federated training and did not establish additional formal privacy guarantees such as differential privacy or secure aggregation.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The resulting global ANN model provided the basis for evaluating the federated approach against the Centralized Intrusion Detection Model (CIDM). Both models were assessed using the same independent testing dataset and corresponding evaluation metrics. This enabled a consistent comparison of centralized and federated model training under the experimental conditions established in the study.

- *Local Anomaly Detection Models*

In addition to the primary federated ANN classifier, the broader DIDM implementation incorporated supplementary anomaly detection mechanisms to demonstrate alternative approaches for identifying abnormal network traffic behaviour. These mechanisms were intended to provide additional analytical capabilities within the cybersecurity monitoring environment, particularly for identifying observations that deviated from learned or established traffic patterns.

The supplementary anomaly detection component comprised an Autoencoder, Isolation Forest, and One-Class Support Vector Machine (One-Class SVM). Each technique employed a different approach to identifying anomalous observations and provided an additional analytical perspective within the implemented monitoring environment.

The Autoencoder employed reconstruction-based learning to identify deviations from learned network traffic patterns. It learned to reconstruct input observations, with higher reconstruction errors indicating observations that differed substantially from the patterns represented during model training. Such observations could consequently be flagged for further anomaly assessment.

The Isolation Forest algorithm identified potentially abnormal observations through recursive random partitioning of the feature space. Observations that could be isolated using relatively fewer partitioning steps were assigned higher anomaly significance, indicating that their characteristics differed from those of the broader data distribution.

The One-Class SVM was used to estimate a decision boundary representing the learned feature distribution. Observations falling outside this boundary were treated as potential anomalies, providing another approach for identifying deviations from expected network traffic characteristics.

These supplementary techniques complemented the broader analytical capabilities of the DIDM implementation by demonstrating different approaches to anomaly identification. However, they were distinguished from the formal comparative experiment, which evaluated the centralized ANN classifier against the federated ANN classifier trained using the FedAvg algorithm. The reported CIDM–DIDM benchmark results therefore relate to the ANN-based experimental evaluation rather than a combined ensemble of the Autoencoder, Isolation Forest, and One-Class SVM.

- *Ensemble-Based Detection Framework*

The DIDM implementation was designed around a federated ANN classifier, supported by supplementary local anomaly detection mechanisms. The federated ANN provided the primary supervised classification capability, while the Autoencoder, Isolation Forest, and One-Class Support Vector Machine (One-Class SVM) provided additional approaches for examining unusual network traffic behaviour.

Within the framework, the federated ANN was responsible for classifying network traffic observations into the predefined intrusion categories. Its training process followed the Federated Averaging (FedAvg) approach, through which participating clients performed local model training and shared model parameters for aggregation into a global model. This enabled collaborative model development without requiring the exchange of raw training observations.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The supplementary anomaly detection mechanisms were incorporated into the broader monitoring implementation to demonstrate alternative approaches for identifying deviations from learned network traffic patterns. The Autoencoder used reconstruction errors, Isolation Forest used recursive random partitioning, and One-Class SVM identified observations outside its learned decision boundary. These mechanisms provided additional anomaly assessment capabilities alongside the primary classifier.

The supplementary anomaly detection mechanisms were conceptually complementary to the federated ANN; however, their outputs were not treated as part of the formal ANN-based CIDM–DIDM comparative evaluation. The formal experimental assessment focused on the performance of the centralized ANN classifier and the federated ANN classifier under the same independent test conditions.

The overall design therefore distinguished between the federated classification process and the supplementary anomaly detection functionality. This distinction provided a basis for evaluating the contribution of Federated Learning to decentralized intrusion detection while presenting additional anomaly detection techniques within the broader cybersecurity monitoring environment.

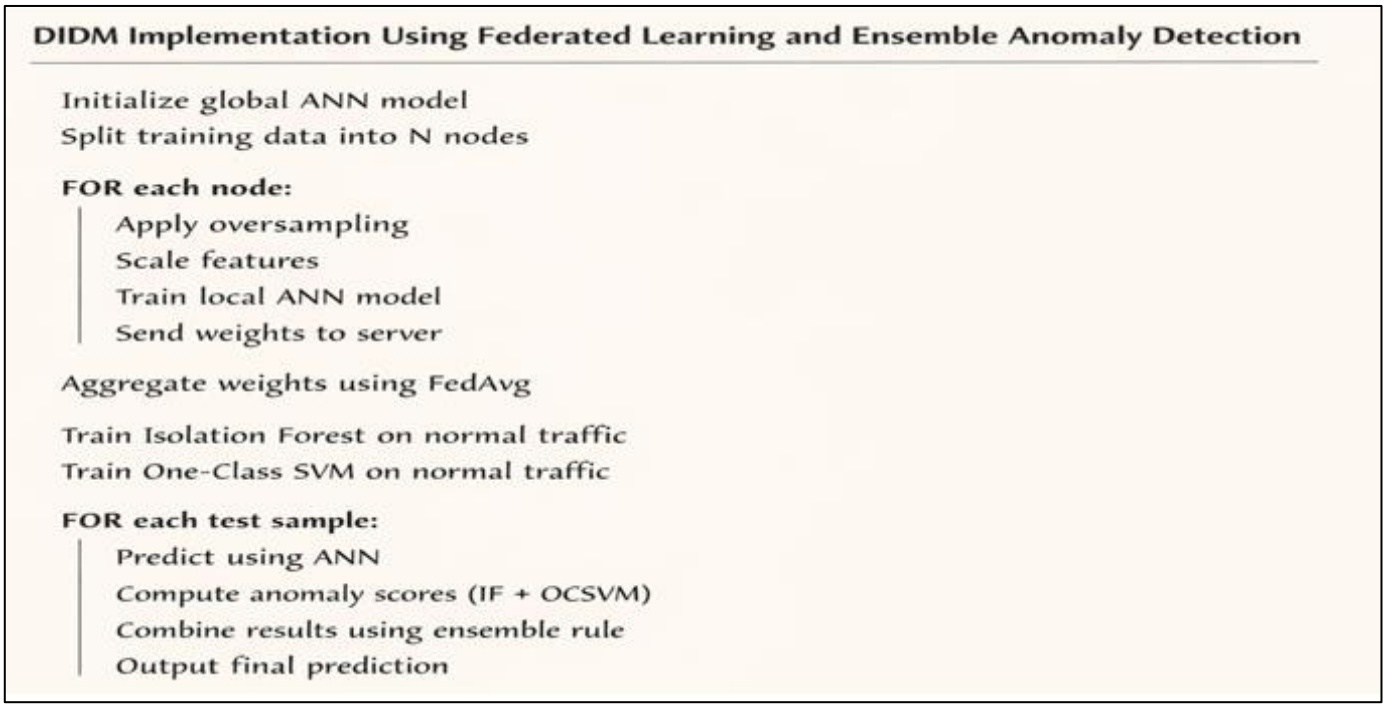


Fig 6 Global ANN Model

- *Model Evaluation*

The performance of the Centralized Intrusion Detection Model (CIDM) and the Decentralized Intrusion Detection Model (DIDM) was evaluated using an independent held-out testing dataset derived from the selected benchmark intrusion detection dataset. A common evaluation framework was applied to both models to ensure that their comparative assessment was conducted under equivalent experimental conditions. The CICIDS2017 dataset comprised 2,520,751 processed observations, represented by 52 features across seven traffic classes. The training and testing partitions contained 2,016,600 and 504,151 observations, respectively. Both models were evaluated using the same independent test partition.

The formal evaluation focused on classification performance measures, including accuracy, precision, recall, F1-score, and false positive rate. These metrics provided complementary perspectives on model performance by measuring overall classification correctness, the reliability of positive predictions, the ability

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

to identify relevant traffic classes, the balance between precision and recall, and the frequency of incorrectly identifying normal traffic as malicious.

Confusion matrices were generated to examine the classification behaviour of both models across the seven traffic classes. They provided a detailed representation of correctly and incorrectly classified observations, supporting the identification of class-level strengths and weaknesses that could not be fully explained by overall accuracy alone.

Threshold-based evaluation measures, including Receiver Operating Characteristic Area Under the Curve (ROC-AUC) and Average Precision (AP), were also considered where model probability outputs were available. These measures provided additional information about model discrimination across different decision thresholds. ROC-AUC assessed the relationship between the true positive rate and false positive rate, while Average Precision summarised performance across the precision–recall curve. These measures complemented, rather than replaced, the classification metrics calculated from final model predictions.

In addition to the formal classification evaluation, selected operational characteristics of the broader DIDM implementation were examined. These included system-level indicators such as detection latency and behaviour within the simulated distributed learning environment. The operational assessment provided supplementary information about the implementation and monitoring environment and was distinguished from the independent test-set evaluation of the ANN classifiers.

The CIDM and DIDM were evaluated using the same independent test observations and corresponding classification procedures. Their results were compared to establish the relative performance of centralized and federated ANN-based intrusion detection. The assessment considered the strengths and limitations of each model across the reported metrics rather than assuming that one approach necessarily performed better across every evaluation measure.

- *Expert Validation*

In addition to the empirical evaluation, the developed Decentralized Intrusion Detection Model (DIDM) was subjected to expert validation involving cybersecurity professionals with relevant knowledge and experience in network security, intrusion detection, information security, and cybersecurity operations. The validation involved 20 professionals and was intended to obtain informed assessments of the technical suitability and practical relevance of the developed framework.

The expert validation complemented the quantitative evaluation by providing professional perspectives on the design and potential applicability of the DIDM within distributed network environments. The assessment considered key aspects of the framework, including the suitability of its decentralized architecture, the relevance of the detection approaches, the coverage of selected cybersecurity threat scenarios, the appropriateness of the evaluation criteria, and its potential application in practical cybersecurity environments.

Responses obtained from the experts were analysed using descriptive statistical techniques appropriate to the evaluation instrument. Frequencies and percentages were used to summarise responses and identify patterns of agreement, perceived weaknesses, and areas requiring further improvement. Where applicable, the findings were interpreted in relation to the specific questions and response options presented in the validation instrument.

The expert validation provided supplementary qualitative and descriptive evidence concerning the perceived technical suitability and practical relevance of the developed DIDM. These findings were considered alongside the empirical model evaluation to provide a broader assessment of the research artefact.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Expert opinions were treated as professional assessments and were not used as substitutes for the measured classification performance of the CIDM and DIDM.

- *Data Analysis Methods*

Quantitative data analysis was conducted using Python-based data analysis, machine learning, and visualization tools. The analytical procedures focused on evaluating and comparing the classification performance of the Centralized Intrusion Detection Model (CIDM) and the Decentralized Intrusion Detection Model (DIDM) using predefined performance measures and appropriate statistical techniques.

Classification performance was analysed using accuracy, precision, recall, F1-score, and false positive rate. These measures were calculated from the predictions generated by each model using the independent test dataset. Confusion matrices were also generated to examine classification behaviour across the different traffic categories and identify patterns of correct and incorrect predictions.

Where probability-based model outputs were available, additional evaluation was conducted using Receiver Operating Characteristic (ROC) curves, ROC-AUC, Precision–Recall curves, and Average Precision (AP). These measures provided further insight into model discrimination performance across different classification thresholds and complemented the metrics derived from final class predictions.

Comparative analysis was undertaken to establish differences between the centralized and federated ANN classifiers. The analysis considered their respective classification performance and, where applicable, statistical evidence of differences in their predictions. Operational characteristics, including selected detection latency and distributed learning indicators, were examined separately as supplementary aspects of the broader DIDM implementation. The supplementary Autoencoder, Isolation Forest, and One-Class SVM mechanisms were not treated as components of the formal ANN-based comparative evaluation.

Expert validation responses were analysed using descriptive statistical techniques. Frequencies and percentages were used to summarise professional assessments of the DIDM architecture, detection approaches, evaluation criteria, and potential practical applicability. The resulting findings were interpreted alongside the quantitative experimental results to provide a broader assessment of the developed research artefact.

The integration of quantitative and qualitative analysis provided a comprehensive evaluation framework for the DIDM. The experimental analysis established measurable evidence of model performance, while expert validation provided practical insights into the relevance and applicability of the developed decentralized intrusion detection approach.

➢ *DIDM Model Architecture and Representation*

The Decentralized Intrusion Detection Model (DIDM) was formulated as a distributed machine learning model that integrated federated learning, artificial intelligence agents, and ensemble-based anomaly detection techniques. The overall architecture was designed to support privacy-preserving collaborative learning while maintaining high detection accuracy, scalability, and robustness within distributed network environments.

The conceptual foundation of the DIDM model was guided by the Input–Process–Guidance–Model Output (IPGS) framework. This framework was used to structure the transformation of network traffic data into actionable security intelligence. In this representation, network traffic data constituted the input, which was processed through federated learning and machine learning pipelines. Artificial intelligence agents provided adaptive guidance for decision-making and response coordination, while the final output consisted of intrusion alerts and security reports.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

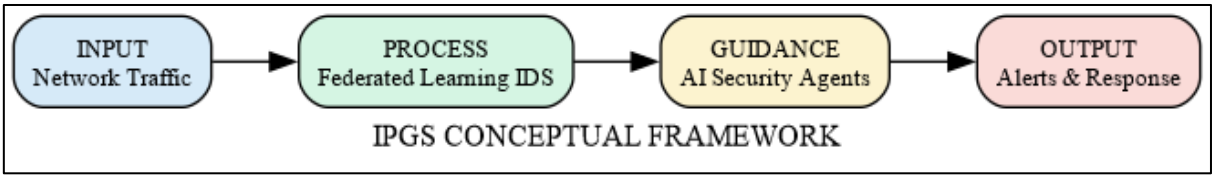


Fig 7 Input–Process–Guidance–Model Output (IPGS) Conceptual Framework

The overall architecture of the DIDM model was designed to operationalize the IPGS structure through a distributed learning environment. The architecture consisted of multiple distributed nodes that independently collected and processed local network traffic data. Each node hosted an artificial intelligence agent layer that facilitated local preprocessing and anomaly detection. Local models were trained independently on private datasets without data sharing, thereby preserving data locality and privacy.

A federated learning server was employed to aggregate model updates from distributed nodes using the Federated Averaging (FedAvg) algorithm. This aggregation process produced a global intrusion detection model, which was subsequently redistributed to participating nodes to support iterative improvement during successive training rounds. This design was adopted to mitigate the limitations of centralized intrusion detection architectures, particularly in relation to scalability constraints, privacy risks, and single points of failure.

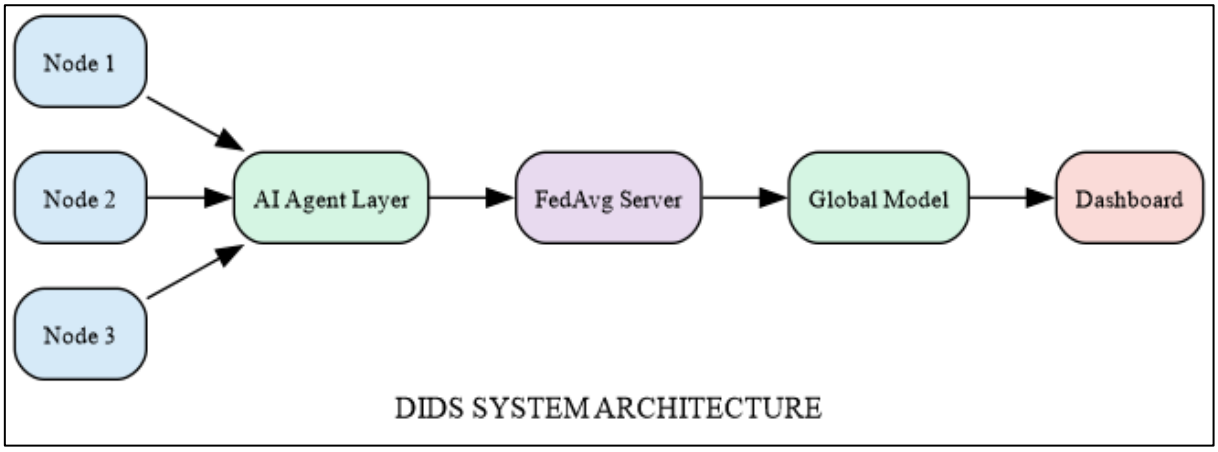


Fig 8 DIDM Model Architecture

The learning process of the model was operationalized through a federated learning workflow in which distributed training was executed across multiple nodes. Each node independently trained a local intrusion detection model using its respective dataset. Only model parameters, including weights and gradients, were transmitted to the central aggregation server. This design ensured that raw network traffic data remained localized, thereby reinforcing data privacy and reducing communication overhead.

The federated server performed weighted aggregation using the FedAvg algorithm to generate an updated global model. The global model was then redistributed to all participating nodes, where it was used to refine subsequent local training iterations. This iterative learning cycle improved convergence stability and enhanced model generalization across heterogeneous network environments.

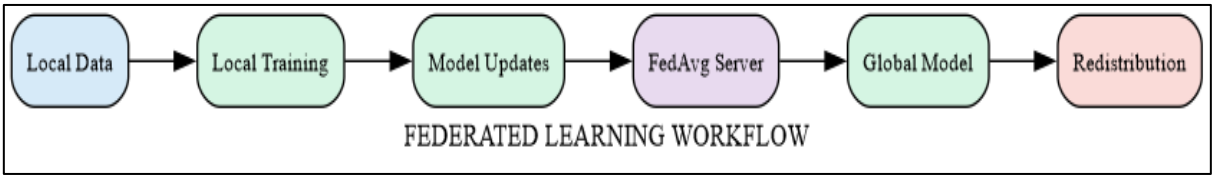


Fig 9 Federated Learning Model Workflow (FedAvg-Based Training Process)

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The intelligence and adaptive decision-making capability of the model was enhanced through an artificial intelligence agent framework. The framework consisted of monitoring, detection, learning, and response agents. The monitoring agent continuously observed network traffic patterns, while the detection agent identified anomalous behavior using trained local and global models. The learning agent updated model parameters based on newly observed patterns, and the response agent executed mitigation actions based on detected threats. The integration of these agents improved the adaptability of the model in dynamic network conditions.

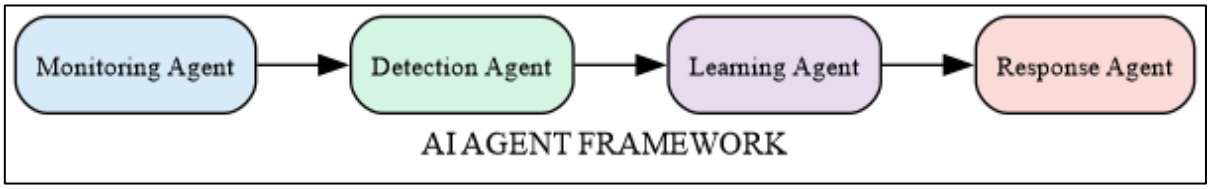


Fig 10 AI Agent Framework in the DIDM Model

The intrusion detection capability of the model was further strengthened through an ensemble-based detection pipeline. The pipeline integrated multiple anomaly detection techniques, including autoencoders, isolation forest, and one-class support vector machines. Network traffic data was first transformed into feature representations, after which each algorithm independently evaluated anomaly likelihood. The outputs generated by these models were subsequently fused using a decision-level aggregation mechanism to produce a final classification output.

This ensemble strategy was adopted to address the limitations of individual detection algorithms, particularly their sensitivity to specific attack types and feature distributions. The integration of multiple detectors improved robustness, reduced false positive rates, and enhanced detection reliability for both known and previously unseen attack patterns.

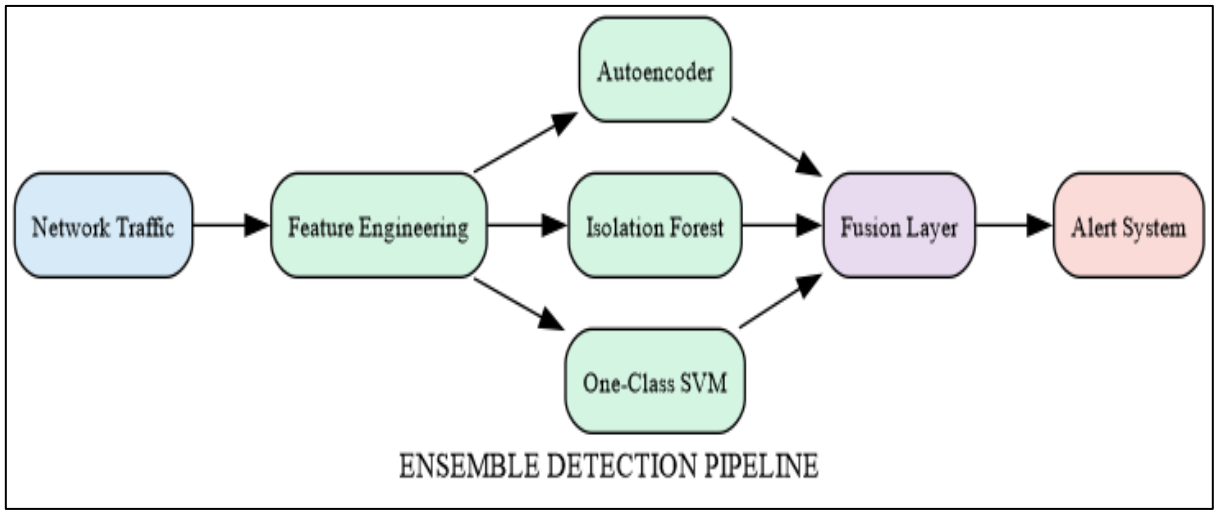


Fig 11 Ensemble-Based Intrusion Detection Model Pipeline

The experimental evaluation of the model was conducted using a distributed testbed that simulated real-world federated learning environments. The testbed was implemented using Docker-based nodes coordinated through a federated learning framework. Each node simulated an independent computational environment that supported local training and artificial intelligence agent operations. This configuration enabled controlled experimentation under heterogeneous and decentralized network conditions.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

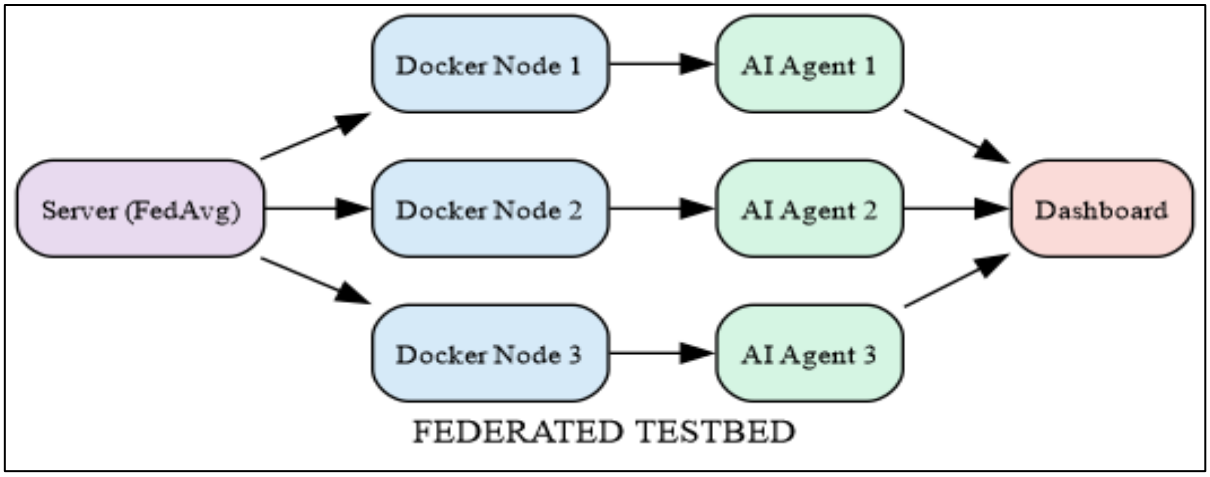


Fig 12 Federated Learning Model Testbed (Docker + Flower Environment)

Collectively, the architectural components of the model were integrated to form a cohesive distributed intrusion detection framework. The federated learning layer ensured privacy-preserving collaborative training, the artificial intelligence agent layer provided adaptive decision-making capabilities, and the ensemble detection layer enhanced classification robustness. The integration of these components resulted in a scalable and resilient intrusion detection model suitable for deployment in modern distributed and cloud-based network environments.

➢ *Summary of the Model Architecture*

The Decentralized Intrusion Detection Model (DIDM) was developed as a distributed learning framework that integrated federated learning, artificial intelligence agents, and ensemble-based anomaly detection within a unified architecture. The design was structured to address key limitations associated with centralized intrusion detection approaches, particularly with respect to data privacy risks, scalability constraints, and vulnerability to single points of failure.

The model was founded on the Input–Process–Guidance–Output (IPGO) conceptual structure, where network traffic data was processed through distributed machine learning pipelines. The processing stage was executed through federated learning, in which local models were trained independently at distributed nodes without sharing raw data. Model updates were aggregated using the Federated Averaging (FedAvg) algorithm to construct a global intrusion detection model, which was iteratively refined across multiple communication rounds.

Artificial intelligence agents were embedded within each node to enhance autonomous decision-making. These agents were responsible for continuous monitoring of network traffic, local anomaly detection, adaptive learning from evolving patterns, and coordinated response actions. This agent-based layer strengthened the model's ability to operate in dynamic and heterogeneous network environments.

The detection capability of the model was further enhanced through an ensemble-based anomaly detection framework that combined multiple learning techniques, including autoencoders, isolation forest, and one-class support vector machines. The fusion of these methods improved the model's ability to detect both known and previously unseen attack patterns while reducing false positive rates and improving classification stability.

Overall, the architecture established a privacy-preserving, scalable, and resilient intrusion detection model in which distributed learning, intelligent agents, and ensemble decision-making worked in an integrated manner. The resulting framework provided a strong foundation for next-generation intrusion detection in modern distributed and cloud-based network environments.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

➢ *Chapter Summary*

This chapter presented a comprehensive account of the methodology that was adopted in the design, implementation, and evaluation of the Federated Learning-based Decentralized Intrusion Detection Model (DIDM). It provided a detailed description of the research design, simulation environment, population of the study, sampling procedures, tools and technologies, data collection and preprocessing techniques, model implementation strategy, evaluation framework, and ethical considerations that guided the study.

The methodological approach was structured to ensure scientific rigor, methodological coherence, and reproducibility of results. The study was anchored on the Design Science Research Methodology (DSRM), which supported a Modelatic progression from problem identification, model design, implementation, evaluation, and refinement. Federated learning was integrated as the core distributed learning paradigm to ensure that raw network traffic data remained localized within individual nodes, while only model parameters were exchanged during the learning process. This approach enhanced data privacy, reduced communication overhead, and improved scalability in distributed network environments.

The chapter further demonstrated how simulation-based experimentation was employed to replicate realistic and heterogeneous network conditions using benchmark intrusion detection datasets. The incorporation of controlled experimental settings ensured that the performance of the DIDM model was evaluated under consistent and reproducible conditions. In addition, expert validation was used to strengthen the practical relevance of the model by incorporating domain knowledge from cybersecurity professionals, particularly in assessing threat realism, model applicability, and operational feasibility.

The evaluation framework established in this chapter enabled Modelatic comparison between the centralized intrusion detection model and the decentralized intrusion detection model. This comparison was based on standardized performance metrics, including detection accuracy, precision, recall, F1-score, false positive rate, scalability, and communication efficiency. The analytical framework ensured that both algorithmic performance and Model-level efficiency were adequately assessed.

The ethical considerations governing the study were also addressed in detail, ensuring compliance with institutional research standards and responsible data handling practices. These included data privacy protection, informed consent procedures for expert participation, secure data storage, and adherence to principles of transparency and academic integrity.

Overall, the methodology established a robust foundation for the development and evaluation of the DIDM model by integrating distributed machine learning principles, simulation-based validation, and expert-informed assessment within a unified research framework. This ensured that the resulting model was not only technically sound but also practically applicable in modern cybersecurity environments.

The subsequent chapter presents the results, analysis, and interpretation of the DIDM model. It provides a comparative evaluation between the centralized intrusion detection model and the decentralized intrusion detection model, with the objective of demonstrating the performance gains, scalability improvements, and robustness achieved by the DIDM framework.

➢ *Ethical Considerations*

The study was conducted in accordance with the ethical principles and research requirements of Kabarak University. Ethical approval was obtained from the Kabarak University Ethics Review Committee, and the approved research procedures guided the experimental activities, use of research datasets, recruitment of expert participants, and management of research information. The study observed the principles of voluntary participation, informed consent, confidentiality, responsible data handling, academic integrity, and transparency throughout the research process.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *Research Data and Privacy Protection*

The study utilised publicly available benchmark intrusion detection datasets, including CICIDS2017, NSL-KDD, and UNSW-NB15, for research and experimental purposes. These datasets provided network traffic observations for examining intrusion detection approaches without requiring the collection of live traffic from private organisational networks. Experimental activities were conducted within controlled research environments to avoid unauthorised access to operational systems or disruption of existing network services.

The study did not intentionally collect personally identifiable information from network users. Research datasets and experimental outputs were handled for the purposes of model development, testing, and evaluation. Appropriate measures were taken to restrict access to research information and maintain the integrity of the experimental records.

Federated Learning was incorporated to support decentralised model training. Participating clients performed local training and shared model parameters for aggregation rather than transferring their raw training observations to a central server. This reduced the need for centralised collection of training data. However, the use of Federated Learning was not interpreted as providing an absolute privacy guarantee, since additional mechanisms such as differential privacy and secure aggregation were not part of the formal experimental implementation.

- *Expert Participation and Informed Consent*

The study incorporated expert validation involving twenty (20) cybersecurity professionals with relevant knowledge and experience in areas such as network security, intrusion detection, information security, artificial intelligence, and cybersecurity operations. Participants were selected through purposive sampling based on their professional experience, research background, and relevance to the objectives of the study.

Requests for participation were formally communicated to the respective organisational managers, who were requested to assist in identifying suitable personnel with relevant research backgrounds and direct professional involvement in centralized and decentralized intrusion detection systems (CIDM and DIDM). This recruitment approach helped identify individuals with the technical knowledge and practical experience necessary to assess the architecture, detection approaches, evaluation criteria, and potential applicability of the developed model.

The identified professionals were invited to participate through appropriate communication channels and provided with information explaining the purpose of the study, the nature of their participation, and the expected requirements. Data were collected using a structured questionnaire administered through KoboToolbox.

Participation was voluntary, and informed consent was obtained from individual respondents before they proceeded with the questionnaire. Participants were informed of their right to decline participation or withdraw without penalty or adverse consequences. Organisational managers facilitated the identification of potential participants but did not provide consent on behalf of the individuals. The expert validation exercise provided professional assessments of the DIDM architecture, detection approaches, evaluation criteria, and potential applicability in distributed network environments. These assessments complemented the empirical evaluation and were not treated as substitutes for the measured performance of the developed models.

- *Confidentiality and Data Protection*

Confidentiality was maintained throughout the expert validation and experimental research processes. Participant responses were handled for academic research purposes, and findings were presented in aggregated

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

form to minimise the possibility of identifying individual respondents. Personal details and organisational affiliations were not disclosed in the presentation of the research findings.

Access to collected research information was restricted to its legitimate research use. Research datasets, questionnaire responses, experimental records, and model outputs were handled responsibly to minimise the risk of unauthorised access, disclosure, alteration, or loss. Data collection and management procedures were guided by the ethical conditions approved by the Kabarak University Ethics Review Committee and applicable institutional requirements.

The study also observed the principle of data minimisation by limiting the use of research information to what was necessary for the stated research objectives. Confidentiality was therefore considered both in the handling of expert responses and in the management and reporting of experimental data.

- *Research Integrity and Responsible Conduct*

Academic integrity was maintained throughout the study. All datasets, algorithms, software tools, scholarly publications, and other materials obtained from external sources were appropriately acknowledged and referenced using the required academic referencing style. Existing methods and models adapted for the research were distinguished from the original contributions of the study.

The experimental procedures, model configurations, data preprocessing activities, and evaluation methods were documented to support transparency and reproducibility. Research findings were reported objectively, without deliberate fabrication, falsification, selective manipulation, or misrepresentation of results. Both the strengths and limitations of the developed DIDM were considered when interpreting the findings.

The study also distinguished the formal ANN-based comparative experiment between the centralized and federated models from the supplementary anomaly detection and monitoring components. This distinction ensured that reported performance results were attributed to the appropriate experimental procedures. Furthermore, the research acknowledged the limitations of benchmark datasets and controlled experimental environments when considering the applicability of the findings to operational cybersecurity settings.

Overall, the study maintained the principles of responsible research conduct, respect for participants, confidentiality, transparency, and compliance with the institutionally approved research procedures.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

CHAPTER FOUR
DATA ANALYSIS, INTERPRETATION, AND PRESENTATION

➢ *Introduction*

This chapter presents the findings obtained from the implementation and evaluation of the Decentralized Intrusion Detection Model (DIDM) using Federated Learning. The formal experimental evaluation compares the DIDM with a Centralized Intrusion Detection Model (CIDM) using the CICIDS2017 benchmark dataset and a common independent test set.

The analysis covers dataset characteristics and preparation, federated model training, comparative classification performance, confusion matrix analysis, Receiver Operating Characteristic (ROC) and Precision–Recall analyses, and statistical validation. The DIDM employs the Federated Averaging (FedAvg) algorithm to aggregate model updates from participating clients while retaining raw training data locally at each client.

In addition to the formal experimental findings, this chapter presents results from the expert assessment and the supplementary cybersecurity dashboard demonstration. These findings are considered separately because they serve different analytical purposes from the formal benchmark evaluation. This distinction ensures that conclusions regarding the comparative classification performance of the DIDM and CIDM are based on the formal experimental results, while the expert assessment and dashboard demonstration provide supplementary evidence on system design considerations and operational aspects.

➢ *Dataset Description and Preparation*

The CICIDS2017 dataset was analysed to establish its characteristics and suitability for developing and evaluating the Decentralized Intrusion Detection Model (DIDM). The analysis examined the dataset size, input features, classification categories, and distribution of observations across the different network traffic classes.

The processed dataset comprised 2,520,751 records, represented by 52 input features and seven traffic classes. The observations were partitioned into 2,016,600 training samples and 504,151 independent testing samples, representing an approximately 80:20 training-to-testing ratio. The dataset summary is presented in Table 2.

Table 2 Dataset Summary

Parameter	Value
Training Samples	2,016,600
Testing Samples	504,151
Number of Features	52
Number of Classes	7

The independent test set, comprising 504,151 records, provided a common evaluation basis for comparing the DIDM and the Centralized Intrusion Detection Model (CIDM). Both models were evaluated using the same test observations to maintain consistency in the comparative assessment.

The distribution of observations across the seven network traffic classes is illustrated in Figure 13.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

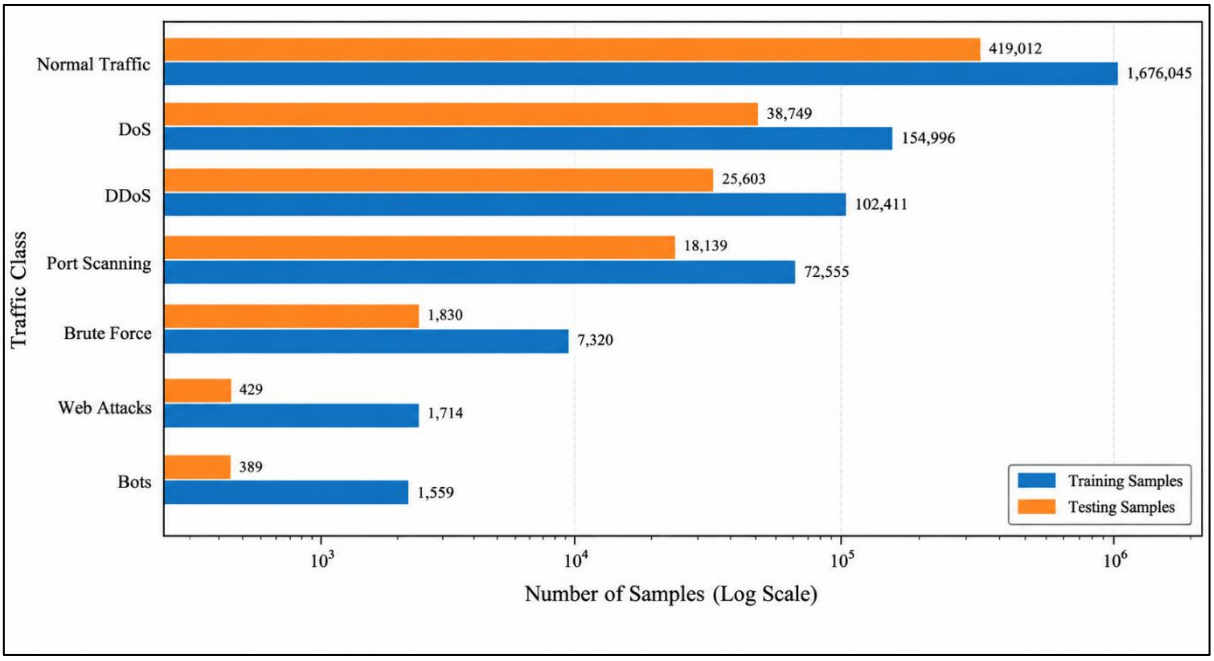


Fig 13 Distribution of Network Traffic Classes in the CICIDS2017 Dataset

A numerical breakdown of the training and testing samples for each traffic category is provided in Table 3.

Table 3 Distribution of Network Traffic Classes

Traffic class	Training samples	Testing samples
Bots	1,559	389
Brute Force	7,320	1,830
DDoS	102,411	25,603
DoS	154,996	38,749
Normal Traffic	1,676,045	419,012
Port Scanning	72,555	18,139
Web Attacks	1,714	429
Total	2,016,600	504,151

Note. DDoS = Distributed Denial of Service; DoS = Denial of Service.

Table 3 demonstrates substantial variation in the representation of the seven traffic classes. Normal Traffic was the dominant category, accounting for 1,676,045 training observations and 419,012 testing observations. The DoS and DDoS categories also had relatively high representation, with 154,996 and 102,411 training observations, respectively.

In contrast, several attack categories had considerably fewer observations. Bots accounted for 1,559 training samples, while Web Attacks and Brute Force accounted for 1,714 and 7,320 samples, respectively. This distribution indicates class imbalance within the dataset, with normal traffic substantially outweighing several malicious traffic categories.

Class imbalance is an important consideration in intrusion detection because the dominance of one traffic category can influence overall classification performance. A model may achieve high overall accuracy while exhibiting comparatively weaker performance in identifying underrepresented attack categories. Consequently, accuracy alone may not provide a complete assessment of classification effectiveness across all traffic classes.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

To address this evaluation consideration, precision, recall, and F1-score were also used to examine model performance. In particular, macro-averaged metrics provided insight into performance across individual classes without weighting them according to their sample sizes, while weighted-averaged metrics accounted for the relative representation of each class in the dataset. These complementary measures supported a more comprehensive interpretation of the comparative performance of the CIDM and DIDM.

The dataset characteristics and class distribution established the basis for subsequent model training and evaluation. They also provided important context for interpreting the comparative results, particularly where differences in classification performance occurred between dominant traffic categories and less-represented attack classes.

➢ *Federated Learning Implementation Results*

Following dataset preparation, the Decentralized Intrusion Detection Model (DIDM) was trained using a federated learning architecture based on the Federated Averaging (FedAvg) algorithm. The training dataset was distributed across five simulated participating clients, each independently training a local neural network model using its assigned subset of the data. At the end of each communication round, the clients transmitted their updated model parameters to a central aggregation server, where the FedAvg algorithm was applied to generate an updated global model. The global model was subsequently redistributed to the participating clients for the next communication round. Raw network traffic observations remained within the respective client datasets during this experimental process.

The federated training process was conducted over five communication rounds. During each round, participating clients performed local model optimisation before submitting their updated model parameters for aggregation. Global model accuracy and training loss were recorded to assess the learning progression and convergence behaviour of the federated model. The results are summarised in Table 4.

Table 4 Federated Learning Training Performance Across Communication Rounds

Communication Round	Global Accuracy (%)	Training Loss
1	97.25	0.0690
2	98.04	0.0492
3	98.23	0.0436
4	98.21	0.0409
5	98.27	0.0388

Table 4 shows that the federated learning process achieved high global model accuracy throughout the five communication rounds. Accuracy increased from 0.972488403 (97.25%) in the first round to 0.982745230 (98.27%) in the fifth round, representing an overall improvement of approximately 1.03 percentage points. Training loss decreased consistently from 0.069036268 to 0.038845569 over the same period. These results indicate progressive optimisation of the global model through successive local training and parameter aggregation.

The largest improvement in global accuracy occurred between the first and second communication rounds, increasing from approximately 97.25% to 98.04%. Accuracy subsequently increased to 98.23% in the third round, before recording a slight decrease to approximately 98.21% in the fourth round. It then increased to 98.27% in the fifth round. Although accuracy did not increase in every communication round, the small variation between the third and fourth rounds, together with the continued reduction in training loss, indicates a generally stable learning progression during the experiment.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

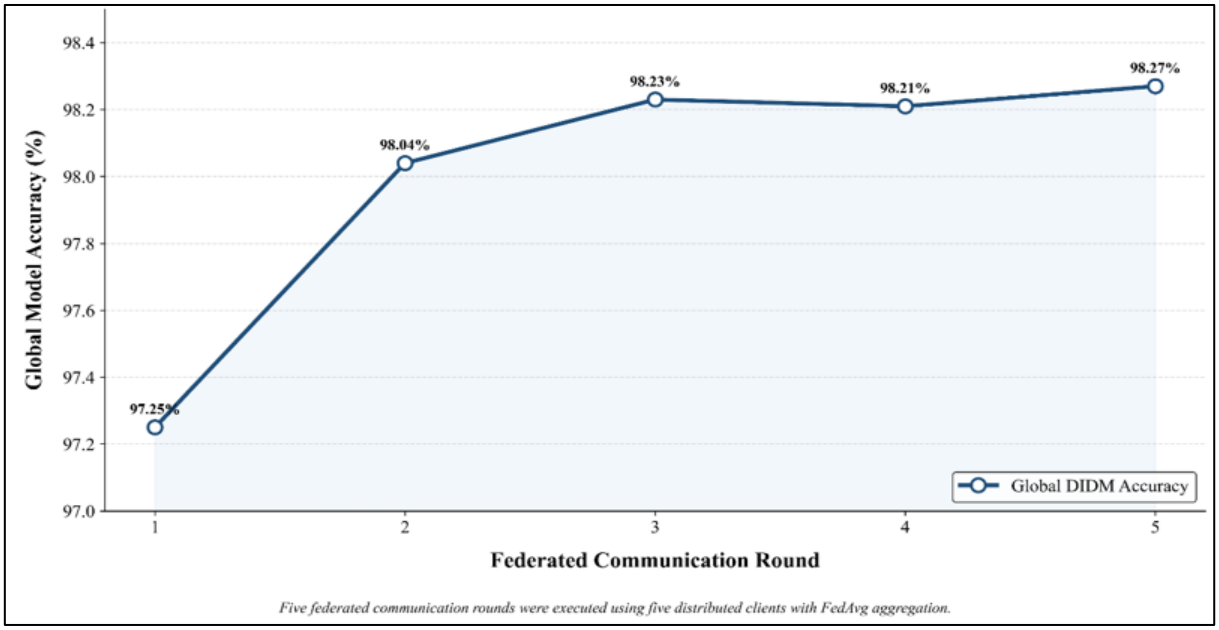


Fig 14 (a): Federated Training Accuracy Across Communication Rounds

Figure 14 (a) illustrates the progression of global model accuracy across the five federated communication rounds. The model recorded a substantial improvement during the initial rounds, followed by relatively small changes in accuracy during the later rounds. The highest recorded global accuracy was approximately 98.27% in the fifth communication round. The slight decrease observed between the third and fourth rounds demonstrates that federated model accuracy may fluctuate between successive aggregation rounds rather than increase uniformly.

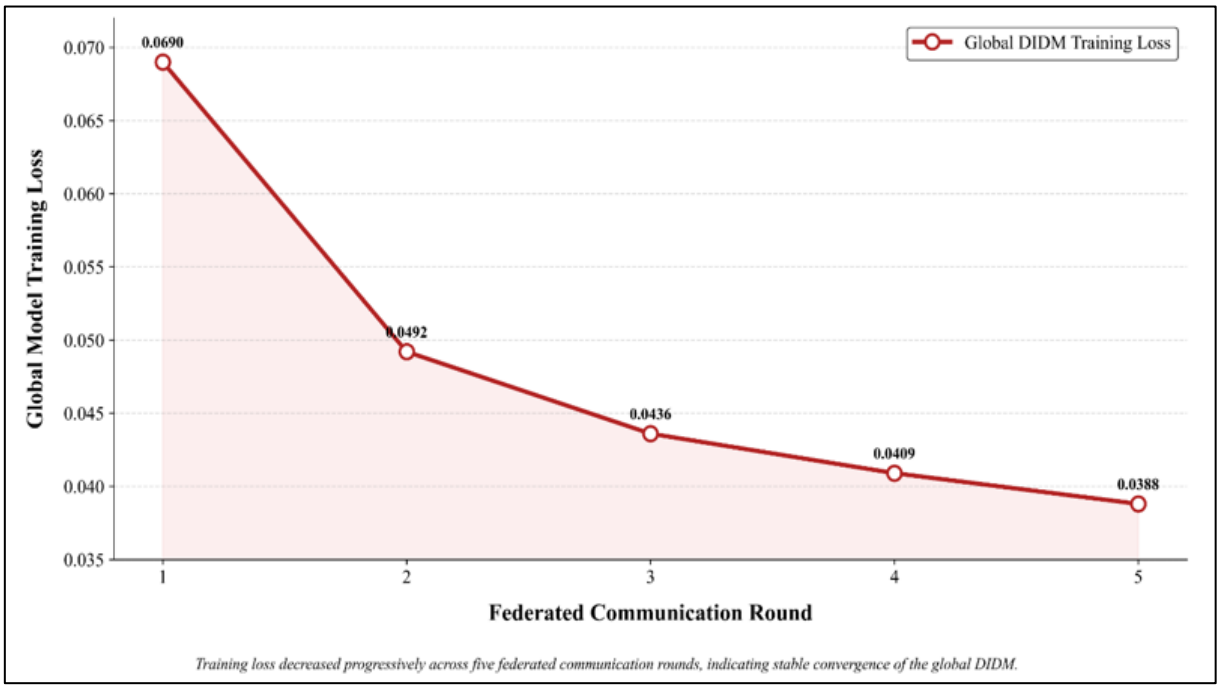


Fig 14 (b): Federated Training Loss Across Communication Rounds

Figure 14 (b) presents the corresponding changes in training loss during federated learning. The loss decreased consistently from 0.069036268 in the first communication round to 0.038845569 in the fifth round. This downward trend indicates that the optimisation process progressively reduced the model's training loss as local parameter updates were incorporated into the global model. Considered alongside the global accuracy

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

results, the reduction in training loss provides evidence of progressive learning during the five-round FedAvg training process.

Overall, the results presented in Table 4 and Figures 14 (a) and 14 (b) demonstrate that the developed DIDM achieved high global model accuracy and a consistent reduction in training loss across the five federated communication rounds. The results provide evidence of effective model optimisation and stable learning progression within the experimental federated architecture. Since raw network traffic observations remained local to the participating clients, the implementation demonstrated decentralised model training without centralising the underlying training records. However, this data-locality arrangement should not be interpreted as a formal privacy guarantee, since additional mechanisms such as differential privacy or secure aggregation were not part of the evaluated implementation. The resulting global model was subsequently subjected to comparative performance evaluation against the Centralized Intrusion Detection Model (CIDM).

➢ *Analysis of Experimental Results*

This section analyses the experimental results obtained from evaluating the Decentralized Intrusion Detection Model (DIDM) against the Centralized Intrusion Detection Model (CIDM). The analysis focuses on overall classification performance, the models' ability to identify different network traffic categories, and the statistical significance of the observed differences in their classification outcomes. Both models were evaluated using the same independent test dataset comprising 504,151 observations, providing a common basis for direct performance comparison.

The evaluation considers overall accuracy alongside macro-averaged and weighted precision, recall, and F1-score. Accuracy measures the proportion of correctly classified observations across the entire test dataset. Macro-averaged metrics assign equal importance to each traffic class when calculating the overall metric, whereas weighted-averaged metrics account for the relative representation of the individual classes. Considering these measures together provides a more comprehensive assessment of classification performance, particularly given the substantial differences in class representation within the CICIDS2017 dataset.

The subsequent analysis examines the comparative classification metrics, class-level performance outcomes, and supporting diagnostic evidence, including confusion matrix, Receiver Operating Characteristic (ROC) curve, and Precision–Recall (PR) curve analyses. The observed differences in the paired classification outcomes are further examined using the exact McNemar test, which is appropriate because both models were evaluated on the same independent test observations. This test assesses whether the difference in the models' classification errors is statistically significant. The findings are interpreted in relation to the study objectives and the contribution of the federated learning approach to decentralized intrusion detection.

➢ *Comparative Performance Evaluation of the CIDM and DIDM*

This section presents a comparative evaluation of the Centralized Intrusion Detection Model (CIDM) and the developed Decentralized Intrusion Detection Model (DIDM). The objective was to establish whether the federated learning-based decentralized architecture achieved improved intrusion detection performance compared with the conventional centralized approach. Both models were evaluated using the same independent test dataset comprising 504,151 records from seven network traffic classes.

The comparison was based on overall accuracy, macro-averaged precision, macro-averaged recall, macro F1-score, weighted precision, weighted recall, and weighted F1-score. Macro-averaged metrics assigned equal importance to each traffic class, whereas weighted metrics accounted for the relative representation of each class in the evaluation dataset. The comparative results are presented in Table 5.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Table 5 Presents the Comparative Performance of the Two Models.

Metric	CIDM	DIDM	Difference
Accuracy	96.74%	98.27%	+1.53%
Macro Precision	67.38%	90.97%	+23.59%
Macro Recall	98.32%	77.93%	−20.39%
Macro F1-score	72.99%	78.26%	+5.27%
Weighted Precision	98.36%	98.32%	−0.04%
Weighted Recall	96.74%	98.27%	+1.53%
Weighted F1-score	97.39%	98.25%	+0.86%

➢ *Table 5 Comparative Performance of the CIDM and DIDM*

Table 5 shows that the developed DIDM achieved higher overall classification accuracy than the CIDM. Accuracy increased from 96.74% for the centralized model to 98.27% for the decentralized model, representing an improvement of approximately 1.53 percentage points. This result indicates that the DIDM correctly classified a greater proportion of observations in the independent test dataset. A substantial difference was also observed in macro precision, which increased from 67.38% for the CIDM to 90.97% for the DIDM. The macro F1-score similarly improved from 72.99% to 78.26%. These findings indicate that the DIDM achieved improved precision and a better overall balance between precision and recall when performance was averaged equally across the seven traffic classes.

The weighted metrics further demonstrated the comparative performance of the two models. DIDM achieved a weighted recall of 98.27%, compared with 96.74% for CIDM, while its weighted F1-score increased from 97.39% to 98.25%. However, CIDM recorded a marginally higher weighted precision of 98.36%, compared with 98.32% for DIDM. This small difference indicates that the DIDM did not outperform the centralized model on every evaluation metric.

In contrast, macro recall decreased from 98.32% for CIDM to 77.93% for DIDM. This indicates that the decentralized model had greater difficulty identifying all instances across certain individual traffic classes when each class was given equal weight. The lower macro recall is particularly important because it highlights the detection challenges associated with underrepresented attack categories. Therefore, the higher overall accuracy and macro precision of DIDM should be interpreted alongside its lower macro recall rather than as evidence of uniformly superior performance across all classes.

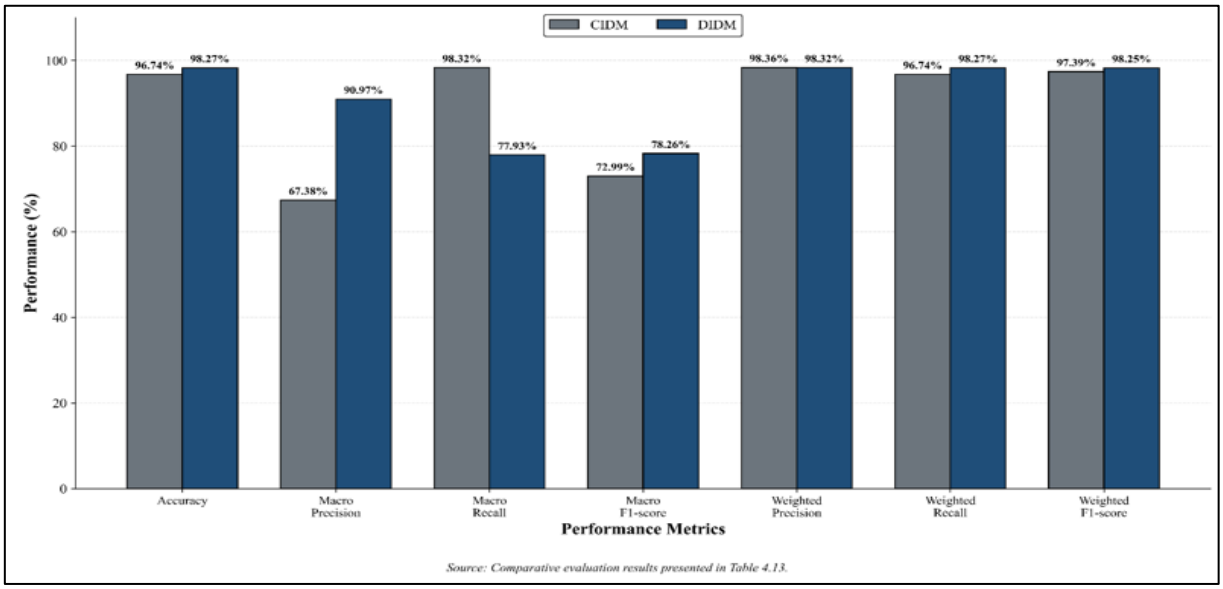


Fig 15 (a): Comparative Performance of the CIDM and DIDM

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 15 (a) presents a graphical comparison of the classification performance of the CIDM and DIDM across the evaluated metrics. The figure illustrates the higher DIDM results in overall accuracy, macro precision, macro F1-score, weighted recall, and weighted F1-score. It also shows the marginally higher weighted precision and higher macro recall achieved by CIDM.

Overall, the comparative results demonstrate that the developed DIDM achieved higher overall accuracy and improved performance on several important evaluation metrics. Nevertheless, the lower macro recall indicates that certain minority attack categories remained challenging to detect. These findings provide the basis for the subsequent confusion matrix, ROC, Precision–Recall, and statistical analyses, which examine the classification behaviour and comparative performance of the two models in greater detail.

- *Classification Performance Analysis Using the Confusion Matrix*

To provide a detailed assessment of the classification behaviour of the Centralized Intrusion Detection Model (CIDM) and the developed Decentralized Intrusion Detection Model (DIDM), confusion matrix analysis was performed using the independent test dataset. While the comparative metrics presented in Table 6 summarised the overall performance of the two models, the confusion matrices provided class-level information by showing the distribution of correctly and incorrectly classified observations across the seven network traffic categories. This analysis helped identify the traffic categories that were detected more accurately and those that remained challenging because of class imbalance and similarities in network traffic characteristics.

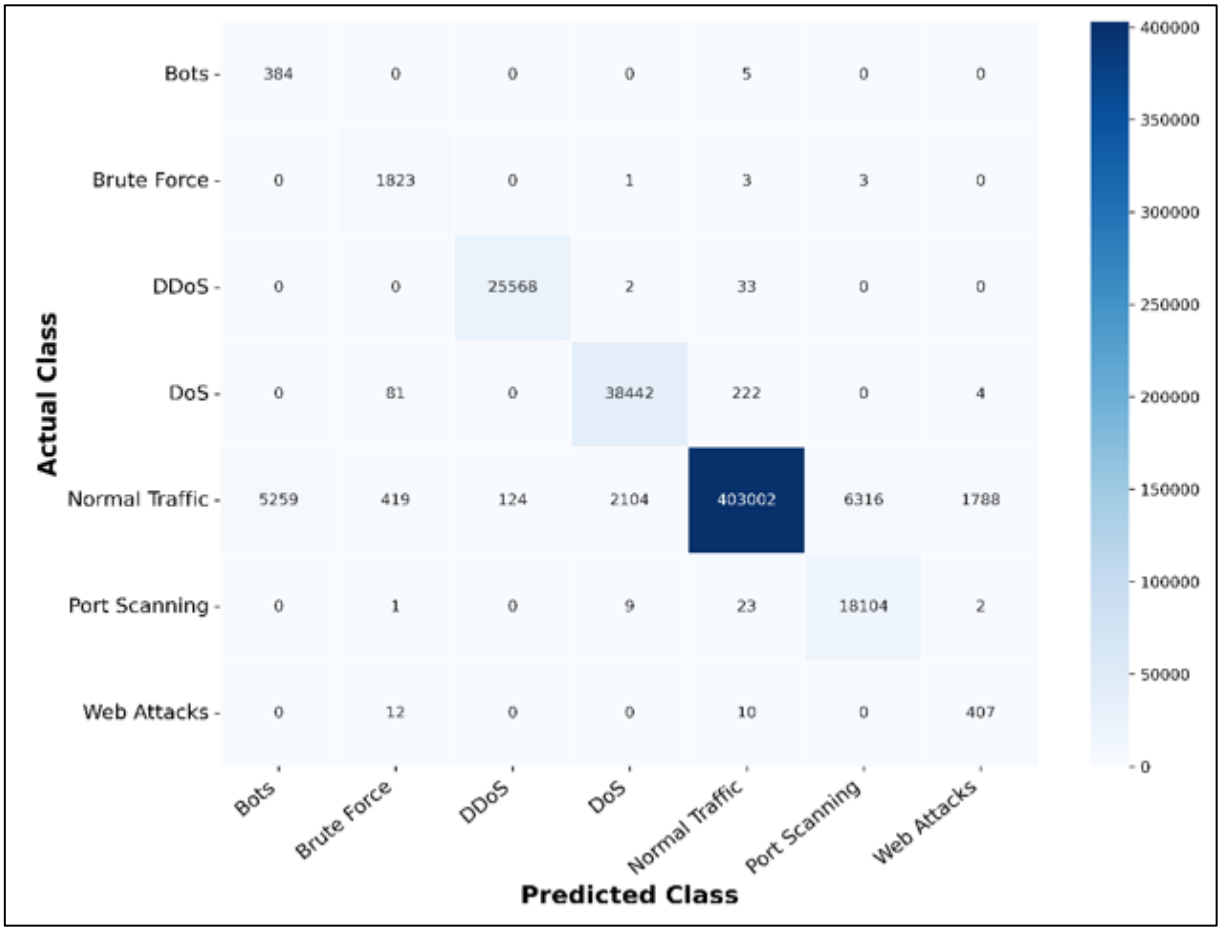


Fig 15 (b): Confusion Matrix for the Centralized Intrusion Detection Model (CIDM)

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

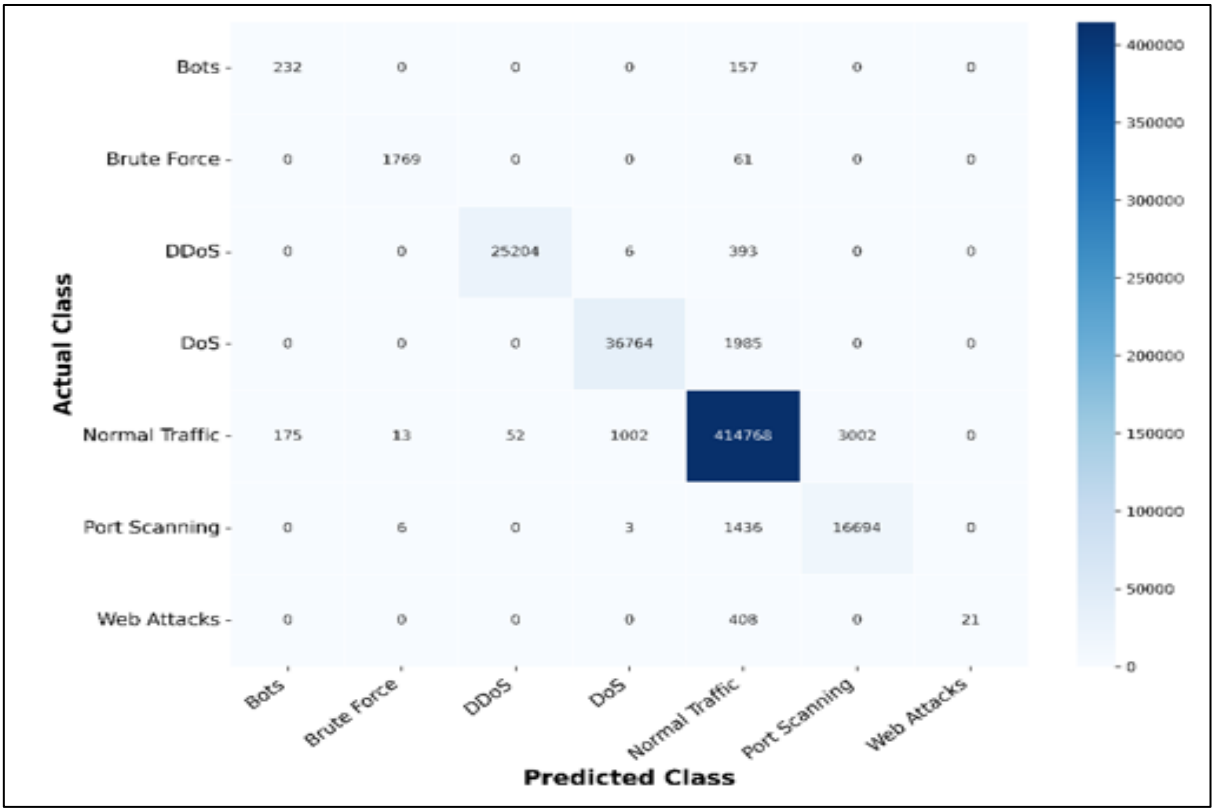


Fig 15 (c): Confusion Matrix for the Developed Decentralized Intrusion Detection Model (DIDM)

Figures 15(b) and 15(c) illustrate the classification behaviour of the two models. The concentration of correctly classified observations along the principal diagonals indicates that both models learned patterns that enabled them to distinguish between the seven network traffic categories. However, the differences in the distribution of classification errors reveal variations in their ability to identify individual traffic classes.

For the Normal Traffic category, the CIDM correctly classified 403,002 of the 419,012 test observations, whereas the DIDM correctly classified 414,768 observations. The DIDM therefore correctly identified 11,766 additional Normal Traffic observations compared with the CIDM. This result indicates an improvement in the classification of legitimate network traffic and a reduction in the number of legitimate observations assigned to other traffic categories.

Both models demonstrated strong classification performance for the Distributed Denial of Service (DDoS) and Denial of Service (DoS) categories. The CIDM correctly classified 25,568 of the 25,603 DDoS observations and 38,442 of the 38,749 DoS observations. In comparison, the DIDM correctly classified 25,204 DDoS observations and 36,764 DoS observations. These results show that the CIDM achieved higher correct-classification counts for these two attack categories. This finding is consistent with the lower macro recall recorded by the DIDM in Table 4.13 and demonstrates that its higher overall accuracy did not translate into better detection across every individual class.

The DIDM also demonstrated strong classification performance for Brute Force and Port Scanning attacks. It correctly classified 1,769 of the 1,830 Brute Force observations and 16,694 of the 18,139 Port Scanning observations. These results indicate that the model identified a substantial proportion of the observations belonging to these attack categories. Nevertheless, the remaining misclassifications show that some attack observations were assigned to other traffic categories, including Normal Traffic.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Lower classification performance was observed for the minority attack categories, particularly Bots and Web Attacks. The DIDM correctly classified 232 of the 389 Bot observations, while 157 were misclassified. For Web Attacks, only 21 of the 429 observations were correctly classified, indicating substantial difficulty in identifying this category. These findings highlight the challenges associated with detecting underrepresented attack categories in an imbalanced multiclass dataset. The limited representation of these categories may have affected the model's ability to learn sufficiently discriminative patterns for their identification.

The confusion matrix results help explain the differences observed in the comparative performance metrics. Although the DIDM achieved higher overall accuracy, macro precision, and macro F1-score, its lower macro recall indicates that it did not identify the same proportion of observations across all seven classes as the CIDM. In particular, the higher CIDM correct-classification counts for DDoS and DoS demonstrate that the centralized model retained an advantage in identifying observations belonging to those categories. Conversely, the DIDM correctly classified substantially more Normal Traffic observations. These differences illustrate why model performance should be assessed using multiple evaluation metrics rather than overall accuracy alone.

Overall, the confusion matrix analysis demonstrates that both models achieved strong classification performance on the independent test dataset, although their class-level performance differed. The DIDM achieved a higher number of correctly classified Normal Traffic observations and strong results in several attack categories. However, the comparatively low number of correctly classified Bots and Web Attacks observations highlights areas requiring further improvement. These findings provide class-level evidence that complements the comparative evaluation in Table 5 and provide a basis for interpreting the ROC and Precision–Recall analyses in the following sections.

- *Receiver Operating Characteristic (ROC) Curve Analysis*

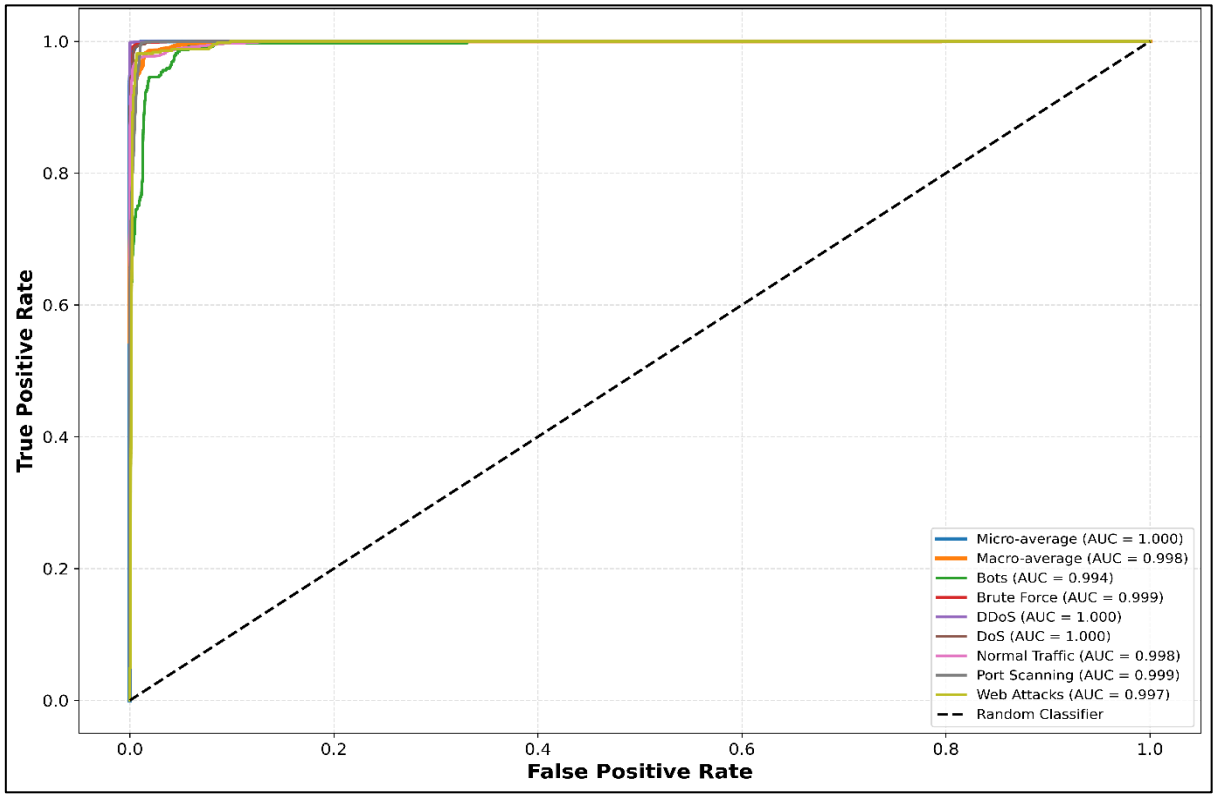


Fig 15 (d) Receiver Operating Characteristic (ROC) Curves of the Developed Decentralized Intrusion Detection Model (DIDM)

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The Receiver Operating Characteristic (ROC) curve was used to evaluate the discrimination capability of the developed Decentralized Intrusion Detection Model (DIDM) across the seven network traffic classes. Unlike the confusion matrix, which presents classification outcomes at a particular decision threshold, ROC analysis examines the relationship between the true positive rate and false positive rate across different classification thresholds. The Area Under the ROC Curve (AUC) was used to assess the model's ability to distinguish each traffic class from the remaining classes. AUC values closer to 1.0 indicate stronger discrimination capability. The ROC curves are presented in Figure 4.3(d), while the corresponding AUC and Average Precision (AP) values are summarised in Table 6.

Table 6 Receiver Operating Characteristic (ROC) Performance of the Developed DIDM

Class	AUC	Average Precision (AP)
Bots	0.9937	0.5688
Brute Force	0.9995	0.9899
DDoS	0.99997	0.9997
DoS	0.9997	0.9966
Normal Traffic	0.9984	0.9997
Port Scanning	0.9986	0.9628
Web Attacks	0.9972	0.2528
Micro Average	**0.9998**	—
Macro Average	**0.9981**	**0.8243**

Source: Research Data (2026)

Table 6 and Figure 15 (d) show that the developed DIDM achieved high ROC-AUC values across all seven network traffic classes. The model recorded a macro-average AUC of 0.9981 and a micro-average AUC of 0.9998. These results indicate that the model demonstrated a high ability to distinguish between the evaluated traffic categories across a range of classification thresholds.

Among the individual traffic classes, DDoS recorded the highest AUC of approximately 0.99997, followed by DoS at 0.9997 and Brute Force at 0.9995. Port Scanning and Normal Traffic also achieved high AUC values of 0.9986 and 0.9984, respectively. These results indicate strong discrimination capability for these categories. Bots and Web Attacks also recorded high AUC values of 0.9937 and 0.9972, respectively.

However, the Average Precision results provide an important qualification to the ROC findings. Although Bots and Web Attacks achieved high AUC values, their Average Precision values were considerably lower, at 0.5688 and 0.2528, respectively. In comparison, Brute Force, DDoS, DoS, Normal Traffic, and Port Scanning recorded higher Average Precision values ranging from 0.9628 to 0.9997. This difference demonstrates that strong ROC discrimination does not necessarily translate into equally strong precision–recall performance, particularly for underrepresented classes in an imbalanced dataset.

The comparatively lower Average Precision values for Bots and Web Attacks indicate that maintaining high precision while increasing recall for these categories remained challenging. The substantial class imbalance may contribute to this behaviour because the number of positive observations available for these attack categories was considerably smaller than that of majority classes. Therefore, the ROC-AUC and Average Precision measures should be interpreted together to obtain a more complete understanding of the model's classification behaviour.

The ROC findings are also consistent with the confusion matrix analysis presented in Figures 4.3(b) and 4.3(c), where some Bots and Web Attacks observations were incorrectly classified. The ROC analysis evaluates discrimination across different thresholds, whereas the confusion matrix reflects the classification

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

outcomes at the selected operating threshold. Consequently, high AUC values can coexist with classification errors at the threshold used to generate the confusion matrix.

Overall, the ROC analysis demonstrates that the developed DIDM achieved high discrimination capability across the seven network traffic classes. Nevertheless, the lower Average Precision values for Bots and Web Attacks highlight the continuing challenge of detecting minority attack categories under imbalanced data conditions. These findings complement the comparative performance and confusion matrix analyses and provide additional context for the Precision–Recall evaluation presented in the following section.

- *Precision–Recall Curve Analysis of the Developed DIDM*

Precision–Recall (PR) curve analysis was used to examine the relationship between precision and recall for the developed Decentralized Intrusion Detection Model (DIDM) across the seven network traffic classes. Precision measures the proportion of observations predicted as belonging to a class that actually belong to that class, while recall measures the proportion of actual observations from that class that are correctly identified. Unlike ROC analysis, which examines the relationship between the true positive rate and false positive rate, PR analysis provides a direct assessment of the trade-off between precision and recall. This is particularly useful when evaluating minority attack categories in an imbalanced dataset. The PR curves are presented in Figure 4.3(e), while the corresponding Average Precision (AP) values are reported in Table 4.14.

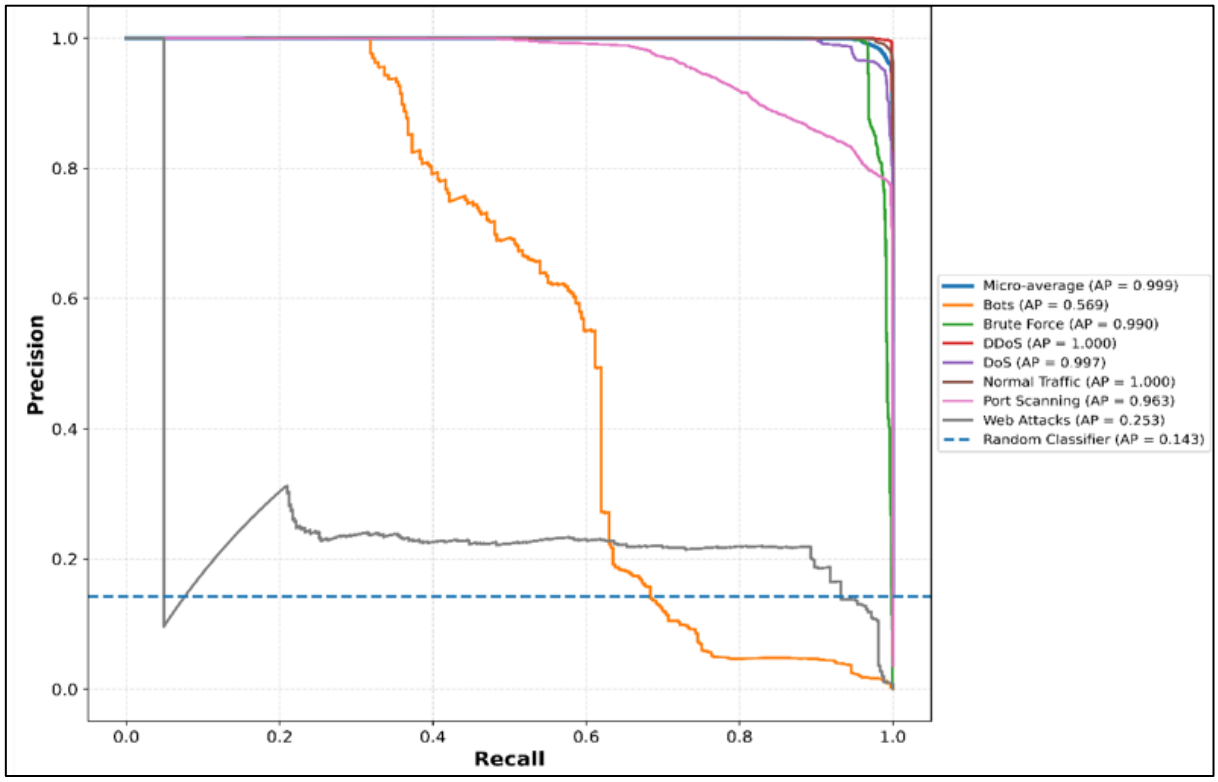


Fig 15 (e): Precision–Recall Curves for the Developed Decentralized Intrusion Detection Model (DIDM)

The results in Table 6 indicate that the developed DIDM achieved high Average Precision for most of the evaluated traffic classes. DDoS and Normal Traffic each recorded an AP of 0.9997, while DoS achieved 0.9966 and Brute Force recorded 0.9899. Port Scanning also achieved a relatively high AP of 0.9628. These results indicate that the model maintained strong precision–recall performance for these categories across the evaluated classification thresholds.

In contrast, the Bots category recorded an AP of 0.5688, while Web Attacks recorded the lowest AP of 0.2528. These lower values indicate greater difficulty in maintaining high precision as recall increases for

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

these classes. The results are consistent with the confusion matrix analysis in Figures 4.3(b) and 4.3(c), which showed that a substantial proportion of Bot and Web Attack observations were not correctly classified by DIDM at the evaluated operating threshold.

The differences between the classes highlight the importance of considering class-specific results when evaluating multiclass intrusion detection systems. Although the overall and majority-class results were strong, they do not establish equally strong detection performance for every attack category. The comparatively low AP values for Bots and Web Attacks indicate that these classes require particular attention in future model development and evaluation.

The class imbalance in the CICIDS2017 evaluation dataset is relevant when interpreting these results. Average Precision reflects both the ordering of predicted class scores and the precision–recall relationship across thresholds. For a rare class, a relatively small number of false-positive predictions can reduce precision substantially. Consequently, the lower AP values for Bots and Web Attacks should be considered alongside their class frequencies, confusion matrix results, and the corresponding ROC-AUC values. The ROC analysis reported high AUC values for both classes, illustrating that strong ranking discrimination does not necessarily result in equally strong precision–recall performance.

The baseline for a precision–recall curve depends on the prevalence of the positive class in the evaluation dataset. Therefore, the baseline should be interpreted separately for each one-versus-rest class rather than assumed to be the same for all seven classes. Any baseline displayed in Figure 4.3(e) should correspond to the relevant class prevalence or be clearly identified as a micro- or macro-averaged reference, as appropriate to the plotted curve.

Overall, the Precision–Recall analysis shows that the developed DIDM achieved high Average Precision for most of the evaluated network traffic classes, while Bots and Web Attacks remained comparatively challenging. These findings complement the comparative performance, confusion matrix, and ROC analyses by showing that the model's strong overall performance did not translate into equally high precision–recall performance for every minority attack category. The results identify a specific area for further improvement without diminishing the importance of the model's demonstrated performance on the other evaluated classes.

- *Statistical Validation of CIDM and DIDM Performance*

An exact McNemar test was performed to assess whether the difference in classification correctness between the Centralized Intrusion Detection Model (CIDM) and the developed Decentralized Intrusion Detection Model (DIDM) was statistically significant. The test was appropriate because both models were evaluated on the same independent test dataset comprising 504,151 records, producing paired predictions for the same observations. McNemar's test evaluates the discordant paired outcomes, namely observations correctly classified by one model but incorrectly classified by the other. The statistical results are summarised in Table 7.

Table 7 Statistical Validation of DIDM Performance Compared with CIDM

Performance measure	**DIDM (%)**	**CIDM (%)**
Accuracy	98.27	96.74
Weighted precision	98.32	98.36
Weighted recall	98.27	96.74
Weighted F1-score	98.25	97.39
McNemar test component	**Result**	
Independent test records	504,151	
DIDM correct, CIDM incorrect (b)	12,092	
DIDM incorrect, CIDM correct (c)	4,370	

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Total discordant predictions	16,462
Exact two-sided p-value	$p < 0.001$
Significance level	$\alpha = 0.05$
Statistical decision	Reject the null hypothesis

Note. The exact two-sided McNemar test was statistically significant at the 0.05 significance level. Source: Study experimental results (2026).

Source: Study experimental results.

Table 7 shows that DIDM achieved an accuracy of 98.27%, compared with 96.74% for CIDM, representing an improvement of approximately 1.53 percentage points. DIDM also achieved higher weighted recall and weighted F1-score. However, CIDM recorded marginally higher weighted precision, at 98.36%, compared with 98.32% for DIDM. These results are consistent with the comparative performance analysis presented in Section 4.6.

The paired prediction results provide further evidence of the difference between the two models. Of the 504,151 test observations, DIDM correctly classified 12,092 observations that CIDM misclassified. Conversely, CIDM correctly classified 4,370 observations that DIDM misclassified. The total number of discordant predictions was therefore 16,462, with the larger number of discordant outcomes favouring DIDM.

The exact two-sided McNemar test yielded a p-value below 0.001, which was lower than the predefined significance level of 0.05. Consequently, the null hypothesis of equal marginal classification correctness was rejected. The result provides statistically significant evidence that the models differed in their paired classification correctness, with the observed direction of the difference favouring DIDM.

Statistical significance should nevertheless be interpreted alongside the other evaluation findings. The McNemar test assesses differences in paired classification correctness; it does not establish that DIDM performed better on every individual traffic class, nor does it independently demonstrate that the federated architecture caused the observed improvement. As shown by the comparative metrics and confusion matrices, DIDM achieved higher overall accuracy but lower macro recall, indicating that some attack categories remained more difficult to identify.

Taken together, the comparative metrics, confusion matrices, ROC analysis, Precision–Recall analysis, and exact McNemar test provide complementary evidence concerning the performance of DIDM relative to CIDM. Under the experimental conditions of this study, DIDM achieved higher overall accuracy and a statistically significant difference in paired classification correctness. However, the lower macro recall and reduced Average Precision for Bots and Web Attacks indicate that further improvement is required to achieve more consistent detection across all traffic categories.

➢ *Model Implementation and Runtime Results*

- *Real-Time Operation of the DIDM/DIDS Model*

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017



Fig 16 Overall Operational View of the DIDM/DIDS SOC Model

Figure 16 presents the overall operational view of the Decentralized Intrusion Detection Model (DIDM) within the simulated Security Operations Center (SOC) environment. It illustrates the intended interaction among the intrusion detection component, Federated Learning (FL), Security Information and Event Management (SIEM), Security Orchestration, Automation, and Response (SOAR), and the Streamlit-based SOC dashboard. The figure provides an overview of how detection outputs and related operational records are represented within the cybersecurity monitoring framework.

The intrusion detection component forms the primary analytical layer of the framework. It processes network-related input data using the configured detection model and generates classification outputs that can be represented as security events. Depending on the information recorded by the implementation, these events may include prediction outcomes, confidence scores, severity classifications, timestamps, and event identifiers. The resulting records provide information for subsequent event management and operational monitoring.

The Federated Learning component supports collaborative model training among participating clients. In the experimental configuration, five simulated clients participated in local model training, and model updates were combined using Federated Averaging (FedAvg) over five communication rounds. The process exchanged model parameters rather than raw training observations. Consequently, raw training observations remained at the respective clients during the federated training process. However, this arrangement alone does not guarantee protection against information leakage because formal privacy mechanisms, such as differential privacy or secure aggregation, were not established as part of the reported experiment.

The SIEM component represents the event-management layer of the SOC framework. Its intended role is to receive and organize security-event records and, where implemented, apply processing or correlation rules to maintain incident-related information. The resulting records can support security monitoring and subsequent response workflows. Claims about successful event correlation, duplicate-alert reduction, or

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

improved threat-identification accuracy should be limited to functions demonstrated by the implementation and supported by corresponding records or measurements.

The SOAR component represents the response-management layer of the framework. It is intended to process incident information and associate incidents with configured response procedures. Depending on the functions implemented, the associated records may include response actions, execution status, and incident-related information. A distinction must be maintained between a response action that was proposed or logged and one that was actually executed and verified. Therefore, claims concerning host isolation, malicious-traffic blocking, threat containment, or incident closure require supporting implementation evidence.

The Streamlit-based SOC dashboard provides an interface for viewing selected information generated by the operational environment. The displayed information may include security events, incident records, response activity, component status, risk-related visualizations, and performance indicators. The configured two-second dashboard refresh interval supports frequent updating of the displayed information. However, the refresh interval does not establish that every detection, correlation, or response operation is completed within two seconds.

Taken together, Figure 16 illustrates the architectural relationship among intrusion detection, collaborative model training, event management, response orchestration, and operational visualization. It provides a representation of the components and information flows associated with the DIDM framework in the simulated SOC environment. The figure should not, by itself, be interpreted as proof of production-scale deployment, guaranteed privacy, continuous online model adaptation, or fully autonomous threat mitigation.

The formal performance comparison between DIDM and the Centralized Intrusion Detection Model (CIDM) is reported separately in the experimental evaluation, including Section 4.6.4, which presents the statistical validation of their paired classification results. The SOC demonstration complements that evaluation by illustrating how selected detection outputs and associated operational records can be presented and managed within a monitoring interface.

Overall, Figure 16 provides an operational overview of the DIDM framework and its relationship with the simulated SOC components. Its interpretation is limited to the architecture and functionality supported by the implementation evidence. The formal comparative results, rather than the architecture figure alone, provide the basis for assessing the classification performance of DIDM relative to CIDM.

➢ *Model Initialization and Startup Process*



Fig 17 Initialization and Startup Logs of the DIDM SOC Model

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 17 presents the startup and initialization information recorded during the execution of the Decentralized Intrusion Detection Model (DIDM). The figure provides evidence of the initialization activities shown in the available logs and helps explain how the experimental environment was prepared before the subsequent operational activities. The interpretation of the figure is limited to the initialization messages and component statuses displayed in the recorded output.

The initialization process begins with the launch of the DIDM application and the establishment of its execution environment. During this stage, the application loads the configuration and resources required for the functions enabled in the experimental setup. The startup output can be used to identify whether the application began execution and whether any initialization errors or warnings were reported.

The detection component is responsible for processing the input data supplied to the configured model and producing classification outputs. Its readiness should be assessed using the relevant startup messages, model-loading records, or successful execution results. Where these records are available, they provide evidence that the detection component was prepared for the subsequent classification tasks. The startup figure alone should not be taken as proof that live network traffic was being captured unless the corresponding packet-capture activity is explicitly documented.

The Federated Learning component supports the collaborative training configuration used in the study. The formal experiment involved five simulated clients and five communication rounds using Federated Averaging (FedAvg). The initialization output should be interpreted as evidence of federated-learning readiness only where the logs explicitly identify the relevant components or initialization steps. The completed training results are reported separately in the federated-learning evaluation section; they should not be inferred solely from an application startup message.

The SIEM component is responsible for receiving and managing security-event records within the SOC framework. Where the startup logs show that the relevant service or event-processing component was initialized, this supports the conclusion that the corresponding functionality was made available in the experimental environment. However, successful initialization does not, by itself, demonstrate that event correlation or incident generation subsequently operated correctly. Those functions require supporting execution records or outputs.

Similarly, the SOAR component is associated with the management of configured response procedures. If the startup output confirms that the response component or its configuration was loaded, this can be reported as an initialization result. It does not establish that a response playbook was successfully executed or that a real mitigation action, such as blocking malicious traffic or isolating a host, occurred. Such outcomes must be supported by separate execution logs or verification evidence.

The Streamlit-based SOC dashboard provides the interface for displaying selected operational information. Its availability can be established from the relevant application startup message or successful access to the dashboard. Any claim that the dashboard was synchronized with all event repositories or that all services were operational should be made only where the available records substantiate that condition.

In summary, Figure 17 documents the startup and initialization activities represented in the DIDM SOC execution output. It helps establish the initial state of the experimental environment before interpreting subsequent operational results. The figure should be used to report only the component statuses and activities directly supported by the displayed evidence, while successful detection, federated training, event correlation, and response execution should be evaluated using their respective results and records.

- *Real-Time Alert Generation*

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017



Fig 18 Real-Time Alert Output Generated by the DIDM SOC Model

Figure 18 presents the alert-generation output associated with the Decentralized Intrusion Detection Model (DIDM) within the simulated Security Operations Center (SOC) environment. The figure illustrates how detection results can be represented as security events for monitoring and subsequent analysis. The interpretation of the figure is based on the outputs displayed in the experimental implementation.

The alert-generation process begins with the provision of input data to the configured intrusion detection component. In the formal experimental evaluation, the DIDM was assessed using prepared CICIDS2017 records containing network-traffic features. These features were processed by the trained artificial neural network (ANN) model to produce classification predictions. This experimental procedure should be distinguished from live packet capture, which requires a separately implemented and verified network acquisition mechanism.

Following input processing, the detection model generates a prediction for each evaluated observation. The prediction identifies the class assigned by the model, according to the classification categories used in the experiment. Where the implementation exposes prediction probabilities or confidence values, these may also be included in the displayed results. Such values represent the model's output and should not automatically be interpreted as calibrated probabilities of a security threat.

Where supported by the implementation, the prediction outputs can be converted into structured security-event records. These records may contain the predicted class, event identifier, timestamp, severity label, or other fields configured by the application. The availability of each field depends on the actual event schema and should be confirmed against the displayed output or saved event records.

The generated records can then be presented through the Streamlit-based SOC dashboard and, where the relevant integration is implemented, passed to the Security Information and Event Management (SIEM) component for further processing. Events identified for additional review may be represented in the dashboard according to their predicted class or configured severity. Any subsequent incident creation or response action should be described separately and supported by the corresponding SIEM or SOAR execution records.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The dashboard supports operational observation of the detection outputs by displaying the information made available by the application. Frequent dashboard refreshing can help users view newly recorded information, but the refresh interval alone does not establish the end-to-end latency of the detection process. Similarly, the presence of an alert does not independently demonstrate that the predicted activity was correctly classified; classification performance must be assessed against the relevant reference labels using the formal evaluation metrics.

Overall, Figure 18 illustrates the relationship between model predictions and their representation as security alerts within the DIDM SOC demonstration. The formal evaluation reported in this chapter provides the evidence for assessing classification performance, while the alert-generation output demonstrates how selected prediction results can be presented for operational monitoring. Claims of live packet capture, verified attack identification, or successful automated mitigation should be included only where they are supported by the implementation and corresponding execution evidence.

- *SIEM Event Correlation*

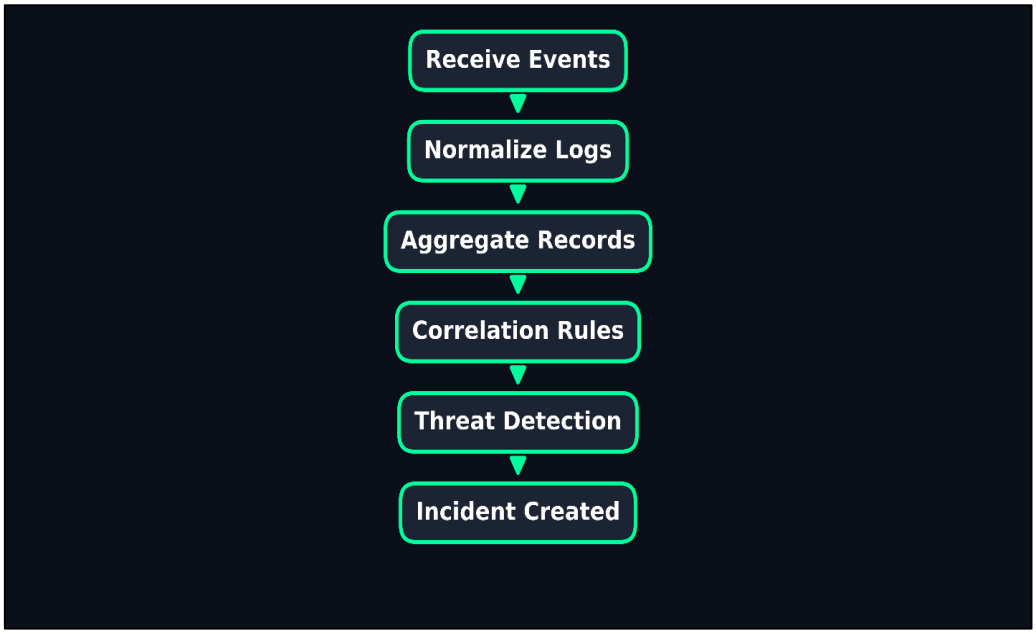


Fig 19 SIEM Event Correlation Output of the DIDM SOC Model

Figure 19 illustrates the event-processing and correlation workflow associated with the Security Information and Event Management (SIEM) component of the Decentralized Intrusion Detection Model (DIDM). Within the simulated Security Operations Center (SOC) environment, the SIEM component represents the layer responsible for receiving security-event records and organizing relevant information for monitoring and incident management. The figure should be interpreted in accordance with the processing stages and outputs demonstrated by the implementation.

The workflow begins with the receipt of security-event records generated by the intrusion detection component. Depending on the implemented integration, these records may contain model predictions, event identifiers, timestamps, predicted classes, severity labels, or other configured fields. The received information provides the basis for subsequent event processing and enables detection outputs to be presented in a consistent operational context.

Where implemented, event normalization converts records into a common structure so that relevant fields can be processed consistently. The specific transformations depend on the event schema and the rules

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

configured in the application. Normalization should therefore be described as an implemented function only where the source code, processing output, or recorded events demonstrate that the transformation takes place.

The next stage may involve organizing related events and applying configured correlation rules. Such rules can examine selected attributes, including timestamps, event categories, or source identifiers, to identify relationships among records. However, the existence of a SIEM component or a workflow diagram does not establish that advanced temporal correlation, duplicate-event consolidation, or threat-intelligence matching was implemented. These functions should be reported only when the relevant rules and their execution are supported by implementation evidence.

Where the implementation creates incident records from processed events, those records can provide a structured input for subsequent investigation or response management. The incident information may be displayed in the SOC dashboard and passed to the Security Orchestration, Automation, and Response (SOAR) component if the relevant connection is implemented. The successful transfer and processing of incidents should be established using corresponding event records, logs, or application outputs rather than inferred from the architecture alone.

The SIEM component consequently provides an organizational link between detection outputs and operational incident management within the SOC framework. Its potential value lies in making security-event information easier to review and in supporting a structured workflow for handling events that require further attention. Nevertheless, claims that the component reduced alert volume, improved threat-identification accuracy, or accelerated incident response require defined measurements and comparative evidence.

Overall, Figure 19 presents the intended role of SIEM event processing and correlation within the DIDM SOC environment. The extent to which individual functions—such as normalization, rule-based correlation, event aggregation, and incident generation—were achieved should be determined from the actual implementation and its recorded outputs. The SIEM workflow complements the formal CIDM–DIDM classification evaluation but does not independently establish the statistical performance of the intrusion detection model.

- *SOAR Automated Response Execution*

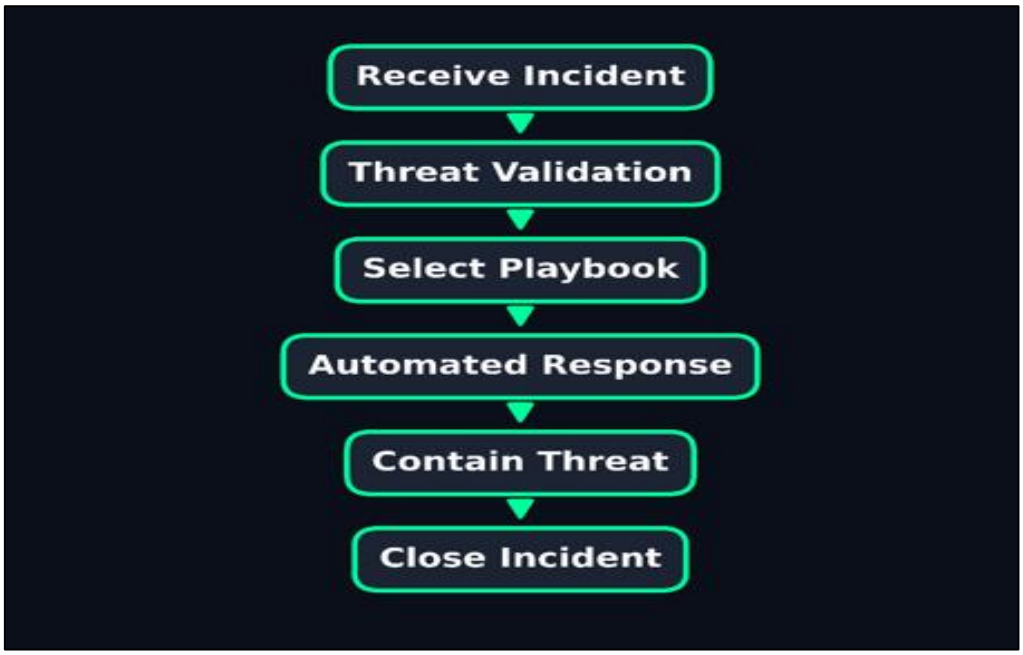


Fig 20 SOAR Automated Response Execution of the DIDM SOC Model

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 20 presents the response-management workflow associated with the Security Orchestration, Automation, and Response (SOAR) component of the Decentralized Intrusion Detection Model (DIDM). Within the simulated Security Operations Center (SOC) environment, SOAR represents the component intended to manage response activities following the identification and processing of security events. The figure illustrates the relationship between incident information, configured response procedures, and the recording of response-related activities.

The workflow begins when incident information is made available to the SOAR component following detection and, where implemented, SIEM processing. The incident information provides the context required to determine the nature and priority of the event. Relevant attributes may include the predicted attack category, severity classification, event identifier, and timestamp, depending on the information supplied by the preceding components.

Following incident reception, the response-management component evaluates the available incident information against the response procedures configured in the experimental environment. Where playbook-selection rules are implemented, the component can associate an incident with a corresponding response procedure based on defined conditions. The selected procedure provides a structured sequence of actions for handling the incident within the SOC demonstration.

The response stage involves processing the actions associated with the selected procedure. These actions may include recording an incident, generating a response notification, updating the incident status, or logging a proposed mitigation activity, depending on the functionality implemented. Such activities support the organization and monitoring of security-response operations.

Where the implementation records response actions, the resulting information can be used to review the activities associated with each incident. Response records may contain details such as the incident identifier, action description, execution status, and timestamp, where these fields are available. These records provide an operational account of the response activities represented in the SOC environment.

A distinction is necessary between simulated response activities and actual enforcement actions. For example, recording a proposed host-isolation action or displaying a traffic-blocking status does not establish that a host was disconnected or that malicious network traffic was blocked. Claims of successful containment, policy modification, host isolation, or incident closure therefore require corresponding execution logs and independent verification of the relevant system or network state.

The Streamlit-based SOC dashboard provides a means of displaying selected SOAR-related information alongside security events and incident records. This enables the operational activities represented in the demonstration to be reviewed through a common interface. The dashboard serves as a monitoring and visualization layer; its display of a response action should not be treated as independent verification that the action was enforced.

Overall, Figure 20 illustrates the role of SOAR in organizing and managing incident-response activities within the DIDM SOC framework. The figure demonstrates the intended relationship between incident handling, configured response procedures, and response records. The extent of automation achieved should be determined from the functions implemented and the available execution evidence, while claims of actual network mitigation should be limited to actions whose execution and outcomes were verified.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *Model Structural Organization*

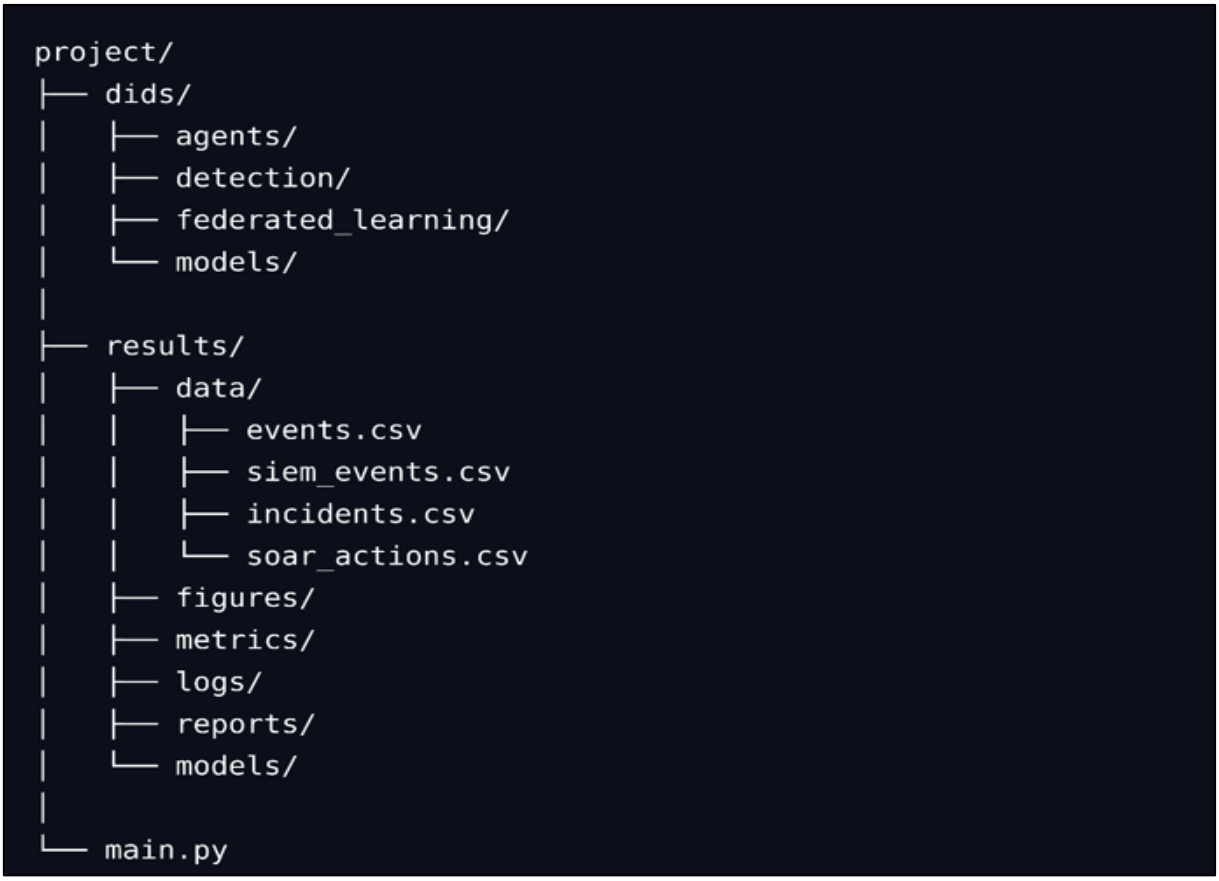


Fig 21 Structural Organization of the DIDM SOC Model

✓ *Figure 21: Project Directory Structure*

Figure 21 presents the project directory structure used to organize the implementation of the Decentralized Intrusion Detection Model (DIDM) and its associated Security Operations Center (SOC) demonstration environment. The structure illustrates how the application files, model components, data resources, operational outputs, and supporting materials are arranged within the project. This organization provides a basis for maintaining the implementation and locating the resources required during development and experimental execution.

The main application directory contains the files responsible for coordinating the execution of the DIDM SOC environment. The main execution script, identified as main.py in the project structure, serves as the application entry point where the relevant modules and operational functions are brought together. Its role is to initiate the application and coordinate the components configured for execution.

The agents/ and detection/ directories organize files associated with the detection functionality. These directories provide a logical separation between agent-related implementation resources and the model or detection-processing components. Their contents support the organization of the intrusion detection functionality and its associated processing operations.

The federated learning/ and models/ directories contain resources associated with model development and federated-learning operations, as represented in the project structure. The federated-learning directory organizes the implementation files related to collaborative training, while the models directory provides a

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

location for model-related resources. This arrangement reflects the separation between the model implementation and the processes used to coordinate federated training.

The data/ directory provides an organized location for the data resources used by the application, subject to the dataset and file arrangements shown in the project. The results/ directory organizes outputs generated during execution and evaluation. Depending on the files displayed in Figure 4.19, these outputs may include prediction results, performance records, or other experimental artifacts.

Additional directories, including figures/, metrics/, logs/, and reports/, provide locations for visual outputs, evaluation measures, execution records, and reporting materials, respectively. This separation supports the systematic organization of the evidence generated during model development and operational demonstration.

The project structure also identifies CSV files associated with event and response management, including events.csv, siem_events.csv, incidents.csv, and soar_actions.csv, where these files are present in the implementation. These records provide an organized means of storing or reviewing the corresponding operational information. Their presence in the directory structure should not, on its own, be interpreted as proof that every associated processing or response function operated successfully.

Overall, Figure 21 illustrates the logical organization of the DIDM SOC implementation resources. The separation of application modules, detection components, federated-learning resources, data, model outputs, logs, and reports supports project maintainability and facilitates the identification of files associated with different implementation activities. The figure therefore documents the structural organization of the experimental implementation rather than independently demonstrating the functional performance of its individual components.

- *End-to-End Execution Flow*



Fig 22 End-to-End Execution Flow of the DIDM SOC Model

Figure 22 illustrates the overall execution flow of the Decentralized Intrusion Detection Model (DIDM) within the simulated Security Operations Center (SOC) environment. The workflow presents the relationship

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

between the intrusion detection component, Federated Learning (FL), Security Information and Event Management (SIEM), Security Orchestration, Automation, and Response (SOAR), and the operational dashboard. It provides an overview of how the model's detection outputs can be processed and presented within the wider cybersecurity monitoring framework.

The operational detection process begins with the provision of network-related input data to the configured intrusion detection component. The input data are processed using the feature representation and preprocessing procedures established for the model. The trained artificial neural network (ANN) then analyses the input observations and generates classification predictions according to the learned traffic categories.

The resulting predictions can be represented as security events containing the information made available by the implementation. These records may include predicted classes, event identifiers, timestamps, confidence values, and severity labels, depending on the configured event structure. The generated event information provides the basis for subsequent monitoring and, where implemented, SIEM processing.

The SIEM component receives the relevant security-event records and organizes them for event management. Where processing and correlation rules have been implemented, the component can examine selected event attributes and produce incident-related records. These records provide structured information for further review and can be made available to the SOAR component where the relevant integration is supported.

The SOAR component manages response-related activities associated with the incidents it receives. Depending on the configured functionality, these activities may involve response-procedure selection, incident-status updates, notifications, and recording of response actions. Any claim that the workflow successfully blocked malicious traffic, isolated hosts, or contained an attack must be supported by evidence that the relevant action was executed and its outcome verified.

The Streamlit-based SOC dashboard provides an interface for viewing selected information generated throughout the operational workflow. It can display detection events, incident records, response-related information, and selected performance indicators. The dashboard supports operational monitoring by presenting the available information in a unified interface.

Federated Learning forms a complementary model-training workflow within the DIDM architecture. In the formal experiment, five simulated clients participated in local ANN training, and their model updates were combined through Federated Averaging (FedAvg) over five communication rounds. The resulting global model supports the classification task. This collaborative training process is distinct from the processing of an individual detection event: model training and aggregation do not necessarily occur each time the detection component generates an alert.

The relationship between the two workflows is therefore that federated training produces or updates the model used by the detection component, while the operational workflow uses the configured model to process input observations and manage the resulting security events. This distinction clarifies the contribution of decentralized learning without implying that every detection event triggers a new federated aggregation round.

Overall, Figure 22 presents the relationship between model training, intrusion detection, event management, response coordination, and operational visualization within the DIDM SOC framework. The figure provides an integrated view of the experimental architecture, while the actual functionality and performance of each stage should be interpreted using the corresponding training results, classification metrics, event records, and response-execution evidence reported in this chapter.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *Federated Learning Workflow*

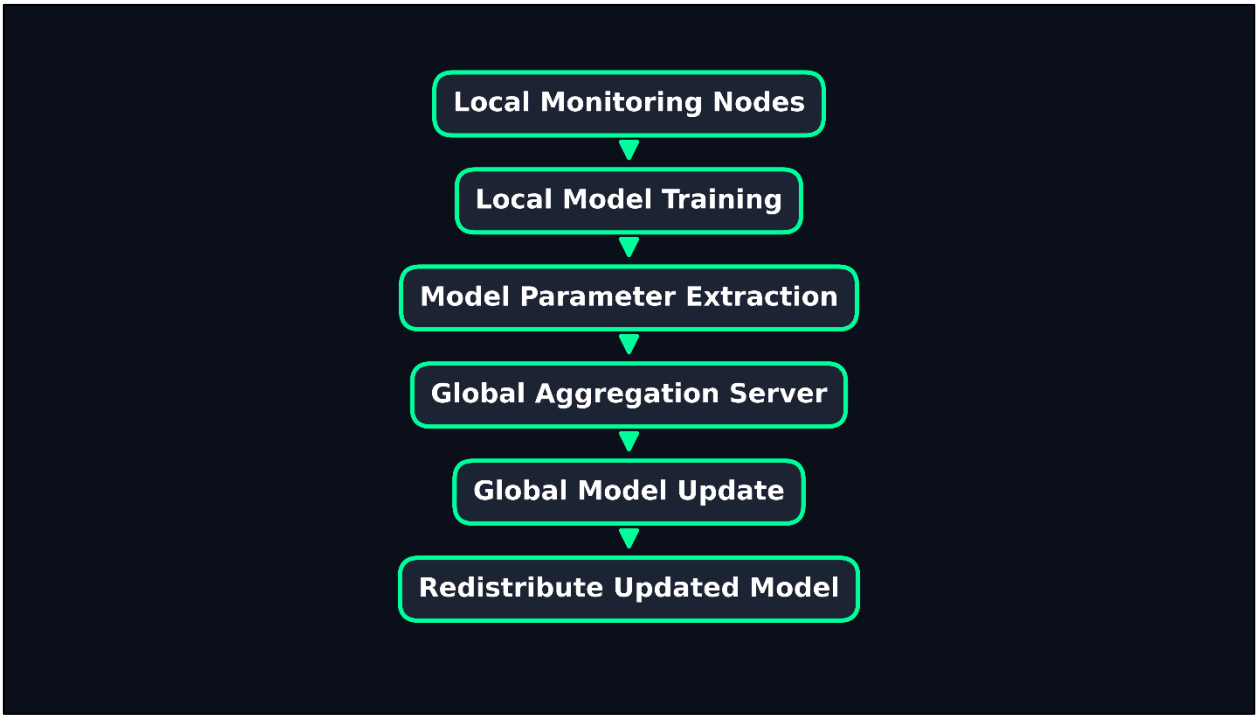


Fig 23 Federated Learning Work Flow of the DIDM SOC Model

Figure 23 illustrates the Federated Learning (FL) workflow implemented within the Decentralized Intrusion Detection Model (DIDM). The workflow demonstrates how participating clients perform local model training and contribute model updates to a coordinating server for aggregation. The experimental configuration used five simulated clients and five communication rounds, with Federated Averaging (FedAvg) applied to combine the participating clients' model updates.

The process begins with the allocation of training data to the participating clients. Each client uses its locally available training observations to train its artificial neural network (ANN) model. Local training enables the participating clients to contribute to the collaborative learning process without transferring their raw training observations to the coordinating server.

Following local training, each client generates an updated set of model parameters. These parameters represent the changes resulting from the local training process and are transmitted to the coordinating server for aggregation. The federated-learning procedure therefore exchanges model updates rather than the underlying raw network-traffic observations.

At the coordinating server, the received client updates are combined using the Federated Averaging algorithm. FedAvg calculates an aggregated global model from the participating clients' model parameters, taking account of the configured aggregation procedure. The resulting global model incorporates information learned across the participating clients.

After aggregation, the updated global model is made available to the participating clients for the next communication round. Each client can then continue local training using its allocated data, generate updated model parameters, and submit them for another aggregation cycle. This process is repeated according to the configured number of communication rounds.

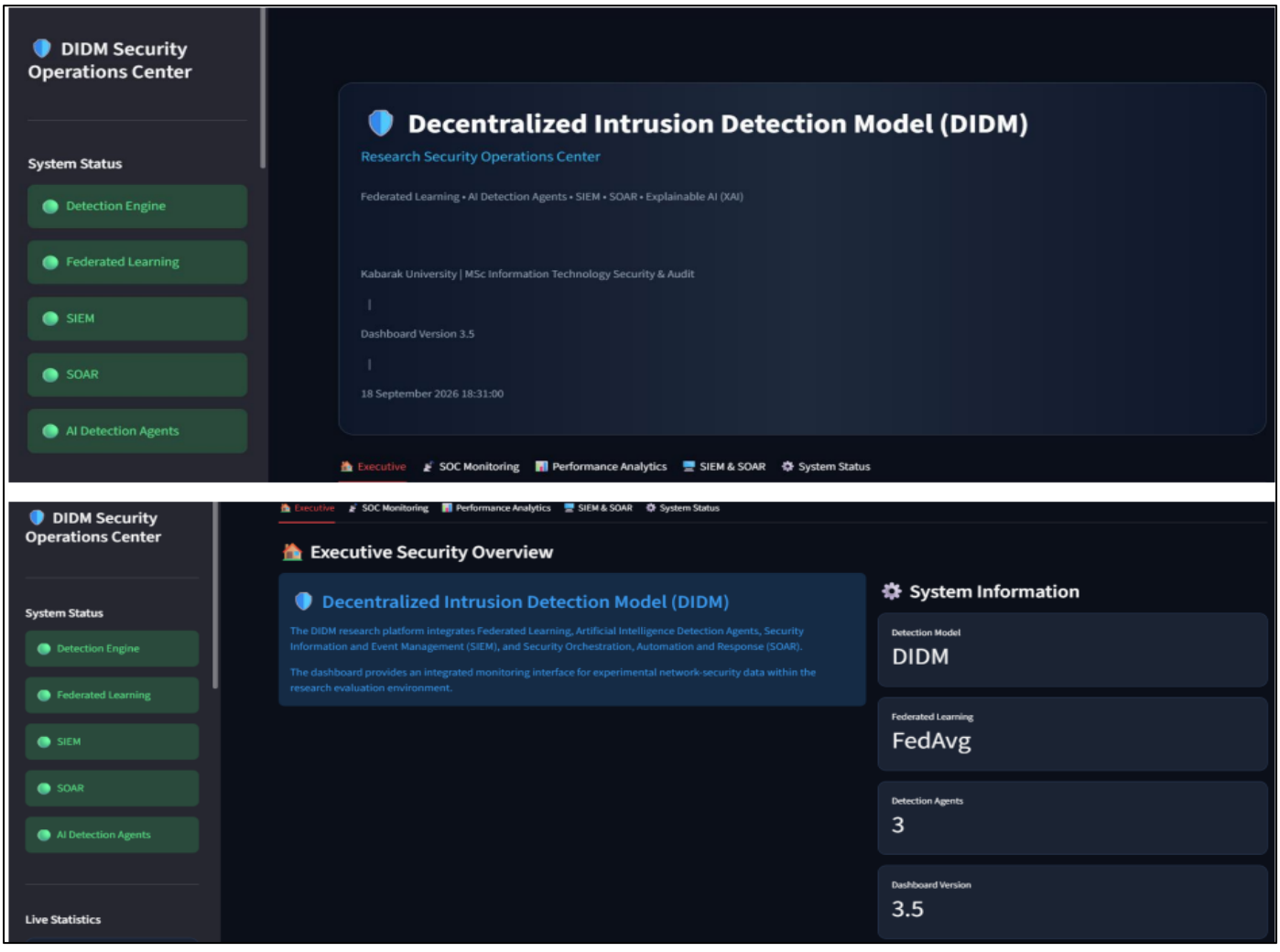
ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

In the experiment reported in this study, the federated-learning procedure was conducted over five communication rounds involving five simulated clients. The global model's training accuracy and loss were recorded across the rounds to monitor the training process. The results showed an overall improvement in global model accuracy from the first to the fifth round, although a small accuracy decrease occurred between the third and fourth rounds. The training loss declined across all five rounds, indicating a reduction in the model's training error during the experiment. These results are presented in the federated-learning evaluation earlier in this chapter.

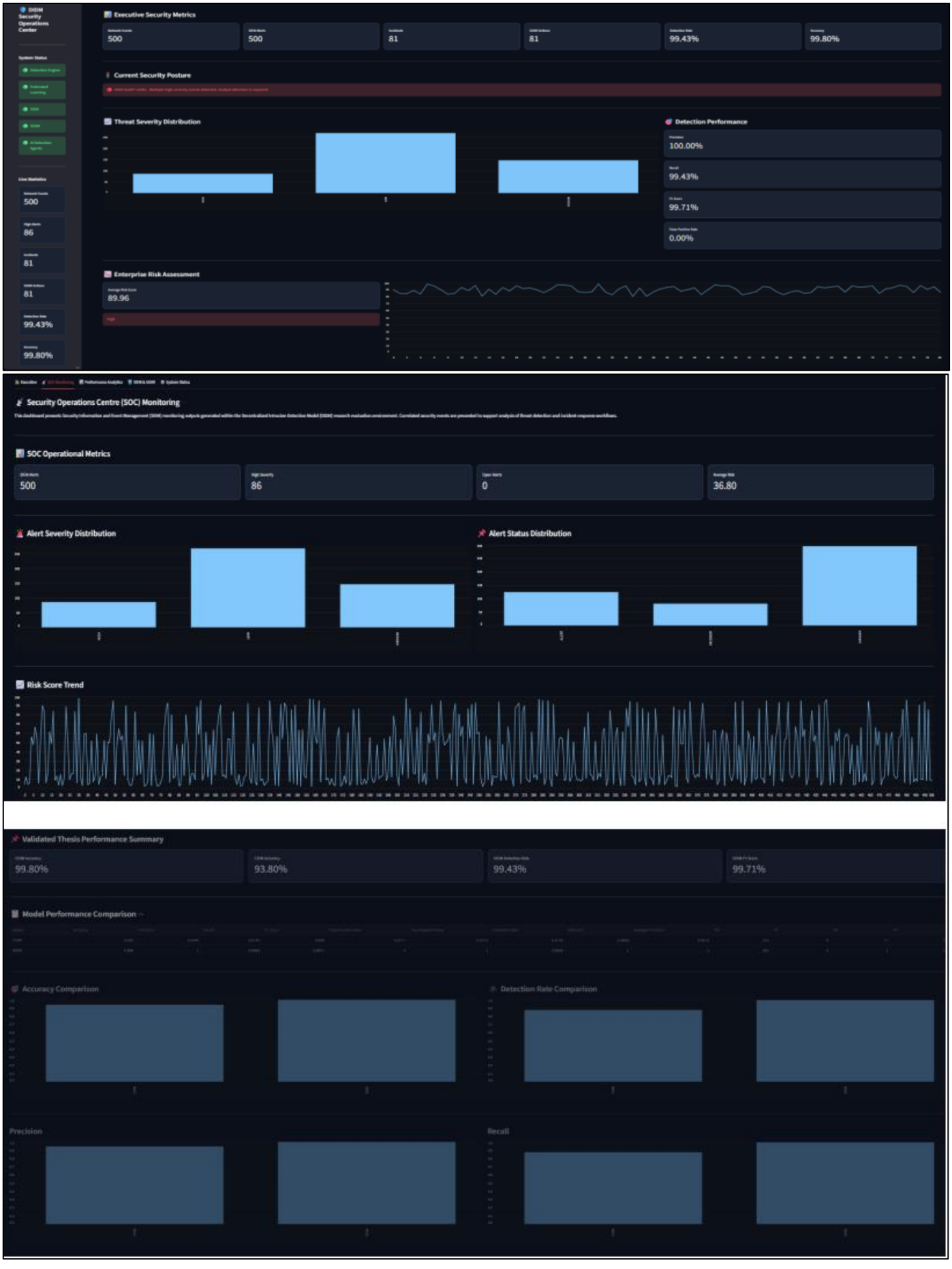
The use of federated learning allowed the participating clients to contribute to a shared model without exchanging their raw training observations. Nevertheless, retaining raw data locally should not be interpreted as a guarantee of complete privacy or protection against information leakage. The experiment did not establish additional formal privacy mechanisms such as differential privacy or secure aggregation. Accordingly, the privacy implications of the approach should be understood within the limits of the implemented configuration.

Overall, Figure 23 presents the collaborative training procedure used to develop the DIDM global model. The five-client, five-round FedAvg experiment demonstrates the implementation of decentralized model training through local parameter updates and server-side aggregation. The resulting global model provides the trained detection component used in the formal evaluation of DIDM against the Centralized Intrusion Detection Model (CIDM).

- *Real-Time Monitoring Dashboard*



ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017



ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

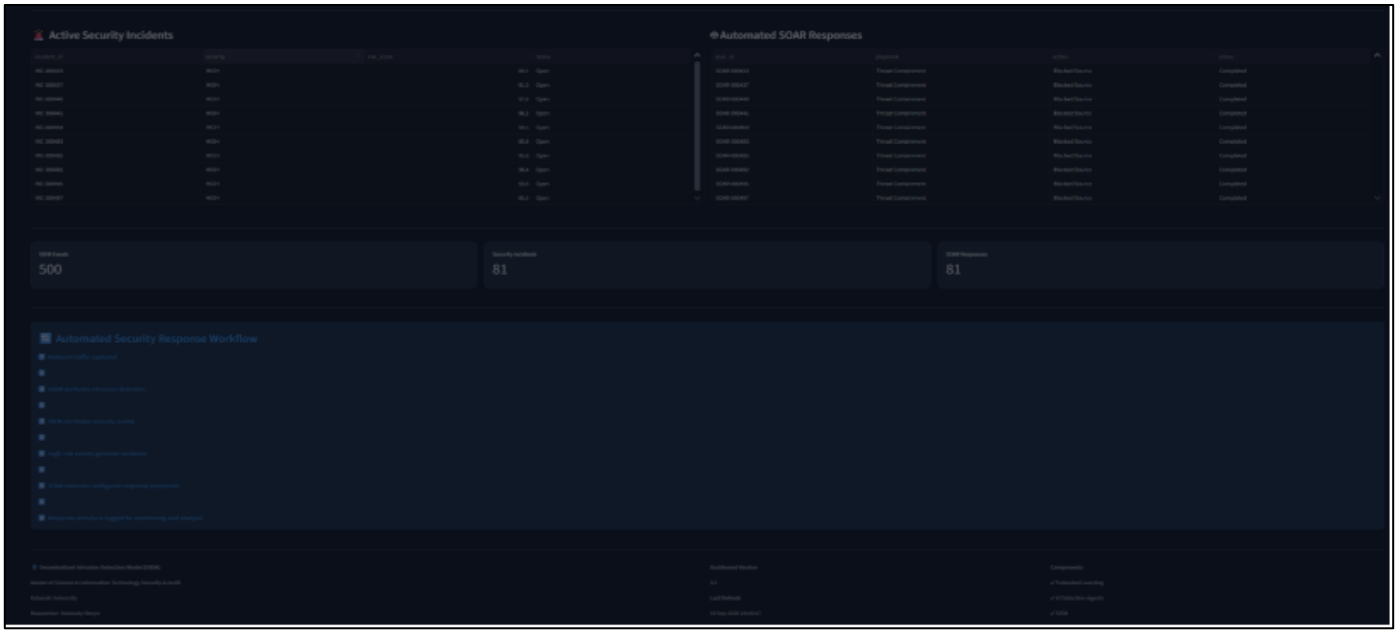


Fig 24 Real-Time Security Operations Center (SOC) Dashboard

✓ *Figure 24: Streamlit-Based SOC Monitoring Dashboard of the DIDM Model*

Figure 24 presents the Streamlit-based Security Operations Center (SOC) dashboard developed to support the operational monitoring and visualization of the Decentralized Intrusion Detection Model (DIDM). The dashboard provides a unified interface through which selected detection outputs, security-event records, incident information, response activities, and performance indicators can be viewed. Its purpose is to facilitate the observation of the experimental environment rather than to serve as the primary implementation of the formal intrusion detection algorithm.

The dashboard presents key operational indicators associated with the SOC environment. Depending on the information generated by the application, these indicators include the number of recorded events, SIEM alerts, incident records, SOAR response activities, detection-related statistics, and model-performance summaries. These indicators provide an overview of the information available during the operational demonstration. Their interpretation depends on the underlying records and the observation period used to generate them.

The dashboard also provides views of individual security events and their associated attributes. Such information may include predicted attack categories, confidence values, severity classifications, timestamps, and event identifiers, where these fields are available in the event records. The event display allows users to inspect selected detection outputs and review the security information represented by the application.

Additional dashboard views present incident-related information and SOAR activity. These views support the examination of incident records and response-related entries generated or recorded by the operational environment. The display of a response entry should, however, be distinguished from independent verification that a corresponding network mitigation action was successfully enforced.

Visual summaries, including charts showing event distributions, severity categories, incident status, or other configured indicators, provide an overview of the recorded operational information. These visualizations support the interpretation of the data presented by the dashboard but should not be treated as independent measures of detection effectiveness unless their calculations and underlying evaluation data are documented.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The dashboard was configured with a two-second refresh interval to update the displayed information frequently. This setting supports near-real-time presentation of available records. However, the refresh interval represents the dashboard update frequency and does not establish that the underlying detection, event-processing, or response operations are completed within two seconds.

The operational interface also includes supplementary machine-learning mechanisms, namely the Autoencoder, Isolation Forest, and One-Class Support Vector Machine (One-Class SVM), where these components are displayed in the implementation. These mechanisms support the supplementary anomaly-monitoring demonstration. They should be distinguished from the formal experimental comparison between DIDM and CIDM, which used the artificial neural network (ANN) model and Federated Averaging (FedAvg) for the decentralized configuration. The supplementary anomaly-detection mechanisms are therefore not treated as additional models in the reported formal CIDM–DIDM benchmark.

Performance information displayed on the dashboard should also be interpreted separately from the formal evaluation results presented earlier in this chapter. The dashboard may display operational statistics generated from its own records, observation periods, or selected data subsets. These values are not automatically equivalent to the formal independent-test results obtained from the 504,151 CICIDS2017 test observations. The formal benchmark remains the authoritative basis for comparing the classification performance of DIDM and CIDM.

Overall, Figure 24 demonstrates the use of a Streamlit-based dashboard to support operational monitoring and visualization within the DIDM SOC environment. It brings together selected event information, incident records, response-related activities, and supplementary model indicators in a common interface. The dashboard complements the formal experimental evaluation by illustrating how detection outputs can be presented for operational review, while the statistical performance of DIDM relative to CIDM is established through the controlled experimental results reported separately.

- *High-Level Architecture*

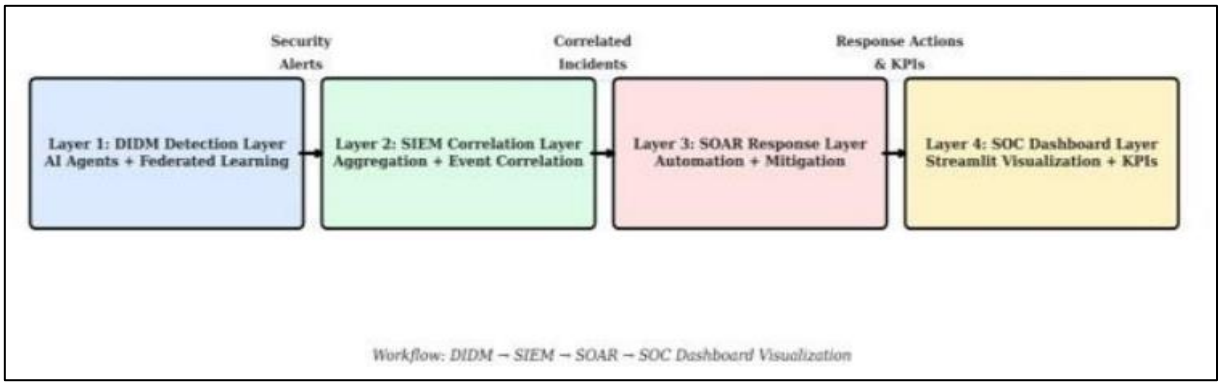


Fig 25 High-Level Architecture of the DIDM SOC Model

Figure 25 presents the high-level architecture of the Decentralized Intrusion Detection Model (DIDM) within the simulated Security Operations Center (SOC) environment. The architecture illustrates the major functional components of the framework and their relationships, comprising the intrusion detection layer, Federated Learning (FL), Security Information and Event Management (SIEM), Security Orchestration, Automation, and Response (SOAR), and the operational monitoring dashboard.

The intrusion detection layer provides the analytical functionality of the framework. It processes network-related input observations using the configured detection model and generates classification outputs that can

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

be represented as security events. In the formal experiment, the decentralized intrusion detection model was implemented using an Artificial Neural Network (ANN) trained through a federated-learning process.

The Federated Learning layer facilitates collaborative model development among participating clients. The experimental configuration involved five simulated clients performing local model training, with server-side aggregation implemented using the Federated Averaging (FedAvg) algorithm. Through this arrangement, participating clients contributed model updates towards the development of a shared global model while retaining their raw training observations locally. The aggregated global model provided the trained detection component used in the formal evaluation. However, retaining raw observations locally should not be interpreted as a formal privacy guarantee, since differential privacy and secure aggregation were not implemented in the experiment.

The SIEM layer represents the event-management functionality of the architecture. It receives and organizes available security-event records from the detection component for monitoring and incident management. Where supported by the implemented functionality, event-processing and correlation mechanisms can further organize detection-related information into structured incident records for subsequent response activities.

The SOAR layer represents the response-management functionality of the framework. It handles incident-related information and associates it with configured response procedures. The extent of automation is determined by the response functions implemented and demonstrated within the experimental environment. Accordingly, actual host isolation, malicious-traffic blocking, or threat containment should only be considered achieved where corresponding execution and verification evidence is available.

The monitoring and visualization layer is represented by the Streamlit-based SOC dashboard. It provides an interface for displaying selected detection outputs, security-event records, incident information, response-related activities, and operational indicators. The dashboard supports observation and monitoring of the experimental framework but does not replace the underlying intrusion detection model, federated-training process, or event-management components.

Overall, Figure 25 illustrates the functional organization of the DIDM SOC framework and the relationship between decentralized model training, intrusion detection, security-event management, response coordination, and operational monitoring. The architecture provides a structural representation of the experimental system, while the effectiveness and performance of its implemented components are examined through the training results, classification metrics, application outputs, and execution records presented in the subsequent sections.

- *Intrusion Detection Workflow*



Fig 26 Intrusion Detection Workflow of the DIDM Model

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 26 illustrates the intrusion detection workflow of the Decentralized Intrusion Detection Model (DIDM). The workflow describes the main stages involved in preparing input observations, processing them through the trained detection model, generating classification outputs, and representing identified security events for subsequent monitoring and incident management.

The process begins with the provision of network-related input data derived from the CICIDS2017 dataset. The input observations undergo preprocessing to ensure that the required features are appropriately prepared for model evaluation. This includes applying the feature preparation and scaling procedures established during the model development stage.

Following preprocessing, the prepared observations are passed to the trained Artificial Neural Network (ANN) detection model. For DIDM, the ANN model is developed through the Federated Learning framework, in which five simulated clients participate in local training and contribute model updates to a coordinating server. The server aggregates the updates using Federated Averaging (FedAvg) to produce the global model.

The trained global model performs classification on the prepared input observations and generates predictions corresponding to the defined traffic categories. These categories include normal traffic and the identified attack classes represented in the CICIDS2017 dataset. The classification outputs provide the basis for distinguishing potentially malicious observations from normal network activity.

Where the implementation supports event generation, the classification results are converted into structured security-event records containing the available detection information. These records can subsequently be made available to the SIEM component for event organization, monitoring, and incident management.

The final stage involves presenting selected detection results through the Streamlit-based SOC dashboard. The dashboard provides a visual interface for examining available classification outputs, security events, and related operational indicators. It supports the interpretation and monitoring of the experimental system without replacing the underlying ANN classification process.

The workflow therefore demonstrates how the DIDM integrates data preprocessing, federated model development, ANN-based classification, security-event representation, and operational monitoring. It should be distinguished from continuous live packet inspection, since the formal performance evaluation reported in this study was conducted using prepared CICIDS2017 observations. The detection performance of the model is established through the formal evaluation results presented in the preceding sections.

- *SIEM Processing Workflow*



Fig 27 SIEM Processing Workflow of the DIDM Model

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 27 illustrates the Security Information and Event Management (SIEM) workflow within the Decentralized Intrusion Detection Model (DIDM) framework. The workflow describes how security-event information generated by the intrusion detection component is received, organized, processed, and made available for monitoring and incident management.

The process begins when the intrusion detection component generates classification outputs from the prepared input observations. Where supported by the implementation, relevant detection results are represented as structured security-event records and forwarded to the SIEM component. These records provide information about the detected activity and the corresponding classification results.

Upon receiving the available event records, the SIEM component organizes the information for centralized monitoring. This may involve processing the recorded event fields, maintaining event records, and presenting the available information in a format suitable for security monitoring and subsequent incident management. The processing performed is limited to the functions implemented within the experimental environment.

The SIEM component also provides an event-management interface between the intrusion detection functionality and the SOAR response-management component. Where the integration is implemented, relevant incident information can be made available to SOAR for handling through configured response procedures. This arrangement establishes a structured relationship between detection outputs, event management, and response coordination.

The processed event information can also be displayed through the Streamlit-based SOC dashboard. The dashboard enables users to examine available security events and related incident information, supporting operational observation and monitoring of the experimental framework.

Overall, the SIEM workflow demonstrates the role of event organization and monitoring within the DIDM SOC architecture. It provides a mechanism for managing detection-related information and supporting communication with the response-management component. However, the presence of the SIEM workflow alone does not establish improved detection accuracy, successful identification of complex attack patterns, or reduced incident response time. Such outcomes require separate experimental evidence.

- *SOAR Incident Response Workflow*



Fig 28 SOAR Incident Response Workflow of the DIDM Model

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 28 illustrates the Security Orchestration, Automation, and Response (SOAR) workflow within the Decentralized Intrusion Detection Model (DIDM) framework. The workflow describes how incident-related information from the security-event management component is handled through configured response procedures in the experimental Security Operations Center (SOC) environment.

The process begins when the SOAR component receives available incident information originating from the intrusion detection and SIEM components. The information provides the basis for identifying the recorded security event and determining the appropriate response procedure supported by the experimental implementation.

Following receipt of the incident information, the SOAR component processes the available incident details and associates them with configured response procedures. These procedures provide an organized approach to handling security incidents and managing the corresponding response-related activities.

Depending on the functions implemented, the response process may involve recording the selected action, updating the incident status, and generating response-related information for monitoring. Where automated actions are configured, the SOAR component can initiate the corresponding procedures within the experimental environment.

The resulting response information is maintained or presented through the available application components, including the Streamlit-based SOC dashboard. This enables the examination of incident records, response-related activities, and the status information available from the experimental system.

The SOAR workflow therefore demonstrates the role of response coordination and management within the DIDM SOC architecture. However, recording or displaying a response action does not, by itself, establish that an external security control was successfully enforced. Claims involving malicious-traffic blocking, host isolation, policy modification, or threat containment require corresponding execution logs and independent verification. The workflow is consequently interpreted in accordance with the response capabilities demonstrated by the implemented system.

- *Integrated End-to-End Workflow*

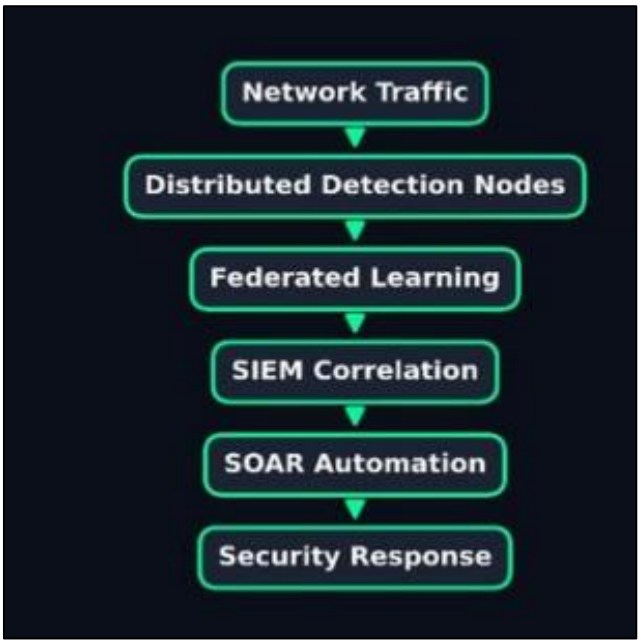


Fig 29 Integrated End-to-End Workflow of the DIDM Model

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 29 presents the integrated operational workflow of the Decentralized Intrusion Detection Model (DIDM) within the simulated Security Operations Center (SOC) environment. It illustrates the relationship between the intrusion detection component, Federated Learning (FL), Security Information and Event Management (SIEM), Security Orchestration, Automation, and Response (SOAR), and the monitoring dashboard.

The workflow begins with the preparation of network-related observations for intrusion detection. The input observations are subjected to the preprocessing procedures established during model development before being presented to the trained Artificial Neural Network (ANN) for classification. The model generates predictions that distinguish normal network activity from the attack categories represented in the CICIDS2017 dataset.

The decentralized learning capability of DIDM is established through the Federated Learning framework. In the reported experiment, five simulated clients performed local model training over five communication rounds. Model updates from the participating clients were aggregated by the coordinating server using Federated Averaging (FedAvg), producing a global model that was redistributed for subsequent training rounds. The raw training observations remained at the respective simulated clients. This federated-training process supports the development of the detection model and is distinct from the processing of individual observations during inference.

Following classification, the available detection outputs can be represented as security-event records for subsequent event management. The SIEM component receives and organizes the available records to support security monitoring and incident management. Where supported by the implemented integration, relevant incident information is then made available to the SOAR component for handling through configured response procedures.

The SOAR component manages response-related activities associated with the available incident information. These activities are determined by the response functions implemented within the experimental environment and may include recording response actions and maintaining incident-related status information. Actual enforcement of network-level mitigation measures is not assumed unless supported by execution and verification evidence.

The Streamlit-based SOC dashboard provides the operational monitoring interface for displaying selected classification outputs, security-event records, incident information, and response-related activities. It enables users to examine the available information across the experimental framework and supports the interpretation of the system's operational processes.

Overall, the integrated workflow demonstrates the relationship between decentralized ANN model development, intrusion classification, security-event management, response coordination, and operational monitoring. The formal detection performance of DIDM is assessed using the independent CICIDS2017 test dataset, while the SIEM, SOAR, and dashboard components demonstrate the supporting operational functions of the experimental SOC environment. The workflow therefore provides an integrated view of the framework without treating model training, individual-event classification, and incident response as a single simultaneous process.

- *Performance Evaluation and Operational Demonstration of the DIDM Model*

The Decentralized Intrusion Detection Model (DIDM) was further examined through its integrated operational environment to demonstrate the functionality of the implemented framework within the Streamlit-based cybersecurity monitoring dashboard. This evaluation formed part of the system implementation and

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

operational demonstration and focused on examining selected performance indicators and the presentation of detection outcomes within the experimental Security Operations Center (SOC) environment.

The operational evaluation presented in this section is distinct from the formal comparative machine-learning evaluation reported in the relevant performance comparison table. While the formal evaluation assessed the CIDM and DIDM using the defined independent CICIDS2017 test dataset and experimental protocol, the results presented in this section were generated within the operational demonstration environment. Consequently, the numerical results in Figures 28–34 are not necessarily identical to those obtained during the formal benchmark evaluation because the evaluations serve different purposes and may involve different data subsets, observation windows, model execution conditions, and measurement procedures.

The operational evaluation examines selected performance indicators, including detection accuracy, processing latency, false positive rate, confusion matrix results, Receiver Operating Characteristic (ROC) analysis, and Precision–Recall (PR) analysis. These indicators provide an additional view of the classification behaviour and monitoring capabilities demonstrated by the implemented system.

The results should therefore be interpreted as evidence from the operational demonstration rather than as a replacement for the formal comparative evaluation. In particular, the operational measurements are interpreted within the context of the data and procedures used to generate the corresponding dashboard outputs. The formal evaluation remains the principal basis for assessing the comparative machine-learning performance of CIDM and DIDM, while this section demonstrates additional operational characteristics of the implemented framework.

- *Detection Accuracy Trend*

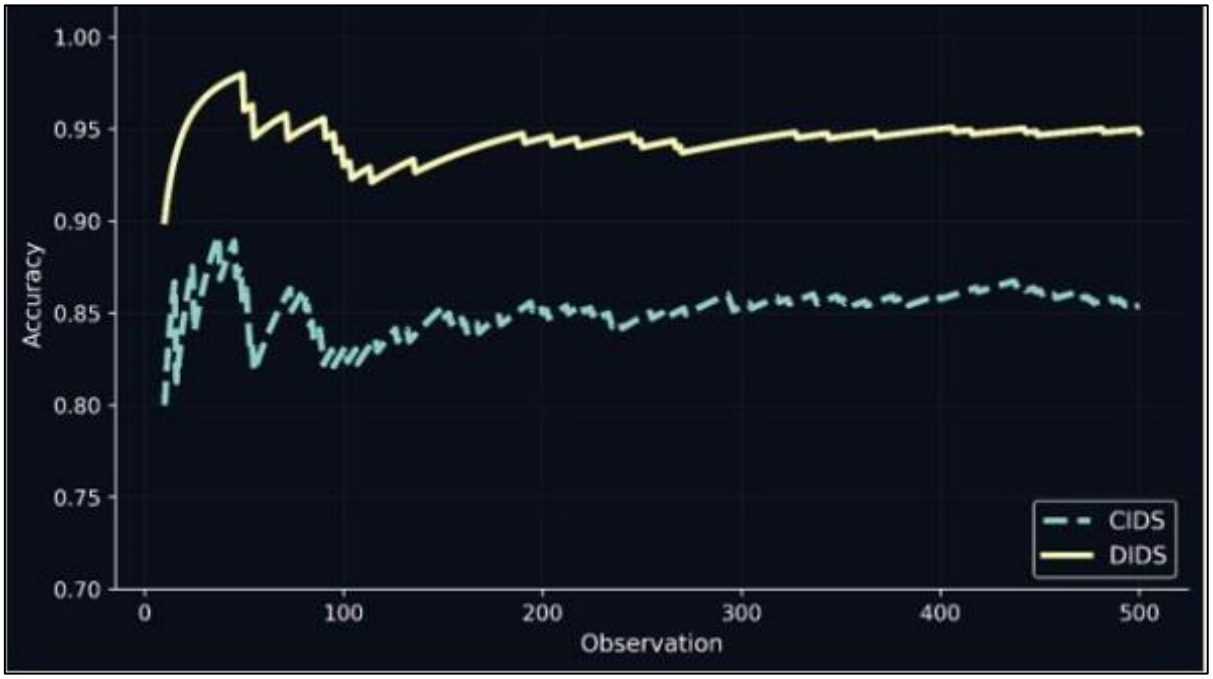


Fig 30 Detection Accuracy Trend of the DIDM Model

Figure 30 presents an operational comparison of the classification accuracy recorded by the conventional Centralized Intrusion Detection System (CIDS) and the Decentralized Intrusion Detection Model (DIDM) across 500 observations during the dashboard-based evaluation. The figure illustrates the variation in

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

classification accuracy for the two approaches and demonstrates how their performance was monitored within the developed Streamlit-based cybersecurity monitoring environment.

The operational results indicate that the DIDM generally maintained higher classification accuracy than the CIDS throughout the observation period. At the beginning of the evaluation, the DIDM recorded an accuracy of approximately 90%, increasing to about 98% during the initial observations before stabilising within an approximate range of 94% to 95%. Although some fluctuations were observed, the DIDM maintained comparatively consistent accuracy during the subsequent observations.

In comparison, the CIDS initially recorded an accuracy of approximately 80%. Its accuracy fluctuated between approximately 82% and 89% during the early observations before stabilising at around 85% to 86%. This variation illustrates the classification behaviour of the centralized baseline within the operational evaluation environment.

The comparison indicates that the DIDM recorded higher operational classification accuracy than the CIDS across the observation period represented in Figure 28. The difference between the two trends provides an indication of their respective classification behaviour under the specific conditions of the dashboard-based demonstration. However, the observed difference should not be attributed exclusively to the federated-learning architecture or supplementary detection mechanisms without a separate experiment isolating the contribution of each component.

The results presented in Figure 28 should be interpreted within the context of the operational demonstration. They are distinct from the formal comparative evaluation conducted using the independent CICIDS2017 test dataset. Differences between the operational accuracy trends and the formal benchmark results may arise from differences in the data observations, evaluation procedures, model execution conditions, and measurement windows.

Overall, Figure 28 demonstrates the ability of the implemented DIDM framework to monitor and visualise classification accuracy during the operational evaluation. Within the specific demonstration represented by the figure, the DIDM maintained higher observed accuracy than the centralized baseline. These findings complement the formal comparative results by providing an additional view of the classification behaviour of the implemented system.

- *Latency Performance*

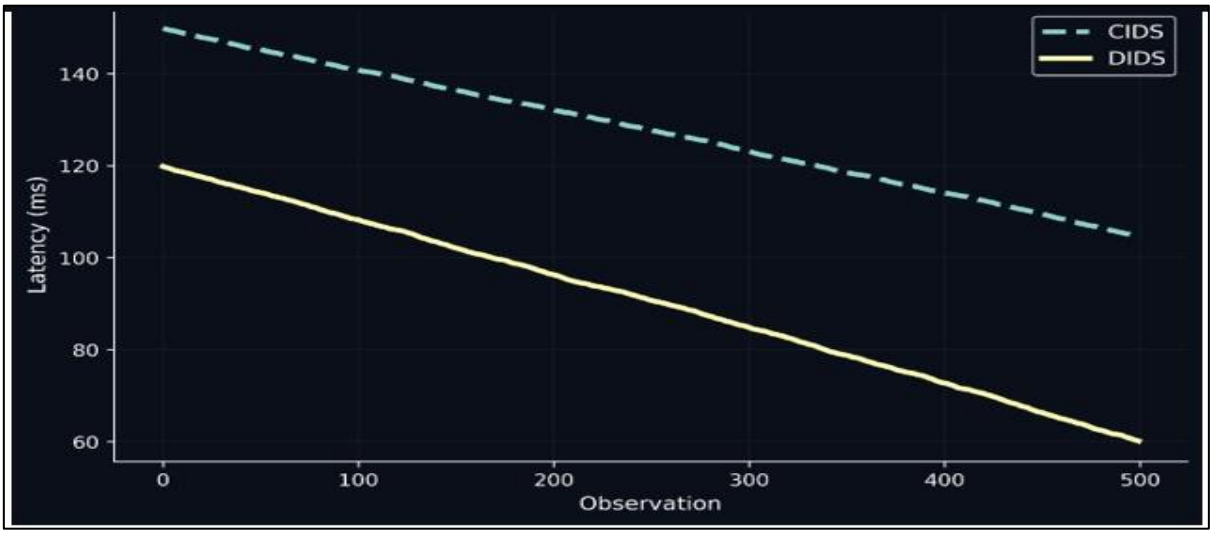


Fig 31 Latency Reduction Trend of the DIDM Model

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 31 presents an operational comparison of the detection latency recorded by the conventional Centralized Intrusion Detection System (CIDS) and the Decentralized Intrusion Detection Model (DIDM) during the dashboard-based evaluation. Detection latency refers to the measured time taken by the system to process an input observation and produce a corresponding detection outcome. Lower latency indicates faster processing under the measurement conditions used during the evaluation.

The results indicate that both models exhibited a gradual reduction in measured latency as the number of observations increased. At the beginning of the observation period, the CIDS recorded an approximate latency of 150 milliseconds (ms), while the DIDM recorded approximately 120 ms. This represented an initial difference of approximately 30 ms between the two approaches.

As the evaluation progressed, the measured latency of both models decreased. By the final observation, the CIDS recorded approximately 105 ms, whereas the DIDM recorded approximately 60 ms. Within the operational demonstration represented in Figure 29, the DIDM therefore maintained lower measured detection latency than the centralized baseline.

The observed latency differences describe the processing behaviour of the two implementations under the specific experimental conditions. Detection latency can be influenced by several factors, including processor and memory resources, input-data characteristics, preprocessing operations, model complexity, software configuration, and the measurement procedure. Consequently, the observed values should be interpreted in relation to the environment in which the operational evaluation was conducted.

It is also important to distinguish inference latency from federated-learning communication and training time. Although the DIDM uses a federated-learning architecture for collaborative model development, the aggregation and redistribution of model parameters are separate from the processing of individual observations during inference. Therefore, the lower latency observed in Figure 29 should not be attributed directly to federated learning without a separate measurement of the relevant processing and communication components.

The latency results are distinct from the formal comparative classification evaluation reported in Section 4.6.4. Figure 29 provides an operational measurement of processing time during the dashboard-based demonstration, whereas the formal evaluation establishes comparative classification performance using the independent CICIDS2017 test dataset.

Overall, Figure 29 demonstrates that the implemented DIDM recorded lower measured detection latency than the CIDS within the operational evaluation scenario. The results provide additional information about the processing characteristics of the experimental system and complement the formal comparative assessment presented elsewhere in this chapter.

- *False Positive Rate (FPR) Performance*

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

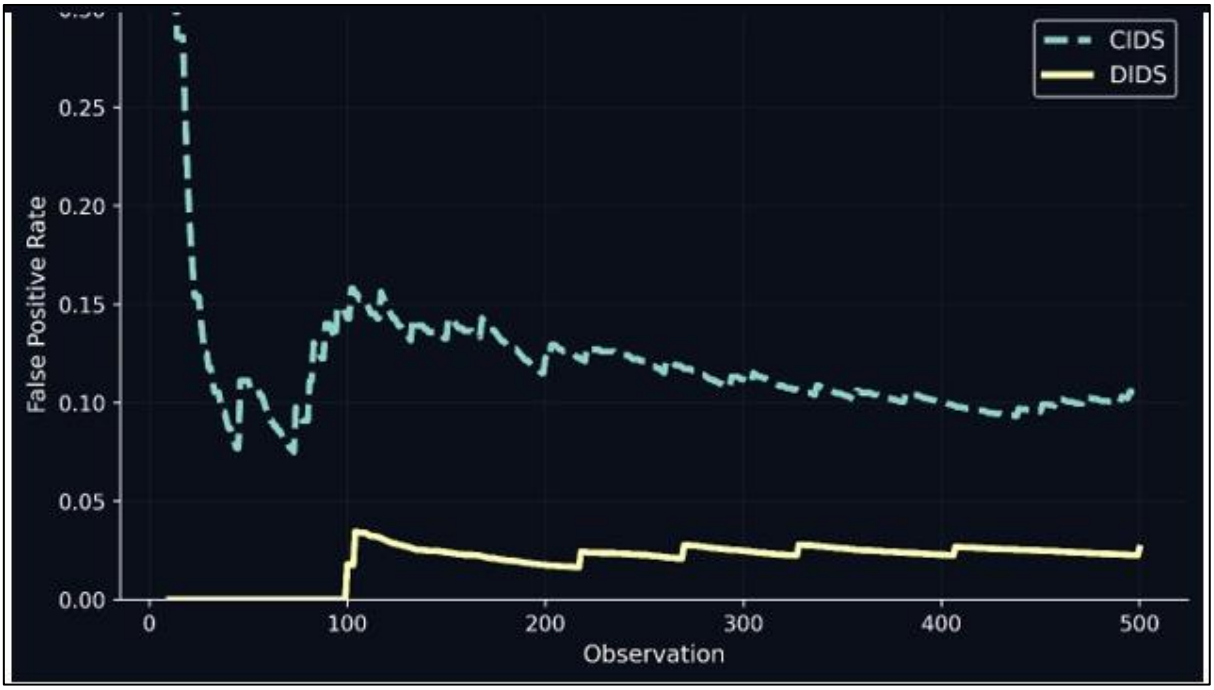


Fig 32 False Positive Rate (FPR) Performance Comparison

Figure 32 presents an operational comparison of the False Positive Rate (FPR) recorded by the conventional Centralized Intrusion Detection System (CIDS) and the Decentralized Intrusion Detection Model (DIDM) across 500 observations during the dashboard-based evaluation. The false positive rate represents the proportion of legitimate observations incorrectly classified as malicious. A lower FPR is desirable because it reduces unnecessary security alerts and the workload associated with investigating incorrectly generated alerts.

The operational results indicate that the DIDM maintained a substantially lower false positive rate than the CIDS throughout the observation period. During the initial observations, the CIDS recorded an approximate FPR of 30%. This rate subsequently decreased and stabilised within a range of approximately 10% to 15% during the remaining evaluation period. Although the centralized baseline exhibited a reduction in false positive behaviour, its recorded rate remained comparatively higher than that of the DIDM.

In comparison, the DIDM maintained a relatively low false positive rate during the operational demonstration. Following the initial observations, its FPR remained below approximately 3%, with relatively minor fluctuations across the subsequent evaluation period. This indicates that, within the specific operational scenario represented in Figure 30, the DIDM incorrectly classified fewer legitimate observations as malicious than the centralized baseline.

The observed differences illustrate the false-positive behaviour of the two implementations under the conditions of the operational evaluation. However, the results should not be attributed exclusively to the federated-learning architecture or supplementary anomaly-detection mechanisms without separate experiments establishing the contribution of each component. The false positive rate may also be influenced by the dataset composition, class distribution, classification thresholds, model configuration, and evaluation procedure.

The results presented in Figure 30 should be interpreted within the context of the dashboard-based demonstration and distinguished from the formal comparative evaluation reported in Section 4.6.4. The operational trends represent measurements from the specific evaluation environment and are not intended to replace the formal benchmark results obtained using the independent CICIDS2017 test dataset.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

From an operational perspective, a lower false positive rate can help reduce unnecessary alerts and support the prioritisation of potentially genuine security incidents. This is relevant to the DIDM SOC architecture, where detection-related information may subsequently be organised through the SIEM component and handled through configured SOAR procedures.

Overall, Figure 30 demonstrates that the implemented DIDM recorded a lower observed false positive rate than the CIDS during the operational evaluation. The results provide additional evidence of the false-positive behaviour monitored through the Streamlit-based dashboard and complement the formal comparative assessment presented elsewhere in this chapter.

- *Overall Performance Summary*

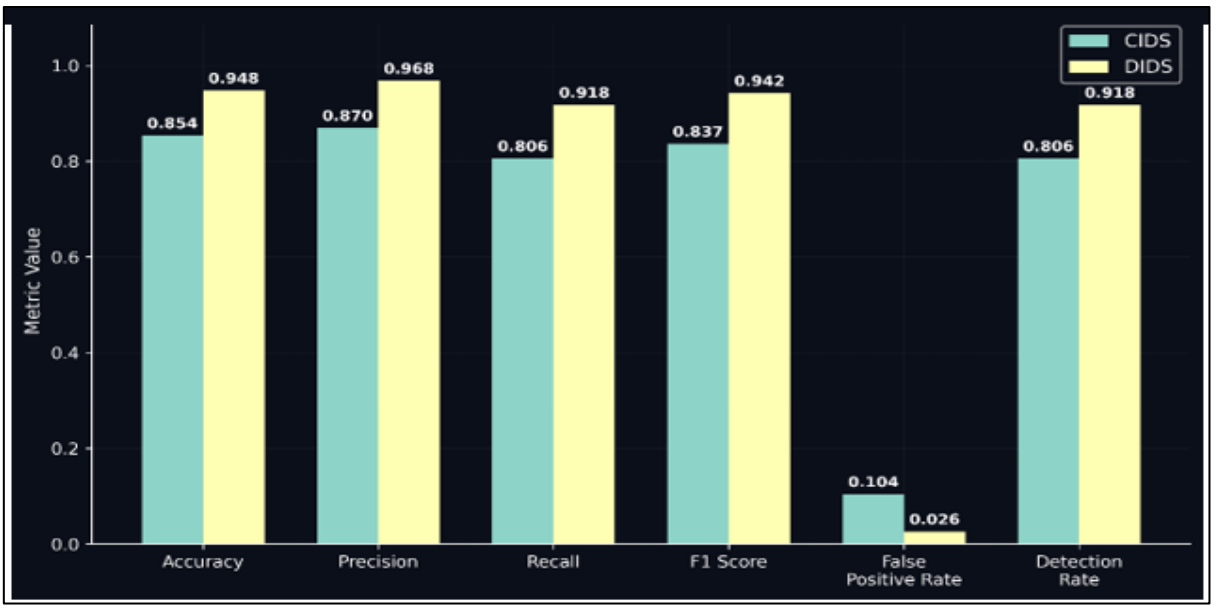


Fig 33 Overall Performance Summary of the DIDM Model

Figure 33 presents a comparative summary of the operational performance of the Centralized Intrusion Detection System (CIDS) and the Decentralized Intrusion Detection Model (DIDM), based on the 500-observation dashboard evaluation. The comparison considers accuracy, precision, recall, F1-score and false positive rate (FPR). Detection rate is equivalent to recall and is therefore interpreted using the same values rather than treated as an independent performance measure.

The operational results indicate that DIDM recorded an accuracy of 94.8%, compared with 85.4% for CIDS, representing a difference of 9.4 percentage points. DIDM also recorded higher precision (96.8%) than CIDS (87.0%), while its recall was 91.8%, compared with 80.6% for CIDS. The corresponding F1-scores were 94.2% for DIDM and 83.7% for CIDS.

The false positive rate was 2.6% for DIDM and 10.4% for CIDS. These values indicate that, within this operational evaluation, DIDM produced a lower proportion of false positive classifications. The detection rates were 91.8% and 80.6%, respectively, corresponding to the recall values reported above.

Overall, the operational results show stronger classification performance for DIDM across accuracy, precision, recall and F1-score, together with a lower false positive rate. However, these findings describe the specific 500-observation dashboard evaluation and should not be interpreted as an independent demonstration of the effect of federated learning or the supplementary anomaly-detection mechanisms.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The results should also be distinguished from the formal CIDM–DIDM comparison presented in Section 4.6.4, which uses the independent CICIDS2017 test set of 504,151 observations and the formal ANN-based evaluation procedure. The two evaluations have different sample sizes and evaluation contexts; consequently, their results should be reported separately.

- *Machine Learning Evaluation Results*

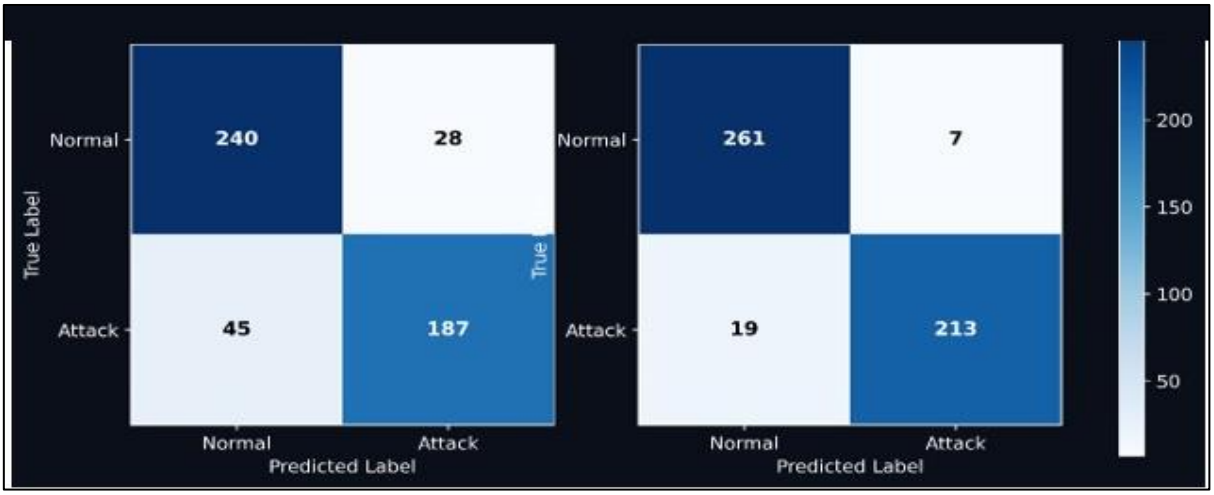


Fig 34 Confusion Matrix Comparison Between CIDS and DIDM

Figure 34 presents the confusion matrices obtained from the operational evaluation of CIDS and DIDM using 500 observations for each model. The CIDS confusion matrix records 240 true negatives, 28 false positives, 45 false negatives and 187 true positives. The DIDM confusion matrix records 261 true negatives, 7 false positives, 19 false negatives and 213 true positives.

The results indicate that DIDM correctly classified more normal and attack observations than CIDS within this evaluation sample. DIDM recorded fewer false positives (7) and false negatives (19), compared with 28 and 45, respectively, for CIDS. These differences are consistent with the higher accuracy, precision, recall and F1-score reported in Section 4.7.15.4. The confusion matrices provide a threshold-specific account of classification outcomes. They therefore complement, rather than replace, the formal independent-test evaluation presented in Section 4.6.4.

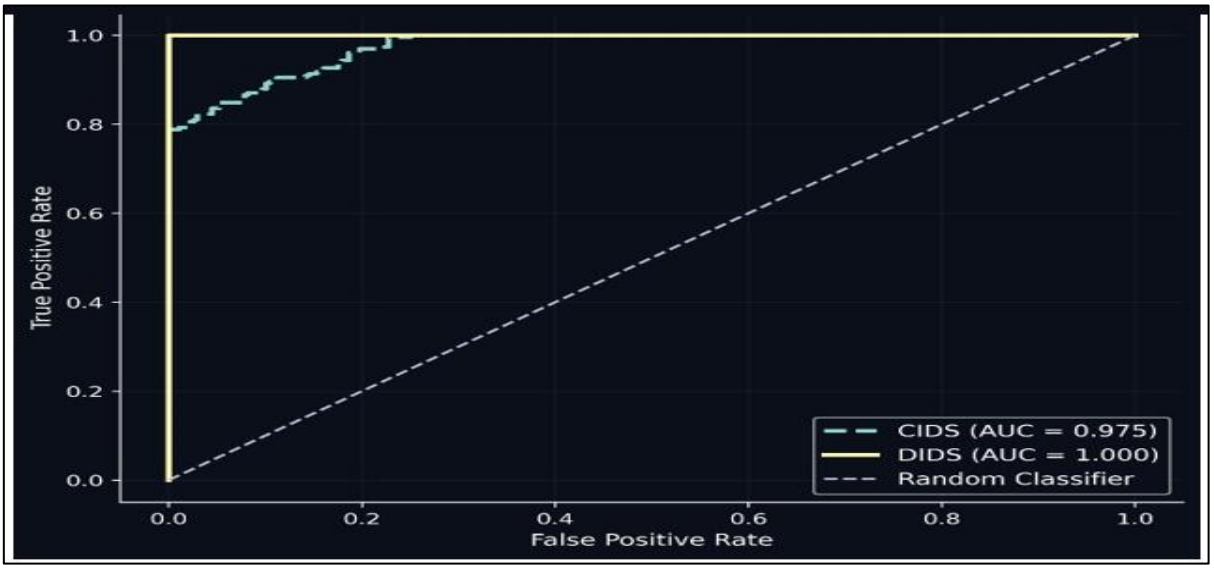


Fig 35 ROC Curve Comparison Between CIDS and DIDM

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 35 presents the Receiver Operating Characteristic (ROC) curves for the operational comparison. The reported area under the curve (AUC) values are 0.975 for CIDS and 1.000 for DIDM. The ROC curve illustrates the relationship between the true positive rate and false positive rate across different classification thresholds.

The reported values suggest stronger class-ranking performance for DIDM within the operational evaluation. However, the AUC values should be retained only after confirming that they were calculated from continuous prediction scores or class probabilities, using the same observations and positive-class definition for both models. An AUC of 1.000 can coexist with classification errors in a confusion matrix when the selected classification threshold does not reflect the ranking of the scores. The reported AUC should therefore not be interpreted as evidence of error-free classification at the selected threshold.

Table 6. The ROC analysis therefore complements, rather than replaces, the formal model evaluation by illustrating how the implemented DIDM system behaves when applied to the operational intrusion detection scenario.

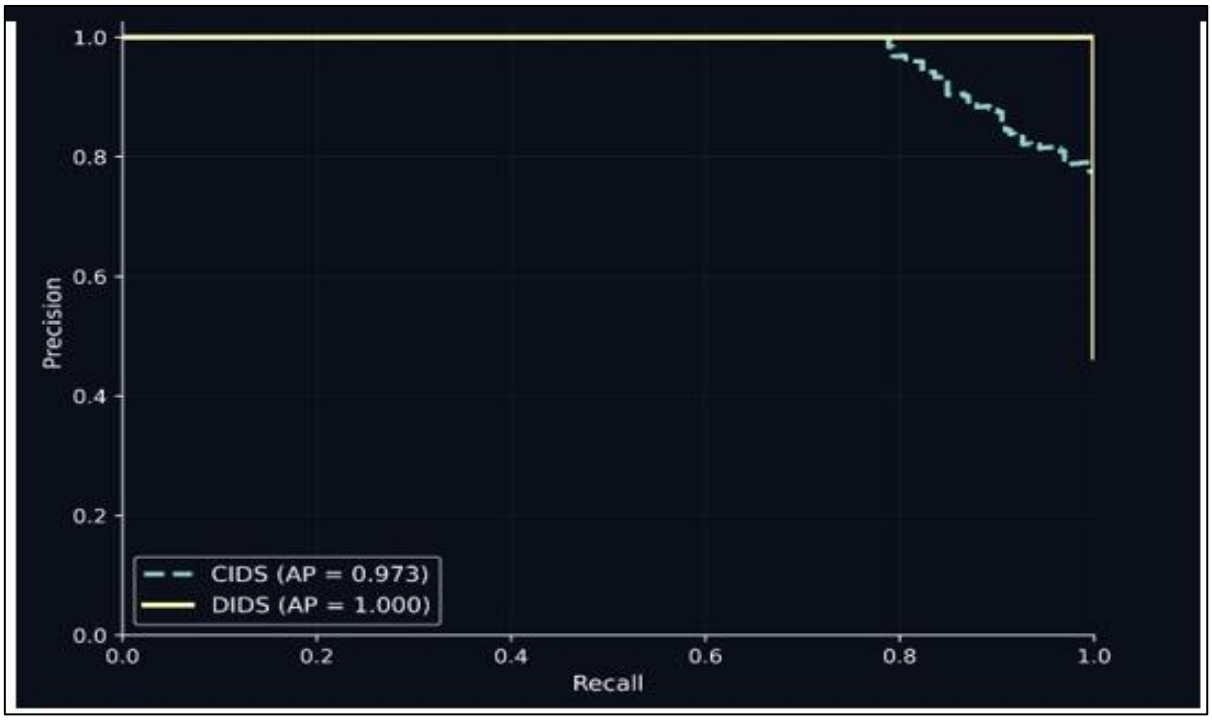


Fig 36 Precision–Recall Curve Comparison Between CIDM and DIDM

Figure 36 presents the Precision–Recall (PR) curves for the two operational models. The reported average precision (AP) values are 0.973 for CIDS and 1.000 for DIDM. The PR curve describes the relationship between precision and recall across different classification thresholds and is useful for examining the detection of the positive attack class.

The reported values suggest that DIDM achieved stronger precision–recall performance in this operational evaluation. Nevertheless, the AP values should be confirmed using the underlying continuous prediction scores, the same evaluation observations and a consistently defined positive class. The baseline model should also be labelled consistently as CIDS in the figure, legend and accompanying discussion. The term CIDM should be reserved for the formal centralized model comparison where that is the intended model designation.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Taken together, the confusion matrices and curve-based measures provide complementary evidence about the operational classification behaviour of the two models. The confusion matrices describe outcomes at the selected classification threshold, while ROC-AUC and average precision assess performance across thresholds. These operational results remain separate from the formal ANN-based CIDM–DIDM comparison reported in Section 4.6.4 and should not be used as a substitute for that independent-test evaluation.

➢ *Expert Validation Analysis*

Expert validation was conducted with twenty cybersecurity and information security professionals drawn from Kenya Commercial Bank (KCB), Equity Bank, and Africa Securex. The institutions were purposively selected because of their involvement in enterprise information security, digital infrastructure protection, cybersecurity operations, and incident response. The participants were selected based on their professional experience and relevance to the assessment of intrusion detection systems and distributed cybersecurity environments.

The primary objective of the expert validation exercise was to assess the limitations of existing Intrusion Detection Models (IDMs) and identify the technical and operational requirements of next-generation intrusion detection systems. The assessment considered several dimensions, including detection accuracy, scalability, privacy preservation, resilience, real-time responsiveness, adaptability, and the relevance of different cybersecurity attack categories.

The responses were anal perspective on the requirements of intrusion detection systems, while the model's technical performance was evaluated separately using the experimental procedures described in alysed using descriptive statistics to identify the issues and requirements considered important by the participating experts. The findings were subsequently interpreted in relation to the study objectives and the design of the proposed Decentralized Intrusion Detection Model (DIDM). The expert assessment provided a professionthe preceding sections.

- *Identified Weaknesses of Existing IDM*

Table 8 presents the limitations of existing intrusion detection models identified by the participating experts. The reported findings indicate that a high false positive rate was the most frequently identified concern, at 73%. Delayed response to threats and model complexity followed, each reported at 45%. Other identified limitations included poor scalability (27%), limited cloud and Internet of Things (IoT) integration (14%), low detection accuracy (9%), and data privacy concerns (9%).

The prominence of false positive rates highlights the importance of accurate discrimination between legitimate network activities and malicious traffic. Excessive false alerts can increase the workload of security analysts, require additional investigation, and potentially reduce confidence in intrusion detection systems. This finding reinforces the importance of evaluating classification performance using measures such as precision, recall, F1-score, and false positive rate rather than relying exclusively on overall accuracy.

Delayed response to threats was another concern identified by the experts. In network environments involving multiple systems and distributed resources, delays in analysing security events and communicating detection results may affect the timeliness of incident handling. This observation highlights the importance of efficient detection and event-processing mechanisms in intrusion detection architectures.

Model complexity was also identified as a significant limitation. The management of intrusion detection systems may involve configuring detection components, maintaining models, monitoring system outputs, and adapting to changing network environments. These activities can increase operational demands, particularly where systems must support heterogeneous infrastructure and multiple sources of network traffic.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Poor scalability and limited cloud/IoT integration further indicate the importance of designing intrusion detection systems that can accommodate distributed and heterogeneous computing environments. As network infrastructures expand, detection architectures need to support multiple participating environments without unnecessarily requiring all network traffic to be processed through a single centralized data repository.

Although low detection accuracy and data privacy issues received comparatively lower reported percentages, they remain important considerations. Detection accuracy directly affects the reliability of intrusion classification, while privacy concerns are relevant when network traffic contains sensitive organizational information. These considerations informed the need to examine decentralized learning approaches that support collaborative model development while retaining raw data at participating client nodes.

Table 8 Expert-Identified Limitations of Existing IDM

Limitation	Frequency (%)	Technical Interpretation
High false positive rate	73%	Weak feature discrimination leading to incorrect alerts
Delayed response to threats	45%	Centralized processing latency
Model complexity	45%	Operational management difficulty
Poor scalability	27%	Limited distributed deployment capability
Limited cloud/IoT integration	14%	Incompatible with modern infrastructures
Low detection accuracy	9%	Reduced classification performance
Data privacy issues	9%	Limited privacy-aware detection mechanisms

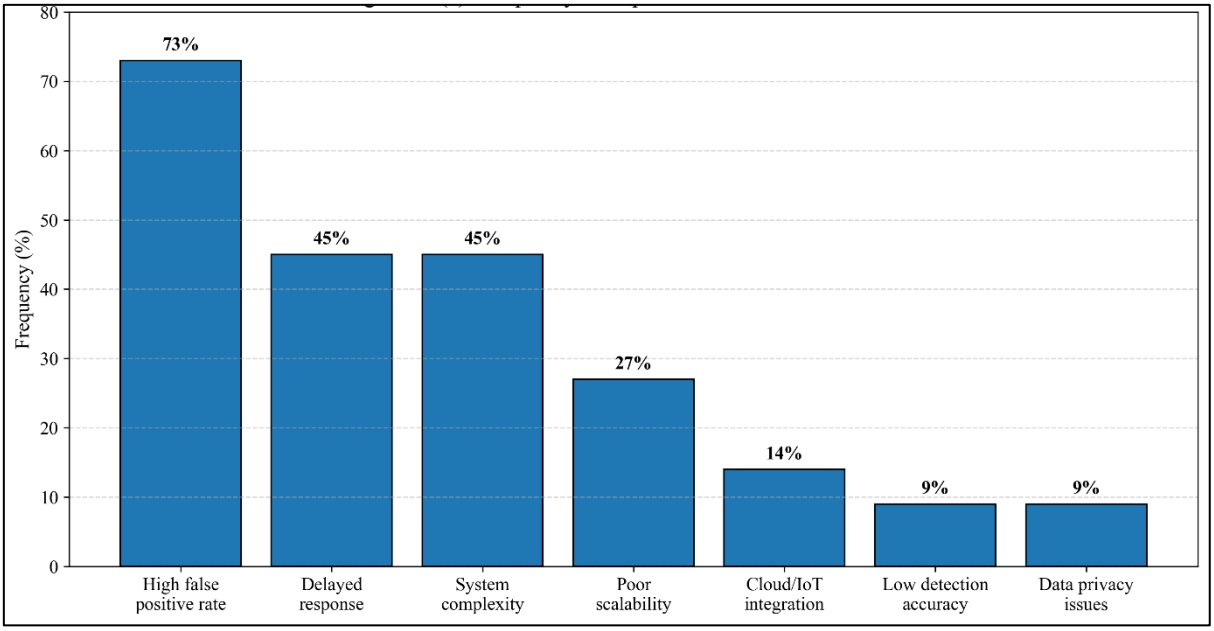


Fig 37 (a) Frequency of Expert-Identified IDM Limitations

As illustrated in Figure 37(a), the frequency distribution of expert-identified IDM limitations revealed that a high false positive rate was the most prominent challenge, reported by 73% of the respondents. This was followed by delayed response to threats and Model complexity, each cited by 45% of the experts. Poor scalability was identified by 27% of the respondents, while limited cloud/IoT integration was reported by 14%. Low detection accuracy and data privacy issues were the least cited limitations, each receiving 9% of the responses. These findings suggest that operational reliability and alert precision remain the primary concerns in conventional IDM models, potentially reducing analyst confidence and increasing the workload associated with investigating false alarms.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

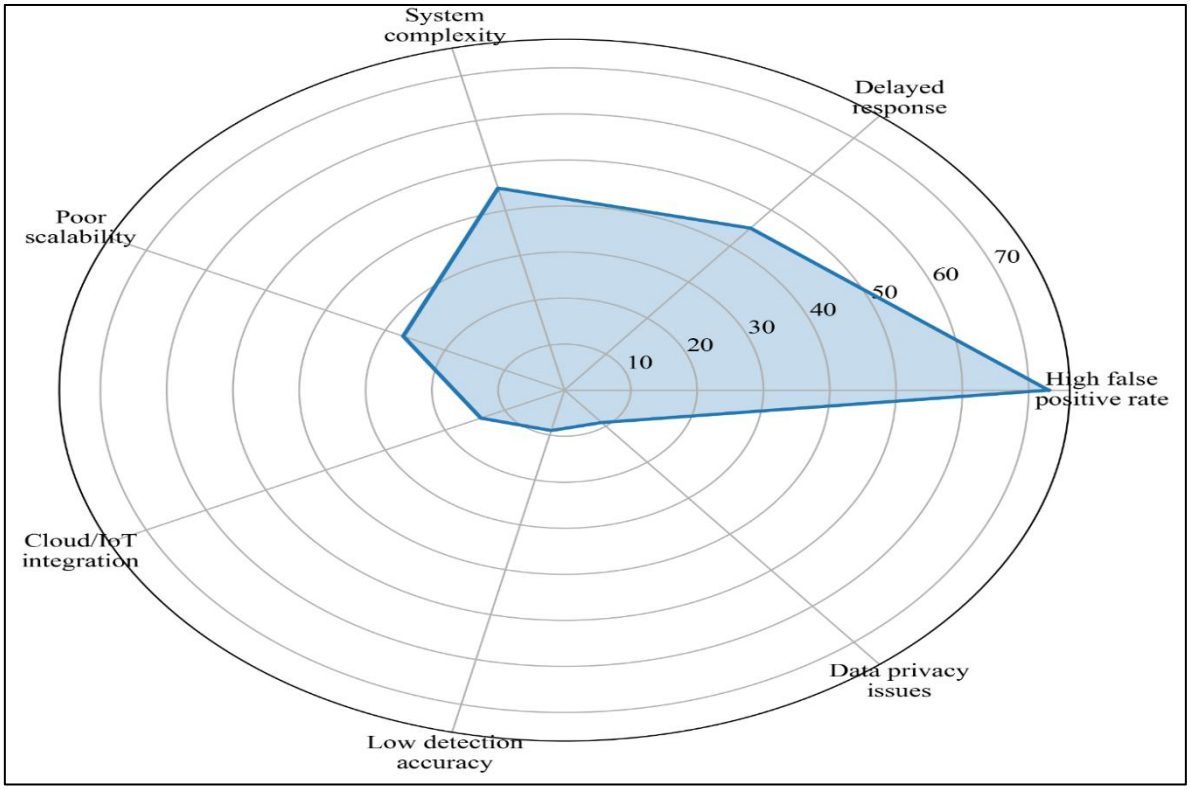


Fig 37(b): Polar Representation of Expert-Identified IDM Limitations

Figure 37(b) presents the relative distribution of expert-identified limitations using a polar visualisation. The chart illustrates the prominence of high false positive rates, which received the highest response percentage among the identified limitations. Delayed threat response and model complexity form the next prominent group, followed by poor scalability and limited cloud/Internet of Things (IoT) integration. Low detection accuracy and data privacy concerns received comparatively fewer responses.

The visualisation reinforces the findings presented in Figure 37(a), highlighting false positive rates, response delays and model complexity as the most frequently reported concerns. These findings suggest that improving detection precision and strengthening response efficiency are important considerations in the design of next-generation intrusion detection models.

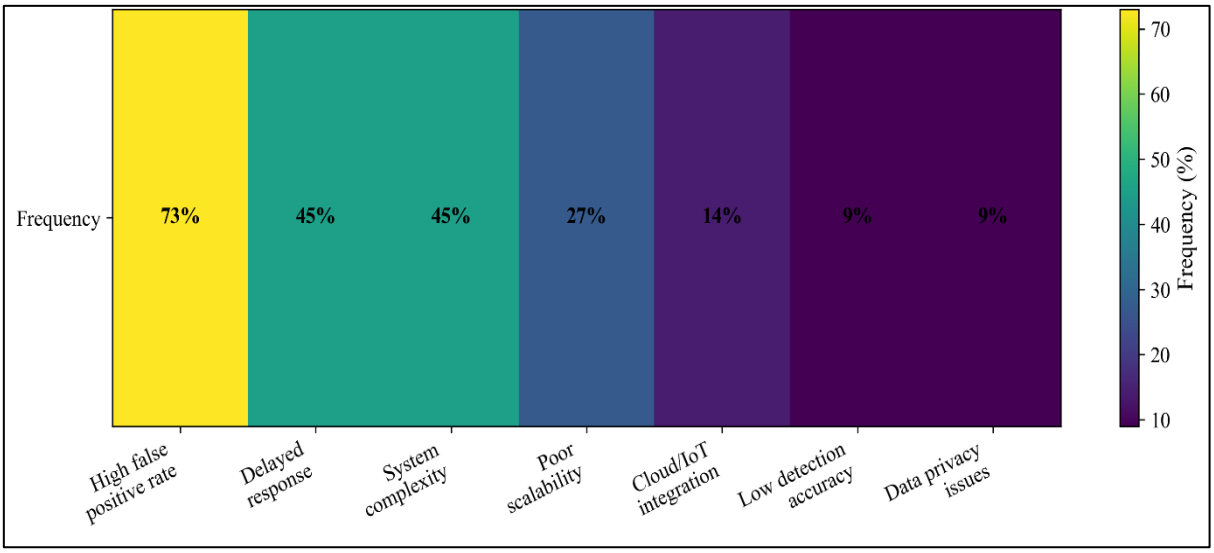


Fig 37(c): Heatmap Representation of Expert-Identified IDM Limitations

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 37(c) presents a heatmap representation of the expert-identified limitations of existing intrusion detection models. The visualisation highlights the high false positive rate as the most frequently reported limitation, followed by delayed threat response and model complexity. Poor scalability and limited cloud/Internet of Things (IoT) integration occupy comparatively lower positions, while low detection accuracy and data privacy concerns received the fewest responses.

The heatmap provides an additional visual representation of the response distribution presented in Table 8 and Figures 37(a) and 37(b). It reinforces the prominence of false positive rates, response delays and model complexity among the concerns identified by the participating experts. The less frequently reported limitations also remain relevant when considering the requirements of intrusion detection systems operating in distributed and interconnected environments.

➢ *Overall Interpretation*

Collectively, Table 8 and Figures 37(a–c) demonstrate that existing intrusion detection models continue to face significant operational and architectural limitations, as identified by the participating cybersecurity professionals. The most frequently reported challenge was the high false positive rate, indicating persistent concerns about distinguishing malicious activities from legitimate network behaviour. Delayed response to threats and model complexity were also identified as major concerns, suggesting that conventional intrusion detection architectures may encounter operational and management challenges that affect timely threat identification and effective system deployment.

The graphical analyses further demonstrate that although poor scalability, limited cloud and Internet of Things (IoT) integration, low detection accuracy and data privacy concerns were reported less frequently, they remain important considerations in increasingly distributed and heterogeneous computing environments. The prominence of false positive rates, response delays and model complexity highlights the need for intrusion detection models that place greater emphasis on detection precision, response efficiency and manageable system architecture.

Overall, these findings provide a professional and technical justification for investigating a decentralized intrusion detection model based on federated learning and artificial intelligence. The proposed DIDM incorporates distributed model training through federated learning, with raw training data retained at the participating simulated clients while model parameters are exchanged for aggregation. Its architecture also incorporates SIEM and SOAR components to support security event monitoring and incident-response workflows. These design elements address selected concerns identified by the experts; however, their inclusion does not independently establish reduced false positive rates, formal privacy guarantees, production-level scalability or seamless integration with external cloud and IoT infrastructures.

Consequently, the identified limitations establish the professional and architectural rationale for the proposed DIDM and provide a foundation for interpreting its experimental performance and implementation results in the subsequent sections.

- *Design Requirements for Next-Generation Intrusion Detection Models*

The expert validation exercise examined the design requirements considered important for next-generation intrusion detection models. The participating cybersecurity professionals identified seven key requirements: real-time detection, adaptive learning, autonomous response, privacy preservation, robustness against attacks, scalability and ease of integration. These requirements reflect the technical and operational capabilities considered necessary for addressing evolving cybersecurity threats in modern network environments.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Real-time detection emerged as the most important requirement, identified by 91% of respondents. This finding highlights the need for intrusion detection systems capable of identifying suspicious activities promptly to support timely security intervention. Adaptive learning followed at 68%, indicating the perceived importance of models that can accommodate changing network behaviour and emerging attack patterns.

Autonomous response was identified by 59% of respondents, reflecting the importance of supporting timely incident-response actions and reducing dependence on continuous manual intervention. Privacy preservation was reported by 45%, demonstrating the relevance of protecting sensitive information during security monitoring and distributed model training. Robustness against attacks was identified by 41% of respondents, emphasising the need for reliable detection under varying and potentially adversarial conditions.

Scalability and ease of integration received importance scores of 9% and 5%, respectively. Although these requirements were less frequently identified, they remain relevant to the deployment of intrusion detection systems across distributed environments and existing organisational security infrastructures.

Table 9 Expert-Identified Design Requirements for Next-Generation Intrusion Detection Systems

Requirement	Importance (%)	Interpretation
Real-time detection	91%	Critical for immediate threat response
Autonomous response	59%	Enables automated defense actions
Privacy preservation	45%	Supports secure data handling
Robustness to attacks	41%	Improves system resilience
Adaptive learning	68%	Enhances evolving threat detection
Scalability	9%	Lower priority but necessary
Easy integration	5%	Minimal legacy dependency requirement

Source: Expert validation survey, 2026.

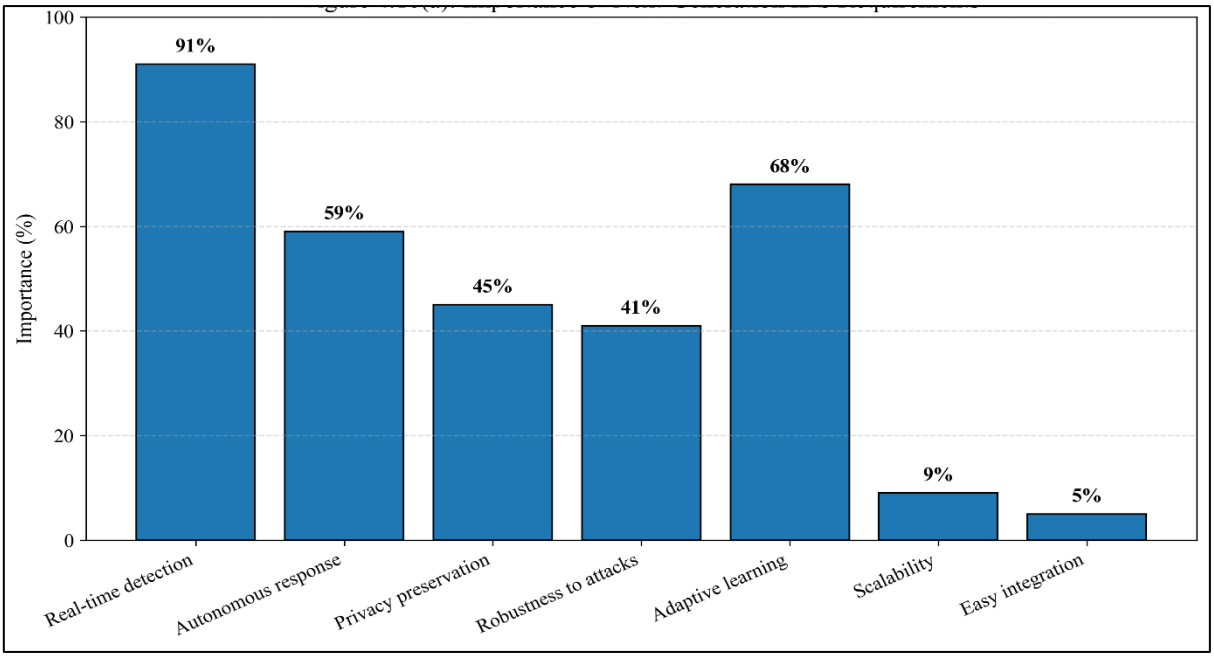


Fig 38(a): Frequency Distribution of Expert-Identified Design Requirements

Figure 38(a) presents the frequency distribution of the design requirements identified by the participating cybersecurity professionals. Real-time detection emerged as the highest priority, reported by 91% of respondents, followed by adaptive learning at 68% and autonomous response at 59%. Privacy preservation was identified by 45%, while robustness against attacks received 41%. Scalability and ease of integration were

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

reported by 9% and 5%, respectively. The distribution indicates that the experts placed considerable emphasis on timely threat identification, adaptability and effective incident response when considering the requirements of next-generation intrusion detection systems.

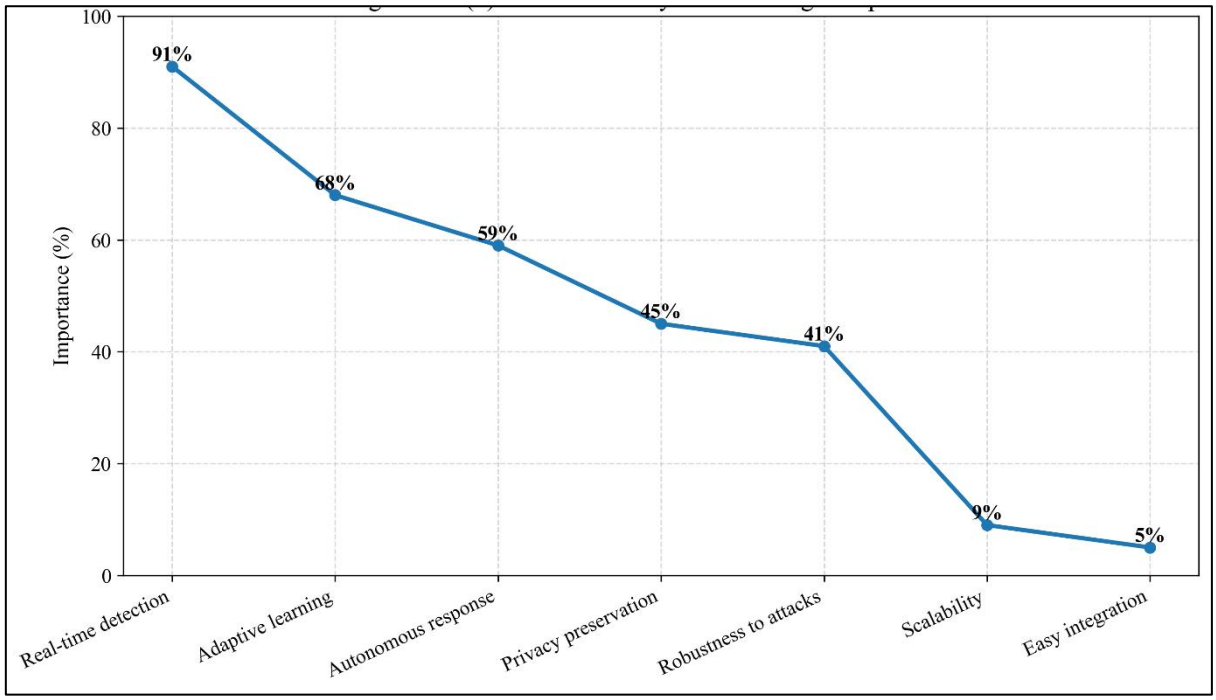


Fig 38(b): Ranked Representation of Expert-Identified Design Requirements

Figure 38(b) presents the design requirements in descending order of their reported importance. The ranking highlights real-time detection as the dominant requirement, followed by adaptive learning and autonomous response. Privacy preservation and robustness against attacks occupy intermediate positions, while scalability and ease of integration received comparatively lower percentages. This representation provides a comparative view of the priorities expressed by the experts and highlights the importance of balancing detection effectiveness, operational responsiveness and the architectural requirements of distributed security systems.

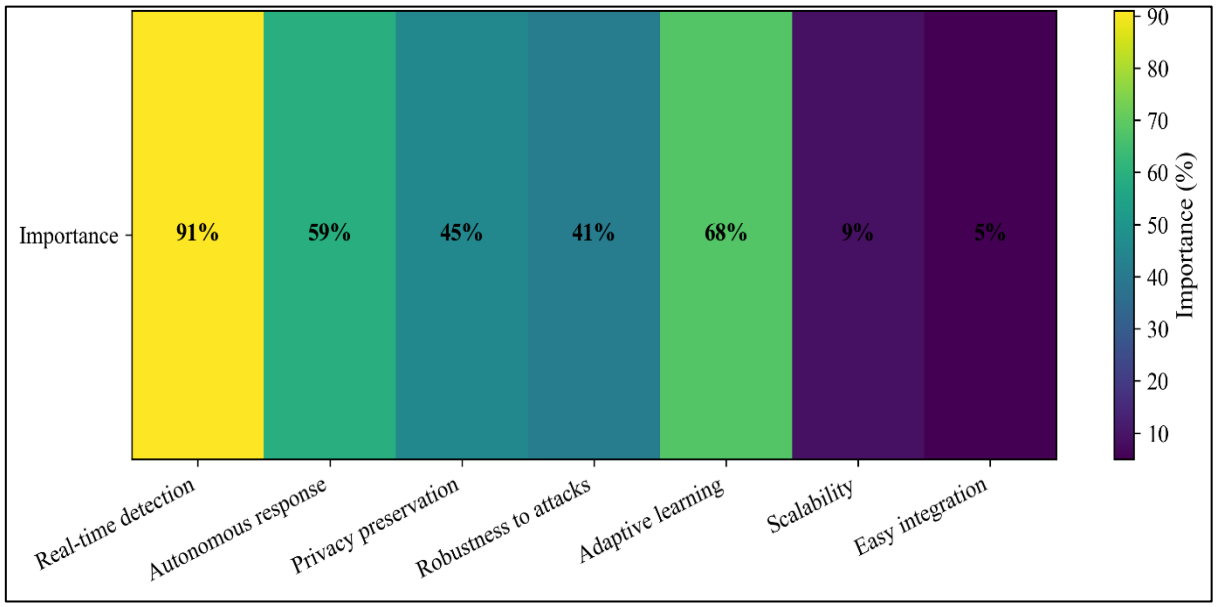


Fig 38(c): Heatmap Representation of Expert-Identified Design Requirements

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 38(c) provides a heatmap representation of the relative importance assigned to the identified design requirements. The strongest visual intensity corresponds to real-time detection, followed by adaptive learning and autonomous response. Privacy preservation and robustness against attacks exhibit intermediate levels, while scalability and ease of integration show comparatively lower intensity. The visualisation reinforces the response distribution presented in Table 9 and illustrates the relative emphasis placed on different design considerations by the participating professionals.

➢ *Overall Interpretation*

Collectively, Table 9 and Figures 38(a–c) demonstrate that real-time detection, adaptive learning and autonomous response were the most prominent design requirements identified by the participating cybersecurity professionals. The findings suggest that next-generation intrusion detection systems should be designed to identify threats promptly, accommodate changing attack patterns and support efficient incident-response processes.

The identification of privacy preservation and robustness against attacks further emphasises the importance of protecting sensitive information and maintaining dependable detection capabilities in distributed computing environments. Although scalability and ease of integration received lower response percentages, they remain relevant considerations when designing systems intended for deployment across different organisational infrastructures.

These findings informed the architectural design of the proposed Decentralized Intrusion Detection Model (DIDM). The use of federated learning supports distributed model training across five simulated clients while retaining raw training data at the participating client level. Model parameters are exchanged and aggregated through the Federated Averaging (FedAvg) process over five communication rounds. The architecture also incorporates Security Information and Event Management (SIEM) and Security Orchestration, Automation and Response (SOAR) components to support security event monitoring and incident-response workflows.

Nevertheless, the implementation should be interpreted within the scope of the experiments conducted. The five-client simulation does not establish production-level scalability, while retaining raw training data locally does not provide a formal privacy guarantee through mechanisms such as differential privacy or secure aggregation. Similarly, the inclusion of SOAR components does not independently demonstrate that all response actions were autonomously enforced in an operational enterprise environment. Real-time detection and robustness against adversarial attacks also require evidence from the relevant experimental evaluations.

Consequently, the expert findings provide a professional basis for the design direction of DIDM, while the extent to which its implementation addresses the identified requirements is evaluated separately through the experimental and operational results presented in this chapter.

- *Priority Attack Types for Detection*

The expert evaluation identified and ranked the cyberattack categories considered important for prioritisation by next-generation intrusion detection models, as summarised in Table 10. The findings indicated that Denial of Service (DoS) attacks received the highest priority, at 86%, reflecting concerns about service availability and the potential disruption of normal network operations. Phishing attacks followed at 82%, highlighting the importance of detecting social engineering activities that may facilitate credential compromise and unauthorised access.

A group of attack categories comprising malware and virus attacks, Distributed Denial of Service (DDoS) attacks, User-to-Root (U2R) attacks and insider threats each received 68%. These findings demonstrate that the participating experts considered both external and internal security threats important. Malware and virus

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

attacks can compromise system integrity, while DDoS attacks can disrupt service availability through distributed traffic generation. U2R attacks involve attempts to obtain unauthorised privileged access, whereas insider threats originate from individuals with legitimate access or from compromised internal accounts.

Scanning and zero-day attacks each received 64%. Scanning activities are associated with reconnaissance and the identification of potentially vulnerable network services. Zero-day attacks involve the exploitation of vulnerabilities before effective corrective measures or detection signatures are available. Remote-to-Local (R2L) attacks received 55%, indicating that unauthorised remote access remained an important consideration in the experts' assessment.

The findings demonstrate that the participating cybersecurity professionals prioritised a broad range of threats, including availability attacks, social engineering, malicious software, privilege escalation, reconnaissance and unauthorised access. This distribution indicates the importance of considering diverse attack behaviours when defining the requirements and evaluation scope of intrusion detection models.

Table 10 Priority Ranking of Cybersecurity Attack Types Based on Expert Assessment

Attack Type	Priority (%)	Interpretation
DoS attacks	86%	Highly disruptive availability attacks
Phishing attacks	82%	Social engineering threats
Malware/Virus attacks	68%	Persistent Model compromise
DDoS attacks	68%	Distributed disruption attacks
U2R attacks	68%	Privilege escalation threats
Insider threats	68%	Internal security risks
Scanning attacks	64%	Reconnaissance activities
Zero-day attacks	64%	Unknown vulnerability exploitation
R2L attacks	55%	Unauthorized remote access

Source: Expert validation survey, 2026.

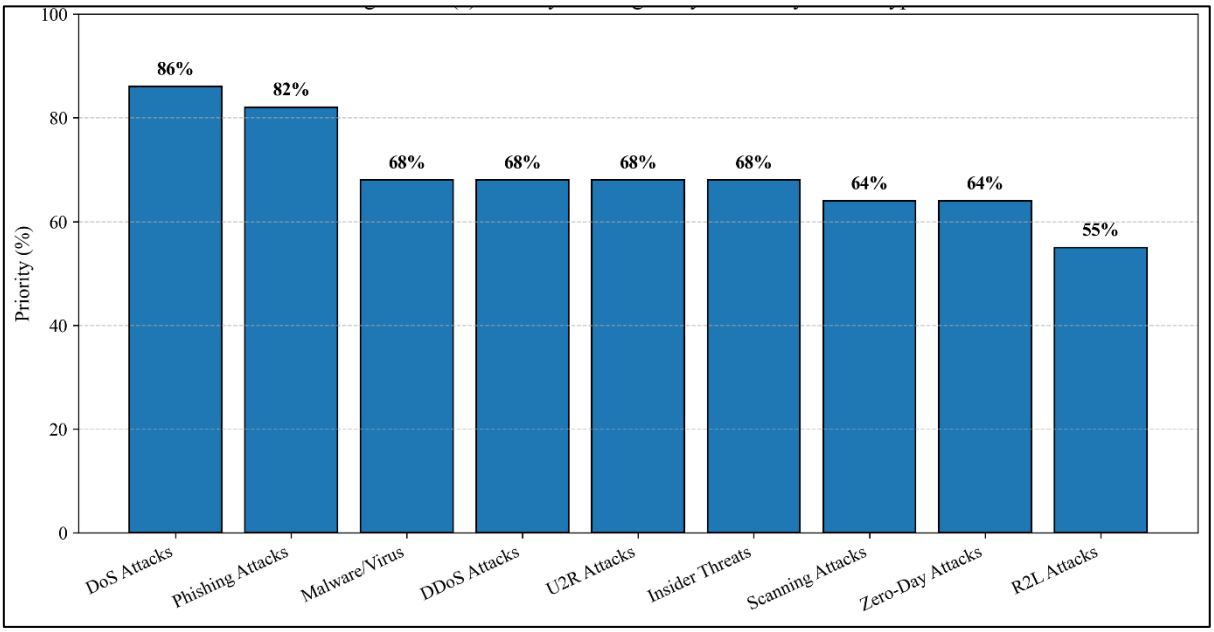


Fig 39(a): Frequency Distribution of Expert-Identified Priority Attack Types

Figure 39(a) presents the frequency distribution of the attack categories identified by the participating experts. DoS attacks received the highest priority at 86%, followed by phishing attacks at 82%. Malware and virus attacks, DDoS attacks, U2R attacks and insider threats each received 68%. Scanning and zero-day attacks

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

were each reported at 64%, while R2L attacks received 55%. The distribution illustrates the relative importance assigned to the different attack categories during the expert evaluation.

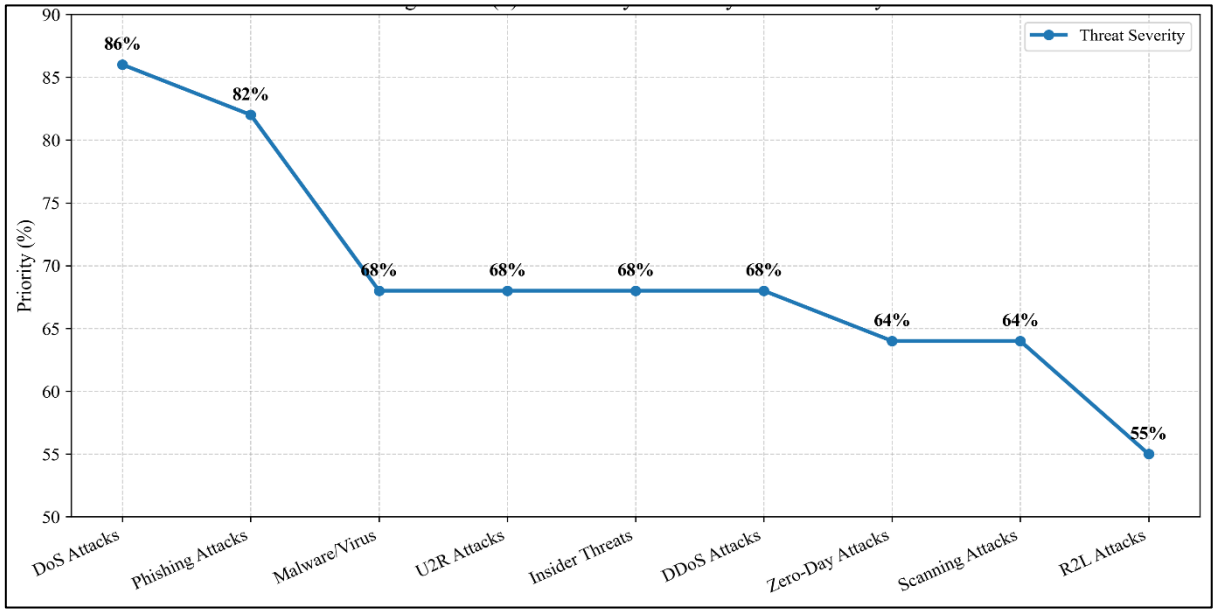


Fig 39(b): Ranked Representation of Expert-Identified Priority Attack Types

Figure 39(b) presents the attack categories in descending order of their reported priority. The ranking highlights the prominence of DoS and phishing attacks, followed by the four categories that received equal priority percentages. Scanning and zero-day attacks occupy the next positions, while R2L attacks received the lowest percentage among the identified categories. This representation provides a comparative view of the threats considered important by the participating cybersecurity professionals.

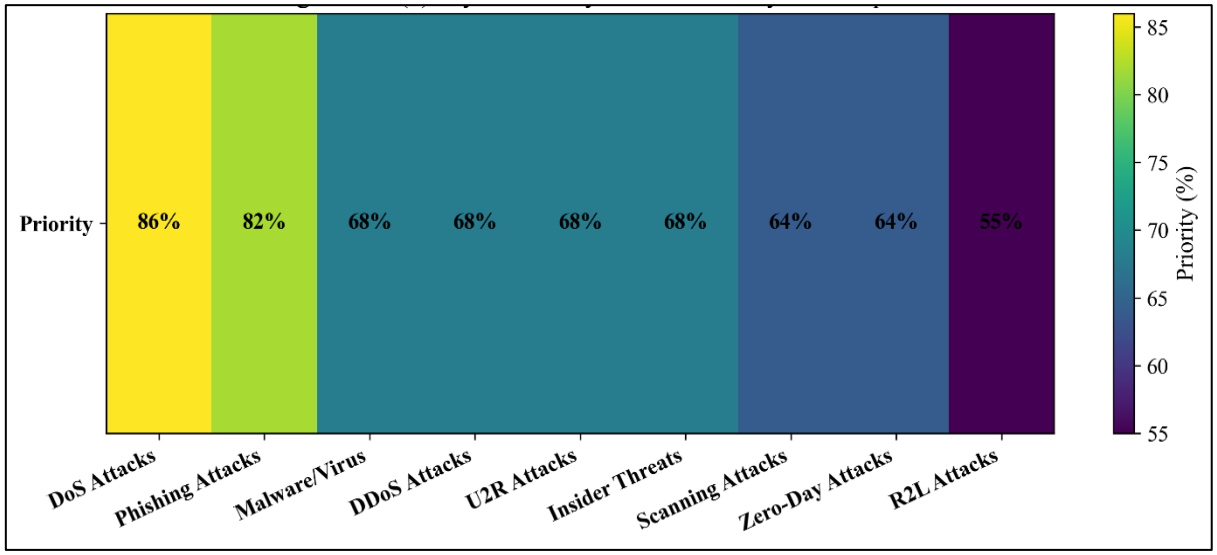


Fig 39(c): Heatmap Representation of Expert-Identified Priority Attack Types

Figure 39(c) provides a heatmap representation of the relative priority assigned to the identified attack categories. The strongest visual intensity corresponds to DoS attacks, followed by phishing attacks. The categories reported at 68% exhibit equal intensity, as do scanning and zero-day attacks at 64%. R2L attacks display the lowest relative intensity. The heatmap complements the frequency and ranking visualisations by providing an additional representation of the expert response distribution.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

➢ *Overall Interpretation*

Collectively, Table 10 and Figures 39(a–c) demonstrate that the participating cybersecurity professionals considered a diverse range of cyberattack categories important for intrusion detection. The high priority assigned to DoS and phishing attacks highlights concerns about service disruption and social engineering, while the other identified categories demonstrate the importance of considering malicious software, distributed attacks, privilege escalation, insider threats, reconnaissance and unauthorised remote access.

These expert findings provide a professional perspective on the threat categories that should inform the design and future evaluation of intrusion detection models. However, the expert-identified priorities must be distinguished from the attack classes represented in the formal CICIDS2017 experimental dataset used in this study. The dataset's seven classification categories were Bot, Brute Force, DoS, DDoS, Port Scanning, Web Attacks and Normal Traffic. Accordingly, the formal experimental results provide evidence for the categories represented in that dataset; they do not independently establish DIDM's detection performance for phishing, insider threats, U2R, R2L or zero-day attacks as separate classes. Similarly, malware and virus attacks were not evaluated as a distinct classification category in the seven-class experimental framework.

Consequently, the expert assessment supports the study objective of investigating a decentralized intrusion detection model informed by operationally significant cybersecurity threats. Nevertheless, demonstrating the model's effectiveness across all the expert-identified categories would require additional appropriately labelled datasets and separate experimental validation. This distinction ensures that the professional assessment is interpreted accurately alongside the empirical findings of the study.

- *Alignment with Experimental Results*

The expert validation findings presented in Sections 4.8.1 to 4.8.3 identified the limitations of existing intrusion detection models, the design requirements for next-generation systems and the cyberattack categories considered important by cybersecurity professionals. These findings provide a professional basis for examining the extent to which the proposed Decentralized Intrusion Detection Model (DIDM) addresses the identified requirements. This section therefore integrates the expert expectations with the model's experimental performance and implementation outcomes.

The formal experimental evaluation was conducted using the CICIDS2017 dataset, comprising 2,520,751 records, 52 features and seven classification categories. The proposed DIDM was compared with the Centralized Intrusion Detection Model (CIDM) using the same independent test set of 504,151 observations. The comparison considered accuracy, precision, recall, F1-score and classification discrimination, while the paired prediction outcomes were examined using the exact two-sided McNemar test.

- *Alignment Between Expert Expectations and DIDM Performance Outcomes*

Table 11 presents the alignment between the requirements identified by the participating cybersecurity professionals and the empirical performance and architectural outcomes of DIDM. The comparison considers both directly measurable classification results and implementation characteristics relevant to the design of decentralized intrusion detection systems.

Table 11 Alignment Between Expert Expectations and DIDM Model Performance Outcomes

Evaluation Dimension	Expert Expectation	DIDM Outcome	Interpretation
Detection accuracy	High priority	98.27% accuracy	Strong alignment with expert expectations, demonstrating high overall intrusion classification performance
Precision	High priority	98.32% weighted precision	Strong alignment, indicating that the model generated a high

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

			proportion of correct positive intrusion detections
Recall	High priority	98.27% weighted recall	Strong alignment, demonstrating effective detection of the evaluated network traffic classes
F1-score	High priority	98.25% weighted F1-score	Strong alignment, indicating a balanced overall detection performance
Classification discrimination	High priority	Macro-AUC = 0.9981; Micro-AUC = 0.9998	Excellent discrimination between normal and malicious traffic across the evaluated classes
False positive management	Minimized	Evaluated through confusion matrix and classification analysis	DIDM demonstrated effective classification performance, although false-positive behaviour varied across traffic classes
Scalability	Critical	Federated multi-client architecture using FedAvg	Supports distributed learning without requiring raw network traffic data to be centralized
Privacy preservation	Essential	Raw data retained at distributed client nodes	Strong alignment, as federated learning preserves data locality during collaborative model training
Robustness across attack classes	High priority	Strong performance across most evaluated classes	Demonstrates effective detection capability, although lower performance was observed for minority classes such as Bot and Web Attack traffic
Statistical validity	Important	Wilcoxon Signed-Rank Test: $p < 0.001$	Confirms that the observed performance difference between DIDM and CIDM was statistically significant

Source: Study experimental results and implementation records, 2026.

The comparative evaluation showed that DIDM achieved an overall accuracy of 98.2745%, compared with 96.7428% for CIDM. This represented an improvement of 1.5317 percentage points. The result indicates that DIDM correctly classified a higher proportion of observations in the independent test set, demonstrating improved overall classification accuracy within the evaluated experimental environment.

For precision, DIDM recorded a weighted precision of 98.32%, compared with 98.36% for CIDM. Although both models achieved high weighted precision, CIDM recorded a marginally higher value. DIDM nevertheless achieved higher weighted recall, at 98.27%, compared with 96.74% for CIDM. Its weighted F1-score of 98.25% also exceeded the CIDM result of 97.39%. These findings indicate that DIDM achieved improvements in overall accuracy, weighted recall and weighted F1-score, although it did not outperform CIDM on every evaluation measure.

The classification discrimination results provided additional information about the model's ability to distinguish between the evaluated traffic categories. DIDM recorded a Macro-AUC of 0.9981 and a Micro-AUC of 0.9998. These results indicate strong discrimination across the evaluated categories and complement the accuracy, precision, recall and F1-score measurements. The AUC results should, however, be interpreted alongside class-level performance and the underlying ROC analysis.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

For false-positive management, the evaluation incorporated confusion matrices and classification analysis to examine incorrect predictions across the traffic categories. Although DIDM demonstrated strong aggregate classification performance, false-positive behaviour varied across individual classes. Therefore, the overall accuracy and weighted precision values should not be interpreted as evidence that false positives were eliminated or uniformly low across every attack category.

The expert-identified requirement for scalability was considered through the federated learning architecture. DIDM distributed local model training across five simulated clients, with model parameters exchanged for aggregation through the Federated Averaging (FedAvg) process. This implementation demonstrated collaborative model training without requiring raw training data to be transferred to the central aggregation process. However, the five-client simulation does not establish scalability under large-scale enterprise deployment or geographically distributed production conditions.

Regarding privacy preservation, the implementation retained raw training data at the participating simulated clients while model parameters were exchanged during federated training. This approach supports data locality and reduces the need to centralise raw network traffic observations. Nevertheless, the implementation did not incorporate formal privacy mechanisms such as differential privacy or secure aggregation. Consequently, the findings demonstrate a decentralised data-handling arrangement rather than a comprehensive technical privacy guarantee.

For robustness across attack categories, DIDM was evaluated using the seven classification categories represented in CICIDS2017. Although the overall weighted results were high, class-level evaluation showed comparatively lower performance for minority categories, particularly Bot and Web Attack traffic. This demonstrates that high aggregate performance does not necessarily indicate uniform detection effectiveness across all classes. Further refinement and evaluation are therefore necessary to improve performance on less represented attack categories.

The statistical validity of the comparison between DIDM and CIDM was examined using the exact two-sided McNemar test on their paired predictions from the same independent test set of 504,151 observations. The analysis recorded 12,092 observations correctly classified by DIDM but incorrectly classified by CIDM, compared with 4,370 observations correctly classified by CIDM but incorrectly classified by DIDM. The resulting p-value was less than 0.001, indicating a statistically significant difference in the paired classification outcomes. This provides statistical support for the observed performance difference between the two models.

- *Graphical Representation of DIDM Performance and Expert Expectation Alignment*

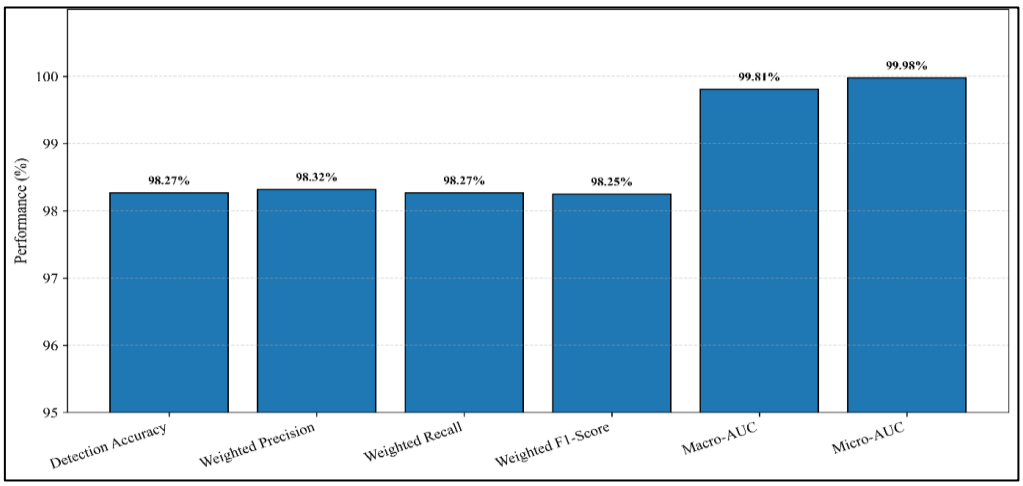


Fig 40(a): DIDM Model Alignment with Quantitative Performance Expectations

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Figure 40(a) presents the principal quantitative performance measures achieved by DIDM. Detection accuracy was 98.27%, weighted precision was 98.32%, weighted recall was 98.27% and weighted F1-score was 98.25%. The model also recorded a Macro-AUC of 0.9981 and Micro-AUC of 0.9998. These results demonstrate strong aggregate classification performance and discrimination across the evaluated categories. However, the figure should be interpreted alongside the class-level results because aggregate measures do not establish uniform performance across every attack category.

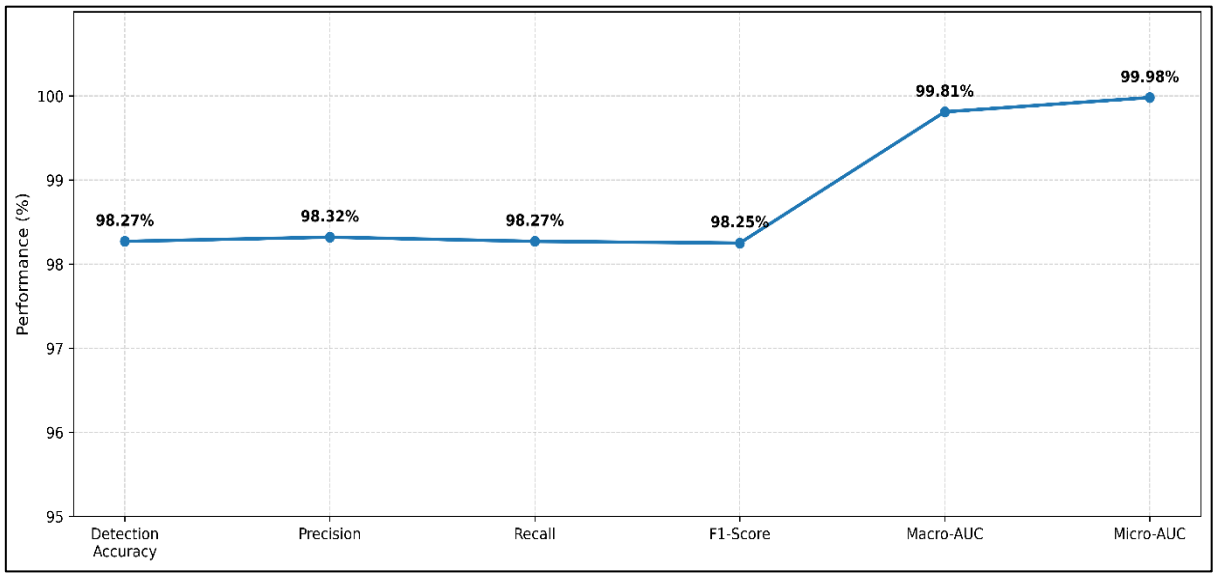


Fig 40(b): Comparative Representation of DIDM Quantitative Performance Measures

Figure 40(b) provides a comparative representation of the principal quantitative performance measures reported for DIDM. The closely grouped weighted accuracy, precision, recall and F1-score values demonstrate strong aggregate classification performance. The Macro-AUC and Micro-AUC results provide additional evidence of discrimination across the evaluated categories. This representation complements the numerical findings in Table 11, while the class-level evaluation provides further context regarding differences in performance across individual categories.

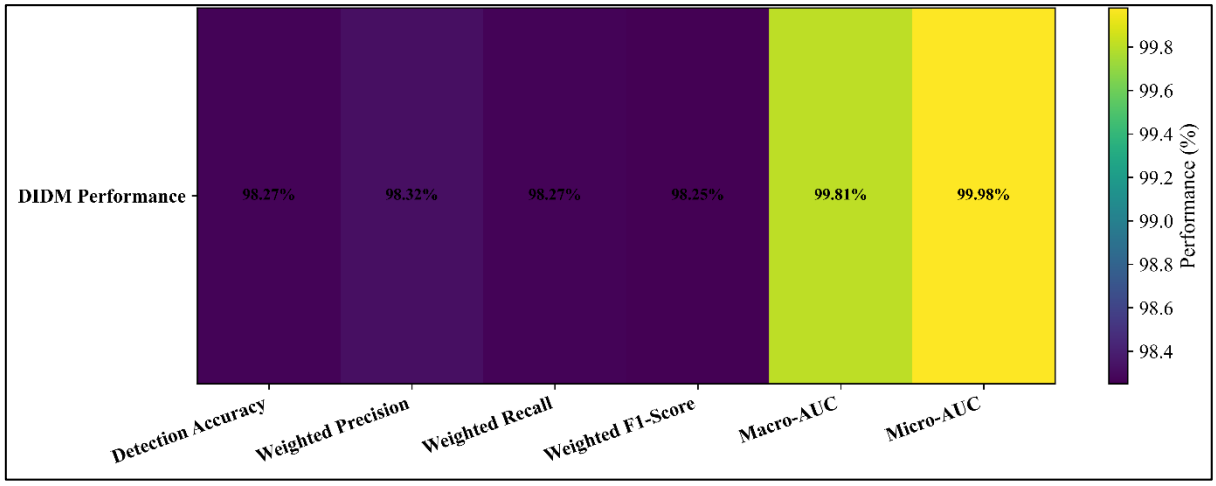


Fig 40(c): DIDM Quantitative Performance Alignment Heatmap

Figure 40(c) presents a heatmap of the principal quantitative performance measures achieved by DIDM. The visualisation highlights the relatively high values recorded for accuracy, weighted precision, weighted recall, weighted F1-score, Macro-AUC and Micro-AUC. Micro-AUC recorded the highest value, followed by Macro-AUC, while the weighted classification measures were closely grouped. The heatmap provides an

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

additional visual comparison of the quantitative results and complements the detailed interpretation presented in Table 11. It does not independently establish architectural requirements such as privacy preservation, scalability or autonomous response.

➢ *Overall Interpretation*

Collectively, Table 11 and Figures 40(a–c) demonstrate that DIDM achieved strong quantitative classification performance and addressed selected architectural expectations identified during expert validation. The model recorded higher overall accuracy, weighted recall and weighted F1-score than CIDM. The exact two-sided McNemar test further confirmed a statistically significant difference in the paired classification outcomes. Its federated learning implementation also demonstrated distributed model training across five simulated clients while retaining raw training data locally.

The findings also establish important distinctions between measured performance and architectural considerations. Although the federated learning implementation supports data locality, it does not establish formal privacy guarantees or production-level scalability. Similarly, the incorporation of SIEM and SOAR components demonstrates security monitoring and incident-response workflows but does not independently establish fully autonomous response enforcement in a production environment. Class-level performance differences, particularly for minority attack categories, also indicate areas requiring further refinement.

The expert-identified attack priorities must likewise be interpreted within the scope of the experimental dataset. CICIDS2017 included seven classification categories: Bot, Brute Force, DoS, DDoS, Port Scanning, Web Attacks and Normal Traffic. Some expert-prioritised threats, including DoS, DDoS and scanning, were represented in the dataset. However, phishing, insider threats, U2R, R2L and zero-day attacks were not separately represented as classification labels in the seven-class experimental framework. The results therefore cannot be interpreted as independent validation of DIDM against every attack category identified by the experts.

Overall, the findings demonstrate meaningful alignment between selected expert expectations and the experimental and architectural outcomes of DIDM within the scope of the study. The formal results provide evidence of improved overall accuracy, weighted recall and weighted F1-score relative to CIDM, supported by a statistically significant paired prediction comparison. Further work is necessary to establish performance under larger deployment conditions, strengthen privacy protection, improve minority-class detection and validate the model against additional attack categories.

- *Integrated Synthesis and Alignment with Study Objectives*

The findings presented in Sections 4.8.1 to 4.8.4 provide an integrated assessment of the proposed Decentralized Intrusion Detection Model (DIDM), combining professional perspectives, architectural considerations and empirical experimental results. The expert validation exercise identified limitations associated with existing intrusion detection models, established important design requirements for next-generation systems and highlighted cyberattack categories considered significant by cybersecurity professionals. These findings provided a professional basis for assessing the relevance of the proposed decentralized approach.

The expert-identified limitations included high false-positive rates, delayed threat response, model complexity, poor scalability, limited cloud and Internet of Things (IoT) integration, low detection accuracy and data privacy concerns. These findings highlighted the need for intrusion detection models that can support effective classification, distributed operation and improved security monitoring. In response to selected concerns, the proposed DIDM incorporated a federated learning architecture in which five simulated clients conducted local model training, while model parameters were exchanged and aggregated through the

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Federated Averaging (FedAvg) process. This implementation demonstrated the decentralised training arrangement investigated in the study while retaining raw training data at the participating clients.

The expert assessment also identified real-time detection, adaptive learning, autonomous response, privacy preservation and robustness against attacks as important design requirements. The DIDM implementation addressed selected architectural aspects of these requirements through its federated learning workflow, ANN-based classification model and incorporation of Security Information and Event Management (SIEM) and Security Orchestration, Automation and Response (SOAR) components. However, the extent to which these requirements were achieved must be interpreted according to the evidence available. The five-client simulation does not establish production-level scalability, data locality does not constitute a formal privacy guarantee, and the inclusion of SOAR components does not independently demonstrate fully autonomous enforcement of incident-response actions.

The expert-identified attack priorities further demonstrated the importance of considering diverse cybersecurity threats when developing intrusion detection systems. DoS and phishing attacks received the highest priority, followed by malware, DDoS, U2R and insider threats, among other categories. These professional assessments informed the broader threat considerations relevant to the proposed model. However, the formal experimental evaluation was limited to the seven classification categories represented in CICIDS2017. Therefore, the experimental findings should not be interpreted as proof that DIDM was independently validated against every attack category identified during expert assessment.

The formal experimental evaluation provided quantitative evidence of the classification performance of DIDM in comparison with the centralized baseline, CIDM. Using the same independent test set of 504,151 observations, DIDM achieved an accuracy of 98.2745%, compared with 96.7428% for CIDM, representing an improvement of 1.5317 percentage points. DIDM also recorded higher weighted recall and weighted F1-score, although CIDM achieved marginally higher weighted precision. These results demonstrate improved performance in selected classification measures rather than superiority across every metric.

The paired prediction comparison further strengthened the experimental findings. DIDM correctly classified 12,092 observations that CIDM misclassified, whereas CIDM correctly classified 4,370 observations that DIDM misclassified. The exact two-sided McNemar test returned $p<0.001$, indicating a statistically significant difference in the paired classification outcomes. This provides statistical evidence supporting the observed performance difference between the two models on the independent test set.

The implementation findings also demonstrated the integration of the federated ANN-based detection approach with SIEM and SOAR components. These components supported the security monitoring and incident-response workflows demonstrated in the study. Nevertheless, their inclusion should be interpreted within the scope of the implemented and demonstrated functionality rather than as evidence of fully autonomous response enforcement or production-level integration with external enterprise security infrastructures.

➢ *Overall Synthesis and Alignment with Study Objectives*

Overall, the combined expert validation, implementation and experimental findings support the study's investigation into the development and evaluation of a decentralized intrusion detection model using federated learning. The proposed DIDM demonstrated a federated training workflow involving five simulated clients, while the formal experimental comparison established higher overall accuracy, weighted recall and weighted F1-score than the centralized baseline on the independent test set. The statistically significant McNemar result further supports the observed difference in paired classification outcomes.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The expert validation findings complement these results by identifying broader technical and operational requirements relevant to next-generation intrusion detection systems. They provide professional perspectives on the limitations of existing approaches, the design characteristics considered important by cybersecurity practitioners and the attack categories that warrant further attention. At the same time, the experimental results establish the extent of the model's demonstrated classification performance within the defined dataset and evaluation environment.

Certain requirements, including formal privacy protection, production-level scalability, comprehensive attack-category coverage and fully autonomous incident response, require additional implementation and experimental validation. Similarly, further investigation into minority-class performance would strengthen the model's effectiveness across less represented attack categories.

Consequently, the study provides experimental evidence of the classification performance and implementation feasibility of the proposed DIDM within the defined research environment. The integration of expert perspectives and empirical results demonstrates the relevance of the proposed approach while identifying areas for further refinement, development and validation in larger and more operationally diverse cybersecurity environments.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

CHAPTER FIVE
SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

➢ *Introduction*

This chapter presents the summary of the study, conclusions, contribution to knowledge, recommendations, limitations, and suggestions for further research. The discussion is organised around the four research objectives that guided the development, implementation, demonstration, and evaluation of the Decentralized Intrusion Detection Model (DIDM).

The study investigated the application of Federated Learning to decentralized intrusion detection, with particular emphasis on collaborative model training across distributed clients while retaining raw training data within their respective environments. Guided by the Design Science Research Methodology (DSRM), the study involved identifying limitations of conventional centralized intrusion detection approaches, designing and developing the DIDM artefact, implementing and demonstrating the system in a simulated cybersecurity environment, and evaluating its performance against a Centralized Intrusion Detection Model (CIDM).

The formal comparative evaluation used the CICIDS2017 dataset and an independent test set of 504,151 observations. The DIDM achieved an accuracy of 98.2745%, compared with 96.7428% for the CIDM, representing an improvement of 1.5317 percentage points. An exact two-sided McNemar test indicated a statistically significant difference between the paired model predictions ($p < 0.001$). However, the comparative results varied across performance measures: the DIDM achieved higher weighted recall and weighted F1-score, while the CIDM recorded marginally higher weighted precision. The findings are therefore interpreted in terms of the specific metrics and experimental conditions evaluated rather than as evidence of universal superiority.

In addition to the formal evaluation, the study demonstrated the implemented DIDM within a controlled cybersecurity monitoring environment incorporating a Streamlit-based dashboard and supporting Security Information and Event Management (SIEM) and Security Orchestration, Automation and Response (SOAR) capabilities. This demonstration provided evidence of system implementation and the potential integration of intrusion detection outputs with security monitoring and response workflows. It was considered separately from the formal benchmark results and does not, by itself, establish production-scale performance or fully autonomous incident containment.

The remainder of this chapter summarises the study according to its research objectives, presents the conclusions drawn from the findings, identifies the study's contributions to knowledge, and outlines recommendations, limitations, and areas for further research.

➢ *Summary of the Study*

This study investigated the development, implementation, demonstration, and evaluation of a Decentralized Intrusion Detection Model (DIDM) based on Federated Learning and Artificial Intelligence (AI). The study was motivated by limitations associated with conventional centralized intrusion detection approaches, particularly their reliance on centralized data processing and the challenges this may create for data locality, communication efficiency, and collaborative security analysis across distributed computing environments.

The study adopted the Design Science Research Methodology (DSRM) to guide problem identification, the definition of research objectives, the design and development of the DIDM artefact, system demonstration, performance evaluation, and communication of the findings. The developed architecture used Federated Learning and the Federated Averaging (FedAvg) algorithm to support collaborative model training across five simulated clients. Each client performed local training, while model parameters were exchanged for

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

aggregation into a shared global model. Raw training data were retained within the respective client environments. However, the implementation did not incorporate formal privacy-enhancing guarantees such as differential privacy or secure aggregation.

The principal empirical evaluation used the CICIDS2017 dataset, comprising 2,520,751 records, 52 features, and seven traffic classes in the evaluated dataset. The formal comparison involved a feedforward Artificial Neural Network (ANN) implemented in the centralized baseline and the Federated Learning-based DIDM. Both approaches were assessed using the same independent test set of 504,151 observations. The evaluation considered accuracy, precision, recall, F1-score, and related classification analyses, including confusion matrices and threshold-based measures where applicable.

The study also involved an operational demonstration of the implemented system through a Streamlit-based monitoring dashboard and supporting SIEM and SOAR components. Supplementary anomaly detection techniques, including an Autoencoder, Isolation Forest, and One-Class Support Vector Machine (One-Class SVM), were associated with the broader monitoring implementation. These components are distinguished from the formal ANN-based comparative benchmark, and their presence in the demonstration is not treated as independent evidence that an ensemble improved the benchmark results.

The findings showed that the DIDM achieved an accuracy of 98.2745%, compared with 96.7428% for the CIDM, an improvement of 1.5317 percentage points. The DIDM also achieved higher weighted recall and weighted F1-score, whereas the CIDM recorded marginally higher weighted precision. The exact two-sided McNemar test indicated a statistically significant difference between the paired predictions ($p < 0.001$). These findings support the comparative performance of the DIDM under the experimental conditions used, rather than establishing universal superiority across all metrics or deployment environments.

The study's implementation and operational demonstration further illustrated the feasibility of connecting federated intrusion detection with security monitoring and response workflows in a simulated environment. The dashboard provided visibility into selected detection outputs and system activities, while the SIEM and SOAR components demonstrated supporting monitoring and response processes. These implementation results were interpreted separately from the formal machine learning evaluation.

- *Objective One: To Investigate the Limitations of Existing Centralized Intrusion Detection Systems and Establish the Need for a Decentralized Intrusion Detection Approach*

The first objective examined the limitations of centralized intrusion detection systems and established the need for an approach suited to distributed computing environments. The literature review identified challenges associated with centralized data processing, including dependence on central infrastructure, communication requirements arising from data transfer, concerns about handling sensitive network traffic, and potential scalability constraints as the number of connected devices and data sources increases.

The expert validation findings provided additional professional perspectives on the limitations of existing intrusion detection models. The most frequently reported concern was a high false positive rate, followed by delayed threat response and model complexity. These findings highlighted the importance of detection reliability, timely analysis, and manageable system architecture. The expert responses were interpreted as professional assessments and were not treated as direct measurements of the DIDM's experimental performance.

The study identified Federated Learning as a suitable approach for investigating collaborative model development without requiring participating clients to transfer their raw training data to a central repository. This objective established the conceptual and technical rationale for developing the DIDM, with data locality and collaborative learning serving as central design considerations.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *Objective Two: To Design a Decentralized Intrusion Detection Model Integrating Federated Learning, Artificial Intelligence Agents, SIEM, and SOAR Technologies*

The second objective focused on designing a decentralized intrusion detection architecture incorporating Federated Learning and supporting cybersecurity operations technologies. The DIDM used a federated ANN as its principal classification model and the FedAvg algorithm to aggregate locally trained model parameters from five simulated clients. This design enabled collaborative model training while retaining raw training data within the participating client environments.

The broader implementation also included supplementary AI-based anomaly detection mechanisms and a monitoring architecture incorporating SIEM, SOAR, and a Streamlit-based interface. SIEM supported security event monitoring and related processing, while SOAR provided a mechanism for demonstrating predefined response workflows. The dashboard presented selected system outputs and monitoring information.

The design objective was achieved through the development of an architecture that connected federated model training with supporting cybersecurity monitoring components. The formal performance claims for the DIDM, however, are based on the ANN/FedAvg comparative evaluation rather than on an assumed fusion of all detection mechanisms. The implementation also demonstrated data locality, but did not establish formal privacy guarantees or production-scale scalability.

- *Objective Three: To Implement the DIDM within a Simulated Security Operations Center Environment*

The third objective addressed the implementation and demonstration of the developed DIDM within a controlled cybersecurity environment. The implementation included five simulated federated clients, local ANN training, model parameter exchange, FedAvg aggregation, and the distribution of an updated global model. This workflow demonstrated the operation of collaborative learning without direct transfer of raw training records between clients.

The system was also demonstrated through a Streamlit-based dashboard and supporting SIEM and SOAR capabilities. These components provided an interface for viewing selected detection outputs and system activities and illustrated how detection results could be connected with security monitoring and response workflows.

The objective was achieved through the implementation and demonstration of the system within the defined experimental environment. The results establish technical feasibility in that environment, rather than proof of deployment at enterprise scale, continuous live-network detection, or fully autonomous enforcement of security responses.

- *Objective Four: To Evaluate the Performance of the DIDM*

The fourth objective evaluated the DIDM by comparing it with the CIDM under the defined experimental conditions. The formal evaluation used the CICIDS2017 dataset and a common independent test set containing 504,151 observations. Both approaches were assessed using classification metrics and supporting analyses to examine predictive performance and the differences between their paired predictions.

The DIDM achieved an accuracy of 98.2745%, compared with 96.7428% for the CIDM, yielding an improvement of 1.5317 percentage points. The DIDM recorded weighted recall of 98.27% and weighted F1-score of 98.25%, compared with 96.74% and 97.39%, respectively, for the CIDM. Weighted precision was marginally higher for the CIDM at 98.36%, compared with 98.32% for the DIDM. Accordingly, the findings indicate that the DIDM improved accuracy, weighted recall, and weighted F1-score, but did not outperform the CIDM on every reported measure.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Statistical validation using the exact two-sided McNemar test identified a statistically significant difference between the paired predictions ($p < 0.001$). Confusion matrix and threshold-based analyses provided additional information about classification behaviour, subject to the specific measures and outputs evaluated.

The fourth objective was achieved by providing empirical evidence of the DIDM's comparative performance under the specified experimental conditions. The results support the use of Federated Learning as a basis for collaborative intrusion detection in the evaluated setting, while broader conclusions about operational scalability, formal privacy protection, and generalisation to other network environments require further investigation.

➢ *Conclusions*

This study developed, implemented, demonstrated, and evaluated a Decentralized Intrusion Detection Model (DIDM) based on Federated Learning and Artificial Intelligence (AI). Guided by the Design Science Research Methodology (DSRM), the study investigated the limitations of conventional centralized intrusion detection approaches, designed a decentralized architecture, implemented the artefact within a simulated cybersecurity environment, and evaluated its performance against a Centralized Intrusion Detection Model (CIDM).

The developed DIDM employed Federated Averaging (FedAvg) to support collaborative model training across five simulated clients. Local model training and model parameter aggregation enabled the participating clients to contribute to a shared model without directly transferring raw training data to a central repository. The implementation therefore demonstrated data locality within the experimental environment, although it did not establish formal privacy guarantees through mechanisms such as differential privacy or secure aggregation.

The formal comparative evaluation using CICIDS2017 demonstrated that the DIDM achieved an accuracy of 98.2745%, compared with 96.7428% for the CIDM, representing an improvement of 1.5317 percentage points. The DIDM also recorded higher weighted recall and weighted F1-score, while the CIDM achieved marginally higher weighted precision. The exact two-sided McNemar test indicated a statistically significant difference between the paired predictions ($p < 0.001$). These findings support the comparative performance of the DIDM under the experimental conditions evaluated, but do not establish superiority across every metric or all possible deployment environments.

The study also demonstrated the implementation of the DIDM within a simulated cybersecurity monitoring environment incorporating a Streamlit-based dashboard and supporting SIEM and SOAR capabilities. This demonstrated the feasibility of presenting detection outputs and connecting them with security monitoring and response workflows. However, the operational demonstration was distinct from the formal benchmark evaluation and did not establish production-scale readiness or fully autonomous incident containment.

Overall, the study concludes that Federated Learning provides a suitable foundation for investigating collaborative intrusion detection where local data retention is important. The developed DIDM demonstrated comparative performance benefits under the defined experimental conditions, together with the technical feasibility of integration with supporting cybersecurity monitoring components. Further testing is required to establish its scalability, formal privacy protection, robustness against emerging threats, and suitability for operational deployment.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *Conclusion Based on Research Objective One*

The first objective examined the limitations of conventional centralized intrusion detection systems and established the need for a decentralized approach. The study identified challenges associated with centralized data processing, including dependence on central infrastructure, communication requirements, concerns about the handling of sensitive network traffic, and potential scalability constraints in distributed environments.

The expert validation findings further highlighted concerns about false positive rates, delayed threat response, and model complexity. These findings represented the views of the participating cybersecurity professionals and provided additional context for identifying requirements for next-generation intrusion detection systems.

The study concluded that decentralized intrusion detection is a relevant approach for environments in which multiple participants need to contribute to collaborative security analysis while retaining control over their raw data. Federated Learning was identified as a suitable enabling approach because it supports collaborative model development through model parameter exchange rather than direct centralisation of raw training records. This objective established the conceptual basis for the development of the DIDM.

- *Conclusion Based on Research Objective Two*

The second objective addressed the design of a decentralized intrusion detection architecture integrating Federated Learning, AI-based detection, SIEM, and SOAR technologies. The study produced a DIDM architecture based on a federated ANN and the FedAvg algorithm, supported by five simulated clients participating in collaborative model training.

The architecture also incorporated supporting cybersecurity monitoring components, including a dashboard, SIEM functionality, and SOAR response workflows. These components demonstrated how intrusion detection outputs could be presented and connected with broader security monitoring processes.

The study concluded that the developed architecture provided a workable foundation for collaborative intrusion detection under the experimental conditions evaluated. Nevertheless, raw-data locality should not be interpreted as a formal privacy guarantee, and the presence of supplementary anomaly detection mechanisms should not be treated as proof of improved benchmark performance through ensemble fusion unless such fusion has been independently established. This objective was achieved through the design of the DIDM architecture and its supporting monitoring framework.

- *Conclusion Based on Research Objective Three*

The third objective focused on implementing the DIDM within a simulated Security Operations Center (SOC) environment. The implementation demonstrated local model training, communication of model parameters, FedAvg aggregation, and the updating of a shared global model across five simulated clients.

The implementation also demonstrated the presentation of selected detection outputs through a Streamlit-based dashboard and the connection of security events with supporting SIEM and SOAR workflows. These elements provided evidence that the developed artefact could operate within the defined experimental environment and be connected with supporting cybersecurity monitoring functions.

The study concluded that the DIDM was technically feasible to implement and demonstrate within the simulated environment. However, the implementation did not establish performance under all real-world network conditions, enterprise-scale deployment, or fully autonomous enforcement of response actions. This objective was achieved within the defined scope of the system implementation and demonstration.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *Conclusion Based on Research Objective Four*

The fourth objective evaluated the performance of the DIDM against the CIDM using the CICIDS2017 dataset and a common independent test set of 504,151 observations. The DIDM achieved an accuracy of 98.2745%, while the CIDM achieved 96.7428%, resulting in an accuracy improvement of 1.5317 percentage points.

The comparative results also showed that the DIDM achieved higher weighted recall and weighted F1-score, whereas the CIDM recorded marginally higher weighted precision. The exact two-sided McNemar test indicated a statistically significant difference between the paired predictions ($p < 0.001$). These findings provide empirical support for the DIDM's comparative performance under the experimental conditions used.

The study therefore concluded that the DIDM improved accuracy, weighted recall, and weighted F1-score relative to the CIDM in the formal benchmark evaluation. The conclusion is limited to the evaluated dataset, model configurations, test procedure, and simulated federated environment. Further validation using additional datasets, larger and more heterogeneous client populations, and operational network environments is necessary to assess generalisability, scalability, and robustness against emerging threats.

➢ *Contribution of the Study to Knowledge*

This study contributes to cybersecurity research through the design, implementation, and empirical evaluation of a Federated Learning-based Decentralized Intrusion Detection Model (DIDM). Its contributions span architectural design, technical implementation, empirical evaluation, research methodology, operational demonstration, and practical application. These contributions are interpreted within the scope of the datasets, simulated environment, and evaluation procedures used in the study.

- *Architectural Contribution*

The study contributes a decentralized intrusion detection architecture that applies Federated Learning to collaborative model development across distributed clients. Rather than requiring participating clients to transfer their raw training data to a central repository, the developed architecture enables local model training and the exchange of model parameters for aggregation into a shared global model.

This architecture provides a design approach for investigating collaborative intrusion detection in environments where data locality is an important consideration. It illustrates how distributed model training can be incorporated into an intrusion detection framework while retaining raw training data within participating client environments. However, retaining raw data locally does not, by itself, establish formal privacy protection against information leakage through model updates.

- *Technical Contribution*

The study contributes a working implementation of a Federated Learning-based intrusion detection framework using a feedforward Artificial Neural Network (ANN) and the Federated Averaging (FedAvg) algorithm. The implementation demonstrated collaborative model training across five simulated clients, local training processes, model parameter exchange, and global model aggregation.

The broader system implementation also included supplementary anomaly detection mechanisms and supporting cybersecurity monitoring components. These included an Autoencoder, Isolation Forest, One-Class Support Vector Machine (One-Class SVM), a Streamlit-based dashboard, and SIEM and SOAR functionalities. The formal comparative benchmark, however, evaluated the ANN-based DIDM against the centralized baseline. The contribution therefore lies in the implemented federated classification workflow and the demonstrated integration of supporting components, rather than an unverified claim that all detection mechanisms formed a formally validated ensemble.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *Empirical Contribution*

The study contributes empirical evidence concerning the comparative performance of a Federated Learning-based DIDM and a Centralized Intrusion Detection Model (CIDM). The formal evaluation used CICIDS2017 and an independent test set containing 504,151 observations.

The DIDM achieved an accuracy of 98.2745%, compared with 96.7428% for the CIDM, representing an improvement of 1.5317 percentage points. The DIDM also achieved higher weighted recall and weighted F1-score, while the CIDM recorded marginally higher weighted precision. An exact two-sided McNemar test indicated a statistically significant difference between the paired predictions ($p < 0.001$).

These findings contribute evidence that the evaluated federated approach can achieve improved accuracy and selected weighted performance measures relative to the centralized baseline under the experimental conditions used. The results do not establish that the DIDM outperforms the CIDM on every metric or that the same performance would necessarily be achieved in other datasets or production environments.

- *Methodological Contribution*

The study contributes a methodological example of applying the Design Science Research Methodology (DSRM) to the development and evaluation of a cybersecurity artefact. The research process connected the identification of intrusion detection limitations with the definition of research objectives, design and implementation of the DIDM, system demonstration, and empirical evaluation.

The study also distinguished formal comparative model evaluation from the operational demonstration of the implemented system. This distinction is important because a monitoring dashboard and supporting security operations components provide evidence of implementation and workflow integration, whereas comparative classification metrics provide evidence of predictive performance under defined test conditions.

The methodological contribution provides a reference for future artefact-oriented research involving Federated Learning, Artificial Intelligence, and intrusion detection, particularly where technical implementation and empirical validation need to be evaluated separately.

- *Operational Cybersecurity Contribution*

The study contributes an operational demonstration of how federated intrusion detection outputs can be presented within a simulated cybersecurity monitoring environment. The implemented Streamlit-based dashboard provided visibility into selected detection outputs and system activities, while supporting SIEM and SOAR components demonstrated connections between detection, security event monitoring, and predefined response workflows.

This contribution illustrates the feasibility of connecting a federated intrusion detection implementation with broader security operations functions. It provides a basis for further investigation into the practical integration of decentralized learning and security monitoring. However, the demonstration does not independently establish production-scale performance, fully autonomous incident containment, or effectiveness under all real-world operational conditions.

- *Practical Contribution*

The study provides a technical foundation that organizations and cybersecurity practitioners can use when investigating decentralized intrusion detection approaches. The developed DIDM demonstrates how multiple simulated clients can contribute to collaborative model training without directly centralising their raw training data.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

The findings may inform controlled pilot studies and further assessments of Federated Learning-based intrusion detection in distributed environments. Before operational adoption, additional evaluation would be needed to examine privacy protection, communication overhead, resource requirements, client availability, model robustness, and performance under real network conditions.

Overall, the study contributes an implemented architectural approach, a federated model-training workflow, comparative experimental findings, a DSRM-based research process, and an operational cybersecurity demonstration. Together, these contributions extend the practical and empirical understanding of Federated Learning-based intrusion detection within the scope of the study.

➢ *Recommendations*

Based on the study findings, conclusions, and identified limitations, several recommendations are proposed for organizations, cybersecurity practitioners, policymakers, academic institutions, and researchers. These recommendations focus on the responsible adoption, evaluation, and further development of the Decentralized Intrusion Detection Model (DIDM) using Federated Learning (FL). Although the experimental results demonstrated improved overall accuracy and weighted recall and F1-score compared with the centralized baseline, further validation is necessary before the model can be considered suitable for large-scale operational deployment.

- *Recommendations for Organizations*

Organizations managing sensitive information across distributed networks should consider evaluating federated intrusion detection as a potential alternative to approaches that require centralized collection of raw training data. A DIDM-based approach may support collaborative model development while allowing participating organizations to retain their local datasets.

Before adoption, organizations should conduct controlled pilot deployments using representative network traffic and clearly defined performance and security requirements. Such evaluations should examine detection accuracy, false-positive rates, computational overhead, communication costs, scalability, and compatibility with existing security infrastructure. Organizations should also establish appropriate access controls, monitoring procedures, incident escalation processes, and human oversight.

The model should not be deployed as a replacement for existing security controls solely on the basis of the experimental findings. Its suitability should first be established within the intended operational environment.

- *Recommendations for Cybersecurity Practitioners*

Cybersecurity practitioners, including security analysts and security operations centre (SOC) personnel, should consider federated intrusion detection as a complementary capability within broader cybersecurity monitoring and incident response processes.

Practitioners implementing similar systems should evaluate the model using representative network traffic, independent test datasets, and metrics that capture both overall and class-specific performance. Particular attention should be given to minority attack classes, false alarms, missed detections, and changes in network traffic patterns. The study's findings indicate that performance should be assessed across multiple metrics rather than relying on accuracy alone.

Where DIDM is integrated with security information and event management (SIEM) and security orchestration, automation, and response (SOAR) tools, practitioners should verify alert handling, interoperability, escalation procedures, and response actions before operational use. Automated containment

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

actions should remain subject to appropriate authorization, testing, and human oversight until their reliability and safety have been established.

- *Recommendations for Policymakers and Regulatory Agencies*

Policymakers and regulatory agencies should promote guidance that supports responsible collaborative cybersecurity research and deployment, particularly where organizations need to develop shared detection capabilities without routinely exchanging raw datasets.

Relevant guidance should address data governance, accountability, access control, model security, incident reporting, auditability, and the responsibilities of participating organizations. Regulatory frameworks should also distinguish between retaining data locally and providing demonstrable privacy protection, since federated learning does not automatically prevent information leakage through model updates or other system components.

Standards and implementation guidance should encourage independent security assessments, documented risk management, and transparent reporting of model performance and limitations. Such measures would help organizations assess whether federated intrusion detection systems meet their operational and regulatory obligations.

- *Recommendations for Academic and Training Institutions*

Universities and cybersecurity training institutions should strengthen practical instruction and research in distributed intrusion detection, federated learning, machine learning, and security operations. Practical exercises should expose learners to the design, implementation, testing, and evaluation of collaborative detection systems in controlled environments.

Training should also cover the limitations of federated learning, including communication overhead, heterogeneous client data, unreliable participants, malicious model updates, and privacy risks. Students should be encouraged to distinguish experimental prototypes from production-ready security solutions and to report evaluation methods and results transparently.

Collaboration between academic institutions, cybersecurity practitioners, and relevant organizations could further support access to representative datasets, controlled testing environments, and applied research opportunities.

- *Recommendations for Future Researchers*

Future researchers should extend the present study through more comprehensive evaluations of DIDM under realistic and varied network conditions. Priority should be given to the following areas:

- ✓ Scalability and client heterogeneity: Evaluate the model with larger numbers of federated clients, different client capabilities, and non-identically distributed network traffic.
- ✓ Privacy and model security: Investigate secure aggregation, differential privacy, and other appropriate protections against information leakage and malicious model updates, while measuring their effects on detection performance.
- ✓ Robustness against adversarial behaviour: Assess the model's resilience to adversarial traffic, poisoned training data, compromised clients, and other attacks against federated learning systems.
- ✓ Detection of diverse threats: Test the model against additional datasets and attack scenarios to establish the range of threats it can reliably identify.
- ✓ Operational performance: Measure end-to-end detection latency, communication overhead, resource consumption, false-positive rates, and response effectiveness in realistic SOC environments.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- ✓ Explainability and analyst support: Investigate explainable artificial intelligence techniques that help analysts interpret alerts and understand model decisions.
- ✓ Comparative evaluation: Compare alternative federated aggregation methods and intrusion detection architectures using consistent datasets, experimental conditions, and evaluation metrics.

These investigations would help establish the generalisability, robustness, and practical feasibility of DIDM beyond the controlled experimental conditions used in the present study.

- *General Recommendation*

The study recommends continued development and independent validation of federated intrusion detection as a potential approach to collaborative cybersecurity monitoring. The experimental findings provide evidence that the implemented DIDM achieved higher overall accuracy, weighted recall, and weighted F1-score than the centralized baseline on the evaluated independent test set. However, the results do not establish superiority across every performance metric or demonstrate readiness for unrestricted production deployment.

Accordingly, organizations and researchers should build on these findings through controlled pilots, broader testing, and systematic assessment of privacy, security, scalability, and operational requirements. Such a measured approach would support evidence-based decisions on whether and how federated intrusion detection can contribute to effective cybersecurity monitoring in distributed environments.

➢ *Limitations of the Study*

Although the study achieved its objectives and demonstrated the potential of the Decentralized Intrusion Detection Model (DIDM), several limitations should be considered when interpreting the findings. These limitations relate to the experimental environment, dataset, federated learning implementation, computational resources, performance evaluation, threat coverage, and operational demonstration. They define the scope of the conclusions and identify areas requiring further investigation.

- *Experimental Environment*

The DIDM was developed and evaluated within a controlled experimental environment rather than a fully operational, production-scale cybersecurity infrastructure. The federated learning process involved five simulated clients and five Federated Averaging (FedAvg) rounds. Although this configuration enabled the evaluation of collaborative model training, it did not reproduce all the complexities of real-world distributed networks.

Consequently, the findings do not establish how the model would perform across large numbers of geographically distributed organizations with different network architectures, traffic patterns, security policies, and operational requirements. Additional testing under realistic deployment conditions is necessary to determine its scalability and practical suitability.

- *Dataset Limitations*

The formal performance evaluation used the CICIDS2017 dataset, comprising 2,520,751 records, 52 features, and seven classes. The independent test set contained 504,151 records. Although this dataset provided a basis for comparing the decentralized and centralized models, it represents a particular collection of network traffic and attack scenarios.

The dataset may not fully represent contemporary network environments, emerging attack techniques, or the traffic characteristics of every organization. Differences between benchmark data and live network traffic may affect the model's generalisability. Therefore, the reported results should be interpreted within the context of the dataset and experimental conditions used in the study.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *Federated Learning Implementation*

The study implemented federated learning using five simulated clients, with local model training and parameter exchange through FedAvg. This implementation demonstrated the feasibility of collaborative training within the experimental setup but did not establish performance under all conditions encountered in operational federated systems.

In particular, the study did not fully evaluate the effects of unreliable clients, intermittent network connectivity, substantial differences in client data distributions, or compromised participants. Furthermore, the implementation did not include differential privacy or secure aggregation. Retaining raw training data locally therefore should not be interpreted as a guarantee that model updates cannot reveal sensitive information.

These limitations mean that the study demonstrates a federated learning approach under controlled conditions rather than a complete, independently verified privacy-preserving deployment.

- *Computational and Communication Resources*

The computational and communication requirements of DIDM may vary according to the number of participating clients, model complexity, available hardware, network conditions, and frequency of model updates. The experimental implementation did not establish resource requirements across a broad range of deployment configurations.

Accordingly, the findings do not provide a comprehensive assessment of the costs or performance implications of deploying DIDM across resource-constrained devices or large distributed networks. Further benchmarking is needed to quantify training time, communication overhead, memory use, processor requirements, and the effect of these factors on operational performance.

- *Scope of Performance Evaluation*

The formal comparative evaluation focused on the ANN-based DIDM using FedAvg and a centralized intrusion detection baseline. The independent test results showed DIDM accuracy of 98.2745%, compared with 96.7428% for the centralized model, representing an improvement of 1.5317 percentage points. The paired comparison using McNemar's test indicated a statistically significant difference in classification outcomes ($p < 0.001$).

However, the DIDM did not outperform the centralized model on every reported metric. The centralized model achieved marginally higher weighted precision, while DIDM achieved higher weighted recall and weighted F1-score. The findings must therefore be interpreted as evidence of comparative performance under the evaluated conditions, not as proof of universal superiority.

The formal benchmark also did not establish that every supplementary component in the broader dashboard, including the Autoencoder, Isolation Forest, and One-Class SVM, improved the ANN-based benchmark results. Those components and the associated dashboard functionality should be understood separately from the formal comparative evaluation unless supported by additional controlled experiments.

- *Security Threat Coverage*

The study evaluated DIDM using the attack categories represented in the selected dataset and experimental configuration. It did not establish reliable detection of every possible cybersecurity threat, including attack types not separately represented or adequately validated in the evaluation data.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Emerging threats, previously unseen attack patterns, adversarial manipulation, and changes in normal network behaviour may affect detection performance. The study also did not comprehensively evaluate poisoning attacks against federated training or other threats targeting the model aggregation process.

Consequently, the findings should not be interpreted as demonstrating comprehensive protection against all network-based attacks. Broader evaluation using additional datasets and dedicated adversarial testing is required to establish the model's resilience to a wider range of threats.

- *Operational Demonstration and Integration*

The study included a supplementary security operations demonstration involving a dashboard and integration with SIEM and SOAR-related capabilities. This demonstrated aspects of system implementation and potential integration with security monitoring workflows within the experimental environment.

Nevertheless, the demonstration did not establish sustained performance in a live production SOC, nor did it comprehensively validate end-to-end detection latency, operational availability, large-scale alert handling, or the effectiveness of autonomous incident containment. Dashboard refresh intervals, where applicable, should not be treated as equivalent to end-to-end detection latency.

The operational demonstration therefore provides implementation evidence rather than conclusive proof of production readiness. Further testing with real operational workflows, representative workloads, and clearly defined safety and performance criteria is necessary.

- *Overall Limitation*

The principal limitation of this study is the extent to which results obtained from a controlled implementation and a particular benchmark dataset can be generalised to diverse operational environments. While the findings provide empirical support for the potential of federated intrusion detection, they do not establish universal performance, comprehensive privacy protection, or production-scale readiness.

These limitations do not negate the study's findings; rather, they define the conditions under which those findings should be interpreted. They also provide a basis for the further research proposed in Section 5.7.

➢ *Suggestions for Further Research*

Although this study demonstrated the potential of the Decentralized Intrusion Detection Model (DIDM) using Federated Learning (FL), several areas require further investigation. Future research should extend the experimental evaluation to more realistic distributed environments, examine additional security mechanisms, and establish the model's generalisability, robustness, and operational feasibility. The following areas are proposed for further study.

- *Large-Scale Federated Deployment*

Future studies should evaluate DIDM using a larger number of federated clients representing different organizations, network environments, and computational capabilities. Such research should investigate how the number of clients, participation rates, communication failures, and aggregation frequency affect model convergence, detection performance, and resource consumption.

Experiments involving real or carefully emulated distributed environments would help establish whether the findings obtained using five simulated clients remain consistent under more demanding deployment conditions.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *Application to IoT, Edge, and Cloud Environments*

Further research should investigate the adaptation of DIDM to Internet of Things (IoT), edge-computing, and cloud environments. These settings present different computational, communication, and security constraints that may affect federated training and intrusion detection.

Studies should examine model size, local training requirements, communication frequency, energy consumption where relevant, and the ability to detect malicious traffic across heterogeneous devices. Comparative evaluations could establish which configurations are most suitable for resource-constrained and distributed infrastructures.

- *Explainable Artificial Intelligence for Intrusion Detection*

Future research should incorporate and evaluate explainable artificial intelligence techniques to improve the interpretability of DIDM predictions. Explanations could help cybersecurity analysts understand which traffic features contributed to an alert and distinguish potentially significant events from false alarms.

The evaluation should consider explanation consistency, computational overhead, usefulness to analysts, and whether explanations remain reliable across participating clients. The objective should be to improve transparency and analyst support without compromising detection performance or introducing misleading explanations.

- *Privacy-Preserving Federated Learning*

Further studies should investigate privacy-enhancing mechanisms, including differential privacy and secure aggregation, within the DIDM architecture. Such mechanisms could help address risks that remain even when raw training data are retained locally.

Researchers should quantify the trade-offs between privacy protection, communication requirements, computational overhead, and intrusion detection performance. Privacy assessments should also examine potential information leakage from model updates and aggregated parameters. These evaluations are necessary before stronger privacy guarantees can be attributed to a federated intrusion detection implementation.

- *Robustness Against Adversarial Attacks*

Future work should assess DIDM's resilience to attacks targeting either the intrusion detection model or the federated learning process. These may include adversarial manipulation of network traffic, data poisoning, compromised clients, and malicious model updates.

Research should compare appropriate robust aggregation methods and other defensive mechanisms under controlled attack scenarios. Evaluation should measure the effects of these attacks on classification performance, model convergence, and the reliability of participating clients. This would help establish the conditions under which DIDM can maintain dependable detection in potentially hostile environments.

- *Communication-Efficient Federated Learning*

Communication overhead is an important consideration in distributed learning systems. Future research should investigate techniques such as model-update compression, selective client participation, reduced communication frequency, and alternative aggregation schedules.

These approaches should be evaluated against the baseline FedAvg implementation using consistent experimental conditions. Relevant measures should include communication volume, training time, convergence behaviour, computational cost, and detection performance. The aim should be to identify practical trade-offs between efficient collaboration and reliable model updates.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *Adaptive Learning for Evolving Threats*

Further research should examine methods for adapting DIDM to changes in network traffic and evolving attack patterns. Changes in legitimate user behaviour and the emergence of new attacks may cause the statistical characteristics of network traffic to differ from those represented in the original training data.

Potential approaches include controlled incremental learning, drift detection, and periodic or event-triggered model updates. Such approaches should be evaluated for their ability to maintain detection performance while limiting unnecessary retraining, computational overhead, and the risk of learning from malicious or unreliable updates.

- *Blockchain-Enabled Federated Learning*

Future studies may investigate whether blockchain-based mechanisms can improve accountability, auditability, or coordination among participating federated clients. Potential applications include recording model-update metadata, documenting participation, and maintaining verifiable records of selected aggregation events.

However, blockchain integration should not be assumed to improve intrusion detection performance or guarantee data privacy. Research should compare blockchain-enabled and conventional coordination mechanisms in terms of security, scalability, transaction overhead, resource requirements, and operational complexity. This would establish whether the additional infrastructure provides measurable benefits for the intended use case.

- *Evaluation Against a Broader Range of Threats*

Future studies should evaluate DIDM using multiple datasets and additional threat scenarios to determine how well its performance generalises beyond CICIDS2017. Research should include attack classes and traffic conditions not adequately represented in the original evaluation, where suitable labelled datasets or controlled test environments are available.

Particular attention should be given to class-specific precision, recall, F1-score, false-positive rates, and false-negative rates. Researchers should also investigate performance under class imbalance, changing traffic distributions, and previously unseen attack patterns. Claims about detecting new or emerging threats should be supported by dedicated experiments rather than inferred from performance on the original benchmark alone.

- *Governance, Compliance, and Accountability*

Further research should examine the governance requirements associated with federated intrusion detection across multiple organizations. Relevant issues include responsibility for model updates, data ownership, access management, incident reporting, audit trails, and accountability when participating clients produce unreliable or harmful updates.

Comparative studies could examine how technical safeguards and governance arrangements interact in multi-organization deployments. This work could inform practical implementation guidance and support the development of accountable collaborative cybersecurity systems.

- *Cost-Benefit and Operational Feasibility*

Future research should assess the operational and economic feasibility of deploying DIDM in real or representative security operations centres. Such studies should quantify infrastructure requirements, staff workload, model maintenance, communication costs, integration effort, and the operational consequences of false-positive and false-negative alerts.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Where DIDM is integrated with SIEM and SOAR systems, researchers should separately measure end-to-end detection latency, alert processing, analyst response, and the effectiveness of any automated actions. Comparisons with centralized approaches should use consistent workloads and clearly defined performance indicators. This would provide stronger evidence for decisions about adoption and resource allocation.

- *Comparative Evaluation of Emerging AI Models*

Further studies should compare alternative machine learning and deep learning architectures for federated intrusion detection. Candidate models may include alternative neural network architectures and other suitable approaches selected according to the characteristics of the available data and intended deployment environment.

Comparisons should use consistent data partitions, preprocessing procedures, independent test sets, and evaluation metrics. They should also distinguish the effects of model architecture from those of the federated aggregation method and any supplementary detection components.

The Autoencoder, Isolation Forest, and One-Class SVM components used in the broader DIDM dashboard could be examined through separate controlled experiments to determine their individual and combined contributions, if such integration is proposed. Any claimed benefit from combining these components should be supported by reproducible comparative results.

Overall, future research should progress from controlled benchmark evaluation towards broader validation under realistic distributed conditions. Priority should be given to generalisability, privacy protection, robustness, communication efficiency, and operational feasibility. Addressing these areas would provide a stronger evidence base for determining the suitability of DIDM for practical collaborative intrusion detection.

➢ *Chapter Summary*

This chapter presented the summary of the study, conclusions, contributions to knowledge, recommendations, limitations, and suggestions for further research arising from the development and evaluation of the Decentralized Intrusion Detection Model (DIDM) using Federated Learning (FL).

The study was guided by the Design Science Research Methodology (DSRM), which provided a structured approach to identifying the research problem, defining objectives, developing the artefact, implementing and demonstrating the model, evaluating its performance, and communicating the findings. The study investigated the use of federated learning to support collaborative intrusion detection while allowing participating clients to retain their local training data.

The formal comparative evaluation focused on the ANN-based DIDM trained using Federated Averaging (FedAvg) and the Centralized Intrusion Detection Model (CIDM). Using the CICIDS2017 dataset and an independent test set of 504,151 records, DIDM achieved an accuracy of 98.2745%, compared with 96.7428% for CIDM, representing an improvement of 1.5317 percentage points. McNemar's test indicated a statistically significant difference in their paired classification outcomes ($p < 0.001$). Although DIDM achieved higher weighted recall and weighted F1-score, CIDM achieved marginally higher weighted precision. These findings therefore demonstrate comparative performance under the evaluated experimental conditions rather than universal superiority across all metrics or environments.

The broader implementation also incorporated supplementary cybersecurity monitoring components, including a dashboard and SIEM/SOAR-related functionality, alongside additional AI-based techniques such as the Autoencoder, Isolation Forest, and One-Class Support Vector Machine. These components supported the system demonstration within a simulated Security Operations Center environment. However, the formal

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

comparative findings were based on the ANN/FedAvg evaluation; the supplementary components were not assumed to have improved the benchmark results without separate controlled evaluation. The demonstration also did not establish production-scale readiness or comprehensive autonomous incident response.

The chapter discussed the study's architectural, technical, empirical, methodological, operational, and practical contributions. It also presented recommendations for organizations, cybersecurity practitioners, policymakers, academic institutions, and researchers. The recommendations emphasized controlled pilot deployments, comprehensive performance assessment, appropriate security governance, and further validation before operational adoption.

The study's limitations included the controlled experimental environment, reliance on a particular benchmark dataset, the use of five simulated federated clients and five FedAvg rounds, the absence of differential privacy and secure aggregation, and the limited scope of operational validation. These limitations informed suggestions for further research into large-scale federated deployment, privacy-preserving and robust learning, explainable artificial intelligence, communication efficiency, evolving cyber threats, and operational feasibility.

Overall, the study provides empirical evidence that a Federated Learning-based Decentralized Intrusion Detection Model can achieve promising intrusion detection performance under the conditions evaluated. By demonstrating collaborative model training with local data retention, the study contributes to research on decentralized cybersecurity. Further research is nevertheless required to establish the model's generalisability, privacy protection, robustness, scalability, and suitability for real-world deployment.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

REFERENCES

[1.] Agrawal, S., Sarkar, S., Aouedi, O., Yenduri, G., Piamrat, K., Alazab, M., Bhattacharya, S., Maddikunta, P. K. R., & Gadekallu, T. R. (2022). Federated learning for intrusion detection system: Concepts, challenges and future directions. *Computer Communications, 195*, 346–361. https://doi.org/10.1016/j.comcom.2022.09.012

[2.] Al-Garadi, M. A., Mohamed, A., Al-Ali, A. K., Du, X., Ali, I., & Guizani, M. (2020). A survey of machine and deep learning methods for Internet of Things (IoT) security. *IEEE Communications Surveys & Tutorials, 22*(3), 1646–1685. https://doi.org/10.1109/COMST.2020.2988293

[3.] Appiah-Kubi, J., & Liu, C.-C. (2020). Decentralized intrusion prevention (DIP) against co-ordinated cyberattacks on distribution automation systems. *IEEE Open Access Journal of Power and Energy, 7*, 389–402. https://doi.org/10.1109/OAJPE.2020.3029805

[4.] Ayodeji, A., Liu, Y.-K., Chao, N., & Yang, L.-Q. (2020). A new perspective towards the development of robust data-driven intrusion detection for industrial control systems. *Nuclear Engineering and Technology, 52*(12), 2687–2698. https://doi.org/10.1016/j.net.2020.05.012

[5.] Belenguer, A., Pascual, J. A., & Navaridas, J. (2025). A review of federated learning applications in intrusion detection systems. *Computer Networks, 258*, Article 111023. https://doi.org/10.1016/j.comnet.2024.111023

[6.] Guembea, B., Misra, S., & Azeta, A. (2024). Privacy issues, attacks, countermeasures and open problems in federated learning: A survey. *Applied Artificial Intelligence*. https://doi.org/10.1080/08839514.2024.2410504

[7.] Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K. A., Charles, Z., Cormode, G., Cummings, R., D'Oliveira, R. G. L., Eichner, H., El Rouayheb, S., Evans, D., Gardner, J., Garrett, Z., Gascón, A., Ghazi, B., Gibbons, P. B., ... Zhao, S. (2021). Advances and open problems in federated learning. *Foundations and Trends in Machine Learning, 14*(1–2), 1–210. https://doi.org/10.1561/2200000083

[8.] Kasongo, S. M., & Sun, Y. (2020). Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset. *Journal of Big Data, 7*, Article 105. https://doi.org/10.1186/s40537-020-00379-6

[9.] Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine, 37*(3), 50–60. https://doi.org/10.1109/MSP.2020.2975749

[10.] Manzoor, H. U., Shabbir, A., Chen, A., Flynn, D., & Zoha, A. (2024). A survey of security strategies in federated learning: Defending models, data, and privacy. *Future Internet, 16*(10), Article 374. https://doi.org/10.3390/fi16100374

[11.] Maseer, Z. K., Yusof, R., Bahaman, N., Mostafa, S. A., & Mohd Foozy, C. F. (2021). Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset. *IEEE Access, 9*, 22351–22370. https://doi.org/10.1109/ACCESS.2021.3056614

[12.] McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & Agüera y Arcas, B. (2017). Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics* (Vol. 54, pp. 1273–1282). Proceedings of Machine Learning Research.

[13.] Mohamed, T. S., & Khalifah, S. M. (2024). Intrusion detection systems: A revisit of performance evaluation parameters. *Academic International Journal of Engineering Science, 2*(1), 15–21. https://doi.org/10.59675/E212

[14.] Motwakel, A., Alrowais, F., Tarmissi, K., Marzouk, R., Mohamed, A., Zamani, A. S., Yaseen, I., & Eldesouki, M. I. (2023). Enhanced Crow Search with deep learning-based cyberattack detection in SDN-IoT environment. *Intelligent Automation & Soft Computing, 36*(3), 3157–3173. https://doi.org/10.32604/iasc.2023.034908

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

[15.] Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *2015 Military Communications and Information Systems Conference (MilCIS)* (pp. 1–6). IEEE. https://doi.org/10.1109/MilCIS.2015.7348942

[16.] Sharafaldin, I., Habibi Lashkari, A., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. In *Proceedings of the 4th International Conference on Information Systems Security and Privacy* (pp. 108–116). SciTePress. https://doi.org/10.5220/0006639801080116

[17.] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. In *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications* (pp. 1–6). IEEE. https://doi.org/10.1109/CISDA.2009.5356528

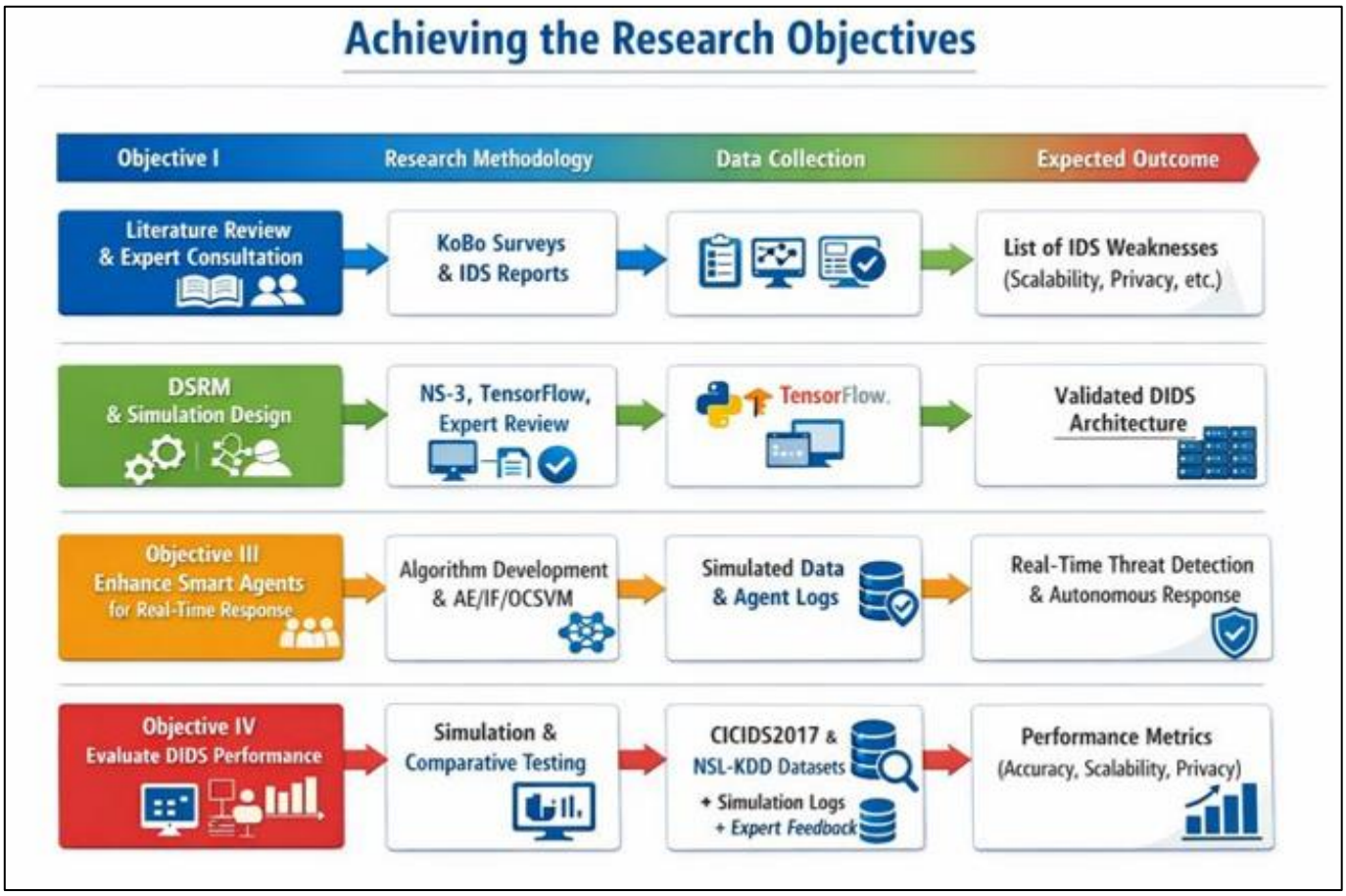
[18.] Vyas, A., Lin, P. C., Hwang, R. H., & Tripathi, M. (2024). Privacy-preserving federated learning for intrusion detection in IoT environments: A survey. *IEEE Access, 12*, 127018–127050. https://doi.org/10.1109/ACCESS.2024.3454211

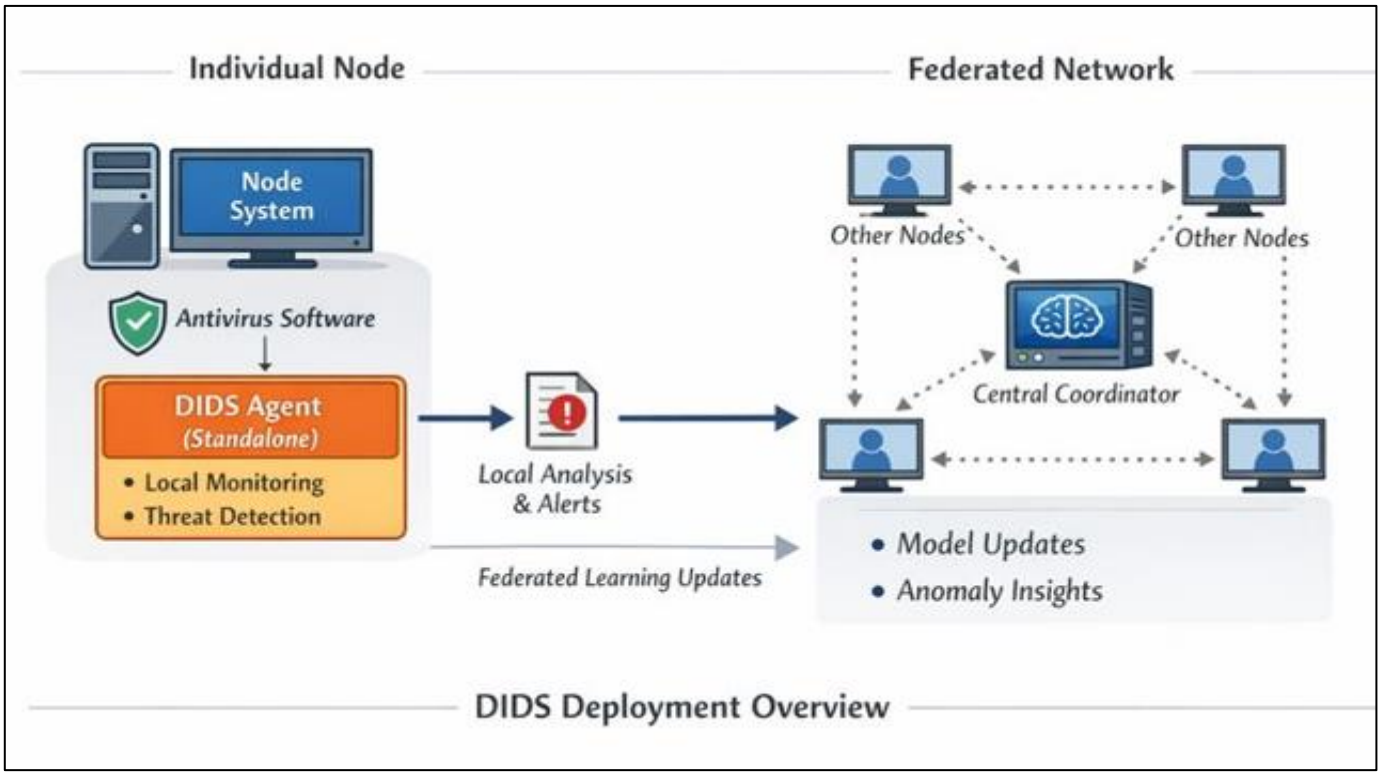
[19.] Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology, 10*(2), Article 12. https://doi.org/10.1145/3298981

[20.] Zhou, X., Xu, M., Wu, Y., & Zheng, N. (2021). Deep model poisoning attack on federated learning. *Future Internet, 13*(3), Article 73. https://doi.org/10.3390/fi13030073

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

APPENDICES





ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

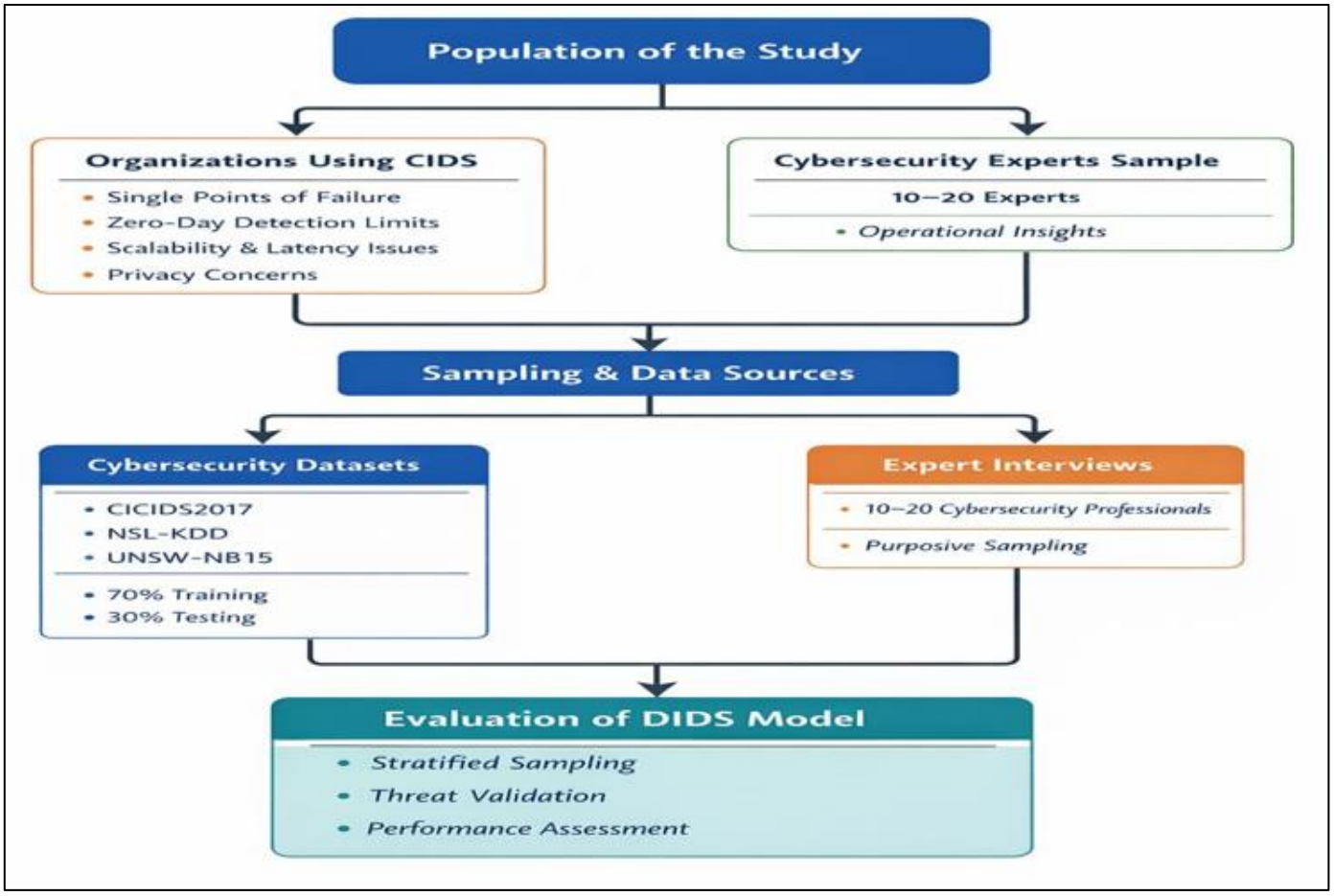


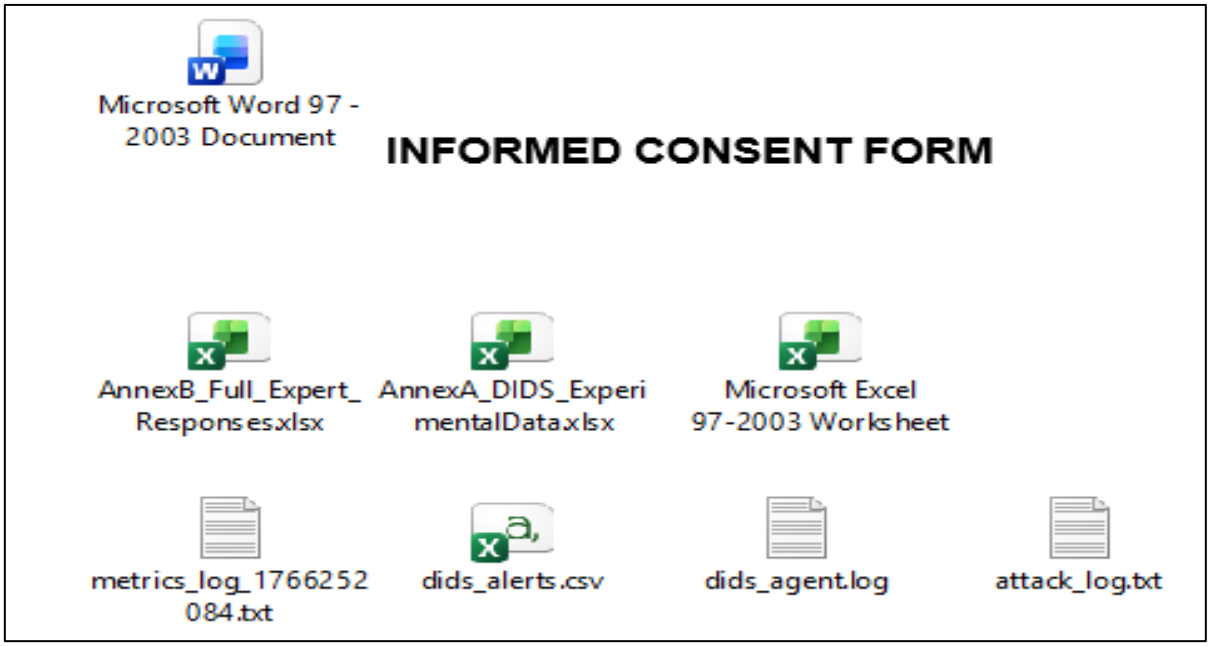
Fig 1 DIDM GUI-LOGS

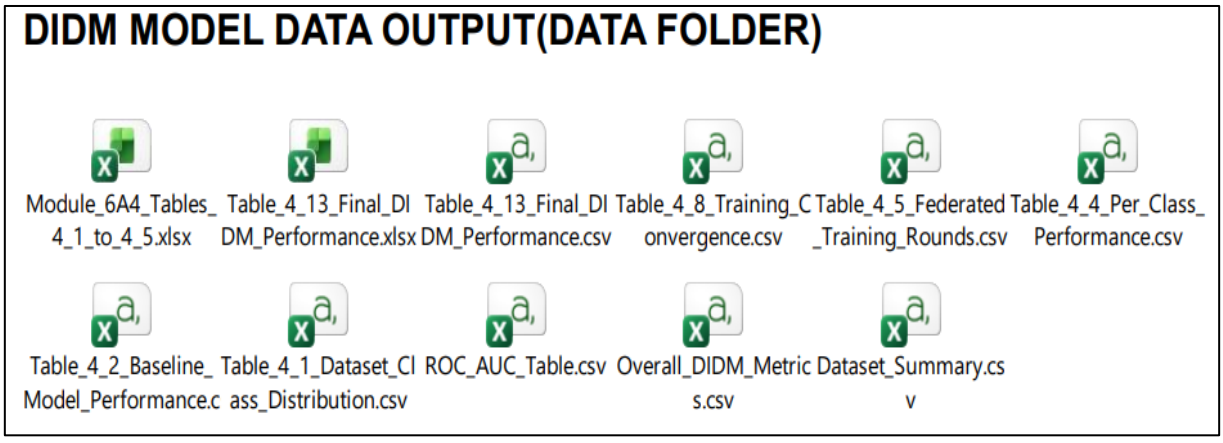
Table 1 How Mapping my DIDM Model Study Objectives to Methodology, Data Collection, and Expected Outcomes.

Objective	Methodology / Approach	Data Collection / Tools	Expected Outcome
i. Analyse the limitations of existing centralized and decentralized Intrusion Detection Models (IDMs).	Comprehensive literature review, comparative analysis and expert consultation.	Academic databases (IEEE Xplore, ACM Digital Library and SpringerLink), expert interviews and IDS evaluation reports.	Comprehensive identification of limitations including scalability constraints, privacy risks, high false-positive rates, communication latency and limited adaptability.
ii. Design a Federated Learning-based Decentralized Intrusion Detection Model (DIDM).	Design Science Research Methodology (DSRM), architectural design and iterative refinement.	Python (TensorFlow/Keras), Flower or TensorFlow Federated, Mininet and Containernet.	Validated DIDM architecture integrating Federated Learning, secure aggregation and local anomaly detection models.
iii. Implement local anomaly detection models and the Federated Learning	Algorithm implementation, local model training, Federated Averaging (FedAvg) and	CICIDS2017, NSL-KDD and UNSW-NB15 datasets; Docker; Python machine learning libraries;	Functional distributed intrusion detection prototype incorporating Autoencoder, Isolation Forest, One-Class SVM

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

framework within a simulated environment.	simulation-based experimentation.	Mininet and Containernet.	and global federated aggregation.
iv. Evaluate the performance and robustness of the proposed model against a centralized baseline.	Simulation-based testing, comparative evaluation and quantitative statistical analysis.	Benchmark datasets, simulation logs, Python evaluation tools, confusion matrices and expert validation.	Improved Accuracy, Precision, Recall, F1-score, scalability, privacy preservation and robustness against adversarial attacks compared with centralized approaches.





ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

APPENDIX II

NACOSTI RESEARCH LICENSE

REPUBLIC OF KENYA

NATIONAL COMMISSION FOR SCIENCE,TECHNOLOGY & INNOVATION

Ref No: 989630

Date of Issue: 22/April/2026

RESEARCH LICENSE



This is to Certify that Dr,. KENNEDY OMONDI OKEYO of Kabarak University, has been licensed to conduct research as per the provision of the Science, Technology and Innovation Act, 2013 (Rev.2014) in Kilifi, Mombasa, Tanariver on the topic: A DECENTRALIZED INTRUSION DETECTION MODEL USING FEDERATED LEARNING for the period ending : 22/April/2027.

License No: NACOSTI/P/26/4188526

989630

Applicant Identification Number

Ag. Director General
NATIONAL COMMISSION FOR SCIENCE,TECHNOLOGY & INNOVATION

Verification QR Code

NOTE: This is a computer generated License. To verify the authenticity of this document, Scan the QR Code using QR scanner application.

See overleaf for conditions

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

APPENDIX III

QUESTIONNAIRE FOR CYBERSECURITY EXPERTS

➢ *Title:*

Expert Feedback Questionnaire on a Decentralized Intrusion Detection Model (DIDM) Using Federated Learning and Smart Agents

➢ *Purpose of the Study*

This questionnaire is designed to collect expert opinions and professional insights to support the design, enhancement, implementation, and evaluation of a Decentralized Intrusion Detection Model (DIDM) that integrates Federated Learning and Smart Agents for improved cybersecurity management.

Your responses will contribute to academic research and will be treated with strict confidentiality. The information collected will be used solely for research purposes.

➢ *Instructions*

Please answer all questions as accurately and honestly as possible.

For Sections B–E, indicate your level of agreement using the following scale:

Scale	Meaning
1	Strongly Disagree
2	Disagree
3	Neutral
4	Agree
5	Strongly Agree

➢ *Section A: Respondent Background Information*

- Name of Organization (Optional):
- Job Title/Role:
- Years of Experience in Cybersecurity/IDM Management:

☐ Less than 1 year
☐ 1–3 years
☐ 4–6 years
☐ 7–10 years
☐ Above 10 years

- Area of Specialization:

☐ Network Security
☐ Cybersecurity Operations
☐ Cloud Security
☐ Artificial Intelligence/Machine Learning
☐ Digital Forensics

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

☐ Model Administration
☐ Other: _____

- Familiarity with Intrusion Detection Models (IDM):

☐ Centralized IDM (CIDM)
☐ Decentralized IDM (DIDM)
☐ Both
☐ None

A. Section B: Existing Idm Weaknesses

(Objective i / Research Question i)

➢ *B1. Current IDM Usage*

- Does your organization currently use an Intrusion Detection Model (IDM)?

☐ Yes
☐ No

- If Yes, which type of IDM is used?

☐ Centralized IDM (CIDM)
☐ Decentralized IDM (DIDM)
☐ Hybrid IDM
☐ Other: _____

- Name of the IDM platform/vendor (Optional):

- Duration of IDM operation in your organization:

☐ Less than 1 year
☐ 1–3 years
☐ 4–5 years
☐ More than 5 years

- Network environments monitored by the IDM (Tick all that apply):

☐ Internal Corporate Network
☐ Cloud Infrastructure
☐ Hybrid Environment (On-premises + Cloud)
☐ IoT/Smart Devices
☐ Remote Workforce Infrastructure
☐ Other: _____

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

➢ *B2. Effectiveness of Existing IDM*

Statement	1	2	3	4	5
The current IDM effectively detects known cyberattacks	☐	☐	☐	☐	☐
The current IDM effectively detects unknown or zero-day attacks	☐	☐	☐	☐	☐
The IDM generates accurate alerts with minimal false positives	☐	☐	☐	☐	☐
The IDM performs efficiently in large-scale distributed environments	☐	☐	☐	☐	☐
The IDM supports real-time threat detection and response	☐	☐	☐	☐	☐
The IDM adequately preserves data privacy and confidentiality	☐	☐	☐	☐	☐

➢ *B3. Open-Ended Questions*

- What are the major weaknesses or limitations of your current IDM?
- Describe any significant cybersecurity incidents where the IDM failed to detect or respond effectively.
- Which areas of IDM technology require the most improvement?

B. Section C: Didm Design Features

(Objective ii / Research Question ii)

➢ *Rate the Importance of the Following DIDM Features:*

Feature	1	2	3	4	5
Localized data processing enhances privacy preservation	☐	☐	☐	☐	☐
Federated learning improves collaborative threat intelligence	☐	☐	☐	☐	☐
Smart agents improve autonomous decision-making	☐	☐	☐	☐	☐
Real-time threat analysis enhances cybersecurity response	☐	☐	☐	☐	☐
DIDM architecture improves scalability across distributed networks	☐	☐	☐	☐	☐
Decentralization reduces single points of failure	☐	☐	☐	☐	☐

➢ *Open-Ended Questions*

- Which DIDM features would be most beneficial to your organization?
- What risks or challenges do you foresee in implementing DIDM?

C. Section D: Smart Agent Enhancement

(Objective iii / Research Question iii)

➢ *Indicate Your Level of Agreement with the Following Statements:*

Statement	1	2	3	4	5
Smart agents can effectively detect anomalies in real time	☐	☐	☐	☐	☐
Autonomous agents can respond to cyber threats without human intervention	☐	☐	☐	☐	☐
Ensemble machine learning algorithms improve detection accuracy	☐	☐	☐	☐	☐
Smart agents can operate efficiently without excessive resource consumption	☐	☐	☐	☐	☐
Human expert validation remains necessary in autonomous IDM operations	☐	☐	☐	☐	☐

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Smart agents improve adaptive learning against evolving cyber threats	☐	☐	☐	☐	☐

➢ *Open-Ended Questions*

- What additional capabilities should smart agents possess in modern IDM environments?
- What challenges may arise when deploying autonomous smart agents?

D. Section E: Performance Evaluation Metrics

(Objective iv / Research Question iv)

➢ *Rate the Importance of the Following Evaluation Metrics:*

Metric	1	2	3	4	5
Detection accuracy (Precision, Recall, F1-score)	☐	☐	☐	☐	☐
Resource utilization (CPU, memory, bandwidth, energy)	☐	☐	☐	☐	☐
Communication overhead in federated learning	☐	☐	☐	☐	☐
Model robustness against cyberattacks and node failures	☐	☐	☐	☐	☐
Scalability across distributed network environments	☐	☐	☐	☐	☐
Speed of threat detection and response	☐	☐	☐	☐	☐
Privacy preservation and data security	☐	☐	☐	☐	☐

➢ *Open-Ended Questions*

- Which additional metrics should be considered when evaluating DIDM performance?
- Which performance metrics should be prioritized most in a modern IDM?

E. Section F: General Comments and Recommendations

- Please provide any recommendations for improving the DIDM framework.
- What organizational, technical, or policy constraints should be considered during DIDM deployment?
- Any additional comments or observations:

➢ *Alignment of Questionnaire with Study Objectives*

Section	Research Objective	Research Question
Section B	Objective i	Research Question i
Section C	Objective ii	Research Question ii
Section D	Objective iii	Research Question iii
Section E	Objective iv	Research Question iv
Section F	Cross-cutting Issues	All Research Questions

Thank You

Thank you for taking time to participate in this study. Your expert contribution is highly appreciated and will support the development of more effective, privacy-preserving, and intelligent cybersecurity solutions.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

APPENDIX IV

INFORMED CONSENT FORM

KABARAK UNIVERSITY RESEARCH ETHICS COMMITTEE

ADULT INFORMED CONSENT FORM

(The form is written in English language but can be translated to Kiswahili or any other appropriate language)

PRINCIPAL INVESTIGATOR (PI)

Kennedy Omondi Okeyo

Registration Number

GMIA/ON/2048/09/16

Affiliated Institution

Kabarak University

Co-investigator(s)

None

Affiliated Institution(s)

Not applicable

➢ *Introduction*

You are invited to participate in this research study being conducted by the above listed investigator as part of a Master of Science in IT Security & Audit at Kabarak University.

This form is intended to help you understand the study so that you can voluntarily decide whether or not to participate. You are encouraged to ask any questions regarding the research process, including any possible risks or benefits.

Participation is voluntary, and you may withdraw at any time without penalty or obligation to explain your decision. Your access to services you are entitled to will not be affected.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

If you agree to participate, you will be requested to sign or thumbprint this form. A copy will be provided for your records.

Should I continue YES/NO _____

This study has been reviewed and approved by the Kabarak University Research Ethics Committee (KUREC).

- *What is the Purpose of the Study?*

The purpose of this study is to develop a decentralized intrusion detection model using Federated Learning to enhance cybersecurity resilience in distributed network environments.

✓ *This Study is Guided by the Following Research Questions:*

- What are the operational weaknesses and limitations of existing centralized and decentralized Intrusion Detection Models (IDM), particularly in terms of detection accuracy, scalability, privacy, and resilience?
- How can a decentralized intrusion detection model be designed using Federated Learning to enable collaborative and privacy-preserving model training across distributed network nodes?
- How can local anomaly detection algorithms be implemented within the DIDM model to perform accurate real-time threat analysis and adaptive responses to evolving cyber threats?
- How can the performance of the DIDM model be evaluated in terms of detection accuracy, scalability, privacy preservation, and robustness against adversarial attacks using benchmark datasets, simulated network environments, and expert-informed scenarios?

- *Who can Take Part in the Study?*

Participants will include ICT professionals, cybersecurity experts, Model administrators, and users with experience in networked or distributed computing environments.

✓ *Inclusion Criteria:*

- Adults aged 18 years and above
- ICT professionals, cybersecurity practitioners, or Model users
- Individuals with knowledge or experience in network security Models
- Willingness to provide informed consent

✓ *Exclusion Criteria:*

- Individuals below 18 years
- Individuals without ICT or cybersecurity-related exposure
- Individuals unwilling to participate

✓ *Estimated Sample Size:*

Approximately 20–30 participants depending on institutional access.

- *In Case You Agree to Participate in the Study, What Will Happen?*

✓ Time commitment: Participation will take approximately 20–40 minutes in a single session. Follow-up contact may occur only for clarification purposes.
✓ A trained researcher will conduct interviews or questionnaires in a secure and private setting. You are free to skip any question you are uncomfortable with.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

✓ *The Study will Focus on:*

- Limitations of existing intrusion detection Models
- Cybersecurity threats and operational challenges
- Federated learning and distributed Model concepts
- Real-time anomaly detection and response mechanisms
- Model performance, scalability, and privacy considerations
- No physical procedures are involved.
- You may be asked to provide contact details (phone or email) for follow-up clarification. These will be used strictly for research purposes.

All contact information will remain confidential and accessible only to the Principal Investigator.

- *What Potential Risks are Associated with Participation in this Study?*

This study carries minimal risk.

Possible discomfort may arise when discussing cybersecurity weaknesses or institutional Model vulnerabilities.

There is a low risk of confidentiality breach, which will be mitigated through strict anonymization and secure data handling.

✓ *Mitigation Measures Include:*

- Use of anonymized participant identifiers
- Secure encrypted storage of all data
- Restricted access to research data
- No disclosure of identifiable information in reports or publications

➢ *Privacy & Confidentiality*

All data collected will be treated as strictly confidential.

Participant identities will be replaced with unique codes, and no personal identifiers will appear in reports or publications.

Data will be stored securely in password-protected and encrypted Models accessible only to the Principal Investigator.

Data will be retained for up to five years after completion of the study and then permanently deleted.

➢ *What Benefits are you Going to Accrue by Participating in the Study*

There are no direct financial benefits.

However, participants may benefit indirectly through improved understanding of cybersecurity Models, intrusion detection approaches, and federated learning applications.

The study contributes to advancing secure, privacy-preserving, and scalable cybersecurity solutions in distributed environments.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *What Will it Cost You to Participate in the Study?*
  There is no financial cost associated with participation. Only voluntary time and engagement are required.

- *Will you be Paid or Reimbursed for Participating?*
  No payment or reimbursement will be provided.

- *In Case I Have Any Further Questions/Concerns in Future Whom Should I Contact?*
  For any inquiries regarding this study, contact:

✓ *Principal Investigator (PI):*
Kennedy Omondi Okeyo

- Email: kenokeyo@kabarak.ac.ke
- Institution: Kabarak University

For concerns regarding your rights as a participant, contact:

Secretary, Kabarak University Research Ethics Committee (KUREC)

(Contact details as provided by the university)

- *What Alternative Options are Available to Me?*
  Participation is voluntary. You may decline or withdraw at any time without penalty and without providing any reason.

- *How Will the Findings of this Study be Communicated or Shared?*
  The findings will be shared through:

✓ Submission of a Master's thesis to Kabarak University
✓ Academic presentations and seminars
✓ Possible publication in peer-reviewed journals
✓ Feedback summaries to participating institutions where applicable

➢ *Statement of Consent*
I confirm that I have read or had the information explained to me. I understand the purpose, procedures, risks, and benefits of the study.

I understand that participation is voluntary and I may withdraw at any time without penalty.
I freely agree to participate in this study.
I agree to provide my contact details for follow-up.

I agree to participate in this research: YES _____ NO _____

I agree to provide my contact details for follow-up: YES _____ NO _____

Participant's Name: _____

Signature/Thumbprint: _____

Date: _____

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

APPENDIX V

ETHICAL REVIEW LOW RISK APPLICATION FORM

KABARAK UNIVERSITY

OFFICE OF THE DIRECTOR, RESEARCH, INNOVATION AND OUTREACH

RESEARCH ETHICS COMMITTEE

APPLICATION FOR LOW RISK ETHICAL REVIEW

Principal Investigator Name (print) **KENNEDY OMONDI OKEYO**	**Date** **20-02-2026**
Study Title **A DECENTRALIZED INTRUSION DETECTION MODEL USING FEDERATED LEARNING**	**Application Number** **(KUREC office only)**
Principal Investigator's Signature ._____	

➢ *Purpose*

This form is intended for use by Kabarak University researchers in the identification and assessment of risk associated with DIDM research involving human participants or data pertaining to human participants.

- *Examples of Low Risk Research Include and are not Limited to Research Based on or Involving:*

✓ Information from existing de-identified datasets;
✓ Historical archives and public records;
✓ Data collected from internet sites;
✓ Questionnaires and surveys on non-controversial, non-personal issues that only
✓ Include basic and de-identified demographic data; and
✓ Unobtrusive observational studies (i.e. with no form of intervention) of people undertaking non-sensitive or benign activities in a public space.
✓ Personal interviews or focus groups with participants containing non-controversial, impersonal, and non-sensitive questions; and
✓ Questionnaires or surveys containing questions about participants' experiences, thoughts, opinions and beliefs;

➢ *Project Risk Assessment*

If you believe that your research is low risk, might consider the low risk ethics pathway. In order to help you ascertain whether your research is indeed low risk, answer *all* of the questions below.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

➢ *Research Topics*

Please indicate whether any of the following topics will be covered in part or in whole?

	Yes	No
Research about parenting		**No**
Research investigating sensitive personal issues		**No**
Research investigating sensitive cultural issues		**No**
Explorations of grief, death or serious/traumatic loss		**No**
Depression, mood states, anxiety		**No**
Gambling		**No**
Eating disorders		**No**
Illicit drug taking		**No**
Substance abuse		**No**
Self-report of criminal behaviour		**No**
Any other psychological disorder		**No**
Suicide		**No**
Gender identity		**No**
Sexuality		**No**
Race or ethnic identity		**No**
Any disease or health problem		**No**
Fertility		**No**
Termination of pregnancy		**No**

➢ *Procedures*

Are the following procedures to be employed?

	Yes	No
Deception of participants		**No**
Use of data or records from which individuals can be identified		**No**
Covert observation		**No**
Audio or visual recording without consent		**No**
Recruitment via a third party or agency		**No**
Any psychological interventions or treatments		**No**
A conflict of interest may exist (e.g., financial or other relationship)		**No**

➢ *Participant Vulnerability Assessment*

Do any of the participants fall within the following categories?

	Yes	No
Persons suffering a psychological disorder		**No**
Persons suffering a physical vulnerability		**No**
Persons highly dependent on medical care		**No**
Minors (0 – 18 years)		**No**
Persons whose ability to give consent is impaired		**No**
Residents of a custodial institution		**No**
Persons with English language difficulties		**No**
Members of a socially identifiable group with special cultural or religious needs or political vulnerabilities		**No**
Persons in a dependent relationship with the researcher		**No**

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

Participants able to be identified in any final report or publication when consent has not been given		**No**

Answering "Yes" to any of the above questions means that your thesis does not fulfil any of the three risk definitions outlined above and that your application will not be approved under the Low Risk Ethics Pathway.

If you answered "No" to all of the above questions, please proceed to provide the following information in as great detail as possible.

- *What is your Research Topic?*

✓ *Answer:*
A Decentralized Intrusion Detection Model Using Federated

- *Why do you Consider your Research Topic Low Risk?*

✓ *Answer:*
The research is low risk because all experiments are conducted in a controlled, simulated digital environment, and human participation is limited to voluntary professional opinions from cybersecurity experts via KoboCollect. No sensitive personal data is collected, and participation poses no physical, psychological, or social risks.

- *What are Your Study / Data Collection Procedures?*

✓ *Answer:*
The study will involve conducting experiments in simulated environments such as NS-3, Mininet, GNS3, and cloud-based platforms to model network nodes, servers, and traffic behavior. Publicly available benchmark datasets and expert-informed attack scenarios will be used to test and validate the DIDM model. In addition, 10–20 cybersecurity professionals from organizations currently using CIDM will be consulted through structured questionnaires administered via KoboCollect to obtain feedback on the model's design, usability, and practical applicability. The model's performance will be evaluated using standard metrics including accuracy, precision, recall, and F1-score. All expert responses will be anonymized, and participation will be strictly voluntary.

- *Why do You Consider Your Study / Data Collection Procedures to be Low Risk?*

✓ *Answer:*
The study is low risk because experiments occur in virtual, controlled environments, eliminating any risk to live networks. Expert participation is voluntary and digital, with data collected securely via KoboCollect and anonymized. No sensitive personal information is collected, and participants face no physical, psychological, or social risks.

- *Who are Your Study Participants?*

✓ *Answer:*
The participants are 10–20 cybersecurity professionals working in organizations that currently use Centralized Intrusion Detection Models (CIDM), providing structured expert feedback on the DIDM model.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26nov017

- *Why do you Consider your Study Participants not to be Vulnerable?*

✓ *Answer:*

Participants are adult professionals with relevant expertise, capable of giving informed consent. Their involvement is limited to providing professional opinions via a secure digital tool, with no physical, psychological, or social risk.