**Detection of Cyber Attack in**

**Network using Machine Learning Techniques**

**Functional Requirement Specification Document**

**09-07-2024**

**Functional Requirements Document**

|  |  |
| --- | --- |
| **Name of Document** | Functional Requirement Specification <Detection of Cyber Attack in Network using Machine Learning Techniques> |
| **Author** | **Piyush Shende, Utkarsha Payghan, Pranshu Mishra, Vedang Jadhav** |
| **Description of Content** | Functional, Technical and Operational Requirements |
| **Reference** | **"Machine Learning for Cybersecurity" by Xiaofeng Chen**  **"Cybersecurity and Applied Machine Learning" by Richard Liu** |
| **Approved by** | **Neha Lahane** |
| **Date of Approval** | **09-07-2024** |

**Distribution**

|  |  |
| --- | --- |
| **Copy No.** | **Distributed to** |
|  | **Piyush Shende** |
|  | **Utkarsha Payghan** |
|  | **Pranshu Mishra** |
|  | **Vedang Jadhav** |

**Document Version Control**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version Number** | **Version Date** | **Prepared by** | **Reviewed by** | **Particulars/ Description for Change** | **Remarks** |
| **1.0** | **09-07-2024** | **Piyush Shende** | **Neha lahane** | **None** | **No** |
|  |  |  |  |  |  |

**List of Requirements Elicitation Discussions**

|  |  |  |
| --- | --- | --- |
| **SN.** | **Date** | **Participating Institutes/ Departments** |
|  | 09-07-2024 | Centre for development of Advance Computing, Bangalore |
|  |  |  |
|  |  |  |
|  |  |  |

**Abbreviations**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**CONTENTS**

[1](#_gjdgxs) **INTRODUCTION** 1

[1.1 Purpose 1](#_30j0zll)

[1.2 Background 1](#_1fob9te)

[1.3 Scope 2](#_3znysh7)

[1.4 References 2](#_2et92p0)

[1.5 Document Overview 2](#_qsh70q)

**2 UAN Allotment System 3**

**3 FUNCTIONAL REQUIREMENTS 3**

3.1 Registration of Departments 4

3.1.1 Registration of Departments 4

3.1.2 Description 4

3.1.3 Functionalities identified 5

3.1.4 Field validations and business rules 5

3.2 API for Generation of UAN 5

3.2.1 UAN Allotment for a single worker 6

3.2.1.1 Description 7

3.2.1.2 Cross Functional Diagram 7

3.2.1.3 Functionalities identified 5

3.2.1.4 Pre-requisites, Assumptions and Dependencies 7

3.2.1.5 Field validations and business rules 7

3.2.2 UAN Allotment for bulk 10

3.2.2.1 Description 10

3.2.2.2 Cross Functional Diagram 10

3.2.1.3 Functionalities identified 5

3.2.2.4 Pre-requisites, Assumptions and Dependencies 11

3.2.2.5 Field validations and business rules 12

[3.3 Updation in UAN Data 13](#_2u6wntf)

3.3.1 Description 13

3.3.2 Cross Functional Diagram 14

3.3.3 Functionalities identified 5

3.3.4 Pre-requisites, Assumptions and Dependencies 14

3.3.5 Field validations and business rules 15

[3.4 Family Member Details 17](#_37m2jsg)

3.4.1 Description 17

3.4.2 Cross Functional Diagram 17

3.3.3 Functionalities identified 5

3.4.4 Pre-requisites, Assumptions and Dependencies 18

3.4.5 Field validations and business rules 18

3.5 Other API Integration with different departments 20

3.5.1 Description 20

3.5.2 Cross Functional Diagram 20

3.3.3 Functionalities identified 5

3.5.4 Pre-requisites, Assumptions and Dependencies 21

3.5.5 Field validations and business rules 21

3.6 Implementation of Aadhaar Data Vault 21

3.7 Download UAN 21

3.8 Print UAN 21

3.9 Report Generation 21

3.10 Data Migration 21

**4 External Interfaces 22**

**5 Audit and Provenance 22**

**6** **Current UAN Schema (Shared by the Departments) 22**

**7 Data Migration of UANs from EPFO to UAN Engine 22**

# INTRODUCTION

Introduction to the component and its usage

## 1.1 Purpose

In the paper "Leveraging Machine Learning Technology to Identify Cyber Attacks in Networks" our main objective was how we can design, develop and use a smart machine that acts instantly as well accurately identifies different versions of attacks. Cyber there by promoting some sexual security. The research aims to leverage machine learning techniques (e.g. artificial neural network, convolutional Neural Network and Random Forest) in detecting traffic patterns of the network, recognizing vulnerabilities or weaknesses posed by such patterns and predict threats based on existing conditions. Today, everyone is using a digital connected world with the help of computer networks but securing these networks are important. The danger and power of cyber-attacks are now stronger than ever. It resulted in significant financial losses, reputational damage, and leakage of sensitive information. Traditional and signature based cyber attack methods are often ineffective against today's threats designed to avoid detection through these methods.

## 1.2 Background

The rapid growth of the Internet and dependence on computer networks makes them an attractive target for cybercriminals. Cyber-attacks have become a major concern for individuals, organizations, and governments as they can cause significant financial losses, reputational damage, and leakage of sensitive information. Global cybercrime losses are expected to rise from $3 trillion in 2015 to $6 trillion by 2021, according to a report by Cybersecurity Ventures. Cyber-attacks have evolved over the years from simple viruses and malware to advanced persistent threats (APT) and ransomware attacks. Modern cyberattacks are often designed to evade traditional security measures that rely on signature-based detection and rules based methods. Which can be managed from any part of the world, so it makes your reply hard to load. Modern Online Threats Bypass Virus Software They are signature based systems, which involves checking incoming traffic patterns with threat knowledge in order to protect the network as such. However, contemporary cyberattacks frequently employ zeroday vulnerabilities or the use of polymorphic malware and other evasion mechanisms to remain undetected. Write is a promising direction in network intrusion detection. These algorithms allow professionals to examine massive volumes of data through networks, uncovering patterns and irregularities that can be used as the basis for later threat predictions. These algorithms are capable of learning from experience, reconfiguring to combat new threats, and getting better at detection over time.

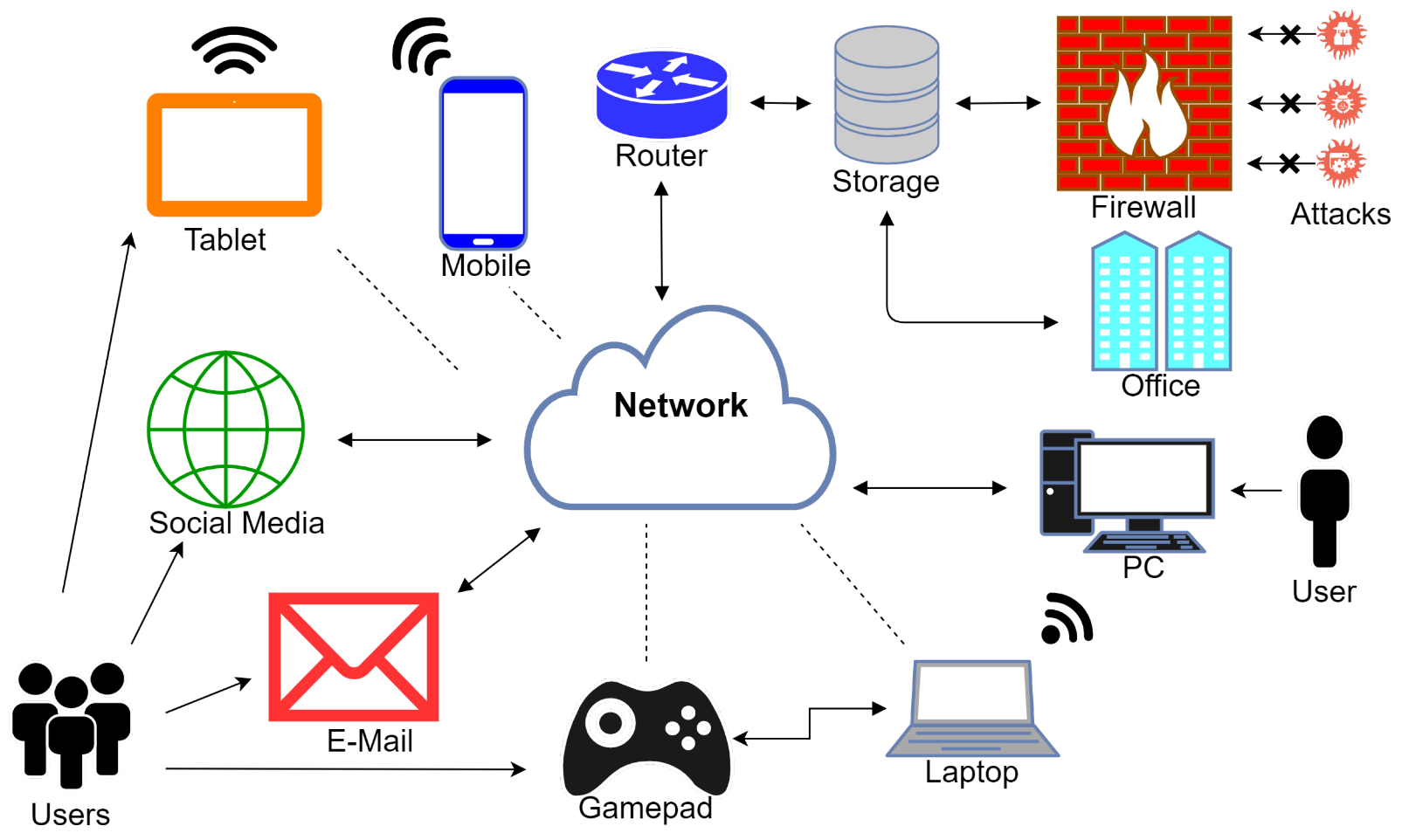
## 1.3 Scope

This research project aims to design, develop and evaluate a machine learning based module for the purpose of detecting cyber-attacks in computer systems. It refers to accurate classification and categorization of cyberattacks as they happen, identification of the most important features in network traffic that indicate a cyberthreat

The stretch will feature a comprehensive literature survey of the past research on machine learning-based cyber attack detection systems to fulfil this purpose. This will be followed by the gathering of network output data from different

sources, counting bundle captures, framework logs, and organize stream information. The collected information will at that point be preprocessed, cleaned, and changed into a organize appropriate for machine learning algorithms.

The extend will too include include building, where important highlights will be recognized and extricated from the preprocessed information. These highlights will at that point be utilized to create machine learning models utilizing different calculations, counting directed and unsupervised learning procedures. The execution of the created models will be assessed utilizing measurements such as precision, exactness, review, and F1-score.

******

***Figure 1: System Architecture***

## 1.4 References

1. "Machine Learning for Cybersecurity" by Xiaofeng Chen
2. "Cybersecurity and Applied Machine Learning" by Richard Liu
3. Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., Al-Nemrat, A., & Venkatraman, S. (2019). Deep Learning approach for Intelligent Intrusion Detection System.

## 1.5 Document Overview

# This Useful Necessities Report (FRD) traces the useful prerequisites for the extend "Discovery of Cyber Assault in Organize utilizing Machine Learning Procedures". The venture points to plan, create, and assess a machine learning-based framework for recognizing cyber assaults in computer systems. This archive gives a nitty gritty portrayal of the system's utilitarian prerequisites, counting its engineering, components, and interfacing. The scope of this extend is restricted to the discovery of cyber assaults in computer systems utilizing machine learning methods. The framework will be planned to identify and classify cyber assaults in real-time, and to supply noteworthy bits of knowledge to security investigators and occurrence responders.FUNCTIONAL REQUIREMENTS

Data Ingestion, Data Preprocessing, Featuring Engineering, Modal development, Modal Evaluation, Realtime detection,

### 2.1 Functionality 1

### 2.1.1 Description

The system's essential objectives are to:

* Detect cyber assaults, counting malware, phishing, DDoS, and other sorts of attacks. Prevent assaults from causing hurt to the organize, frameworks, and data. Alert security investigators and occurrence responders to require incite action.
* Provide actionable insights and visualization to assist examiners get it the assault characteristics and react effectively.
* The framework will utilize machine learning calculations to analyze the information and recognize designs, irregularities, and dangers.
* The information will be utilized to prepare and assess the machine learning models, guaranteeing that they can precisely distinguish and anticipate cyber-attacks.

The benefits of this framework include:

* Improved security: Upgraded location and anticipation of cyber attacks.
* Reduced hazard: Minimized hazard of information breaches, budgetary misfortunes, and reputational damage.
* Increased productivity: Mechanized discovery and reaction to cyber assaults, diminishing the workload of security analysts.
* Better decision-making: Data-driven experiences and visualization to back educated decision-making.

### 2.1.2 Functional Requirements Identified

|  |  |  |  |
| --- | --- | --- | --- |
| **Functionality 1** | | | |
| **SN** | **Functionality** | **Process** | **Additional Info/Remarks** |
| 1 | Data Ingestion | Ingest network traffic data from various sources (packet captures, system logs, network flow data) | Handle large volumes of data, scale to meet network needs |
| 2 | Data Preprocessing | Clean, filter, and transform data into format suitable for machine learning algorithms | Handle noisy and incomplete data, perform data normalization and feature scaling |
| 3 | Feature Engineering | Identify and extract relevant features from preprocessed data (network traffic, system logs, network flow) | Select most informative features for cyber attack detection, perform feature selection and dimensionality reduction |
| 4 | Model Development | Develop machine learning models using various algorithms (supervised, unsupervised, deep learning) | Train and evaluate models using preprocessed data, perform model selection and hyperparameter tuning |
| 5 | Model Evaluation | Evaluate model performance using metrics (accuracy, precision, recall, F1-score, ROC-AUC) | Compare performance of different models, select best-performing model, perform model validation and testing |
| 6 | Real-time Detection | Detect cyber attacks in real-time using trained models | Provide alerts and notifications to security analysts and incident responders, perform anomaly detection and outlier detection |
| 7 | Visualization | Provide visualization framework for actionable insights | Display detected cyber attacks and characteristics (attack type, severity, source, target) |
| 8 | Integration | Integrate with existing SIEM systems, incident response systems, and threat intelligence feeds | Ensure seamless integration and data exchange |
| 9 | Security | Ensure confidentiality, integrity, and availability of data | Protect against unauthorized access and data breaches, perform encryption and access control |
| 10 | Scalability | Scale to meet network needs, handle increasing volumes of data and traffic | Perform load balancing and distributed processing |
| 11 | Maintenance | Perform automated updates and maintenance, backups and data recovery | Provide logging and auditing capabilities |
| 12 | User Interface | Provide user-friendly interface for security analysts and incident responders | Offer dashboard for monitoring and analyzing cyber attacks, reporting and analytics capabilities |

### 2.1.3 Fields Validations

This table lists the various fields that will be there in the form for the requirement

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SN** | **Field Name** | **Field Description** | **Validations** | **Remarks** |
| 1 | Source IP Address | IP address of the source device | * IP address format (IPv4 or IPv6) | Identify the source of the traffic, potential attacker's IP address |
| 2 | Destination IP Address | IP address of the destination device | * IP address format (IPv4 or IPv6) | Identify the target of the traffic, potential victim's IP address |
| 3 | Source Port Number | Port number used by the source device | * Integer value between 0 and 65535 | Identify the source port used by the attacker, potential vulnerability |
| 4 | Destination Port Number | Port number used by the destination device | * Integer value between 0 and 65535 | Identify the destination port used by the attacker, potential vulnerability |
| 5 | Protocol | Network protocol used (e.g., TCP, UDP, ICMP) | * TCP, UDP, ICMP, or other protocol types | Identify the protocol used by the attacker, potential vulnerability |
| 6 | Packet Size | Size of the packet in bytes | * Integer value representing the packet size in bytes | Identify unusual packet sizes, potential indicator of malicious activity |
| 7 | Packet Count | Number of packets sent or received | * Integer value representing the number of packets | Identify unusual packet counts, potential indicator of malicious activity |
| 8 | Time Stamp | Date and time of the packet capture | * Date and time format (e.g., YYYY-MM-DD HH:MM:SS) | Identify the time of the attack, potential correlation with other events |
| 9 | HTTP Request Method | HTTP request method used (e.g., GET, POST, PUT, DELETE) | * GET, POST, PUT, DELETE, or other HTTP methods | Identify unusual HTTP request methods, potential indicator of malicious activity |
| 10 | HTTP Request URL | URL of the HTTP request | * URL format (e.g., https://example.com/path) | Identify unusual HTTP request URLs, potential indicator of malicious activity |
| 11 | User Agent | User agent string (e.g., browser type, OS) | * String value representing the user agent | Identify unusual user agents, potential indicator of malicious activity |
| 12 | Payload | Binary or text data representing the packet payload | * Binary or text data | Identify malicious payload, potential indicator of malware or exploit |
| 13 | Flow Duration | Duration of the network flow in seconds | * Integer value representing the flow duration in seconds | Identify unusual flow durations, potential indicator of malicious activity |
| 14 | Bytes Sent/Received | Number of bytes sent or received | * Integer value representing the number of bytes sent or received | Identify unusual byte counts, potential indicator of malicious activity |
| 15 | Packet Loss | Percentage of packet loss | * Integer value representing the packet loss percentage | Identify unusual packet loss, potential indicator of malicious activity |
| 16 | TCP Flags | TCP flags used (e.g., SYN, ACK, FIN) | * String value representing the TCP flags | Identify unusual TCP flags, potential indicator of malicious activity |

### 

### 3.2.1.4 Pre-requisites, Assumptions and Dependencies

# Pre-requisites:-

* Network Information: Get to to organize activity information, counting parcel captures, logs, and other significant information sources.
* Computational Assets: Adequate computational assets, counting CPU, memory, and capacity, to prepare and analyze huge datasets.
* Machine Learning Ability: Information and involvement in machine learning, counting information preprocessing, highlight designing, demonstrate determination, and hyperparameter tuning. Domain Information: Understanding of arrange conventions, danger insights, and cybersecurity standards to illuminate include building and show development.
* Data Capacity: A versatile and secure information capacity arrangement to store and oversee expansive datasets.

# Assumptions:- Data Quality: The organize information is precise, total, and agent of the organize traffic.Data Stationarity: The organize information is stationary, meaning that the basic dispersion of the information does not alter over time.

# Attack Dispersion: The dissemination of cyber assaults is known or can be approximated, permitting for compelling modeling and detection.Feature Significance: The chosen highlights are important and enlightening for identifying cyber attacks.Model Generalizability: The machine learning show generalizes well to modern, inconspicuous information and isn't overfitting to the preparing data.

# Dependencies: Network Framework: The arrange framework, counting switches, switches, and firewalls, is arranged to gather and forward organize activity data.

# Data Collection Apparatuses: Instruments such as parcel sniffers, log collectors, and arrange taps are utilized to gather arrange activity data.

# Data Preprocessing: Information preprocessing strategies, such as highlight scaling, normalization, and encoding, are utilized to get ready the information for modeling. Machine Learning Libraries: Machine learning libraries, such as scikit-learn, TensorFlow, or PyTorch, are utilized to actualize and prepare machine learning models.

# Evaluation Measurements: Assessment measurements, such as exactness, exactness, review, and F1-score, are utilized to degree the execution of the machine learning model. Threat Insights: Danger insights nourishes and databases are utilized to illuminate the machine learning demonstrate around known dangers and assault patterns.

# Model Overhauls: The machine learning show is overhauled frequently to adjust to modern dangers, assault designs, and changes in organize activity.

# External Interfaces

Information Ingestion Interfaces

To collect arrange activity information, interfacing to organize taps, such as SPAN ports or organize interface cards (NICs), are required. Also, interfacing to gather log information from different arrange gadgets, such as firewalls, interruption discovery frameworks (IDS), and switches, are fundamental. Furthermore, APIs are required to gather information from cloud-based administrations, such as AWS CloudWatch or Google Cloud Logging. These interfacing empower the collection of different information sources, which are at that point utilized to prepare and convey machine learning models for cyber assault detection.

Data Capacity Interfaces

To store and oversee huge datasets, interfacing to social databases (e.g., MySQL) or NoSQL databases (e.g., MongoDB) are required. In addition, interfacing to information distribution centers (e.g., Amazon Redshift) or information lakes (e.g., Apache Hadoop) are essential to store and oversee amassed information. These interfacing empower the productive capacity and recovery of information, which is basic for machine learning demonstrate preparing and deployment.

Machine Learning Interfaces

To prepare machine learning models, interfacing to well known machine learning libraries, such as scikit-learn, TensorFlow, or PyTorch, are required. Besides, interfacing to send prepared models, such as demonstrate serving stages (e.g., TensorFlow Serving) or cloud-based administrations (e.g., AWS SageMaker), are fundamental. These interfacing empower the improvement, preparing, and arrangement of machine learning models for cyber assault detection.

Visualization and Announcing Interfaces

To visualize and report on location comes about, interfacing to dashboards, such as Grafana, Kibana, or Scene, are required. Also, interfacing to cautioning frameworks, such as e-mail, SMS, or informing stages (e.g., Slack), are fundamental to send alarms and notices to security groups. These interfacing empower the compelling communication of discovery comes about and encourage opportune occurrence response.

Integration Interfaces

To coordinated with existing security frameworks, interfacing to security data and occasion administration (SIEM) frameworks, such as Splunk or ELK Stack, are required. Besides, interfacing to danger insights bolsters, such as OpenPhish or URLhaus, are vital to remain educated almost developing dangers. Furthermore, interfacing to occurrence reaction frameworks, such as Ghost or Demisto, are required to encourage occurrence reaction and remediation. These interfacing empower the integration of machine learning-based cyber assault discovery with existing security frameworks and services.

Authentication and Authorization Interfaces

To guarantee secure get to to information and frameworks, interfacing to verification frameworks, such as LDAP or Dynamic Registry, are required. Besides, interfacing to authorization frameworks, such as role-based get to control (RBAC) or attribute-based get to control (ABAC), are essential to control get to to delicate information and frameworks. These interfacing empower secure and controlled get to to machine learning-based cyber assault discovery systems.

APIs and SDKs

To give adaptable integration alternatives, interfacing to Serene APIs are required. Furthermore, interfacing to computer program improvement packs (SDKs) for programming dialects, such as Python or Java, are essential to empower consistent integration with existing frameworks and administrations. These interfacing empower the improvement of custom integrative and applications that use machine learning-based cyber assault location capabilities.