



VELAMMAL
INSTITUTE OF TECHNOLOGY
Chennai - Kolkatta Highway, Panchetti, Ponneri



**INSTITUTION'S
INNOVATION
COUNCIL**
(Ministry of HRD Initiative)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NETWORK TRAFFIC CYBER ATTACKS CLASSIFICATION USING SUPERVISED MACHINE LEARNING TECHNIQUES

TEAM MEMBERS :

1. FINO FRANKLIN J(113320104029)
2. FRINO FREDY(113320104030)
3. DHARANI KUMAR M(113320104023)

GUIDE :

Ms. PRATHEEBA R S, M.E.,
Assistant Professor

OBJECTIVES



=> The objective is to develop a machine learning model for cyber attack prediction.

=> This model aims to potentially replace updatable supervised machine learning classification models by predicting results with the best accuracy.

=> The prediction results are obtained by comparing supervised algorithms.

ABSTRACT



- ❑ Cyberattack classification through the utilization of supervised machine learning methods. The system is designed to categorize diverse cyber attacks by employing a meticulously curated dataset encompassing a wide array of attack types, including but not limited to malware, phishing, and distributed denial-of-service (DDoS) attacks.
- ❑ Feature extraction techniques are applied to both network traffic data and behavioral attributes, facilitating the training of a robust classification model. Various supervised learning algorithms, such as decision trees, support vector machines, and neural networks, are evaluated for their efficacy in accurately predicting attack categories.

ABSTRACT



- ❑ The training process involves labeling historical attack instances, enabling the model to discern intricate patterns and subtle differentiators among attack types. Regular model updates and retraining with new attack data ensure its relevance in dynamically evolving threat landscapes.
- ❑ The system's predictive accuracy empowers cybersecurity teams to swiftly identify and respond to cyber threats, thereby bolstering overall defense strategies. Through this research, we contribute to the proactive identification and mitigation of cyber attacks, ultimately fortifying digital security frameworks.

PROBLEM STATEMENT



With the rising complexity and diversity of cyber threats, there is a critical need for an advanced system utilizing supervised machine learning techniques to accurately categorize network traffic cyber-attacks. This system aims to empower cybersecurity teams in swiftly responding to and mitigating evolving cyber threats.

EXISTING SYSTEM



- **Security Approach:**

Employs invariants in Cyber-Physical Systems (CPS) for preventing and detecting cyber-attacks.

- **Invariant Generation:**

Derives invariants from operational data or system requirements/design documents.

- **Attack Detection Strategy:**

Addresses data-driven invariant shortcomings through adversarial attack demonstrations and proposes a solution by complementing them with design-driven invariants, showcasing effectiveness on a real water treatment testbed.

DISADVANTAGES OF EXISTING SYSTEM



- Higher time complexity for implementation process.
- Complexity and usability.
- Accuracy was low.
- Limited scalability.

PROPOSED SYSTEM



- Holistic Cyber Attack Classification:**

- Utilizes supervised machine learning and an extensive dataset for comprehensive classification of various cyber attacks.

- Robust Feature Set and Model Training:**

- Constructs a strong feature set from network data, logs, and attack patterns.
- Trains a classification model using algorithms like decision trees, support vector machines, or neural networks, refined with labeled historical data.

- Real-time Monitoring and Proactive Defense:**

- Swiftly deploys the model for real-time network monitoring, enabling rapid and accurate identification of cyber threats.
- Ensures continuous efficacy through regular updates and retraining to adapt to evolving attack methodologies, empowering proactive cyber defense.

ADVANTAGES OF PROPOSED SYSTEM



- We compared more than two algorithms to get a better accuracy level.
- We build a user-friendly web application.
- We improved the accuracy level and performance level.
- We implemented Machine Learning properly.

LITERATURE SURVEY

- Title : Cyber restoration of power systems: Concepts and methodology for resilient observability
- Author: Rui Yao and Bo chen
- Year : 2023
- Social network analysis is a basic mechanism to observe the behavior of a community in society. In the huge and complex social network formed using cyberspace or telecommunication technology, the identification or prediction of any kind of socio-technical attack is always difficult. This challenge creates an opportunity to explore different methodologies, concepts and algorithms used to identify these kinds of community on the basis of certain pattern, properties, structure and trend in their linkage. This paper tries to find the hidden information in huge social network by compressing it in small networks through apriori algorithm and then diagnosed using viterbi algorithm to predict the most probable pattern of conversation to be followed in the network and if this pattern matches with the existing pattern of criminals, terrorists and hijackers then it may be helpful to generate some kind of alert before crime.

LITERATURE SURVEY



- ❑ **Title** : An Investigation of Learning Model Technologies for Network Traffic Classification Design in Cyber Security Exercises
- ❑ **Author**: Younghoan Jang, Dong-wook Kim, Gun-yoon Shin, Seungjae Cho, Kwangsoo Kim , Jaesik Kang And Myung-mook Han.
- ❑ **Year** : 2023
- ❑ Since the 2000s, research on cyber training has been a major focus among cybersecurity and networking researchers. This study investigated network traffic classification technology for creating environments and scenarios for cyber training. The research categorized learning models into supervised, unsupervised, and reinforcement learning, conducting a detailed analysis of network traffic classification technology, methods, procedures, result interpretation, and approaches for each model. The study identified limitations and provided recommendations based on the learning method. It analyzed port-based, payload-based, and learning model-based network traffic classification methods, highlighting the necessity for further research in this area.

LITERATURE SURVEY



- **Title** : Deep Learning Based Attack Detection for Cyber-Physical System Cybersecurity: A Survey
- **Author** : Jun Zhang, , Lei Pan, Qing-Long Han, Chao Chen, Sheng Wen and Yang Xiang.
- **Year** : 2022
- This survey offers insights into the detection of cyber attacks in Cyber-Physical Systems (CPSs). A six-step deep learning (DL) methodology is proposed to summarize and analyze twenty recent papers. The methodology involves examining CPS scenarios, identifying cybersecurity issues, translating research problems into the ML/DL domain, constructing DL models, preparing datasets, and evaluating models. Cyber attacks continue to pose a significant threat to CPS security and safety. The reviewed works demonstrate the potential of DL models in exploiting CPS cyber data, with promising performance results. High-quality datasets contribute to the success of these models. Additionally, promising research topics include integration with blockchain, detection of advanced persistent threats, adoption of new ML and DL paradigms, prevention of adversarial and model extraction attacks, dataset enrichment, and application of additional performance metrics. Overall, there is optimism and confidence in the flourishing of research in this field.

LITERATURE SURVEY



❑ **Title** : Cyber security: Study on Attack, Threat, Vulnerability

❑ **Author:** Tushar P. Parikh, Dr. Ashok R. Patel

❑ **Year** : 2017

❑ Cybersecurity incidents often target computer-literate users, with new employees in organizations being particularly vulnerable. Attackers often seek personal identifiable information from these individuals. Psychological variables also contribute to user and network vulnerability. While technology plays a role in mitigating cyber attacks, human behavior, impulses, and psychological predispositions remain significant factors. Education can influence these aspects to reduce cyber attacks, although a comprehensive solution to overcome such threats is yet to be developed. Future work should focus on implementing cybersecurity models to decrease cyber attack threats and vulnerabilities in networks.

LITERATURE SURVEY



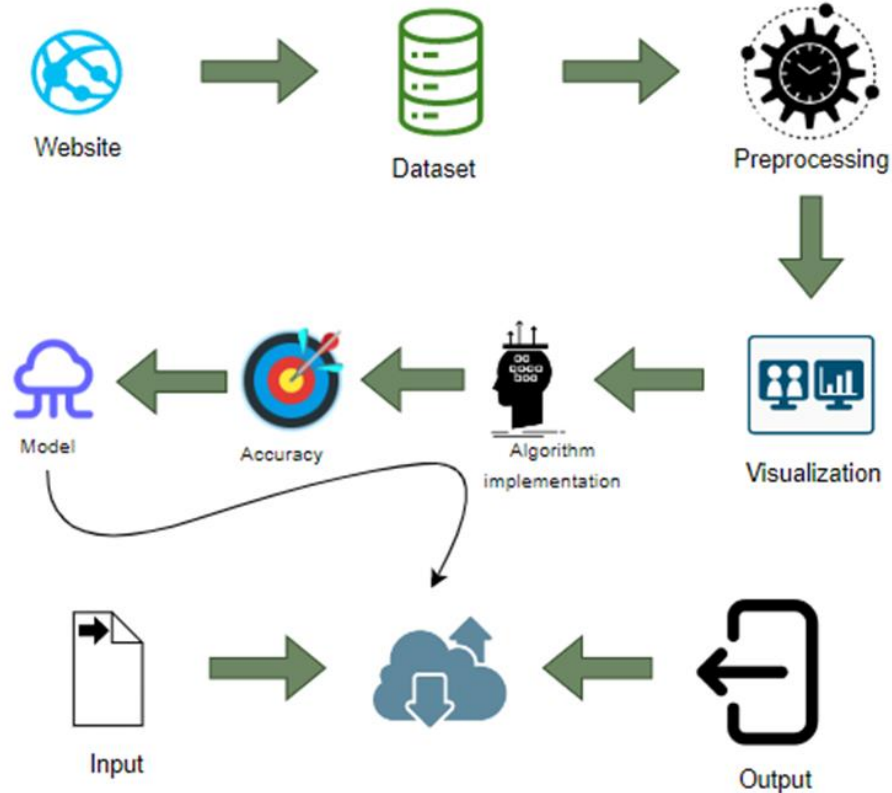
- ❑ **Title** : Cyber-Attacks – Trends, Patterns and Security Countermeasures
- ❑ **Author**: Andreea Bendovschi
- ❑ **Year** : 2015
- ❑ This paper emphasizes the urgent need for global awareness about cybercrime and the importance of legal frameworks at an international level to combat cyber threats effectively. It stresses the responsibility of individuals, companies, and authorities to ensure adequate security measures to protect data privacy rights. Moving forward, the study aims to monitor the evolving trends in cybercrime and countermeasures, focusing on enhancing universal awareness and regulatory decisions to bolster cybersecurity worldwide.

TECHNOLOGIES USED IN PROPOSED SYSTEM



- ❑ **Machine Learning and Deep Learning:** Deep learning techniques, particularly convolutional neural networks (CNNs)
- ❑ **Python Programming Language:** Python is widely used for machine learning and deep learning tasks
- ❑ **Feature extraction:** The system utilizes Scikit-learn for various methods and custom scripts tailored to cybersecurity data, while employing cross-validation techniques and evaluation metrics (Precision, Recall, F1-score) for model training and assessment.
- ❑ **Cybersecurity Response Interface:**
 - Web-based interface for cybersecurity teams (built with Django)
 - API integration for seamless interaction with the model.

ARCHITECTURE DIAGRAM



IDENTIFIED MODULES



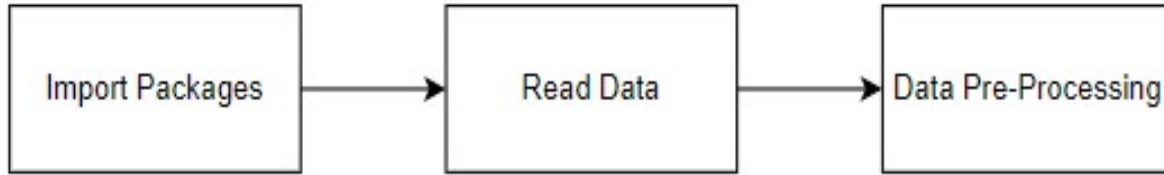
1. Data Pre-processing
2. Data Analysis of Visualization
3. Comparing and Implementing Algorithms
4. Deployment

MODULE 1: Data Preprocessing



- Objective: Ensure data quality and prepare the dataset for machine learning by addressing missing values, handling duplicates, and conducting variable identification through Uni-variate, Bi-variate, and Multi-variate analysis.
- Tasks:
 - Employ Python's Pandas library for efficient handling of missing values, employing techniques such as imputation and statistical approaches.
 - Implement data validation techniques to evaluate model performance, especially crucial in real-world scenarios with non-representative samples.
 - Use renaming and dropping strategies as needed, and specify value types for enhanced data understanding.
- Importance: High-quality and consistent data preprocessing is crucial for improving model accuracy and generalization

MODULE 1: Module Diagram



- **GIVEN INPUT EXPECTED OUTPUT:**
 - input : data
 - output : removing noisy data

MODULE 1: Data Preprocessing

```
In [14]: df["Attack_type"].value_counts()
```

```
Out[14]: Attack_type
DOS_SYN_Hping      94659
Normal              8108
ARP_poisoning      7750
MQTT_Publish       4146
NMAP_UDP_SCAN      2590
NMAP_XMAS_TREE_SCAN 2010
NMAP_OS_DETECTION  2000
NMAP_TCP_scan      1002
DDOS_Slowloris     534
Wipro_bulb         253
Metasploit_Brute_Force_SSH 37
NMAP_FIN_SCAN      28
Name: count, dtype: int64
```

```
In [15]: pd.Categorical(df["Attack_type"]).describe()
```

```
Out[15]:
```

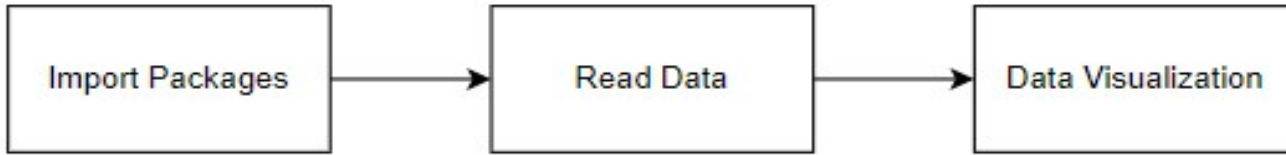
	counts	freqs
categories		
ARP_poisoning	7750	0.062948
DDOS_Slowloris	534	0.004337
DOS_SYN_Hping	94659	0.768854
MQTT_Publish	4146	0.033875
Metasploit_Brute_Force_SSH	37	0.000301
NMAP_FIN_SCAN	28	0.000227
NMAP_OS_DETECTION	2000	0.016245
NMAP_TCP_scan	1002	0.008139
NMAP_UDP_SCAN	2590	0.021037
NMAP_XMAS_TREE_SCAN	2010	0.016326
Normal	8108	0.065856
Wipro_bulb	253	0.002055

MODULE 2: Feature Extraction



- Objective: Develop proficiency in data visualization for applied statistics and machine learning, using charts and plots to gain qualitative insights, identify patterns, and convey key relationships in a dataset.
- Tasks:
 - Master the creation of line plots for visualizing time series data and bar charts for representing categorical quantities in Python.
 - Develop skills in summarizing data distributions through the use of histograms and box plots.
 - Explore the application of data visualization to gain qualitative insights, identify patterns, and effectively communicate key relationships within a dataset.
- Significance: Data visualization is a crucial skill for enhancing the interpretability of data in statistics and machine learning, enabling better decision-making through clear and impactful representation of information.

MODULE 2: Module Diagram

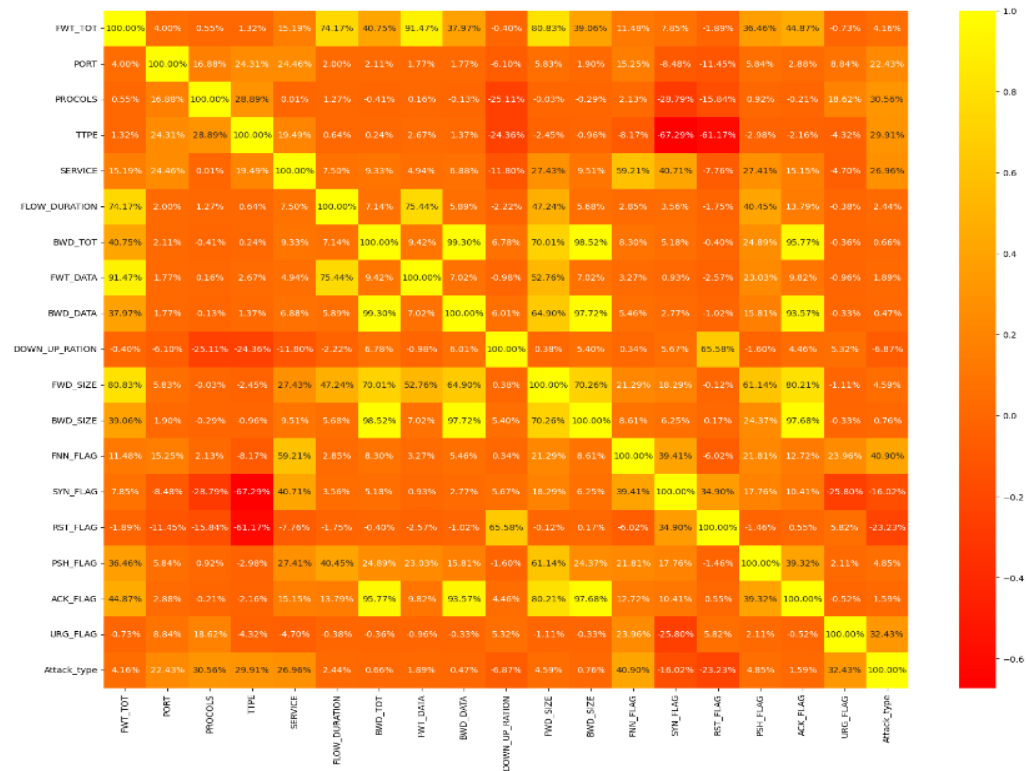


- **GIVEN INPUT EXPECTED OUTPUT :**
 - input : data
 - output : visualized data

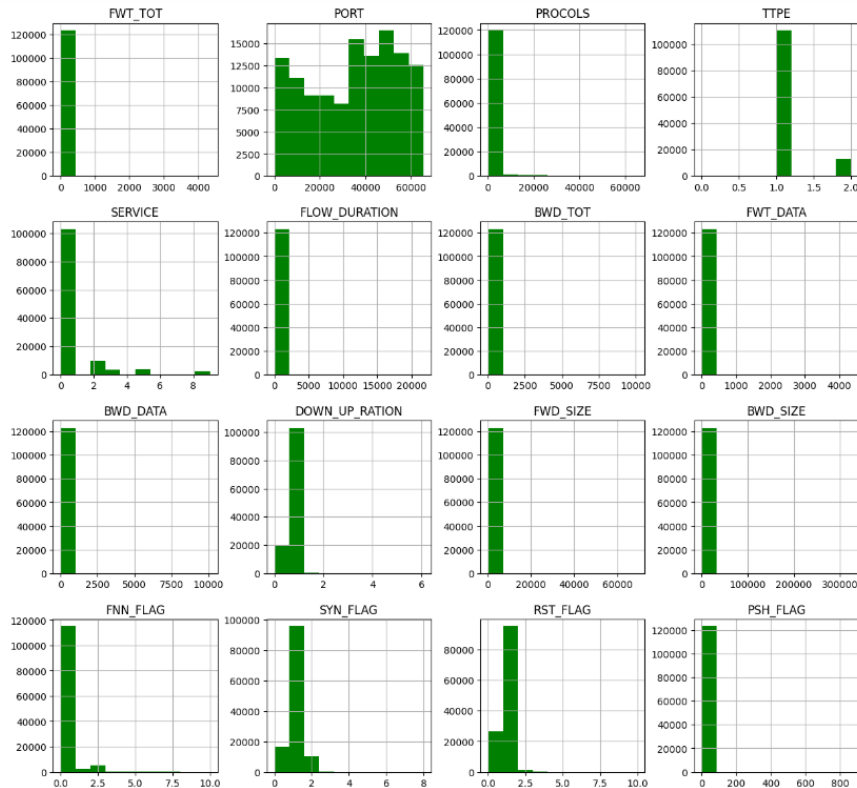
MODULE 2: DATA VISUALIZATION

```
In [14]: fig, ax = plt.subplots(figsize=(20,15))
sns.heatmap(df.corr(),annot = True, fmt='0.2%', cmap = 'autumn',ax=ax)
```

Out[14]: <Axes: >



MODULE 2: DATA VISUALIZATION

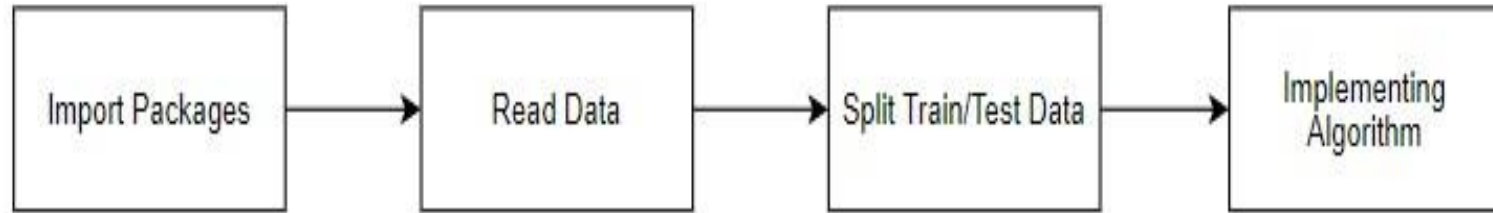


MODULE 3: Model Training and Evaluation



- Objective: Compare and implement three machine learning algorithms—Adaboost Classifier, Random Forest Classifier, and Naïve Bayes—in Python using scikit-learn, focusing on consistent evaluation metrics and visualizations.
- Tasks:
 - Implement a test harness for consistent comparison of Adaboost, Catboost, and Naïve Bayes.
 - Calculate key performance metrics, such as TPR, FPR, Accuracy, Precision, Recall, and F1-Score.
 - Visualize average accuracy and variance for each algorithm, aiding in effective model selection.
- Importance: Algorithm implementation and comparison are crucial for selecting the most suitable model for a given machine learning problem. Understanding the principles and characteristics of each algorithm contributes to informed decision-making, ensuring optimal model selection and performance.

MODULE 3: Module Diagram



- **GIVEN INPUT EXPECTED OUTPUT:**

- input : data
- output : getting accuracy

MODULE 3:

In [16]:

```
from sklearn.metrics import accuracy_score
a = accuracy_score(y_test,predicted)
print("THE ACCURACY SCORE OF ADA BOOST CLASSIFIER IS :",a*100)
```

THE ACCURACY SCORE OF ADA BOOST CLASSIFIER IS : 37.16051447737937

In [17]:

```
from sklearn.metrics import hamming_loss
hl = hamming_loss(y_test,predicted)
print("THE HAMMING LOSS OF ADA BOOST CLASSIFIER IS :",hl*100)
```

THE HAMMING LOSS OF ADA BOOST CLASSIFIER IS : 62.83948552262063

In [19]:

```
from sklearn.model_selection import cross_val_score
accuracy = cross_val_score(ADA, x, y, scoring='accuracy')
print('THE CROSS VALIDATION TEST RESULT OF ACCURACY :\n\n\n', accuracy*100)
```

THE CROSS VALIDATION TEST RESULT OF ACCURACY :

[37.04210721 37.03374387 37.07115881 37.07132199 37.13954952]

MODULE 3:



```
In [16]: from sklearn.metrics import accuracy_score  
a = accuracy_score(y_test,predicted)  
print("THE ACCURACY SCORE OF GAUSSIAN NAIVE BAYES CLASSIFIER IS :",a*100)
```

THE ACCURACY SCORE OF GAUSSIAN NAIVE BAYES CLASSIFIER IS : 72.7385972480214

```
In [17]: from sklearn.metrics import hamming_loss  
hl = hamming_loss(y_test,predicted)  
print("THE HAMMING LOSS OF GAUSSIAN NAIVE BAYES CLASSIFIER IS :",hl*100)
```

THE HAMMING LOSS OF GAUSSIAN NAIVE BAYES CLASSIFIER IS : 27.261402751978594

```
In [19]: from sklearn.model_selection import cross_val_score  
accuracy = cross_val_score(GNB, x, y, scoring='accuracy')  
print('THE CROSS VALIDATION TEST RESULT OF ACCURACY :\n\n\n', accuracy*100)
```

THE CROSS VALIDATION TEST RESULT OF ACCURACY :

[73.08061378 72.70338319 72.74475971 72.80494408 72.79437981]

MODULE 3:

```
In [16]: from sklearn.metrics import accuracy_score
a = accuracy_score(y_test,predicted)
print("THE ACCURACY SCORE OF RANDOM FOREST CLASSIFIER IS :",a*100)

THE ACCURACY SCORE OF RANDOM FOREST CLASSIFIER IS : 99.82613059133206
```

```
In [17]: from sklearn.metrics import hamming_loss
hl = hamming_loss(y_test,predicted)
print("THE HAMMING LOSS OF RANDOM FOREST CLASSIFIER IS :",hl*100)

THE HAMMING LOSS OF RANDOM FOREST CLASSIFIER IS : 0.17386940866794023
```

```
In [18]: from sklearn.metrics import classification_report
C = classification_report(y_test,predicted)
print("THE CLASSIFICATION REPORT SCORE OF RANDOM FOREST CLASSIFIER IS :\n\n",C)

THE CLASSIFICATION REPORT SCORE OF RANDOM FOREST CLASSIFIER IS :
```


	precision	recall	f1-score	support
0	1.00	0.99	1.00	18932
1	1.00	1.00	1.00	18932
2	1.00	1.00	1.00	18932
3	1.00	1.00	1.00	18932
4	0.98	1.00	0.99	18932
5	1.00	1.00	1.00	18932
6	1.00	1.00	1.00	18932
7	1.00	1.00	1.00	18932
8	1.00	1.00	1.00	18931
9	1.00	1.00	1.00	18932
10	1.00	0.99	1.00	18932
11	1.00	1.00	1.00	18931
accuracy			1.00	227182
macro avg	1.00	1.00	1.00	227182
weighted avg	1.00	1.00	1.00	227182

MODULE 4: Deployment



- Objective: Deploy the cyberattack classification system for real-time usage using Django, a micro web framework written in Python.
- Tasks:
 - Integrate the cyberattack classification models with Django, ensuring seamless compatibility and scalability within the web framework.
 - Implement robust monitoring and logging mechanisms to track the performance of the deployed models, enabling timely detection of anomalies or failures.
 - Continuously update and maintain the deployed models, incorporating enhancements to adapt to emerging cyber threats and improve accuracy over time.
- Relevance: Successful deployment using Django ensures the accessibility and scalability of the cyberattack classification system in real-world scenarios. Regular updates and monitoring mechanisms contribute to the system's resilience and effectiveness in dynamically evolving threat landscapes.

SAMPLE CODING



```
#!/usr/bin/env python
"""Django's command-line utility for administrative tasks."""
import os
import sys

def main():
    """Run administrative tasks."""
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'PROJECT.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)

if __name__ == '__main__':
    main()
```

SAMPLE CODING

```
def Landing_1(request):
    return render(request, '1_Landing.html')

def Register_2(request):
    form = UserRegisterForm()
    if request.method == 'POST':
        form = UserRegisterForm(request.POST)
        if form.is_valid():
            form.save()
            user = form.cleaned_data.get('username')
            messages.success(request, 'Account was successfully created. ' + user)
            return redirect('Login_3')

    context = {'form': form}
    return render(request, '2_Register.html', context)

def Login_3(request):
    if request.method == 'POST':
        username = request.POST.get('username')
        password = request.POST.get('password')

        user = authenticate(username=username, password=password)

        if user is not None:
            login(request, user)
```


SAMPLE OUTPUT



SAMPLE OUTPUT

A login form mockup centered on a brown background with a fine grid pattern. The form is contained within a rounded rectangle with a white border and a subtle drop shadow. At the top of the form, the word "WELCOME" is written in a small, white, italicized sans-serif font. Below it, the word "LOGIN" is displayed in a larger, white, bold sans-serif font. The form contains two input fields: the first is labeled "USERNAME" in a small, white, bold sans-serif font, and contains the text "zap"; the second is labeled "PASSWORD" in a small, white, bold sans-serif font, and contains seven dots. Below the password field is a white button with the word "LOGIN" in a small, white, bold sans-serif font. At the bottom of the form, the text "Don't Have an Account?" is written in a small, white, italicized sans-serif font, followed by a white button containing the word "REGISTER" in a small, purple, bold sans-serif font.

WELCOME

LOGIN

USERNAME

zap

PASSWORD

.....

LOGIN

Don't Have an Account?

REGISTER

SAMPLE OUTPUT



SAMPLE OUTPUT

CYBER SECURITY NETWORK ATTACKS PREDICTION USING SUPERVISED MACHINE LEARNING TECHNIQUES

HOME

FWT_TOT

1

PORT

1

PROCOLS

38171

TTYPE

TCP

SERVICE

-

FLOW_DURATION

1.00E-06

BWD_TOT

1

FWT_DATA

RESULT

ABSTRACT
TECHNOLOGY

SAMPLE OUTPUT



CYBER SECURITY NETWORK ATTACKS PREDICTION USING SUPERVISED MACHINE LEARNING TECHNIQUES

[HOME](#)

RESULT

THE DOS_SYN_FLOODING CYBER SECURITY NETWORK ATTACK MIGHT BE OCCUR IN THIS CONDITIONS.

PREVENTIONS : Network Monitoring: Continuously monitor network traffic and logs to detect anomalies and potential security breaches. User Awareness Training: Educate users about security best practices, such as avoiding phishing emails and practicing good password hygiene.

CONCLUSION



In conclusion, we began by cleaning and processing data, addressing missing values, and exploring key insights. After model building and evaluation, the algorithm with the highest accuracy on the public test set was identified. This selected algorithm is integrated into the application, aiding in the identification of various cyberattacks for enhanced cybersecurity.

FUTURE WORK



- **Cloud Deployment:**
 - Explore deploying the system on cloud platforms, enhancing scalability and accessibility for a broader user base.
- **Optimizing for IoT Systems:**
 - Investigate strategies to optimize the system for integration into Internet of Things (IoT) environments, expanding its applicability and efficiency.

REFERENCES



- [1] Mrs. Ashwini Sheth, Mr. Sachin Bhosale, Mr. Farish Kurupkar, "Research Paper on Cyber Security," Contemporary Research in India (ISSN 2231-2137): Special Issue: April, 2021.

- [2] Andreea Bendovschi, "Cyber-Attacks – Trends, Patterns and Security Countermeasures," 7th International Conference on Financial Criminology 2015.

- [3] Tushar P. Parikh, "Cyber Security: Study on Attack, Threat, Vulnerability," International Journal of Research in Modern Engineering and Emerging Technology, Vol. 5, Issue: 6, June 2017 (IJRMEET) ISSN: 2320-6586.

- [4] Shamsun Nahar Edib, Yuzhang Lin, Vinod M. Vokkarane, Feng Qiu, and Bo Chen, "Cyber Restoration of Power Systems: Concept and Methodology for Resilient Observability," IEEE Transactions on Systems, Man, and Cybernetics: Systems, Vol. 53, No. 8, August 2023.

- [5] H. Wang and Y. Wu, "Adversarial Attack Detection in Wireless Networks Using Transfer Learning," IEEE Internet of Things Journal, vol. 8, no. 8, pp. 6554-6563, 2021.

REFERENCES



- [6] Z. Chen and H. Liu, "Feature Selection and Ensemble Learning for Wireless Network Attack Detection," IEEE Transactions on Network and Service Management, vol. 18, no. 4, pp. 3236-3248, 2021.

- [7] S. Kim and J. Park, "Wireless Network Intrusion Detection System Using Machine Learning and Software-Defined Networking," IEEE Transactions on Mobile Computing, vol. 20, no. 3, pp. 1042-1055, 2021.

- [8] Y. Huang and Q. Zhang, "Deep Reinforcement Learning for Adaptive Wireless Network Defense against Attacks," IEEE Transactions on Cognitive Communications and Networking, vol. 8, no. 1, pp. 58-71, 2022.

- [9] L. Zhao and X. Li, "Federated Learning-Based Wireless Network Intrusion Detection," IEEE Transactions on Dependable and Secure Computing, vol. 19, no. 1, pp. 74-86, 2022.

- [10] J. Xu and Z. Wang, "Anomaly Detection in Wireless Networks Using Long Short-Term Memory Networks," IEEE Transactions on Vehicular Technology, vol. 71, no. 3, pp. 2392-2403, 2022.



THANK YOU