Project Charter

EECE3093 Software Engineering

Brayden Osborne, Austin Reule, Alex Mezcua, Hoang Vu

5/21/2019

**Phishing Email Detector**

**Description:**

We are making a program that will use words or phrases common in phishing emails to be able to detect if an email sent to the user has potential of being a phishing email. The idea is that the user will be able to copy the contents of the email into a program that will determine if the email is most likely safe or not by having a database of phishing emails and the commonality in them. Based on the contents of the email it will return a rating based on the potential of how dangerous the email could be based on sentence structure, grammar, etc.

**Purpose:**

Phishing is defined as “the fraudulent practice of sending emails purporting to be from reputable companies in order to induce individuals to reveal personal information, such as passwords and credit card numbers.” It is a practice with a multitude of victims and it often preys on the most vulnerable of internet users. We aim to develop an application capable of identifying the likelihood of an email being illegitimate in order to allow those who are the most susceptible to phishing to have a better chance of catching malicious emails. An application such as this has the potential to stop some of the “$12 billion in losses” reported by the FBI.

**Background:**

* 92.4% of malware is still sent via email.
* In 2017, the average user was receiving 16 email attacks per month.
* 76% of organizations say that they experienced phishing attacks in 2017.
* The average financial cost of a data breach is $3.86m

(According to www.alertlogic.com)

**Scope:**

INCLUDED

* Manual input of email contents into an application
* Automatic detection of common phishing terms
* Return of some sort of danger rating
* Give useful information to the user about how to check if the email is potentially malicious on their own (“Is there a clickable link?”, “Is this email expected?”, “Did you sign up for email from the sender?”, etc.)

POSSIBLY INCLUDED

* Use of a neural network to detect common phishing email features (such as spelling and sentence structure)

EXCLUDED

* Automatic email reading (web scraping)
* Features specific to certain email providers (Yahoo, Google, etc.)

**Objectives:**

* Learn about Project Management and Software Development
* Develop software capable of identifying phishing emails

**Participants:**

Brayden Osborne - Project Lead/Backend Developer

Austin Reule - Programmer/App Designer

Alex Mezcua - Programmer/Tester

Hoang Vu - Programmer/Frontend Developer

**Risks:**  
 False negatives have a high potential for user harm. If a user is told there is a “0%” chance of an email being a phishing attempt and acts accordingly, while in fact it is a malicious email, they could be seriously impacted.

**Benefits:**

* Increased awareness for phishing emails
* Less people getting scammed

**Cost:**

* $5 Chrome Store Fee
* Time
* Potential cost to access databases

**Timeline:**

5/16: Brainstorm topic

5/23: Completed Project Charter

6/6: Have database of known phishing emails gathered,

Define loose project structure (backend, frontend, etc.)

Clearly define Github layout

Create project files

6/13: Beginning of backend development, implementations with dummy functions

6/20: Continued backend development, some functions implemented

6/27: Roughly implemented backend workflow

7/4: Test-ready backend code, working functions etc.

7/11: Implemented frontend workflow with dummy functions

7/18: Test-ready frontend code, working functions etc.

7/25: Testing

End of Semester: Completed Software on 2 platforms

**Constraints:**  
 Those who write and send phishing emails are constantly evolving their methods of tricking their victims. It will be difficult to develop a service that is capable of keeping up with the latest scamming trends.

Another potential constraint is that users may not be willing to input emails into the system unless they are already suspicious of their contents. This is somewhat of a paradox since the whole purpose of the application is to raise the user’s awareness of a potential phishing email.

We are not experts, and any sort of danger rating that we produce is at-best an aggregation of potential risks that we identify. We are not able to do any sort of statistical analysis on actual likelihood of risk.

**Assumptions:**

We assume that while phishing emails are constantly changing, they have common trends amongst them that we are capable of identifying. We also assume that users would be willing to manually enter an email into the system, and that mose phishing emails are of a low enough quality that even the most uninformed user would be somewhat suspicious.