Low Level Design (LLD)

Phishing Domain Detection

Written By	Abhijeet R. Waghchaure
Document Revision Number	1.0
Last Date of Revision	15/08/2022

Document Version Control

Date Issued	Version	Description	Author
15th Aug 2022	1.1	First Draft-LLD	Abhijeet Waghchaure

Reviews:

Version	Review Date	Reviewer	Comments

Approval Status:

Version	Review Date	Reviewer	Approved By	Comments

Contents

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. This paper presents two dataset variations that consist of 58,645 and 88,647 websites labelled as legitimate or phishing and allow the researchers to train their classification models, build phishing detection systems, and mining association rules.

1. INTRODUCTION

1.1 Why this Low-Level Design Document?

The purpose of this Low-Level Design (LLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

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

1.2 Scope

The LLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The LLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system. This software system will be a Web application. This system will be designed to detect unusual activity, and fire disasters.

1.3 Constraints

We will only be selecting numerical features and not complete URL's for this projects, detecting malicious domains are malicious or not.

1.4 Risks

Document specific risks that have been identified or that should be considered.

1.5 Out of Scope

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

2. Technical specifications of the Dataset

- The 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 Dataset overview

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:

2.2 Input schema:

• Attributes based on the whole URL properties presented in Table 1,

• Attributes based on the domain properties presented in Table 2,

• Attributes based on the URL directory properties presented in Table 3,

• Attributes based on the URL file properties presented in Table 4,

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

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

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]

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]

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 1. The last group attributes are based on the URL resolve metrics as well as on the external services such as Google search index.

The dataset in total features 111 attributes excluding the target phishing attribute, which denotes whether the particular instance is legitimate (value 0) or phishing (value 1). We prepared two variations of the dataset, the one where the total number of instances is 58,645 and the balance between the target classes in more or less balanced with 30,647 instances labelled as phishing websites and 27,998 instances labelled as legitimate.

The second variant of the dataset is comprised of 88,647 instances with 30,647 instances labelled as phishing and 58,0 0 0 instances labelled as legitimate, the purpose of which is to mimic the real-world situation where there are more legitimate websites present. The distribution between the classes of both dataset variants is presented in Figure 2.

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	

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	

Fig. 1. Separation of the whole URL string into sub-strings.

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	TLD[1] present in parameters	Boolean	[0, 1]
20	qty_params	Number of parameters	Numeric	

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 NS[4]	Numeric	
9	qty_mx_servers	Number of MX [5]servers	Numeric	
10	ttl_hostname	Time-To-Live (TTL)	Numeric	
11	tls_ssl_certificate	Has valid TLS [6]/SSL [7]certificate	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]**

2.3 Logging

We should be able to log every activity done by the developer in the code.

- 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.

- The User chooses the activity dataset.
- The User gives required information.
- 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 whether MySQL, SQLite etc.

3. Deployment

1. MS Azure
2. Google Cloud
3. AWS

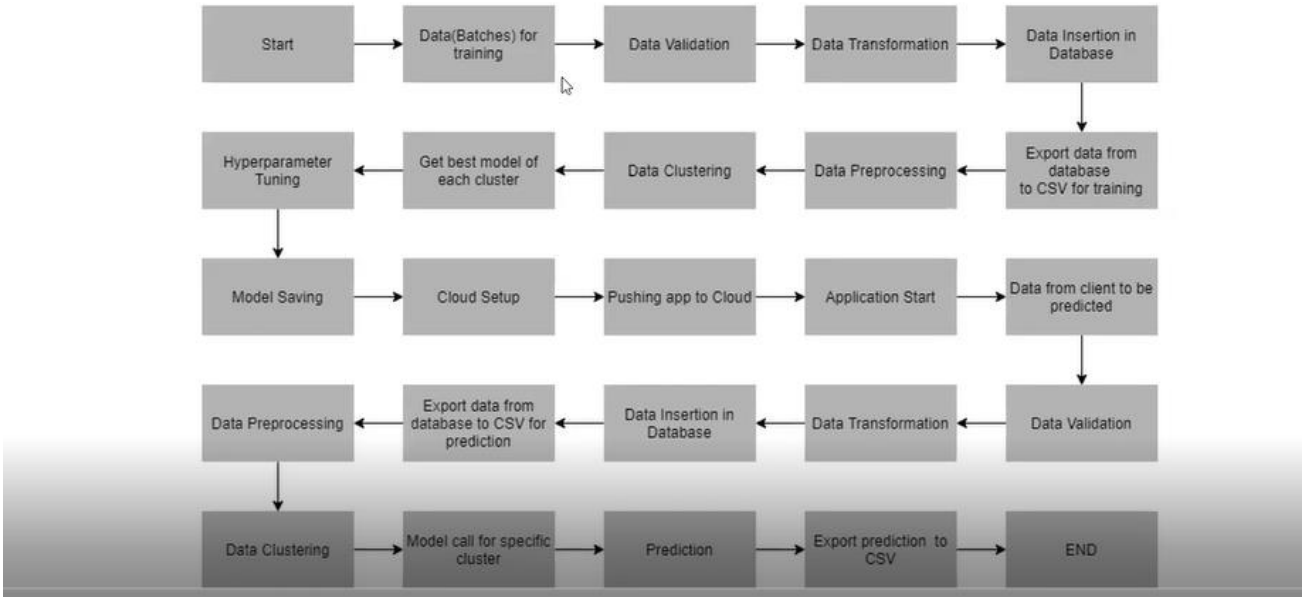
4. Technology stack

Front End	HTML/CSS/JSt
Backend	Python
Database	SQLite
Deployment	Flask, Heroku

5. Proposed Solution

The solution proposed here is a Phishing Domain detection can be implemented to perform above mention use cases. In first case, if where an attacker tries to obtain sensitive information from the victim. Usually, these kinds of attacks are done via emails, text messages, or websites. 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.

6. Model training/validation workflow



Developer will host this application in the web. User must use this application on the website.

8. Error Handling

Should errors be encountered, an explanation will be displayed as to what went wrong?

An error will be defined as anything that falls outside the normal and intended usage.

9. Test cases

Use case	Module	Accuracy
Model Performance	XGBoost Classifier	97.264
Model Performance	Random Forest	90.334

10. Key performance indicators (KPI)

- Key indicators displaying a summary of the phishing domain detection.
- To detect malicious activities and inform cyber security team.
- Taking adequate evidence of the URL.
- Send URL details to concerned authorities.
- Length of the URL.
- Character in the URL

11. Conclusion

- The final take away from this project is to explore various machine learning models, perform Exploratory Data Analysis on phishing dataset and understanding their features.
- Creating ipynb notebook or .py files will help me to learn a lot about the features affecting the models to detect whether URL is safe or not, also I will came to know how to tuned model and how they will affect the model performance.
- The final conclusion on the Phishing dataset is that the some feature like "HTTTPS", "AnchorURL", "WebsiteTraffic" will have more importance to classify URL is phishing URL or not.
- Some Classifiers correctly classify URL up to some percentage with respective to classes and hence reduces the chance of malicious attachments.