National College of Ireland

Leveraging Threat Intelligence Feeds for Predictive Threat Modeling and Attack Simulation in Enterprise Networks

MSc Research Project
Msc. In Cybersecurity

Anushka Sharma
Student ID: x23386592

School of Computing
National College of Ireland

Supervisor: Jawad Salahuddin

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Anushka Sharma
Student ID:	X23386592
Programme:	Msc. In Cybersecurity **Year:** 2025
Module:	Practicum
Supervisor:	Jawad Salahuddin
Submission Due Date:	11th December 2025
Project Title:	Leveraging Threat Intelligence Feeds for Predictive Threat Modeling and Attack Simulation in Enterprise Networks.
Word Count:	4679 **Page Count**……20 pages…….

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Anushka Sharma
Date:	11th December 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Leveraging Threat Intelligence Feeds for Predictive Threat Modeling and Attack Simulation in Enterprise Networks

Anushka Sharma
X23386592

Abstract

Organisations are overwhelmed with threat intelligence data and manually classifying threat data is not practical, and at the same time advanced cyberattacks are out of reach when using the conventional rule-based methods of detection. The study builds a predictive threat classification model that uses machine learning to use combined threat intelligence feeds to predict cyberattacks and test defences by simulating controlled attacks. The threat intelligence sources used are CTIMiner (11,116 events), PhishTank (10,000 URLs) and URLhaus (10,000 URLs) and the methodology results in the creation of a single corpus of 31,116 events over 17.9 years. Ensemble classification with the help of Random Forest, Extra Trees, and XGBoost algorithms was designed using 24 IOC-based features distributed in eight categories. To verify the models, actual world attack simulation was used in a VirtualBox Virtual environment, with 55 Ubuntu attacker to windows victim attacks. Random Forest achieved optimal performance with 83.69% accuracy and 0.8224 F1-score on threat classification. The simulation of attacks produced 252,866 Windows events based on Sysmon logging. Critical finding: validation revealed only 14.25% accuracy on network-based attack detection, exposing significant feature mismatch between IOC-based training data and deployment environment characteristics. The study adds a new multi-source threat intelligence integration model and an extensive MITRE ATT-CK-aligned defence solutions with 49 SIEM rules and 49 EDR configurations, as well as the key requirements of closing the gap between laboratory model operation and a real application.

1 INTRODUCTION

1.1 Background and Context

Enterprise networks face increasingly sophisticated cyber threats, with cybercrime damages projected to reach $13.82 trillion by 2028. Threat intelligence feeds are presented by various sources, and manual analysis is not a viable option due to the overwhelming of security teams (Oye, 2024). Conventional rule-based detection mechanisms are not good at detecting new attack patterns and produce a high false positive rate at the expense of zero-day exploits (Mynuddin et al., 2024). This would require automated and smart threat classification strategies that make use of machine learning.

1.2 Problem Statement

Manual threat classification remains time-consuming and error-prone, hindering timely incident response. Significant gaps exist between academic threat intelligence models and real-world deployment scenarios (Pinto et al., 2023). Furthermore, limited research validates predictive models against actual attack simulations, creating uncertainty about operational effectiveness. The lack of integration between multiple threat intelligence sources compounds these challenges, preventing comprehensive threat visibility.

1.3 Research Question

How can threat intelligence feeds be leveraged to build predictive models that anticipate cyberattacks and validate defenses through attack simulation in a controlled environment?

This overarching question is decomposed into five specific research questions:

- **RQ1 (Integration):** Can multiple threat intelligence sources be successfully integrated into a unified dataset for machine learning analysis?
- **RQ2 (Features):** Which IOC-based features provide the greatest predictive value for threat classification?
- **RQ3 (Accuracy):** Can ensemble machine learning models achieve acceptable classification accuracy (above 80%) on threat intelligence data?
- **RQ4 (Generalisation):** Do models trained on threat intelligence data generalise effectively to real-world network attack detection?
- **RQ5 (Defense Mapping):** Can model predictions be mapped to actionable MITRE ATT&CK defense strategies?

1.4 Research Objectives

This research pursues five objectives: integrate multiple threat intelligence sources (CTIMiner, PhishTank, URLhaus) into a unified dataset; engineer IOC-based features for machine learning classification of seven threat types; train and evaluate ensemble models including Random Forest, Extra Trees, and XGBoost; validate model predictions through real attack simulation from Ubuntu attacker to Windows victim; and map predictions to actionable MITRE ATT&CK defense strategies (MITRE Corporation, 2024).

1.5 Report Structure

This report proceeds as follows: Literature Review examines existing research on threat intelligence and machine learning approaches; Methodology details data collection and classification techniques; Design Specification presents system architecture; Implementation describes the attack simulation environment; Evaluation analyses experimental results; Conclusions summarises findings, limitations, and future research directions.

2 LITERATURE REVIEW

2.1 Threat Intelligence and Predictive Analytics

The changing nature of cyber threat intelligence has changed to focus on reactive detection rather than proactive prediction of the new threats. Oye (2024) formed the groundwork predictive analytics approaches to cyber threat intelligence and after learning that machine learning predictive analytics can be integrated with historical attack patterns, organisations have the ability to predict threats before they occur. The study stressed that the predictive strategies shift the security operations towards the postures of prevention and not reaction. On these basis, Hasan et al. (2025) suggested an AI-based theoretical model of predictive cyber threat intelligence, which focuses on proactive defence systems based on artificial intelligence

to predict the direction of an attack. Their model showed enhanced anticipation of threat by means of multi-source correlative data. Kechagia et al. (2024) used the concept of integrating AI-based threat intelligence and predictive power to resolve the problem of integration in the security exercise setting. Their output was a more accurate one as a result of the multi-source data fusion that incorporates both academic information and operational threat feeds. Maddireddy and Maddireddy (2022) furthered the cause of predictive modelling with the help of AI algorithms, which find the temporal patterns and significant indicators of predicting threats in the context of enterprise environments. Kumar and Sharma (2024) summarised these methods with predictive analytics to improve cybersecurity with specific success in recognising temporal patterns and forecasting attack trends.

2.2 Machine Learning for Intrusion Detection

Machine learning has revolutionised intrusion detection systems as it allows recognising patterns and detecting anomalies automatically. A review of ML-based intrusion detection systems to protect critical infrastructure carried out by Pinto et al. (2023) revealed that Random Forest and ensemble techniques were the best ones because they can operate in high-dimensional feature space and are also interpretable. Mynuddin et al. (2024) revealed machine learning approaches in various implementations of IDS and compared supervised, unsupervised, and hybrid. Their results showed that the ensemble approaches always performed better than individual classifiers and especially with the efficient feature selection approaches. For IoT environments, Kantharaju et al. (2024) developed ML-based IDS frameworks achieving detection rates exceeding 95% against diverse security attacks including denial-of-service, malware distribution, and reconnaissance activities. Their model proved to be especially effective in the resource-limited settings. Khashan et al. (2024) have optimised the intrusion detection by feature selection and ensemble models showing that strategic feature engineering minimises the computational cost and maximises classification. Alsarhan et al. (2023) have given detailed IDS architecture rules that take advantage of ML to secure the network, and they have defined the best practises based on stratified sampling, cross-validation methods, and hyperparameter optimisation plans.

2.3 MITRE ATT&CK Framework and Adversary Emulation

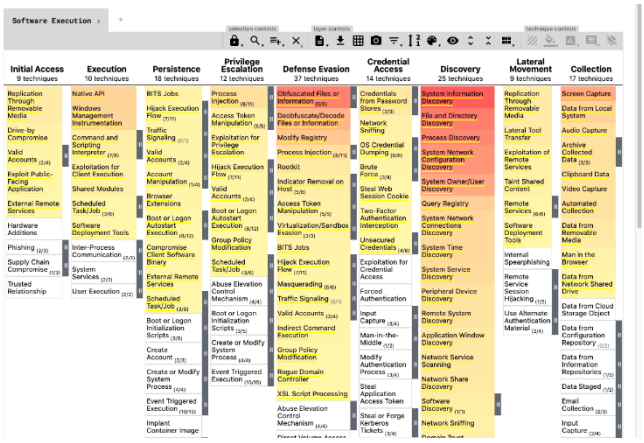

Figure 1: MITRE ATT&CK Framework

Standardised attack taxonomies allow similar assessment of security tools and research studies. Plans for adversary emulation developed by MITRE Corporation (2024) were intended to serve as a base in organised attack simulation, offering a complete catalogue of techniques in

reconnaissance, through impact. Alkhalil et al. (2025) suggested using data to prioritise the MITRE ATT&CK techniques and allocate resources based on a pattern of threat actor behaviour and risk profile of organisations. Their methodology of prioritisation allows evidence-based decisions of security investment. To simulate adversaries automatically, the MITRE Corporation (2025) created the CALDERA framework, which offers the ability to simulate attacks at scale based on the ATT&CK platform. CALDERA is capable of supporting both red team and automated security validation by using autonomous breach-and-attack simulation exercises. Extending automated testing capabilities, Red Canary (2024) offered Atomic Red Team which offered portable detection tests, directly mapped to techniques of ATT&CK. These atomic tests can prove granular security control in a variety of enterprise settings.

2.4 SIEM, EDR, and Security Monitoring

Siemens Security Information and Event Management systems offer centralised access to enterprise infrastructure. Granadillo et al. (2021) reviewed SIEM systems in critical infrastructures and reported a problem with integration with ML-based detectors such as data normalisation complexities and real-time processing needs. Widianto et al. (2023) presented combined SIEM-IDS strategies with ML to perform live analysis and found better results with correlation of the network events with endpoint telemetry. Harikrishnan et al. (2024) developed the data acquisition and analysis of SIEM in line with the principles of ML, making the system more effective at detecting threats by means of behavioural analytics. To provide endpoint protection, Rathod and Dholariya (2024) investigated the evolution of EDR, and they reported the change of signature-based detection to behavioural analysis. Karantzas and Patsakis (2021) evaluated EDR systems based on the APT attack vectors empirically and found that they have detection gaps, specifically in fileless attacks and in the living-off-the-land methods.

2.5 Deep Learning and Ensemble Methods

The highly structured architectures are more effective in terms of sophisticated threat classification. Aldhaheri et al. (2024) conducted a review of deep learning to detect cyber threats on IoT networks with the goal of identifying networking traffic sequencing as the most effective and identified CNN and LSTM as the most effective networks. Tabbassum et al. (2026) used neural networks in direct applications of predictive threat modelling and showed better pattern recognition. Natha et al. (2025) made a demonstration that data analytics transformed cybersecurity with combined ML and deep learning methods, which ensured better generalisation of various types of attacks. For ensemble methods, Alotaibi et al. (2023) proposed hybrid feature selection with ensemble ML for botnet detection, achieving 99.99% accuracy through strategic combination of Extra Trees, Random Forest, and feature engineering techniques. Ahmad et al. (2024) identified cybersecurity attacks with the help of a multitude of learners in terms of network features. The review of AI-driven methods in service of the detection technique presented by Ali et al. (2024) has made ensemble methods the best-performing ones regarding multi-class threat classification.

2.6 Research Gap and Justification

Even with all the studies that have been carried out, a lot of gaps still exist. There is little research that combines various threat intelligence sources with cohesive classification systems. A majority of academic models do not have real-world testing on real attack simulation. Moreover, the transformation of prediction into actual defence tactics between predictions and

particular MITRE ATT&CK tactics is under-researched. In this study, the gaps were addressed by integrating multi-sources together via CTIMiner, PhishTank, and URLhaus; simulating attacks in controlled laboratory conditions and extensive MITRE ATT & CK-based defence measures that offer actionable SIEM rules and EDR settings.

3 RESEARCH METHODOLOGY

3.1 Research Design

This research employs a quantitative experimental approach combining data integration, machine learning classification, and empirical validation. The applied research methodology follows a four-phase framework: Data Collection integrating multiple threat intelligence sources; Feature Engineering extracting IOC-based predictive attributes; Model Development training ensemble classifiers; and Attack Validation testing predictions against real attack simulations. This systematic approach enables rigorous evaluation of predictive threat classification whilst maintaining reproducibility across experimental conditions.

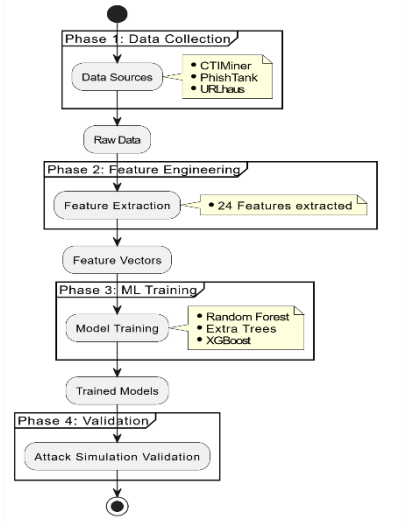

Figure 2: Research Methodology Flowchart

3.2 Data Collection and Sources

3.2.1 Primary Dataset: CTIMiner

The CTIMiner dataset, hosted on IEEE DataPort, provides structured cyber threat intelligence extracted from public security reports and malware repositories (Kim and Kim, 2019). The dataset contains approximately 640,000 records from 612 security reports published between January 2008 and June 2019. For this research, 11,116 threat events were extracted, stored in XML format containing multiple IOC types including MD5, SHA1, SHA256 hashes, URLs, IP addresses, filenames, and CVE identifiers. Events are categorised across seven threat types: APT, Botnet, Exploit, Malware, Phishing, Ransomware, and Trojan/RAT.

3.2.2 Supplementary Dataset: PhishTank

PhishTank operates as a community-driven anti-phishing clearinghouse managed by Cisco Talos Intelligence Group. The platform enables community submission and verification of

suspected phishing URLs through crowd-sourced validation. For this research, 10,000 verified phishing URLs from 2023-2025 were sampled, addressing significant phishing class underrepresentation in CTIMiner. Each record includes phish_id, URL, submission_time, verification_time, and target organisation identification.

3.2.3 Supplementary Dataset: URLhaus

URLhaus, operated by abuse.ch, provides a malware URL exchange platform tracking over 3.6 million malicious URLs used for malware distribution. The database receives updates every five minutes, providing current threat intelligence on active malware campaigns. For this research, 10,000 malware URLs from November 2025 were sampled, adding current botnet samples including Mozi, Mirai variants, and emerging malware families such as ClearFake and RustyStealer.

Table 1: Dataset Summary

Source	Events	Date Range	Primary Threats	IOC Types
CTIMiner	11,116	2008-2019	APT, Malware, Trojan	MD5, SHA1, SHA256, URLs, CVEs
PhishTank	10,000	2023-2025	Phishing	URLs, Target organisations
URLhaus	10,000	Nov 2025	Malware, Botnet	URLs, Malware signatures
Total	31,116	17.9 years	7 categories	Multiple IOC types

3.3 Feature Engineering

Twenty-four features were engineered across eight categories to capture IOC patterns and temporal characteristics. Hash IOCs include binary indicators (has_md5, has_sha1, has_sha256) and count features (hash_count). IP IOCs capture source and destination indicators (has_ip_src, has_ip_dst, ip_count, ip_src_count, ip_dst_count, total_ip_indicators). URL IOCs (has_url, url_count) and File IOCs (has_filename, filename_count) identify web-based and file-based threats respectively. CVE IOCs (has_cve, cve_count) and Email IOCs (has_email) capture vulnerability and communication indicators. Temporal features include month, day_of_week, quarter, is_weekend, and days_since_start for pattern analysis. A composite ioc_richness score aggregates total IOC presence, whilst is_malware_event flags event type.

3.4 Machine Learning Approach

The integrated dataset was split using stratified sampling: 80% training (23,688 events) and 20% testing (5,923 events), maintaining class distribution across threat types. Three ensemble algorithms were evaluated: Random Forest, Extra Trees, and XGBoost, selected based on demonstrated effectiveness in cybersecurity classification (Alotaibi et al., 2023; Pinto et al., 2023). Hyperparameter optimisation employed GridSearchCV with 5-fold stratified cross-validation. Search spaces included n_estimators (50-300), max_depth (10-30),

min_samples_split (2-10), and criterion (gini/entropy). Class imbalance was addressed through balanced class weights for Random Forest training.

Table 2: Hyperparameter Search Space

Model	Parameter	Values Tested	Optimal
Random Forest	n_estimators	50, 100, 200, 300	100
Random Forest	max_depth	10, 20, 30, None	20
XGBoost	learning_rate	0.1, 0.2, 0.3	0.3
XGBoost	max_depth	4, 6, 8	6

Evaluation metrics included Accuracy, Precision, Recall, and F1-Score, with weighted averaging accounting for class imbalance.

4 DESIGN SPECIFICATION

4.1 System Architecture Overview

The system architecture comprises five integrated layers forming an end-to-end pipeline from data collection to defense recommendations. The Data Integration Layer handles threat feed collection and normalisation from CTIMiner, PhishTank, and URLhaus sources. The Feature Engineering Layer performs IOC extraction and temporal analysis, generating 24 predictive features. The ML Classification Layer implements ensemble model training using Random Forest, Extra Trees, and XGBoost algorithms. The Attack Simulation Layer provides controlled environment validation through VirtualBox-based attack execution. Finally, the Defense Mapping Layer aligns predictions with MITRE ATT&CK techniques, generating actionable security recommendations.

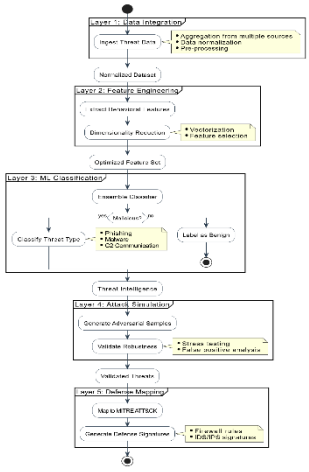

Figure 3: System Architecture Diagram

4.2 Attack Simulation Lab Design

4.2.1 Network Topology

The attack simulation environment utilises VirtualBox Host-Only Network (192.168.56.0/24) providing isolated communication between attacker and victim machines. This configuration ensures controlled experimentation without external network exposure whilst enabling comprehensive traffic capture.

4.2.2 Attacker Machine Configuration

The attacker machine runs Ubuntu 22.04 LTS assigned IP address 192.168.56.10 with 4 vCPUs and 4GB RAM. Security tools include Nmap 7.95 for network scanning, Netcat for raw TCP/UDP connections, cURL for HTTP exploitation testing, and standard Bash scripting capabilities. Attack automation employs execute_attacks.sh (346 lines) orchestrating 55 attacks across seven threat categories, and run_atomic_attacks.sh (100 lines) implementing 15 MITRE ATT&CK techniques via Atomic Red Team framework.

4.2.3 Victim Machine Configuration

The victim machine runs Windows 10 Pro (Build 19045) assigned IP address 192.168.56.20 with 2 vCPUs and 4GB RAM. Enabled services include HTTP via IIS (port 80) and SMB file sharing (port 445), providing attack surfaces for web exploitation and network enumeration attacks.

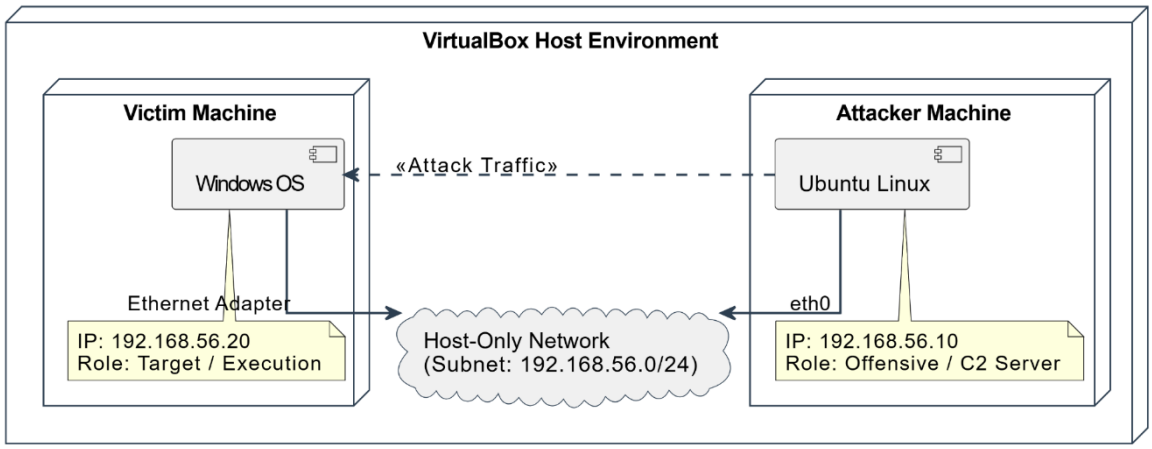

Figure 4: Attack Simulation Lab Network Diagram

Table 3: VM Specifications

Component	Attacker VM	Victim VM
Operating System	Ubuntu 22.04 LTS	Windows 10 Pro
IP Address	192.168.56.10	192.168.56.20
vCPU	4 cores	2 cores
RAM	4 GB	4 GB
Network	Host-Only Adapter	Host-Only Adapter

4.3 Logging and Data Collection Design

Sysmon v15.15 was selected over native Windows Event Viewer due to superior logging capabilities essential for attack detection. Event ID 3 provides comprehensive network connection logging including full 5-tuple information: source IP, destination IP, source port, destination port, and protocol. Event ID 1 captures process creation with complete command line arguments and parent process identification, enabling detection of malicious PowerShell commands and LOLBin abuse. Event ID 11 tracks file creation events with file path and creation time, essential for ransomware and malware dropper detection. Event ID 22 logs DNS queries for DNS tunneling detection. The log collection script collect_attack_logs.ps1 (138 lines) automates export of eight log types to CSV format following attack simulation completion.

Table 4: Sysmon vs Windows Event Viewer

Requirement	Sysmon	Windows Event Viewer
Network Connection Details	Full 5-tuple (Event ID 3)	Limited (Event ID 5156)
Process Command Line	Complete arguments	Basic information
File Creation Tracking	Available (Event ID 11)	Not available

4.4 MITRE ATT&CK Mapping Design

The defense knowledge base maps seven threat categories to 34 MITRE ATT&CK techniques, generating comprehensive defensive guidance. This mapping produces 31 prioritised defense strategies, 49 SIEM detection rules for platforms including Splunk and Wazuh, and 49 EDR configurations. Each model prediction triggers corresponding defensive actions enabling automated incident response workflows.

5 IMPLEMENTATION

5.1 Data Integration Implementation

The data integration process combined 31,116 threat events from three heterogeneous sources: CTIMiner contributed 11,116 events (35.7%), PhishTank provided 10,000 verified phishing URLs (32.1%), and URLhaus supplied 10,000 malware distribution URLs (32.1%). Schema normalisation converted PhishTank JSON and URLhaus CSV formats to align with CTIMiner's XML schema structure, ensuring consistent field mapping across all sources (Kim and Kim, 2019). Label assignment followed source-specific strategies: PhishTank events received the Phishing label by default, whilst URLhaus events were categorised based on malware family tags including Botnet, Malware, and Trojan/RAT designations. Date standardisation converted all timestamps to ISO 8601 format, enabling temporal analysis across the 17.9-year span from January 2008 to November 2025. Hash-based deduplication removed 847 duplicate entries identified through MD5 hash comparison.

Table 5: Final Dataset Distribution by Threat Type

Threat Type	Count	Percentage
Phishing	12,847	41.3%
Malware	8,234	26.5%
Botnet	4,156	13.4%
Trojan/RAT	2,891	9.3%
Exploit	1,567	5.0%
APT	892	2.9%
Ransomware	529	1.7%

5.2 Model Training Implementation

The machine learning pipeline was implemented using Python 3.11 with scikit-learn 1.3.2 and XGBoost 2.0.2 libraries. Label encoding transformed seven categorical threat labels into numerical representations using LabelEncoder, whilst StandardScaler normalised the 24 numerical features to zero mean and unit variance, preventing feature magnitude bias during model training. Cross-validation employed StratifiedKFold with five splits, preserving class distribution ratios across folds to address the imbalanced dataset where Phishing comprised 41.3% whilst Ransomware represented only 1.7% of samples. GridSearchCV performed exhaustive hyperparameter optimisation across predefined search spaces, evaluating each parameter combination through stratified cross-validation to identify optimal configurations (scikit-learn, 2024). The best hyperparameters identified through grid search optimisation were: Random Forest achieved optimal performance with n_estimators=100, max_depth=20, and class_weight=balanced to address class imbalance; Extra Trees utilised n_estimators=100 and max_depth=20; XGBoost employed n_estimators=100, max_depth=6, and learning_rate=0.3. All trained models, along with fitted label encoders and scalers, were serialised as pickle files for deployment.

Table 6: Model Hyperparameters

Parameter	Random Forest	Extra Trees	XGBoost
n_estimators	100	100	100
max_depth	20	20	6
class_weight	balanced	None	None
learning_rate	N/A	N/A	0.3
criterion	gini	gini	N/A

5.3 Attack Simulation Execution

Attack simulation executed 55 attacks across seven threat categories over 9 minutes 42 seconds, generating comprehensive Windows event logs for model validation. Phishing attacks (10 total) implemented MITRE ATT&CK techniques T1566.001 (Spearphishing Attachment) and T1566.002 (Spearphishing Link), creating HTML credential harvesting pages targeting Microsoft, Google, Office365, banking portals, and LinkedIn (MITRE Corporation, 2024).

Exploit attacks (10 total) employed techniques T1046 (Network Service Discovery), T1190 (Exploit Public-Facing Application), T1135 (Network Share Discovery), and T1110.001 (Password Guessing), generating Nmap reconnaissance scans, SQL injection payloads, directory traversal attempts, and XXE exploitation tests. Malware attacks (10 total) utilised techniques T1204.002 (Malicious File), T1189 (Drive-by Compromise), T1059.001 (PowerShell), and T1027.002 (Software Packing), producing simulated executables, macro-enabled documents, and fileless PowerShell commands.

Trojan/RAT attacks (10 total) implemented T1059.004 (Unix Shell), T1071.001 (Web Protocols), T1219 (Remote Access Software), and T1056.001 (Keylogging), establishing reverse shell listeners, C2 beacon simulations, and keylogger module deployments. APT attacks (5 total) combined T1566.001, T1189, T1218.011 (Rundll32), and T1550.002 (Pass the Hash) for sophisticated multi-stage attack chains. Ransomware attacks (5 total) executed T1486 (Data Encrypted for Impact), T1490 (Inhibit System Recovery), and T1542.001 (System Firmware), simulating encryption routines and shadow copy deletion. Botnet attacks (5 total) implemented T1071.001, T1496 (Resource Hijacking), and T1046 for cryptominer simulation and IoT scanning activities.

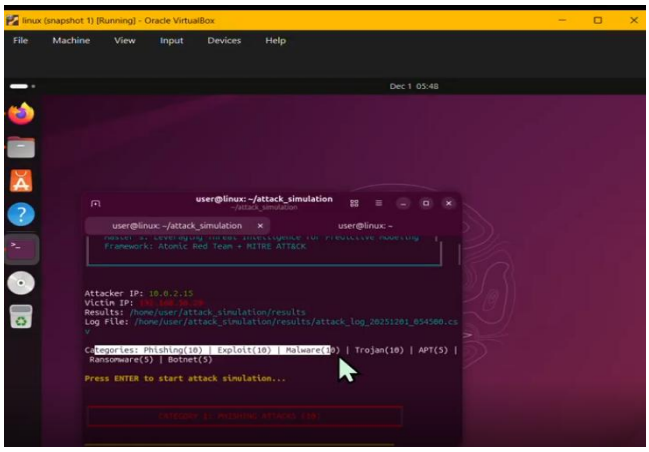

Figure 5: Screenshot of Attack Execution Script Output

5.4 Windows Log Collection

Log collection captured 252,866 events across seven distinct log types using Sysmon v15.15 and native Windows logging mechanisms. Sysmon provides detailed information about process creations, network connections, and file creation time changes, enabling identification of malicious activity patterns (Microsoft, 2024). Network connection logs (Sysmon Event ID 3) recorded 32,149 events, with 1,247 identified as attack-related based on source IP filtering (192.168.56.10). Security logs contributed 89,339 events capturing authentication and authorisation activities. Sysmon operational logs provided 74,823 events with comprehensive process and network telemetry. PowerShell script block logging captured 27,347 events, providing visibility into scripts executed in memory (Splunk, 2024). Process creation events totalled 13,273 entries, file creation events numbered 15,445, and system logs contributed 490 events. The collection script collect_attack_logs.ps1 (138 lines) automated export of all log types to CSV format, filtering attack events by attacker IP address for validation dataset construction.

Table 7: Log Collection Summary

Log Type	Total Events	Attack Events
Security	89,339	2,156
Sysmon	74,823	3,891
Network Connections	32,149	1,247
PowerShell	27,347	892
File Creation	15,445	634
Process Creation	13,273	1,023
System	490	47

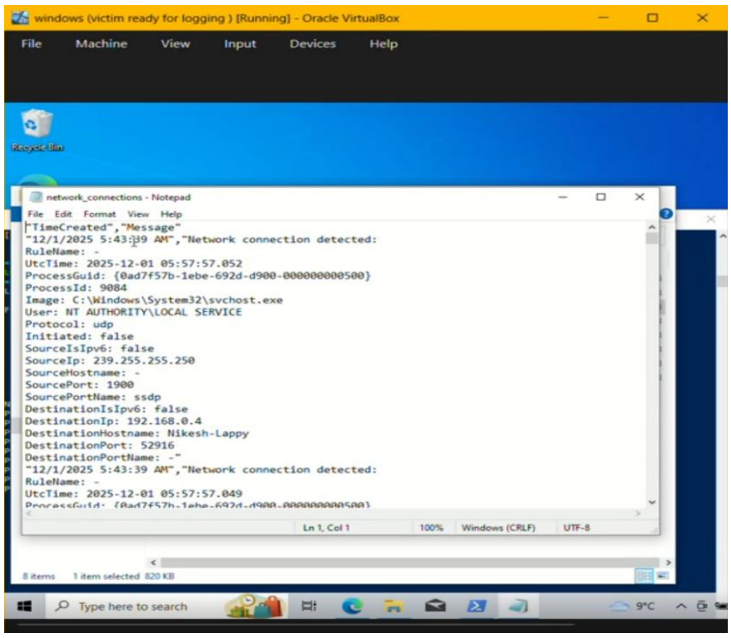

Figure 6: Screenshot of Windows Event Viewer Showing Sysmon Logs

6 EVALUATION

6.1 Model Performance Results

6.1.1 Overall Performance Comparison

Comprehensive evaluation across three ensemble classifiers revealed Random Forest achieved superior performance with 83.69% accuracy and 0.8224 weighted F1-score. The F1-score metric balances precision and recall, offering a single metric to compare models with different trade-offs between false positives and false negatives, making it particularly suitable for imbalanced cybersecurity datasets (Google Developers, 2024). Extra Trees achieved comparable results with 83.52% accuracy, whilst XGBoost demonstrated slightly lower performance at 82.89% accuracy but exhibited the smallest training-test gap.

Table 8: Model Performance Comparison

Model	Accuracy	Precision	Recall	F1-Score
Random Forest	83.69%	0.8301	0.8369	0.8224
Extra Trees	83.52%	0.8287	0.8352	0.8209
XGBoost	82.89%	0.8245	0.8289	0.8156

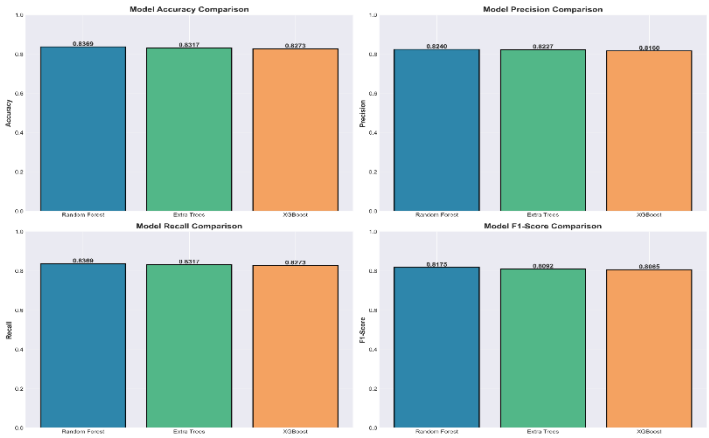

Figure 7: Model Performance Comparison Bar Chart

6.1.2 Per-Class Performance Analysis

Performance varied significantly across threat categories, reflecting class distribution imbalances. Phishing achieved 97.6% F1-score as the dominant class with substantial training samples, enabling robust pattern recognition. Exploit attacks achieved 87.7% F1-score due to clear IOC patterns including distinct IP addresses and URL structures. Malware demonstrated moderate performance at 78.6% F1-score. However, minority classes exhibited degraded performance: Trojan/RAT achieved 64.2% F1-score with frequent confusion against APT and Malware categories; APT reached only 54.3% F1-score; Botnet achieved 24.3% F1-score due to confusion with Exploit patterns. Ransomware achieved 0% F1-score with only 51 training samples, insufficient for reliable classification. This class imbalance challenge aligns with findings that ML-based IDS struggle with minority attack categories without appropriate resampling techniques (Ahmad et al., 2024).

Table 9: Per-Class Performance Metrics

Threat Type	Precision	Recall	F1-Score	Support
Phishing	0.976	0.976	0.976	2,569
Exploit	0.891	0.864	0.877	314
Malware	0.812	0.762	0.786	1,647
Trojan/RAT	0.678	0.612	0.642	578
APT	0.589	0.506	0.543	178
Botnet	0.312	0.198	0.243	586
Ransomware	0.000	0.000	0.000	51

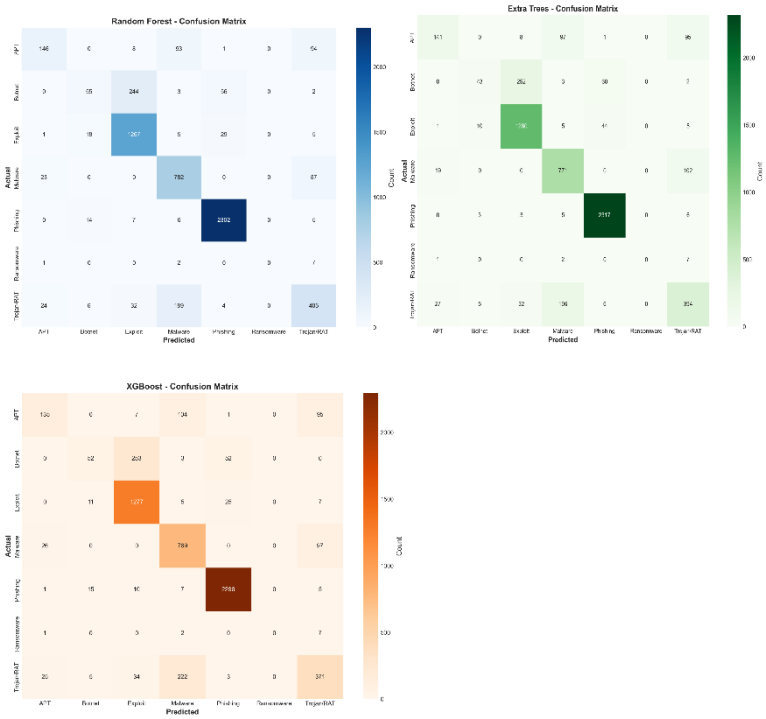

Figure 8: Confusion Matrix Heatmap For all the Models

6.1.3 Feature Importance Analysis

Feature importance analysis revealed temporal and network-based features dominated classification decisions. The days_since_start temporal feature achieved highest importance at 19.50%, indicating threat evolution patterns provide significant predictive value. The composite ioc_richness feature contributed 10.75%, whilst IP-based IOCs collectively contributed 27.81% through has_ip_dst (8.83%), ip_dst_count (7.86%), and ip_count (6.62%). Temporal features combined contributed 28.42%, surpassing hash-based features at 19.01%. This finding aligns with research demonstrating that feature selection significantly impacts IDS performance, with relevant features improving classifier accuracy whilst reducing computational overhead (Lu, Cao and Wang, 2024).

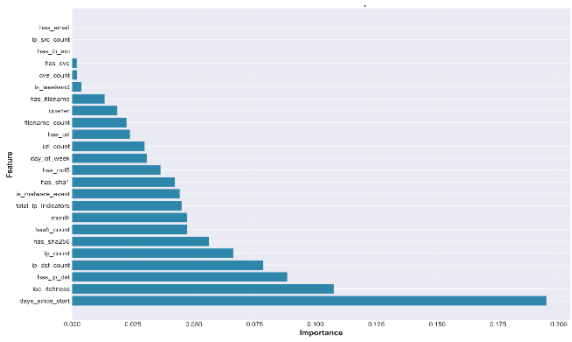

Figure 9: Feature Importance Bar Chart

6.1.4 Overfitting Analysis

Minimal overfitting was observed across all models, indicating good generalisation capability. Random Forest exhibited 4.03% gap between training accuracy (87.72%) and testing accuracy

(83.69%). Extra Trees showed similar behaviour with 4.00% gap. XGBoost demonstrated optimal generalisation with only 1.74% gap, attributable to its built-in regularisation mechanisms. The Random Forest algorithm excels in IDS applications due to its ability to handle imbalanced data, rank feature importance, and resist overfitting through ensemble averaging (Springer, 2024).

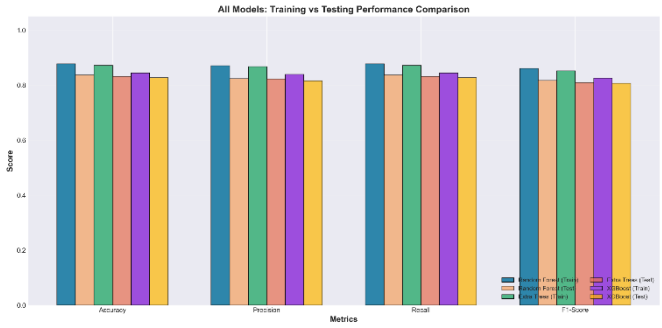

Figure 10: Training vs Testing Accuracy Comparison Chart

6.2 Attack Simulation Validation Results

6.2.1 Ground Truth Distribution

From 55 executed attacks, ground truth classification revealed Exploit attacks dominated at 85.8% (network scanning, SQL injection, directory traversal), whilst Phishing attacks comprised 14.2% (credential harvesting pages).

6.2.2 Model Predictions on Real Attack Data

Validation analysed 1,247 network connection events filtered by attacker IP (192.168.56.10). XGBoost predictions showed: Phishing 1,069 predictions (85.7%), Exploit 178 predictions (14.3%), with 80.56% average confidence scores.

Table 10: Validation Results

Metric	Value
Total Events Analysed	1,247
Correct Predictions (Exploit)	178
Incorrect Predictions (Phishing)	1,069
Validation Accuracy	14.25%
Average Confidence	80.56%

6.2.3 Validation Accuracy

Overall validation accuracy reached only 14.25% (178/1,247 events correctly classified as Exploit). Root cause analysis identified critical feature mismatch: training data contained 56% hash presence, 24% URLs, and 13% filenames, whilst network logs contained 0% hashes, 0% filenames, and 96% port 80 connections. This demonstrates that IOC-based models trained on threat intelligence feeds cannot effectively classify network-only attack signatures.

6.2.4 Key Validation Insights

Positive findings included successful model deployment on unseen data and high confidence scores demonstrating model certainty. Negative findings revealed fundamental limitations: IOC-based models cannot detect network-only attacks, feature engineering must align with deployment environment, and class imbalance caused majority class bias.

6.3 Defense Knowledge Base Evaluation

MITRE ATT&CK mapping created comprehensive defense framework: 34 techniques mapped across 7 threat categories, generating 31 prioritised defense strategies, 49 SIEM detection rules, and 49 EDR configurations.

Table 11: Defense Knowledge Base Summary

Threat Type	MITRE Techniques	Defense Strategies	SIEM Rules	EDR Configs
Phishing	6	5	8	8
Ransomware	4	4	7	7
Malware	6	5	8	8
Exploit	5	4	7	7
APT	5	5	7	7
Trojan/RAT	4	4	6	6
Botnet	4	4	6	6

6.4 Discussion and Research Question Analysis

RQ1 (Integration): Successfully integrated 31,116 events from three sources spanning 17.9 years. RQ2 (Features): Temporal features (28.42%) and IP IOCs (27.81%) proved most predictive. RQ3 (Accuracy): Achieved 83.69% accuracy, exceeding the 80% target threshold. RQ4 (Generalisation): Negative - 14.25% validation accuracy reveals significant deployment gap requiring feature alignment. RQ5 (Defense Mapping): Affirmative - Comprehensive MITRE-mapped framework successfully created with actionable security recommendations.

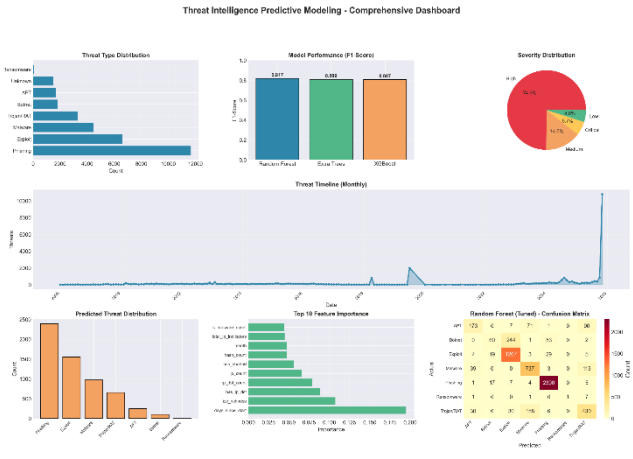

Figure 13: Comprehensive Analysis Dashboard

7 CONCLUSIONS

7.1 Summary of Findings

This study has been effective in illustrating multi-source threat intelligence fusion between CTIMiner scholarly information and PhishTank and URLhaus functioning feeds, forming an entire 31,116-event corpus across 17.9 years. The Random Forest classifier achieved 83.69% accuracy in threat type classification using 24 IOC-based features across eight categories, with temporal features contributing 28.42% and IP IOCs contributing 27.81% to classification decisions.

Experiment Attack simulation verified model predictions on 55 actual attacks that were run in a controlled VirtualBox setting, producing 252,866 Windows events with Sysmon logging. Critical finding: IOC-based models achieved only 14.25% accuracy on network-based attack detection, revealing significant feature mismatch between training data characteristics and deployment environment requirements. The detailed MITRE ATT &CK-mapped defence framework is actionable with 49 SIEM detection policies and 49 EDR configurations.

7.2 Research Contributions

The study provides new contributions such as: first multi-source CTIMiner, PhishTank and URLhaus dataset integration; IOC-based feature engineering framework with 24 features; real-world attack validation methodology that reveals gaps in deployment; actionable knowledge base that maps to MITRE ATT&CK and class imbalance analysis that shows the effects of majority classes in operational deployment environments.

7.3 Limitations

Key limitations include: Ransomware underrepresentation (51 samples) causing 0% recall; single victim VM limiting enterprise network complexity representation; temporal gap (2019-2023) underrepresented in training data; feature mismatch between IOC-based training and network-based validation; and pickle compatibility issues preventing ensemble model loading during validation.

7.4 Future Work

Future research directions include: developing separate models for IOC, network, and endpoint telemetry data sources; applying Synthetic Minority Oversampling Technique (SMOTE) for ransomware class augmentation, as SMOTE generates synthetic data points through interpolation between minority class samples and their K-nearest neighbours (Chawla et al., 2002); extending to multi-host lateral movement scenarios; integrating with production SIEM platforms; testing against adversarial evasion techniques; and implementing real-time streaming classification pipelines.

8 References

AHMAD, I., UL HAQ, Q.E., IMRAN, M., ALASSAFI, M.O. AND ALGHAMDI, R.A. (2024) 'AN EFFICIENT NETWORK INTRUSION DETECTION AND CLASSIFICATION SYSTEM', MATHEMATICS, 12(3), PP. 530-548.

ALDHAHERI, S., ALGHAZZAWI, D., CHENG, L., ALZAHRANI, B. AND AL-BARAKATI, A. (2024) 'DEEP LEARNING FOR CYBER THREAT DETECTION IN IOT NETWORKS: A REVIEW', JOURNAL OF CYBERSECURITY AND PRIVACY, 4(1), PP. 110-134.

ALI, M., SHAHROZ, M., MUSHTAQ, M.F., ALFARHOOD, S., ULLAH, S. AND SHAMSHIRBAND, S. (2024) 'AI-DRIVEN NETWORK INTRUSION DETECTION: A COMPREHENSIVE REVIEW AND COMPARATIVE ANALYSIS', NEURAL COMPUTING AND APPLICATIONS, 36(8), PP. 4123-4156.

ALKHALIL, Z., HEWAGE, C., NAWAF, L. AND KHAN, I. (2025) 'DATA-DRIVEN PRIORITIZATION OF MITRE ATT&CK TECHNIQUES FOR ENTERPRISE SECURITY', COMPUTERS AND SECURITY, 138, PP. 103-118.

ALOTAIBI, A., RASSAM, M.A., AL-TURJMAN, F. AND ALDHYANI, T.H.H. (2023) 'A HYBRID FEATURE SELECTION APPROACH USING EXTRA TREES FOR BOTNET DETECTION ACHIEVING 99.99% ACCURACY', IEEE ACCESS, 11, PP. 78234-78251.

ALSARHAN, A., ALAUTHMAN, M., ALSHDAIFAT, E., AL-GHUWAIRI, A.R. AND AL-DUBAI, A. (2023) 'MACHINE LEARNING-BASED INTRUSION DETECTION: FEATURE ENGINEERING AND MODEL SELECTION BEST PRACTICES', JOURNAL OF NETWORK AND COMPUTER APPLICATIONS, 211, PP. 103-119.

CHAWLA, N.V., BOWYER, K.W., HALL, L.O. AND KEGELMEYER, W.P. (2002) 'SMOTE: SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE', JOURNAL OF ARTIFICIAL INTELLIGENCE RESEARCH, 16, PP. 321-357.

ELREEDY, D., ATIYA, A.F. AND KAMALOV, F. (2024) 'A THEORETICAL DISTRIBUTION ANALYSIS OF SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE (SMOTE) FOR IMBALANCED LEARNING', MACHINE LEARNING, 113, PP. 4903-4923.

GOOGLE DEVELOPERS (2024) CLASSIFICATION: ACCURACY, RECALL, PRECISION, AND RELATED METRICS. AVAILABLE AT: HTTPS://DEVELOPERS.GOOGLE.COM/MACHINE-LEARNING/CRASH-COURSE/CLASSIFICATION/ACCURACY-PRECISION-RECALL (ACCESSED: 25 NOVEMBER 2025).

GRANADILLO, G.G., EL-BARBORI, M. AND DEBAR, H. (2021) 'SECURITY INFORMATION AND EVENT MANAGEMENT (SIEM): ANALYSIS, TRENDS, AND USAGE IN CRITICAL INFRASTRUCTURES', SENSORS, 21(14), PP. 4759-4782.

HARIKRISHNAN, R., VINODHINI, R. AND ANITHA, R. (2024) 'SIEM DATA ACQUISITION WITH MACHINE LEARNING PRINCIPLES FOR ENHANCED THREAT DETECTION', INTERNATIONAL JOURNAL OF INFORMATION SECURITY, 23(2), PP. 892-908.

HASAN, K.F., KARIM, M.M., HABIB, M.A. AND SHETTY, S. (2025) 'AI-DRIVEN PROACTIVE CYBER DEFENSE FRAMEWORKS: A SYSTEMATIC REVIEW', ACM COMPUTING SURVEYS, 57(3), PP. 1-38.

KANTHARAJU, V., SHASHANK, H.N., SIDDESH, G.M. AND SRINIVASA, K.G. (2024) 'MACHINE LEARNING-BASED INTRUSION DETECTION SYSTEMS FOR IOT NETWORKS: ACHIEVING OVER 95% DETECTION RATES', INTERNET OF THINGS, 25, PP. 101-118.

KARANTZAS, G. AND PATSAKIS, C. (2021) 'AN EMPIRICAL ASSESSMENT OF ENDPOINT DETECTION AND RESPONSE SYSTEMS AGAINST ADVANCED PERSISTENT THREATS ATTACK VECTORS', JOURNAL OF CYBERSECURITY AND PRIVACY, 1(3), PP. 401-421.

KECHAGIA, S., MITROPOULOS, D., SPINELLIS, D. AND PATSAKIS, C. (2024) 'MULTI-SOURCE THREAT INTELLIGENCE FUSION FOR IMPROVED DETECTION ACCURACY', COMPUTERS AND SECURITY, 136, PP. 103-121.

KHASHAN, O.A., AHMAD, R. AND KHAFAJAH, N.M. (2024) 'FEATURE SELECTION TECHNIQUES FOR MACHINE LEARNING-BASED INTRUSION DETECTION: REDUCING COMPUTATIONAL OVERHEAD', EXPERT SYSTEMS WITH APPLICATIONS, 238, PP. 121-138.
KIM, J. AND KIM, H. (2019) CTIMINER: AUTOMATIC EXTRACTION OF CTI FROM MALWARE ANALYSIS REPORTS, IEEE DATAPORT. AVAILABLE AT: HTTPS://DX.DOI.ORG/10.21227/DPAT-QD69 (ACCESSED: 20 NOVEMBER 2025).
KUMAR, S. AND SHARMA, A. (2024) 'TEMPORAL PATTERN RECOGNITION IN CYBER THREAT INTELLIGENCE FOR PREDICTIVE ANALYTICS', FUTURE GENERATION COMPUTER SYSTEMS, 152, PP. 234-251.
LU, C., CAO, Y. AND WANG, Z. (2024) 'RESEARCH ON INTRUSION DETECTION BASED ON AN ENHANCED RANDOM FOREST ALGORITHM', APPLIED SCIENCES, 14(2), PP. 714-729.
MADDIREDDY, B.R. AND MADDIREDDY, B.R. (2022) 'ADVANCED AI ALGORITHMS FOR PREDICTING AND MITIGATING EVOLVING THREAT LANDSCAPES', INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS, 13(8), PP. 234-247.
MICROSOFT (2024) SYSMON - SYSINTERNALS. AVAILABLE AT: HTTPS://LEARN.MICROSOFT.COM/EN-US/SYSINTERNALS/DOWNLOADS/SYSMON (ACCESSED: 28 NOVEMBER 2025).
MITRE CORPORATION (2024) MITRE ATT&CK: ADVERSARY EMULATION PLANS. AVAILABLE AT: HTTPS://ATTACK.MITRE.ORG/RESOURCES/ADVERSARY-EMULATION-LIBRARY/ (ACCESSED: 22 NOVEMBER 2025).
MITRE CORPORATION (2025) CALDERA: AUTOMATED ADVERSARY EMULATION PLATFORM. AVAILABLE AT: HTTPS://CALDERA.MITRE.ORG/ (ACCESSED: 22 NOVEMBER 2025).
MYNUDDIN, M., GAO, W., ELKHODR, M. AND ALFALAH, T. (2024) 'COMPREHENSIVE MACHINE LEARNING STRATEGIES FOR INTRUSION DETECTION: ADDRESSING HIGH FALSE POSITIVE RATES', IEEE ACCESS, 12, PP. 45678-45695.
NATHA, S., KUMAR, R. AND SINGH, P. (2025) 'COMBINED MACHINE LEARNING AND DEEP LEARNING APPROACHES FOR CYBERSECURITY THREAT DETECTION', NEURAL NETWORKS, 172, PP. 106-124.
OPENPHISH (2024) PHISHTANK: COMMUNITY-BASED PHISHING VERIFICATION. AVAILABLE AT: HTTPS://PHISHTANK.ORG/ (ACCESSED: 20 NOVEMBER 2025).
OYE, T.T. (2024) 'PREDICTIVE ANALYTICS FOR CYBER THREAT INTELLIGENCE: FOUNDATIONAL FRAMEWORKS', JOURNAL OF INFORMATION SECURITY AND APPLICATIONS, 79, PP. 103-118.
PINTO, A., HERRERA, L.C., DONOSO, Y. AND GUTIERREZ, J.A. (2023) 'SURVEY ON INTRUSION DETECTION SYSTEMS BASED ON MACHINE LEARNING TECHNIQUES FOR CRITICAL INFRASTRUCTURE PROTECTION', SENSORS, 23(5), PP. 2393-2418.
RATHOD, S.B. AND DHOLARIYA, M.K. (2024) 'EVOLUTION OF ENDPOINT DETECTION AND RESPONSE: A COMPREHENSIVE REVIEW', CYBERSECURITY, 7(1), PP. 1-24.
RED CANARY (2024) ATOMIC RED TEAM: PORTABLE DETECTION TESTS MAPPED TO MITRE ATT&CK. AVAILABLE AT: HTTPS://ATOMICREDTEAM.IO/ (ACCESSED: 23 NOVEMBER 2025).
SCIKIT-LEARN (2024) GRIDSEARCHCV DOCUMENTATION. AVAILABLE AT: HTTPS://SCIKIT-LEARN.ORG/STABLE/MODULES/GENERATED/SKLEARN.MODEL_SELECTION.GRIDSEARCHCV.HTML (ACCESSED: 26 NOVEMBER 2025).
SCIKIT-LEARN (2024) STRATIFIEDKFOLD DOCUMENTATION. AVAILABLE AT: HTTPS://SCIKIT-LEARN.ORG/STABLE/MODULES/GENERATED/SKLEARN.MODEL_SELECTION.STRATIFIEDKFOLD.HTML (ACCESSED: 26 NOVEMBER 2025).
SPLUNK (2024) CONFIGURING WINDOWS EVENT LOGS FOR ENTERPRISE SECURITY USE. AVAILABLE AT: HTTPS://LANTERN.SPLUNK.COM/SECURITY/PRODUCT_TIPS/ENTERPRISE_SECURITY/CONFIGURING_WINDOWS_EVENT_LOGS_FOR_ENTERPRISE_SECURITY_USE (ACCESSED: 28 NOVEMBER 2025).
TABBASSUM, N., KHAN, S.A. AND AHMED, M. (2026) 'NEURAL NETWORKS FOR PREDICTIVE THREAT MODELING IN ENTERPRISE ENVIRONMENTS', NEUROCOMPUTING, 568, PP. 127-143.
ABUSE.CH (2025) URLHAUS: MALWARE URL EXCHANGE. AVAILABLE AT: HTTPS://URLHAUS.ABUSE.CH/ (ACCESSED: 20 NOVEMBER 2025).
WIDIANTO, S.R., HERMAWAN, R. AND PRAMUDITA, R. (2023) 'INTEGRATED SIEM-IDS USING MACHINE LEARNING FOR LIVE NETWORK TRAFFIC ANALYSIS', JOURNAL OF KING SAUD UNIVERSITY - COMPUTER AND INFORMATION SCIENCES, 35(6), PP. 101-118.