Low Level Design (LLD)

Phishing Domain Detection

Revision Number: 1.0

Last date of revision: 11/03/2021

Triveni Medici

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Date Issued | Version | Description | Author |
| 03th Nov 2022 | 1.0 | First Draft | Triveni Medici |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Contents**

[Document Version Control 2](#_gjdgxs)

[**Abstract** 4](#_30j0zll)

[1](#_1fob9te) Introduction 5

[1.1](#_3znysh7) Why this Low-Level Design Document? 5

[1.2](#_2et92p0) Scope 6

[1.4](#_3dy6vkm) 1.3 Risks 6

[1.5](#_1t3h5sf) 1.4 Out of Scope 7

[2](#_4d34og8) Technical specifications 7

[2.1 Predicting Disease 7](#_2s8eyo1)

[2.2 Logging 7](#_17dp8vu)

[2.3 Database 7](#_3rdcrjn)

[3](#_26in1rg) Technology stack 8

[4](#_lnxbz9) Proposed Solution 9

[5](#_35nkun2) Model training/validation workflow 10

[6](#_1ksv4uv) User I/O workflow 12

[7](#_44sinio) Exceptional scenarios 13

**Abstract**

Phishing stands for a fraudulent process, where an attacker tries to obtain sensitive information from the victim. Usually, these kinds of attacks are done via emails, text messages, or websites. Phishing websites, which are nowadays in a considerable rise, have the same look as legitimate sites. However, their backend is designed to collect sensitive information that is inputted by the victim. Discovering and detecting phishing websites has recently also gained the machine learning community’s attention, which has built the models and performed classifications of phishing websites.

The mail goal is to predict whether the domains are real or malicious.

# Introduction

## Why this Low-Level Design Document?

The purpose of this document is to present a detailed description of the Phishing Domain Detection Model . It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which url can be detected as phishing. This document is intended for both the stakeholders and the developers of the system and will be proposed to the higher management for its approval.

Phishing is a type of fraud in which an attacker impersonates a reputable company or person in order to get sensitive information such as login credentials or account information via email or other communication channels. Phishing is popular among attackers because it is easier to persuade someone to click a malicious link that appears to be authentic than it is to break through a computer's protection measures.

The main objective of the project is to predict if a url is phishing url or not

This project shall be delivered in two phases:

Phase 1: All the functionalities with PyPi packages.

Phase2: Integration of UI to all the functionalities.

## Scope

This software system will be a Web application This system will be designed to detect the phishing websites and prevent the cyber attacks and improve cyber security

## Constraints

For the phishing websites, only the ones from the Phish Tank registry were included, which are verified from multiple users. For the legitimate websites, we included the websites from publicly available, community labelled and organized lists [1], and from the Alexa top ranking websites.

## Out of Scope

Delineate specific activities, capabilities, and items that are out of scope for the project.

# Technical specifications

## 2.1 Dataset

Data were acquired through the publicly available lists of phishing and legitimate websites, from which the features presented in the datasets were extracted.

Data source: <https://doi.org/10.17632/72ptz43s9v.1>

Data format Raw: csv file

**Value of the Data**

* These data consist of a collection of legitimate, as well as phishing website instances. Each website is represented by the set of features that denote whether the website is legitimate or not. Data can serve as input for the machine learning process.
* Machine learning and data mining researchers can benefit from these datasets, while also computer security researchers and practitioners. Computer security enthusiasts can find these datasets interesting for building firewalls, intelligent ad blockers, and malware detection systems.
* This dataset can help researchers and practitioners easily build classification models in systems preventing phishing attacks since the presented datasets feature the attributes which can be easily extracted.
* Finally, the provided datasets could also be used as a performance benchmark for developing state-of-the-art machine learning methods for the task of phishing websites classification.

## 2.1.1 Data Description

The presented dataset was collected and prepared for the purpose of building and evaluating various classification methods for the task of detecting phishing websites based on the uniform resource locator (URL) properties, URL resolving metrics, and external services. The attributes of the prepared dataset can be divided into six groups:

The data is comprised of the features extracted from the collections of websites addresses. The data in total consists of 111 features, 96 of which are extracted from the website address itself, while the remaining 15 features were extracted using custom Python code.

attributes based on the whole URL properties presented in Table 1:

**Table 1:** Dataset attributes based on URL.

| **Nr.** | **Attribute** | **Format** | **Description** | **Values** |
| --- | --- | --- | --- | --- |
| 1 | qty\_dot\_url | Number of ”.” signs | Numeric |  |
| 2 | qty\_hyphen\_url | Number of ”-” signs | Numeric |  |
| 3 | qty\_underline\_url | Number of ”\_” signs | Numeric |  |
| 4 | qty\_slash\_url | Number of ”/” signs | Numeric |  |
| 5 | qty\_questionmark\_url | Number of ”?” signs | Numeric |  |
| 6 | qty\_equal\_url | Number of ”=” sings | Numeric |  |
| 7 | qty\_at\_url | Number of ”@” signs | Numeric |  |
| 8 | qty\_and\_url | Number of ”&” signs | Numeric |  |
| 9 | qty\_exclamation\_url | Number of ”!” signs | Numeric |  |
| 10 | qty\_space\_url | Number of ” ” signs | Numeric |  |
| 11 | qty\_tilde\_url | Number of ″′˜′signs | Numeric |  |
| 12 | qty\_comma\_url | Number of ”,” signs | Numeric |  |
| 13 | qty\_plus\_url | Number of ”+” signs | Numeric |  |
| 14 | qty\_asterisk\_url | Number of ”\*” signs | Numeric |  |
| 15 | qty\_hashtag\_url | Number of ”#” signs | Numeric |  |
| 16 | qty\_dollar\_url | Number of ”$” signs | Numeric |  |
| 17 | qty\_percent\_url | Number of ”%” signs | Numeric |  |
| 18 | qty\_tld\_url | Top level domain character length | Numeric |  |
| 19 | length\_url | Number of characters | Numeric |  |
| 20 | email\_in\_url | Is email present | Boolean | [0, 1] |

attributes based on the domain properties presented in Table 2

**Table 2** Dataset attributes based on domain URL.

| **Nr.** | **Attribute** | **Format** | **Description** | **Values** |
| --- | --- | --- | --- | --- |
| 1 | qty\_dot\_domain | Number of ”.” signs | Numeric |  |
| 2 | qty\_hyphen\_domain | Number of ”-” signs | Numeric |  |
| 3 | qty\_underline\_domain | Number of ”\_” signs | Numeric |  |
| 4 | qty\_slash\_domain | Number of ”/” signs | Numeric |  |
| 5 | qty\_questionmark\_domain | Number of ”?” signs | Numeric |  |
| 6 | qty\_equal\_domain | Number of ”=” signs | Numeric |  |
| 7 | qty\_at\_domain | Number of ”@” signs | Numeric |  |
| 8 | qty\_and\_domain | Number of ”&” signs | Numeric |  |
| 9 | qty\_exclamation\_domain | Number of ”!” signs | Numeric |  |
| 10 | qty\_space\_domain | Number of ” ” signs | Numeric |  |
| 11 | qty\_tilde\_domain | Number of ”signs | Numeric |  |
| 12 | qty\_comma\_domain | Number of ”,” signs | Numeric |  |
| 13 | qty\_plus\_domain | Number of ”+” signs | Numeric |  |
| 14 | qty\_asterisk\_domain | Number of ”\*” signs | Numeric |  |
| 15 | qty\_hashtag\_domain | Number of ”#” signs | Numeric |  |
| 16 | qty\_dollar\_domain | Number of ”$” signs | Numeric |  |
| 17 | qty\_percent\_domain | Number of ”%” signs | Numeric |  |
| 18 | qty\_vowels\_domain | Number of vowels | Numeric |  |
| 19 | domain\_length | Number of domain characters | Numeric |  |
| 20 | domain\_in\_ip | URL domain in IP address format | Boolean | [0, 1] |
| 21 | server\_client\_domain | ”server” or ”client” in domain | Boolean | [0, 1] |

attributes based on the URL directory properties presented in Table 3,

**Table 3:** Dataset attributes based on URL directory.

| **Nr.** | **Attribute** | **Format** | **Description** | **Values** |
| --- | --- | --- | --- | --- |
| 1 | qty\_dot\_directory | Number of ”.” signs | Numeric |  |
| 2 | qty\_hyphen\_directory | Number of ”-” signs | Numeric |  |
| 3 | qty\_underline\_directory | Number of ”\_” signs | Numeric |  |
| 4 | qty\_slash\_directory | Number of ”/” signs | Numeric |  |
| 5 | qty\_questionmark\_directory | Number of ”?” signs | Numeric |  |
| 6 | qty\_equal\_directory | Number of ”=” signs | Numeric |  |
| 7 | qty\_at\_directory | Number of ”@” signs | Numeric |  |
| 8 | qty\_and\_directory | Number of ”&” signs | Numeric |  |
| 9 | qty\_exclamation\_directory | Number of ”!” signs | Numeric |  |
| 10 | qty\_space\_directory | Number of ” ” signs | Numeric |  |
| 11 | qty\_tilde\_directory | Number of ”signs | Numeric |  |
| 12 | qty\_comma\_directory | Number of ”,” signs | Numeric |  |
| 13 | qty\_plus\_directory | Number of ”+” signs | Numeric |  |
| 14 | qty\_asterisk\_directory | Number of ”\*” signs | Numeric |  |
| 15 | qty\_hashtag\_directory | Number of ”#” signs | Numeric |  |
| 16 | qty\_dollar\_directory | Number of ”$” signs | Numeric |  |
| 17 | qty\_percent\_directory | Number of ”%” signs | Numeric |  |
| 18 | directory\_length | Number of directory characters | Numeric |  |

attributes based on the URL file properties presented in Table 4,

**Table 4:** Dataset attributes based on URL file name.

|  |
| --- |
|  |
| **Nr.** | **Attribute** | **Format** | **Description** | **Values** |
| 1 | qty\_dot\_file | Number of ”.” signs | Numeric |  |
| 2 | qty\_hyphen\_file | Number of ”-” signs | Numeric |  |
| 3 | qty\_underline\_file | Number of ”\_” signs | Numeric |  |
| 4 | qty\_slash\_file | Number of ”/” signs | Numeric |  |
| 5 | qty\_questionmark\_file | Number of ”?” signs | Numeric |  |
| 6 | qty\_equal\_file | Number of ”=” signs | Numeric |  |
| 7 | qty\_at\_file | Number of ”@” signs | Numeric |  |
| 8 | qty\_and\_file | Number of ”&” signs | Numeric |  |
| 9 | qty\_exclamation\_file | Number of ”!” signs | Numeric |  |
| 10 | qty\_space\_file | Number of ” ” signs | Numeric |  |
| 11 | qty\_tilde\_file | Number of ”signs | Numeric |  |
| 12 | qty\_comma\_file | Number of ”,” signs | Numeric |  |
| 13 | qty\_plus\_file | Number of ”+” signs | Numeric |  |
| 14 | qty\_asterisk\_file | Number of ”\*” signs | Numeric |  |
| 15 | qty\_hashtag\_file | Number of ”#” signs | Numeric |  |
| 16 | qty\_dollar\_file | Number of ”$” signs | Numeric |  |
| 17 | qty\_percent\_file | Number of ”%” signs | Numeric |  |
| 18 | file\_length | Number of file name characters | Numeric |  |

attributes based on the URL parameter properties presented in Table 5, and

Table 5: Dataset attributes based on URL parameters.

| **Nr.** | **Attribute** | **Format** | **Description** | **Values** |
| --- | --- | --- | --- | --- |
| 1 | qty\_dot\_params | Number of ”.” signs | Numeric |  |
| 2 | qty\_hyphen\_params | Number of ”-” signs | Numeric |  |
| 3 | qty\_underline\_params | Number of ”\_” signs | Numeric |  |
| 4 | qty\_slash\_params | Number of ”/” signs | Numeric |  |
| 5 | qty\_questionmark\_params | Number of ”?” signs | Numeric |  |
| 6 | qty\_equal\_params | Number of ”=” signs | Numeric |  |
| 7 | qty\_at\_params | Number of ”@” signs | Numeric |  |
| 8 | qty\_and\_params | Number of ”&” signs | Numeric |  |
| 9 | qty\_exclamation\_params | Number of ”!” signs | Numeric |  |
| 10 | qty\_space\_params | Number of ” ” signs | Numeric |  |
| 11 | qty\_tilde\_params | Number of ”signs | Numeric |  |
| 12 | qty\_comma\_params | Number of ”,” signs | Numeric |  |
| 13 | qty\_plus\_params | Number of ”+” signs | Numeric |  |
| 14 | qty\_asterisk\_params | Number of ”\*” signs | Numeric |  |
| 15 | qty\_hashtag\_params | Number of ”#” signs | Numeric |  |
| 16 | qty\_dollar\_params | Number of ”$” signs | Numeric |  |
| 17 | qty\_percent\_params | Number of ”%” signs | Numeric |  |
| 18 | params\_length | Number of parameters characters | Numeric |  |
| 19 | tld\_present\_params | TLD1present in parameters | Boolean | [0, 1] |
| 20 | qty\_params | Number of parameters | Numeric |  |

attributes based on the URL resolving data and external metrics presented in Table 6.

**Table 6:** Dataset attributes based on resolving URL and external services.

| **Nr.** | **Attribute** | **Format** | **Description** | **Values** |
| --- | --- | --- | --- | --- |
| 1 | time\_response | Domain lookup time response | Numeric |  |
| 2 | domain\_spf | Domain has SPF 2 | Boolean | [0, 1] |
| 3 | asn\_ip | ASN 3 | Numeric |  |
| 4 | time\_domain\_activation | Domain activation time (in days) | Numeric |  |
| 5 | time\_domain\_expiration | Domain expiration time (in days) | Numeric |  |
| 6 | qty\_ip\_resolved | Number of resolved IPs | Numeric |  |
| 8 | qty\_nameservers | Number of resolved NS4 | Numeric |  |
| 9 | qty\_mx\_servers | Number of MX 5servers | Numeric |  |
| 10 | ttl\_hostname | Time-To-Live (TTL) | Numeric |  |
| 11 | tls\_ssl\_certificate | Has valid TLS 6/SSL 7certificate | Boolean | [0, 1] |
| 12 | qty\_redirects | Number of redirects | Numeric |  |
| 13 | url\_google\_index | Is URL indexed on Google | Boolean | [0, 1] |
| 14 | domain\_google\_index | Is domain indexed on Google | Boolean | [0, 1] |
| 15 | url\_shortened | Is URL shortened | Boolean |  |
| 16 | phishing | Is phishing website | Boolean | [0, 1] |

## 2.2 Detecting Phishing URL

The first group is based on the values of the attributes on the whole URL string, while the values of the following four groups are based on the particular sub-strings, as presented in Figure. The last group attributes are based on the URL resolve metrics as well as on the external services such as Google search index.

Fig. 1

## 2.3 Logging

We should be able to log every activity done by the user.

* The System identifies at what step logging required
* The System should be able to log each and every system flow.
* Developers can choose logging methods. You can choose database logging/ File logging as well.
* System should not be hung even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

## 2.4 Database

System needs to store every request into the database and we need to store it in such a way that it is easy to retrain the model as well.

1. The User chooses the disease.

2. The User gives required information.

3. The system stores each and every data given by the user or received on request to the database. Database you can choose your own choice (Cassandra).

**2.5 Deployment**

1. AWS



# Technology stack

|  |  |
| --- | --- |
| **Front End** | HTML/CSS/JS/React |
| **Backend** | Python Django |
| **Database** | Cassandra |
| **Deployment** | AWS |

# Proposed Solution

refer: <https://www.sciencedirect.com/science/article/pii/S2352340920313202>

The classical machine learning tasks like Data Exploration, Data Cleaning, Feature Engineering, Model Building and Model Testing. Try out different machine learning algorithms that’s best fit for the above case.

For Feature Engineering show:-

1. URL-Based Features

2. Domain-Based Features

3. Page-Based Features

4. Content-Based Features

Baseline Model: Logistic Regression, since this is a classification problem.

Actual model: XG Boast

# Model training/validation workflow



# User I/O workflow

Diagram

Description automatically generated

# Exceptional scenarios

|  |  |  |  |
| --- | --- | --- | --- |
| Step | Exception | Mitigation | Module |
|  |  |  |  |
|  |  |  |  |

# Test cases

|  |  |  |  |
| --- | --- | --- | --- |
| Test case | Steps to perform test case | Module | Pass/Fail |
|  |  |  |  |

# Key performance indicators (KPI)

* Detecting the malicious websites will help in preventing cyber attacks
* Comparison of accuracy of model prediction and real time data prediction.