A Smart Email Client to Help User Identify Malicious URLs

Project Plan

Supervised By

Dr. T.W. Chim

Researched By

Wu Zekai (3035468586) & Yau Lai Shun (303541179)

04 - Oct - 2020
Contents
1 Background ........................................................................................................................................ 3
  1.1 Overview of malicious URLs ........................................................................................................ 3
  1.2 Machine learning approach to detect malicious URLs .................................................................... 3
2 Objective ............................................................................................................................................... 4
  2.1 To develop a lightweight, web-based email client .......................................................................... 4
  2.2 To implement an accurate, effective malicious URLs detector ....................................................... 4
3 Methodology ........................................................................................................................................ 5
  3.1 Implementation of the email client .................................................................................................. 5
    3.1.1 Front-end implementation ......................................................................................................... 5
    3.1.2 Back-end implementation ......................................................................................................... 5
  3.2 URLs classification ........................................................................................................................ 6
    3.2.1 Data processing ......................................................................................................................... 6
    3.2.2 Training the machine learning models ....................................................................................... 6
4 Schedule and Milestones .................................................................................................................... 7
  4.1 Project schedule ............................................................................................................................ 7
  4.2 Project milestones .......................................................................................................................... 8
Reference ................................................................................................................................................ 9
1 Background

1.1 Overview of malicious URLs

Nowadays, with the popularity of the Internet, email has become a popular way for people to communicate. According to the email statistics report of Radicati Group, there are 3.9 billion email users all over the world in 2019, and 293 billion emails had been sent and received every day in 2019 [1]. Unfortunately, some criminals begin to use email to cheat and attack email users include individuals and companies, by inserting malicious URLs (Malicious Uniform Resource Locators).

The majority categories of malicious URLs attacks include phishing and drive-by download [2]. Phishing URLs is a type of email fraud that redirects users to a fake website instead of well-known and trusted organizations. Victims may input sensitive information such as credit card numbers, passwords, identity card numbers, addresses on websites. Drive-by download attacks will download malicious software or virus to users’ computers. Malicious software may steal information, push advertisements, or damage computers and servers. Users may leak sensitive intellectual property, personal information, even national-security secrets [3] and hence to suffer financial loss.

1.2 Machine learning approach to detect malicious URLs.

In order to provide a safe mailing environment, researchers have applied machine learning to detect malicious URLs. The machine learning approach is able to analyze the information of a URL and its corresponding websites [2]. This approach extracts the features which represent a URL such as the lexical features from URL string, the information about the host, the web page source content, and so forth [2]. Hence, we can firstly train a prediction model by analyzing a huge number of malicious and harmless URLs and then use such a mature model to detect whether a new URL is malicious or not.
2 Objective

In this project, we aim to build a smart email client who not only includes the basic functionalities of an email client but also has the ability to read through the contents of the emails and perform URLs detection based on the fetched contents. Hence, the objectives of this project are composited by two main parts, the development of the email client and the implementation of the malicious URLs detector.

2.1 To develop a lightweight, web-based email client

Although plenty of excellent, open-source email clients are available in the market for modification and improvement, we still seek to build our own email client for this project in order to be more compatible with our malicious URLs detector. During the preliminary implementation of the email client, we aspire to develop a lightweight version, and hence only some basic features will be included, such as writing emails, sending emails, and receiving emails. We choose to build a web-based email client as the cost is relatively low, and we can use it more freely on any device with any operating system. Additionally, we will inspect our preliminary product when we reach the finalized tested implementation, and more features or platforms may be included if needed.

2.2 To implement an accurate, effective malicious URLs detector

As stated in Chapter 1.2, we could train a prediction model to analyze the essence of a URL without directing to the website behind it. This is exactly what we want to help the user to identify malicious URLs, and thus we aim to implement such malicious URLs detector based on the prediction model. During the preliminary implementation of the malicious URLs detector, we expect that the detector is able to work in the background, i.e., running on a terminal and outputting the detection result to the console. We will continuously improve the performance of the detector, such as improving the detection accuracy and reducing the time needed for URLs detection. After that, we will merge it with the developed email client to act as a finalized tested product.
3 Methodology

3.1 Implementation of the email client
The final product of the email client in this project is a lightweight, web-based application. The system architecture (tentative) for the email client is provided in Figure 3.1. Please refer to the sub-chapters below for more details.

![Figure 3.1: the system architecture of the final product](image)

3.1.1 Front-end implementation
For the front-end implementation of the email client, some basic components in web development such as HTML (Hypertext Markup Language), CSS (Cascading Style Sheets), and JavaScript will definitely be included to form the main body of the application. In addition, we will also use some extra frameworks for improving development efficiency. For instance, we may use React for boosting productivity, ensuring faster rendering, and so forth. We may also use Bootstrap for providing responsive styles, building a beautiful user interface, and so forth.

3.1.2 Back-end implementation
For the back-end implementation of the email client, Python will be used as our back-end programming language. The project will also import the Django framework to help us develop rapidly and efficiently. Besides, we will also use the Gmail API, which is provided by Google for authenticating Gmail users and obtaining their email data. The email client will read the content of the emails and fetch and the information of the URLs included in the emails to our server. After that, our machine learning model will perform URLs classification and return the result back to the email client.
3.2 URLs classification
We will apply machine learning to classify whether the URLs in the email are malicious URLs or benign URLs. A general process of malicious URL classification has several steps (Figure 3.2).

![Figure 3.2: the URLs classification model](image)

3.2.1 Data processing
Firstly, data of URLs will be crawled from the Internet and saved into the URL database. There are some websites which already contain a database of malicious or benign URLs. Secondly, the features of the URLs will be extracted. An URL contains several features include a hostname, protocol, hostname, top-level domain, path, some lexical features. And then, we label whether the URLs are malicious or benign corresponding to their features.

3.2.2 Training the machine learning models
After collecting and extracting data, model training can be performed to train a prediction model. There are many machine learning algorithms that are suitable for our projects, such as logistic regression and random forest. a Python library called scikit-learn will be imported. The library provides tools for model fitting, data pre-processing, model selection, and evaluation.
4 Schedule and Milestones

4.1 Project schedule
The project lasts from September of 2020 to May of 2021. During this period, the tasks scheduled for each month are listed in Table 4.1.

<table>
<thead>
<tr>
<th>Time</th>
<th>Scheduled Tasks</th>
</tr>
</thead>
</table>
| 2020, Sep | • Review on existing open-source email clients  
• Review on existing online URL checkers  
• Research on the options for system architecture                                                |
| 2020, Oct | • Define the system architecture for both front-end and back-end implementations  
• Further research and study on the defined system architecture and set up and development environment correspondingly |
| 2020, Nov | • Design and implement the UI for the email client  
• Design and implement the database for URLs dataset storing  
• Design and implement different machine learning models using different algorithms |
| 2020, Dec | • Implement the basic functionalities of the email client  
• Test and compare machine learning models                                                  |
| 2021, Jan | • Test and refine on the preliminary product  
• Prepare and deliver the first presentation  
• Prepare and deliver the interim report                                                   |
| 2021, Feb | • Improve the product based on the feedback got from the supervisor                                                                       |
| 2021, Mar | • Integrate the email client and the malicious URLs detector  
• Further, test and refine on the product                                                  |
| 2021, Apr | • Prepare and deliver the final presentation  
• Prepare and deliver the final report                                                     |
| 2021, May | • Prepare for the project exhibition                                                    |

*Table 4.1: project schedule*
4.2 Project milestones
Apart from the monthly project schedule, we also plan to organize the project in a systematic manner. The project is divided into four main milestones (phases), which are the inception phase, elaboration phase, construction phase, and transition phase. The duration and deliverables for each phase are shown in Figure 4.1.

Figure 4.1: project milestones
Reference

