

Secure Message Scan Url

Dr.D. Pravin Kumar ¹, Deekshitha Gayathri K.S ², Dhanusha T.A ³

¹ Associate Professor, Dept of CSE

^{2,3} Dept of CSE

^{1,2,3} K.L.N.College of Engineering,Sivagangai, India

Abstract- The increasing use of digital messaging has led to a rise in phishing and smishing attacks through malicious URLs. The proposed project, Secure Message Scan URL, combines secure message communication with phishing URL detection. The system uses Python Tkinter for the user interface, MongoDB for storing scanning records, and AES encryption for message confidentiality. Regular expressions extract URLs from messages, while a rule-based mechanism analyzes suspicious keywords, HTTP usage, IP addresses, and blacklist matches. Based on the calculated risk score, URLs are classified as Safe, Suspicious, or Malicious. The system displays the results and helps users identify potentially harmful links, improving digital communication security.

I. INTRODUCTION

Nowadays, people use messaging applications to share information and links. However, attackers use phishing and smishing to steal personal information through malicious URLs. The Secure Message Scan URL project helps protect users by combining message encryption and URL detection. AES is used to encrypt and decrypt messages, while regular expression extract URLs from messages. A rule-based method checks suspicious keywords, HTTP usage, IP addresses, and blacklist matches to identify harmful links. The system classifies URLs as Safe, Suspicious, or Malicious based on their risk scores. Python Tkinter is used to create the interface, and MongoDB stores scanning records. This project helps users identify suspicious links and communicate more safely.

II. LITERATURE SURVEY

Previous studies have focused on detecting smishing attacks phishing URLs to protect users from online threats. Researchers have developed URL analysis techniques using features such as suspicious keywords, IP addresses, HTTP usage, and URL structure. AES encryption is also widely used to protect messages from unauthorized access and maintain confidentiality. Python libraries such as re and urlparse help extract and analyze URLs from messages

The proposed Secure Message Scan URL system combines AES encryption. Python-based URL analysis, rule-based risk scoring, and MongoDB storage. It analyzes URLs

based on predefined security rules and classifies them as safe, suspicious, or malicious. MongoDB stores the scan results and related information for future reference. This system aims to improve message security and help users identify potentially harmful links.

III. PROPOSED METHODOLOGY

The proposed Secure Message Scan URL system is developed using Python and Tkinter to provide a user-friendly interface for secure messaging and URL scanning. The system uses the AES algorithm to encrypt and decrypt messages, ensuring message confidentiality. When a message contains a URL, Python libraries such as re and urlparse are used to extract and analyze the link. The system checks for suspicious characteristics, including HTTP usage, IP-based domains, suspicious keywords, and known blacklisted URLs.

Based on the detected characteristics, the system assigns a risk score to each URL and classifies it as Safe, Suspicious, or Malicious. Rule-based detection is used to calculate the score without requiring machine learning training.

MongoDB is used to store the scanned URLs, risk scored, detection reasons, and results. Finally, the system displays the analysis results through the graphical user interface, helping users identify potentially harmful links and communicate more securely.

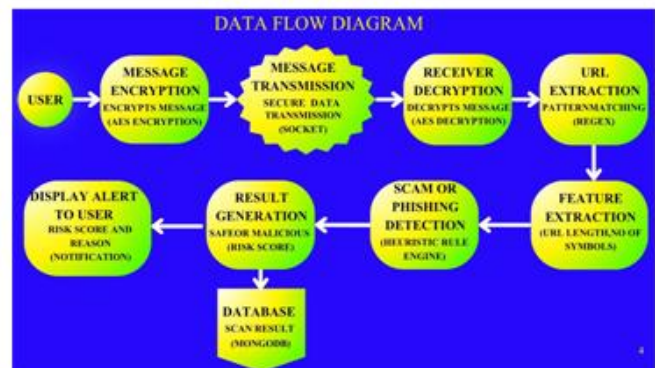


Figure 1 : Dataflow diagram of our proposed module

IV. SYSTEM PROCESS FLOW

1. Message Encryption:

The user's message is encrypted using the AES algorithm to protect it from unauthorized access.

2. Message Transmission:

The encrypted message is securely transmitted to the receiver using python socket programming.

3. Decryption and URL Extraction:

The receiver decrypts the encrypted message and URLs are extracted using Regex.

4. Feature Extraction:

URL features such as length,symbols,IP addresses,and suspicious keywords are identified.

5. Phishing Detection and Result Generation:

A rule-based engine analyzes the URL,calculates its risk score,and classifies it as safe or malicious.

6. Alert and Database Storage:

The result and risk score are displayed to the user,and scan details are stored in MongoDB.

V. RESULT AND DISCUSSIONS

The Secure Message Scan URL system combines Python,AES encryption,rule-based URL detection,and MongoDB to improve message security.The system extracts URLs from messages and checks for suspicious feature such as HTTP usage,IP addresses,suspicious keywords,and blacklisted links.Based on the risk score,URLs are classified as safe,suspicious,or malicious.AES encryption protects messages from unauthorized access,while MongoDB stores the scanning results for future reference.The Tkinter interface helps users view the results easily.The system is simple to implement and does not require machine learning training.However,some new malicious URLs may not be detected if they do not match the predefined rules.Overall,the project helps users identify potentially harmful links and improves the security of digital communication.

VI. CONCLUSION

The Secure Message Scan URL project provides a simple and effective approach to secure message communication and phishing URL detection.The system uses AES encryption to protect messages,socket programming for secure transmission,and Python libraries to extract and analyze URLs.A rule-based heuristic engine identifies suspicious links and classifies them based on their risk scores.The results are displayed through a user-friendly Tkinter interface,while MongoDB stores the scan details for future reference.Overall,this project helps users identify potentially harmful URLs,improves message confidentiality,and promotes safer digital communication.

REFERENCES

- [1] S.F.Schwarz et al.,”Smishing Detection From a Messaging Platform View”,IEEE Access,2025.
- [2] B.V.Pavani et al.,”Phishing URL Detection Using Machine Learning and Explainable AI”,IEEE ICCCNT 2024.
- [3] “Detecting Phishing URLs Using BERT Transformer Model”,IEEE BigData,2023.
- [4] M.Sa’nchez-Paniagua et al.,”Phishing URL Detection:A Real-Case Scenario Through Login URLs”,IEEE Access,2022.
- [5] NIST,”Advanced Encryption Standard(AES)”,FIPS PUB 197,2001.
- [6] MongoDB Documentation,Database Design and Management Concepts.
- [7] Python Documentation,Regular Expressions and socket programming
- [8] A.K.Jain and B.B.Gupta,”Phishing Detection:Analysis of Methods and Approaches”,International Journal of Computer Applications,2018.