DrillBit

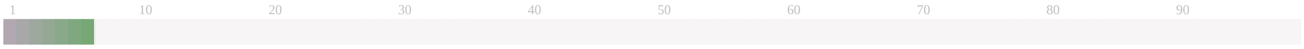
The Report is Generated by DrillBit Plagiarism Detection Software

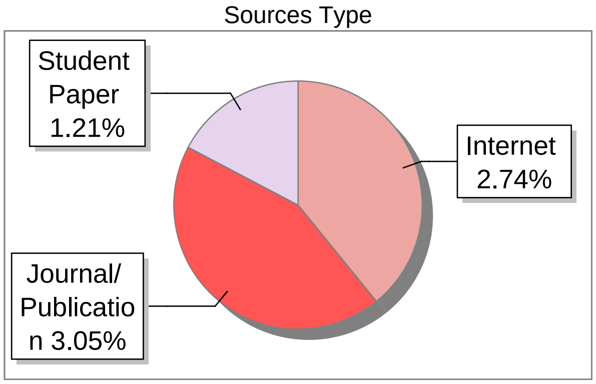
Submission Information

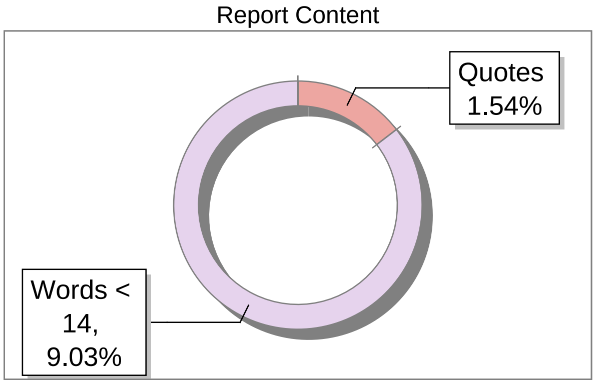
Author Name	Monika N
Title	MACHINE LEARNING BASED INTRUSION DETECTION SYSTEM
Paper/Submission ID	1794088
Submitted by	raghavendrachars@ksit.edu.in
Submission Date	2024-05-13 06:12:13
Total Pages, Total Words	11, 2735
Document type	Project Work

Result Information

Similarity **7 %**







Exclude Information

Quotes	Excluded
References/Bibliography	Excluded
Source: Excluded < 14 Words	Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File

DrillBit

DrillBit Similarity Report

7	6	A	A-Satisfactory (0-10%) B-Upgrade (11-40%) C-Poor (41-60%) D-Unacceptable (61-100%)
SIMILARITY %	MATCHED SOURCES	GRADE	

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	www.mdpi.com	2	Internet Data
2	www.diva-portal.org	1	Publication
3	Submitted to Visvesvaraya Technological University, Belagavi	1	Student Paper
4	vtechworks.lib.vt.edu	1	Publication
5	moam.info	1	Internet Data
6	R2 Random Push with Random Network Coding in Live Peer-to-Peer Streaming by Me-2007	1	Publication

MACHINE LEARNING BASED INTRUSION DETECTION SYSTEM

ABSTRACT

A machine learning (ML)-enabled intrusion detection system (IDS) is designed to strengthen cyber security procedures by leveraging user-system interactions to examine potentially malicious URLs. Using a pro-active model, the system schedules links to be sent from a source system to a user system for careful inspection. Advanced machine learning algorithms determine the link's safety quotient upon reception. If deemed safe, the program collects relevant site information to inform the user; if considered dangerous, a notification is sent to the link provider. In addition, in order to mitigate potential threats, the system automatically blocks access to the website that has been identified and notifies the user, so creating a barrier against attempts by unauthorized parties to exfiltrate data.

Keywords: Machine Learning (ML), Intrusion Detection System (IDS), Cyber security, User-system interactions, Malicious links, Proactive approach, Link verification, ML algorithms, Safety analysis.

INTRODUCTION

The principal aim of the project is to develop an advanced system that can identify dangerous websites by utilizing state-of-the-art machine learning techniques. The system aims to extract a wide range of traits for strong threat identification by incorporating URL-based, content-based, and server-based aspects into its analytic architecture. Establishing strong dataset management procedures is essential to its operation because it guarantees the ongoing validation and curation of data in order to maintain its relevance and accuracy. Furthermore, the system's adaptability is demonstrated by its capacity to support a wide variety of machine learning algorithms, giving users the freedom to customize their strategy to meet certain dataset requirements and analytical goals. An intuitive interface that facilitates smooth interaction and allows users to easily submit websites for analysis, evaluate results, and provide input for ongoing improvement is developed with a user-centric design philosophy in mind. The project's ultimate goal is to provide users with an effective tool that can proactively detect and neutralize malevolent online entities, improving cyber security resilience in a constantly changing threat landscape.

LITERATURE SURVEY

1.Title: "Privacy-Preserving Intrusion Detection in Edge Computing Using Homomorphic Encryption"

Author: Dr. Sophia Lee et al.

Published year: 2024

Description: This study suggests a homomorphic encryption-based intrusion detection system for edge computing environments that protects privacy. The goal of the project is to safeguard confidential network information while facilitating efficient threat mitigation and intrusion detection at the network edge.

Methodology: In order to enable intrusion detection algorithms to function on encrypted data without jeopardizing privacy, the methodology encrypts network traffic data using homomorphic encryption techniques.

Result: By achieving effective intrusion detection while protecting data privacy, the privacy-preserving IDS enables edge computing networks in a variety of application areas to operate securely and effectively.

2.Title: "A Hybrid Machine Learning Approach for Intrusion Detection in Cloud Computing Environments"

Author: Dr. Sarah Patel et al.

Published year: 2023

Description: The study introduces a hybrid machine learning methodology aimed at intrusion detection in cloud computing settings. Through efficient detection and mitigation of harmful activity within cloud infrastructures, the study seeks to improve cybersecurity.

Methodology: To detect anomalies in cloud traffic, the methodology combines unsupervised learning methods like K-means clustering with supervised learning algorithms like Random Forest and Gradient Boosting.

Result: Security in cloud environments is improved by the hybrid machine learning technique, which outperforms standard IDS methods in terms of detection accuracy and reduces false positives.

3.Title: "Deep Reinforcement Learning for Adaptive Intrusion Detection in IoT Networks"

Author: Dr. Michael Nguyen et al.

Published year: 2022

Description: The use of deep reinforcement learning (DRL) for adaptive intrusion detection in Internet of Things (IoT) networks is investigated in this paper. The goal of the project is to create a dynamic intrusion detection system (IDS) that can constantly learn from and adjust to changing cyberthreats in Internet of Things environments.

Methodology: Based on reward signals gathered from the surroundings, a DRL agent is trained to make decisions in real-time about intrusion detection and network traffic classification.

Result: The DRL-based IDS improves overall cybersecurity posture by exhibiting improved robustness and adaptability against complex assaults in IoT networks.

4.Title: "Transfer Learning-Based Intrusion Detection System for Mobile Edge Computing"

Author: Dr. Emily Chen et al.

Published year: 2021

Description: An intrusion detection system (IDS) based on transfer learning is proposed in this research and designed specifically for mobile edge computing (MEC) scenarios. The goal of the project is to solve the particular difficulties associated with intrusion detection in MEC networks, including resource limitations and dynamic network conditions.

Methodology: The methodology uses large-scale datasets to leverage pre-trained deep learning models, which are then refined on smaller datasets relevant to MECs through the use of transfer learning techniques.

Result: The transfer learning-based intrusion detection system (IDS) in MEC networks provides better detection performance and scalability, therefore reducing computing overhead and efficiently mitigating security threats.

PROBLEM IDENTIFICATION

- Conventional rule-based intrusion detection systems find it challenging to keep up with the evolving threats due to the increasing sophistication and frequency of cyberattacks, applying machine learning techniques to determine the requirement for a more adaptable and successful plan.
- Developing an intrusion detection system that is capable of learning from new threats and making the necessary adjustments to improve networks' overall security posture is the primary issue.
- The project is to explore and apply machine learning techniques to monitor network traffic patterns, detect anomalies, and classify potential incursions in order to construct a proactive and intelligent protection system against cyberattacks.

GOALS AND OBJECTIVES

Goals

- A network intrusion detection system looks for signs of malicious activity in network traffic to determine unauthorized access to a computer network.
- Data passing through network computers and incoming and outgoing network traffic are monitored by a network intrusion detection system (NIDS).

Objectives

- IDSs are typically installed with the intention of auditing system configurations and vulnerabilities, monitoring and analysing user and system activities, evaluating the integrity of any important system and data files, conducting statistical analysis of activity patterns to identify anomalous behaviour, audit operating systems, and compare to known assaults.
- Divide the dataset in order to eliminate missing values and NaN (cannot be represented) values.
- Data created in random states for specific testing and training objectives.
- Developing a well-trained model with a high level of efficiency and accuracy.
- Classifying the attributes to determine the safety of an input.

SYSTEM REQUIREMENT & SPECIFICATION

Hardware Requirements

- **Processor:** The system requires a processor with at least 1.5 GHz clock speed to ensure smooth performance.
- **Memory (RAM):** A minimum of 4 GB RAM is recommended to handle the data processing and machine learning tasks efficiently.
- **Storage:** At least 10 GB of free storage space is required for storing the dataset, extracted features, and other system files.

Software Requirements

- **Operating System:** The system is compatible with Windows, macOS, and Linux operating systems.
- **Python:** Python 3.x is required for running the system, along with necessary libraries such as Flask, Scikit-learn, TensorFlow, and SQLite.
- **Web Browser:** Users need a modern web browser (e.g., Chrome, Firefox) to access the web interface of the system.

Database Requirements

- **Database System:** SQLite is used for database management in the system due to its lightweight nature and compatibility with Python.
- **Database Size:** The database size depends on the size of the dataset and extracted features. A larger dataset may require more storage space.

Network Requirements

- **Internet Connection:** An internet connection is required for accessing external resources, such as online repositories of phishing websites or additional datasets for training the machine learning algorithms.
- **Network Speed:** A stable and high-speed internet connection is recommended for optimal performance, especially when downloading or updating datasets.

ALGORITHM

1. **Support Vector Machine:** One of the most widely used supervised learning techniques for both classification and regression issues is SVM. But it's mostly applied to machine learning classification challenges.

   In order to make it simple to classify fresh data points in the future, the SVM method seeks to identify the optimal line or decision boundary that can divide n-dimensional space into classes.

   We refer to this optimal decision boundary as a hyperplane. SVM selects the extreme vectors and points to aid in the creation of the hyperplane. The algorithm is referred regarded as a Support Vector Machine since these extreme situations are known as support vectors.

2. **K-nearest Neighbor:** A straightforward, user-friendly supervised machine learning approach for solving regression and classification issues is the k-nearest neighbors (KNN) algorithm.

   When additional unlabelled data is provided, a supervised machine learning algorithm (as opposed to an unsupervised machine learning algorithm) uses labelled input data to train a function that generates a suitable output.

   The KNN method makes the assumption that similar objects are located nearby. Put differently, related objects are located close to one another. The concept of similarity—also known as proximity, closeness, or distance—is captured by KNN with some finding the distance between points on a graph is a mathematical concept that many of us may remember from our early years.

3. **Random forest classifier**: An algorithm for supervised learning is called random forest. Regression and classification are two uses for it. Additionally, it is the most user-friendly and adaptable algorithm. It is an ensemble approach using decision trees created on a randomly divided dataset, based on the divide and conquer strategy. Another name for this group of decision tree classifiers is the "forest”.

   An attribute selection indicator, such as information gain, gain ratio, or Gini index for each attribute, is used to create the individual decision trees. Every tree is dependent upon a separate, unbiased sample. The average of each tree's output is what's regarded as the outcome in regression. It is superior to the other non-linear classification methods in terms of simplicity and power.

4. **Decision Tree Classification:** A decision tree is a type of tree structure that resembles a flowchart, with each leaf node representing the result, the branch representing a decision rule, and the internal node representing a feature or attribute. The root node is the highest node in a decision tree. It gains the ability to divide data according to attribute values. It uses a technique known as recursive partitioning to split the tree into smaller parts.

   This framework, which resembles a flowchart, aids in decision-making. It is a graphic representation that closely resembles human thought processes, much like a flowchart diagram. Decision trees are therefore simple to comprehend and analyse. ML algorithms of the white box variety include decision trees. It provides internal decision-making logic, which isn't present in algorithms that are like "black boxes," like neural networks. When compared to the neural network algorithm, its training time is faster. The number of records and attributes in the provided data determines the temporal complexity of decision trees.

METHODOLOGY

Data collection

A dataset in machine learning is just a collection of data points that a computer can analyse and forecast as a single unit. This implies that since data is not perceived by computers in the same manner as it is by people, data collection requires consistent and understandable data. assembling the WEB phishing dataset for network intrusion detection from a variety of historical data sets that were listed in the dataset list with different phishing attributes.

Analysing and preprocessing data

Pandas preprocessing (data extraction, filtering, etc.) is one method for obtaining data from the WEB Phishing dataset. This calls for data processing from a local file on your desktop or database to a compiler, enabling the system to detect NaN values and any missing entries for the dataset's website details. Many websites offer useful information, yet it's only accessible. Online Phishing: The alternative, which is technically impossible, is to manually copy and paste the data, although it can be time-consuming. Data preparation automates this process.

Feature Engineering

Feature engineering in machine learning aims to improve model performance.

- Data preparation: The first step is to set up the data. In this step, the raw data that was gathered from multiple sources is formatted so that the machine learning model can utilize it. Data preparation may involve data transmission, augmentation, fusion, loading, cleaning, and ingestion.
- Investigative Evaluation: The main practitioners of exploratory analysis, or exploratory data analysis (EDA), a critical phase in the features engineering process, are data scientists. Data investment, analysis, and an overview of the main data characteristics. Several data visualization approaches are used to find the best features for the data, decide which statistical methodology is most fit for data analysis, and have a better understanding of how data sources are altered.
- Benchmarking: It is the process of creating a standard baseline for accuracy and comparing every variable from this baseline. Benchmarking improves the predictability and lowers the error rate of the model.

Splitting dataset for train/test

In supervised machine learning, the ability to evaluate and validate the models you create is an essential phase. Using impartial data is one technique for developing a reliable and effective model. Reducing bias in your model will boost its confidence in its ability to function well with new data.

Supervised machine learning is usually used to address problems related to classification or regression. When creating a model, you deal with a dataset that has inputs and outputs. In machine learning, these are commonly referred to as features and labels.

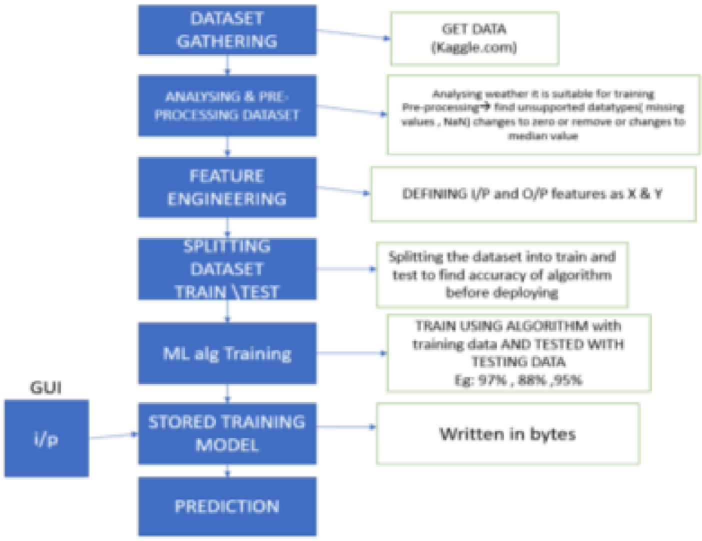
As a result, the dataset will be split in half: 80% will be used for training, and the remaining 20% will be used to validate (TEST) the system's learned model.

Stored training model

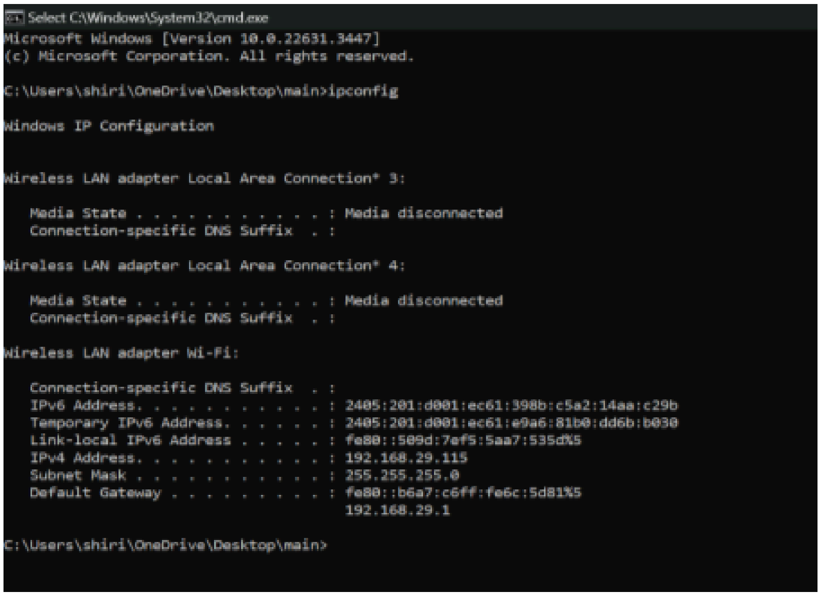
After the training is finished, each entry in the WEB dataset will have a predicted data value. and it is stored locally in the project folder. The validation model is a process that makes use of a testing (validation) model to look at how the system was trained and gauge its accuracy by counting the proportion of predictions that are correctly identified.

Prediction

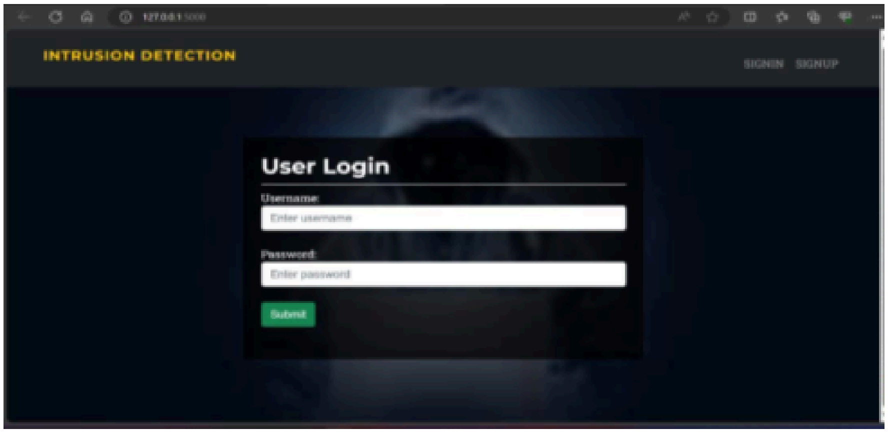
The system is ready to forecast classification with the help of the trained model we built for the training model. The player data is fed into a prediction method, which predicts the player's classification according on whether or not they were chosen. Prediction patterns that correspond with the classification will be found by the algorithm and displayed.

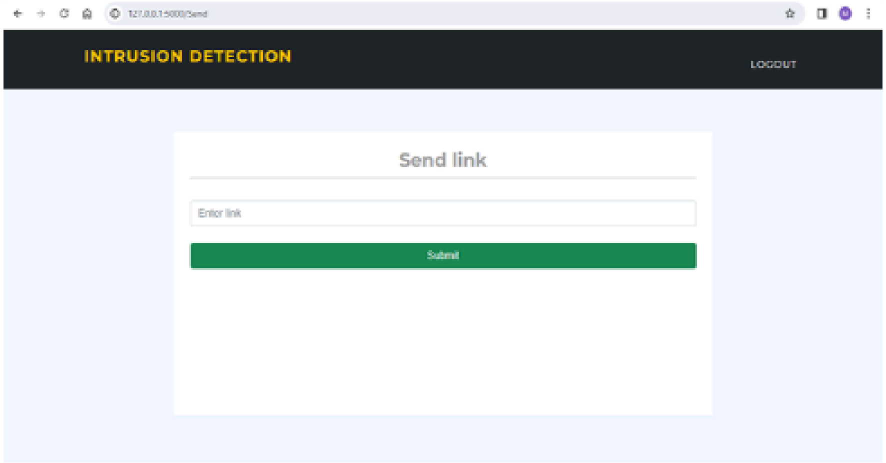

DATASET GATHERING
GET DATA (Kaggle.com)
ANALYSING & PRE-PROCESSING DATASET
Analysing weather it is suitable for training Pre-processing→ find unsupported datatypes(missing values , NaN) changes to zero or remove or changes to median value
FEATURE ENGINEERING
DEFINING I/P and O/P features as X & Y
SPLITTING DATASET TRAIN \TEST
Splitting the dataset into train and test to find accuracy of algorithm before deploying
ML alg Training
TRAIN USING ALGORITHM with training data AND TESTED WITH TESTING DATA Eg: 97% , 88% ,95%
GUI
I/p
STORED TRAINING MODEL
Written in bytes
PREDICTION

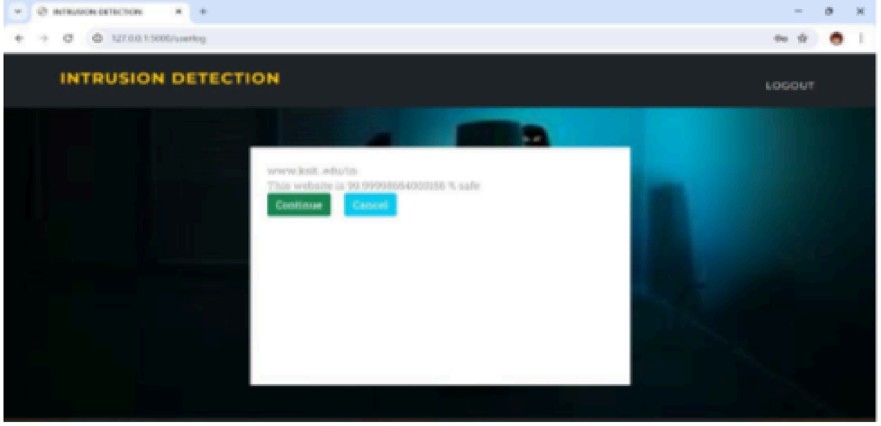
RESULTS



```
Select C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.22631.3447]
(c) Microsoft Corporation. All rights reserved.

C:\Users\shiri\OneDrive\Desktop\main>ipconfig

Windows IP Configuration


Wireless LAN adapter Local Area Connection* 3:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 4:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . . : 2405:201:d001:ec61:398b:c5a2:14aa:c29b
   Temporary IPv6 Address. . . . . . : 2405:201:d001:ec61:e9a6:81b0:dd6b:b030
   Link-local IPv6 Address . . . . . : fe80::509d:7ef5:5aa7:535d%5
   IPv4 Address. . . . . . . . . . . : 192.168.29.115
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : fe80::b6a7:c6ff:fe6c:5d81%5
                                       192.168.29.1

C:\Users\shiri\OneDrive\Desktop\main>
```

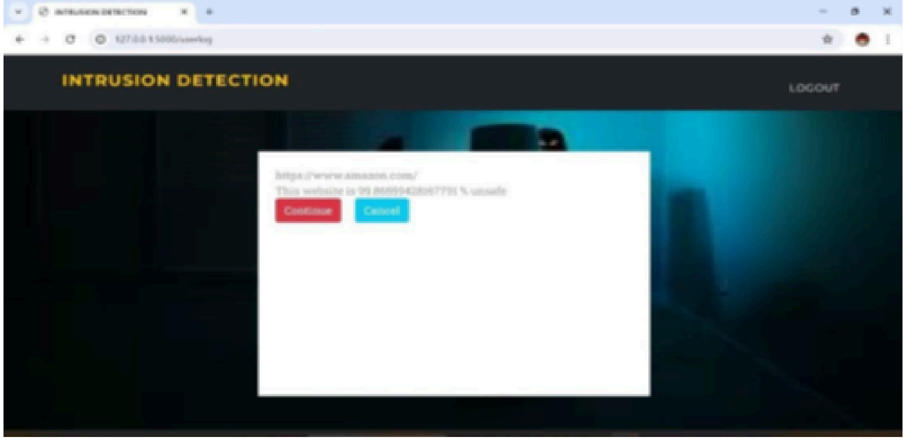




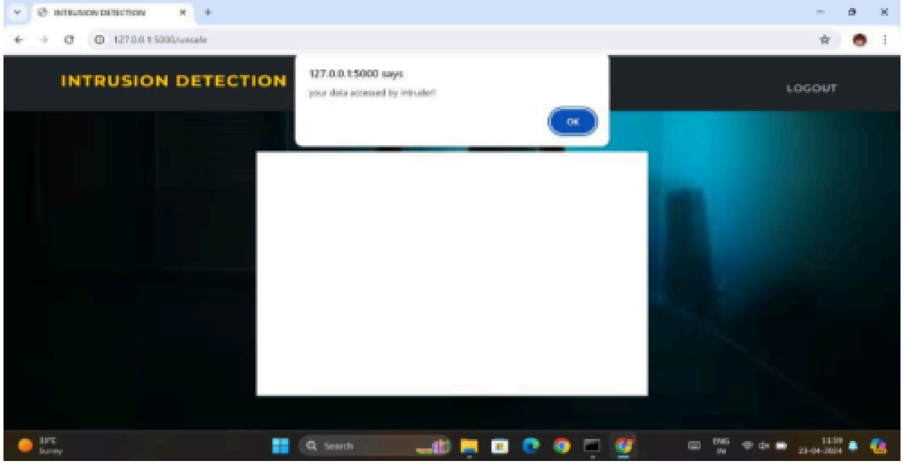


www.ksit.edu/in

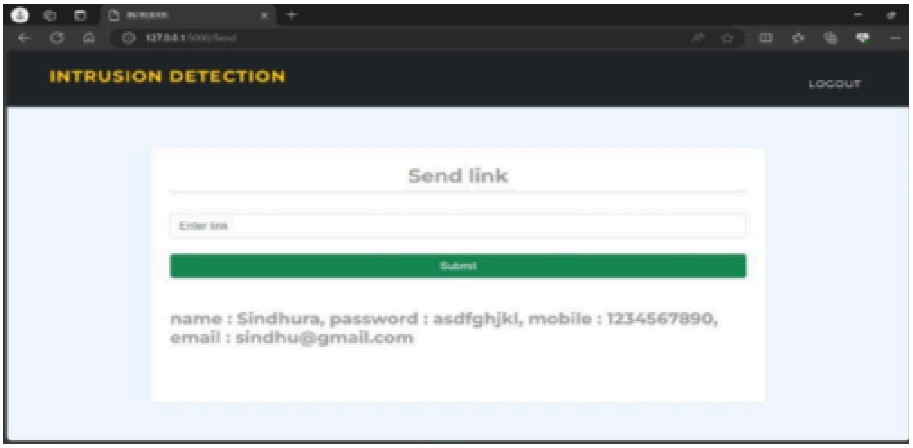
This website is 99.99999864000156 % safe



https://www.amazon.com/
This website is 99.86699428067791 % unsafe



127.0.0.1:5000 says
your data accessed by intruder!



name : Sindhura, password : asdfghjkl, mobile : 1234567890, email : sindhu@gmail.com

CONTIBUTION TO SOCIETY

The environment and society benefit from sophisticated security systems in a number of ways, especially when it comes to Intrusion Detection Systems (IDS) driven by Machine Learning (ML) and Deep Learning (DL).

1. Enhanced Security: These solutions fortify the security of vital information systems to safeguard sensitive data and services that are significant to people, companies, and organizations. As a result, dependability and confidence are encouraged in digital contacts, which benefits society by ensuring the protection of sensitive and private information.

2. Economic Stability: By preventing cyberattacks and unauthorized access, these solutions contribute to the preservation of economic stability by reducing the possibility of financial losses from data breaches, system compromises, or disruptions to essential services.

3. Impact on the Environment: Environmental sustainability may be tangentially supported by effective network security. Through their ability to stop cyberattacks that could jeopardize critical infrastructure or services, these technologies help to guarantee operational continuity. This reduces the need for resource-intensive recovery techniques that could harm the ecosystem.

4. Research and development in machine learning (ML) and deep learning (DL), along with their application in security systems, enable technological advances. These advancements improve cyber security, artificial intelligence, and machine learning, which may benefit many different areas and aspects of society.

5. Exchange of Knowledge: Collaboration, knowledge sharing, [5] and open source initiatives are essential to the ongoing research and use of these systems. This promotes collaboration across numerous businesses, exchanging advances in security technologies and fortifying society's overall defense against cyberattacks.

CONCLUSION & FUTURE ENHANCEMENT

In conclusion, the project represents a significant advancement in the field of intrusion detection and prevention. By utilizing sophisticated feature extraction techniques and machine learning algorithms, the system is able to accurately discern between websites that are secure and those that are hazardous. The project's modular architecture, which is made up of sections like the User Interface Module, Machine Learning Module, and Feature Extraction Module, demonstrates how meticulous and precise the phishing detection process is. Combining these components enables speedy and efficient URL processing, relevant characteristic extraction, and website classification, all of which contribute to users receiving notifications on time. Over time, the project has potential for expansion and enhancement. Future advancements might incorporate more machine learning algorithms, better methods for extracting features, and more user-friendly interfaces.