

**PHISHING EMAIL ANALYSIS AND BASIC ATTRIBUTION**

MWANGIRI SABASTIAN MWAIWO

23/08058

**Supervised by:** MR. CHARLES MALUNGU

**Date of Submission:** 22 June 2025

Table of Contents

[Introduction 3](#_Toc201499585)

[1.1 Purpose 3](#_Toc201499586)

[1.2 Intended Audience and Reading Suggestions 3](#_Toc201499587)

[1.3 Project Scope 3](#_Toc201499588)

[2. Overall Description 3](#_Toc201499589)

[2.1 User Classes and Characteristics 3](#_Toc201499590)

[2.2 Operating Environment 4](#_Toc201499591)

[2.3 Design and Implementation Constraints 4](#_Toc201499592)

[2.4 User Documentation 4](#_Toc201499593)

[2.5 Assumptions and Dependencies 4](#_Toc201499594)

[3. Functional Requirements 5](#_Toc201499595)

[3.1 System Features 5](#_Toc201499596)

[3.1.1 Email Parsing Module 5](#_Toc201499597)

[3.1.2 Analysis Engine 5](#_Toc201499598)

[3.1.3 Reporting Module 6](#_Toc201499599)

[4. External Interface Requirements 6](#_Toc201499600)

[4.1 User Interfaces 6](#_Toc201499601)

[4.2 Hardware Interfaces 6](#_Toc201499602)

[4.3 Software Interfaces 6](#_Toc201499603)

[4.4 Communications Interfaces 7](#_Toc201499604)

[5. Other Nonfunctional Requirements 7](#_Toc201499605)

[5.1 Performance Requirements 7](#_Toc201499606)

[5.2 Safety Requirements 7](#_Toc201499607)

[5.3 Security Requirements 7](#_Toc201499608)

[5.4 Software Quality Attributes 7](#_Toc201499609)

[6. Other Requirements 8](#_Toc201499610)

[7. References 8](#_Toc201499611)

[8.Appendix A: Glossary 8](#_Toc201499612)

# Introduction

## 1.1 Purpose

In this document, I aim to specify the software requirements for my project titled **"Phishing Email Analysis and Basic Attribution."** This system is designed to automate the detection and analysis of phishing emails specifically for Elite Luxe Imports, with the goal of enhancing their cybersecurity measures. This SRS will cover the entire system, including the email parsing, analysis and reporting modules, to provide a clear roadmap for development.

## 1.2 Intended Audience and Reading Suggestions

I have written this document primarily for developers, project managers, IT security staff and testers who will be involved in the project. To get the most out of this document, I suggest starting with the overall description to understand the project's scope and then proceeding to the functional and non-functional requirements for detailed insights.

## 1.3 Project Scope

With this project, I intend to develop a system that provides automated tools to detect, analyze and report on phishing emails. The system will focus on email header analysis, URL and attachment scanning, and generating detailed reports for the IT and security teams at Elite Luxe Imports. This project aligns with the organization's goal of enhancing cybersecurity and reducing the risk of phishing attacks, which have been a significant concern.

# 2. Overall Description

## 2.1 User Classes and Characteristics

Based on my analysis, the system will cater to the following user classes:

**IT Security Staff:** These are technical experts who will use the system for forensic analysis and incident response. They require detailed analytical data and robust tools to effectively counter phishing threats.

**General Staff:** Employees who may interact with the system through awareness reports and alerts. Their needs are more about understanding risks and receiving clear, actionable information.

**Administrators:** Users responsible for system maintenance and updates. They need comprehensive access and control over the system's configurations and settings.

## 2.2 Operating Environment

I plan to deploy the system in the following operating environment:

**Hardware Platform:** Standard PC or server hardware to ensure compatibility and ease of deployment.

**Operating System:** Windows Server 2022 and Linux (Ubuntu 22.04 LTS) to cover the most commonly used server environments.

**Software Components:** Java Runtime Environment (JRE) 11 and Python 3.8 for development and execution, along with VirtualBox/VMware for creating isolated analysis environments.

## 2.3 Design and Implementation Constraints

I have identified several constraints that will guide the design and implementation of the system:

**Regulatory Policies:** Compliance with data protection and privacy regulations to ensure legal and ethical handling of data.

**Hardware Limitations:** Adequate memory and processing power will be necessary for efficient email analysis, especially when dealing with large volumes of emails.

**Interfaces**: The system will need to integrate seamlessly with existing email systems and threat intelligence APIs to enhance its analytical capabilities.

**Security Considerations:** Secure handling and storage of email data and analysis results are paramount to prevent data breaches and ensure user privacy.

## 2.4 User Documentation

I will provide the following user documentation to support the system:

User Manuals: Comprehensive guides tailored for IT security staff and administrators, detailing system operations and maintenance.

On-line Help: Context-sensitive help within the system interface to assist users in real-time.

Tutorials: Step-by-step tutorials designed to help new users get up to speed quickly and effectively.

## 2.5 Assumptions and Dependencies

I have made the following assumptions and identified key dependencies for the project:

Availability of Threat Intelligence APIs: Access to APIs like VirusTotal is crucial for enhancing the system's analytical capabilities.

Access to Phishing Email Samples: A diverse set of phishing email samples will be necessary for thorough testing and training of the system.

Collaboration with Security Communities: Engaging with security communities will facilitate data sharing and improve the system's effectiveness.

# 3. Functional Requirements

## 3.1 System Features

### 3.1.1 Email Parsing Module

**Description and Priority**: This is a high-priority module designed to extract and analyze email headers, body content and attachments. It forms the foundation of the system's analytical capabilities.

**Stimulus/Response Sequences:**

User Uploads an Email: The user initiates the process by uploading an email for analysis.

System Parses the Email: The system processes the email, extracting relevant data such as headers, URLs and attachments.

System Displays Parsed Data: The parsed data is presented to the user for review, providing insights into potential phishing indicators.

### 3.1.2 Analysis Engine

Description and Priority: This high-priority module is responsible for scanning URLs and attachments and performing header forensics to identify malicious indicators.

Stimulus/Response Sequences:

User Initiates Analysis: The user requests an analysis of a specific email.

System Scans for Threats: The system scans the email for malicious URLs, attachments and other phishing indicators.

System Generates Report: A comprehensive report is generated, highlighting phishing techniques and providing threat attribution details.

### 3.1.3 Reporting Module

Description and Priority: This medium-priority module focuses on generating automated reports that summarize the findings of the analysis.

Stimulus/Response Sequences:

User Requests a Report: The user requests a report based on the analysis performed.

System Compiles Data: The system compiles the analytical data and generates a comprehensive report.

System Provides Downloadable Report: The report is made available to the user in a downloadable format, ensuring easy access and distribution.

**Functional priority table**

|  |  |  |
| --- | --- | --- |
| **Module** | **Description** | **Priority** |
| Email Parsing | Extracts email components | High |
| Analysis Engine | Scans for malicious content | High |
| Reporting Module | Generates downloadable analysis reports | Medium |
| User Management | Role-based access control | Medium |
| Threat API Lookup | Integrates VirusTotal, WHOIS for threat insights | High |

# 4. External Interface Requirements

## 4.1 User Interfaces

GUI Standards: I will adhere to modern GUI design principles to ensure the interface is intuitive and user-friendly.

Screen Layout: The layout will be designed to provide clear navigation and easy access to all system features.

Error Messages: User-friendly error messages will be implemented to guide users in resolving issues effectively.

## 4.2 Hardware Interfaces

Supported Devices: The system will be compatible with standard PCs and servers, ensuring broad accessibility.

Communication Protocols: Secure communication protocols will be used for data transfer to protect sensitive information.

## 4.3 Software Interfaces

Databases: The system will integrate with SQL databases for efficient data storage and retrieval.

Operating Systems: Compatibility with Windows and Linux will be ensured to support diverse deployment environments.

Libraries: I will utilize the JavaMail API and external WHOIS/IP libraries to enhance the system's email parsing and analysis capabilities.

## 4.4 Communications Interfaces

Email Protocols: Support for standard email protocols such as SMTP and IMAP will be included to facilitate seamless email analysis.

APIs: Integration with threat intelligence APIs will be implemented to enhance the system's analytical capabilities and provide up-to-date threat information.

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

**Response Time:** The system will be designed to process email analysis requests quickly, ensuring timely responses to potential threats.

**Throughput:** The system will be capable of handling multiple analysis requests simultaneously, ensuring scalability and efficiency.

## 5.2 Safety Requirements

**Data Protection:** Robust measures will be implemented to ensure the secure handling and storage of sensitive data, protecting user privacy and preventing data breaches.

**User Authentication**: Strong authentication mechanisms will be in place to control access to the system and ensure that only authorized users can perform sensitive operations.

## 5.3 Security Requirements

**Data Encryption:** Data will be encrypted both at rest and in transit to protect against unauthorized access and ensure data integrity.

**Access Control**: Role-based access control will be implemented to restrict access to system features based on user roles, enhancing security and accountability.

## 5.4 Software Quality Attributes

Reliability: The system will be designed for high availability and minimal downtime, ensuring consistent performance and dependability.

Maintainability: The system will be easy to update and maintain, with a modular design that facilitates quick modifications and enhancements.

Usability: The interface will be intuitive and user-friendly, ensuring that users can effectively utilize the system's features with minimal training.

# 6. Other Requirements

Database Requirements: The database will be designed for efficient storage and retrieval of analysis results, ensuring quick access to critical data.

Legal Requirements: The system will comply with relevant cybersecurity and data protection laws, ensuring legal and ethical operation.

# 7. References

1. Alsharnouby, M., Alaca, F., & Chiasson, S. (2015). *Why phishing still works: User strategies for combating phishing attacks*. International Journal of Human-Computer Studies, 82, 69-82.
2. Marchal, S., Francois, J., State, R., & Engel, T. (2014). *PhishStorm: Detecting phishing with streaming analytics*. IEEE Transactions on Network and Service Management, 11(4), 458-471.
3. Jakobsson, M., & Myers, S. (2006). *Phishing and countermeasures: Understanding the increasing problem of electronic identity theft*. Wiley-Interscience.

# 8.Appendix A: Glossary

* Phishing: Cyber deception aimed at stealing credentials or sensitive data.
* Spoofing: The act of falsifying sender information to mislead recipients.
* Attribution: The process of tracing the source of phishing attempts.
* VirusTotal: A service used for scanning and analyzing malware and suspicious links.
* Email Header: Metadata included in emails that can be used to trace the email's path and origin.