ENS 491 – Graduation Project

(Design) (Draft) Proposal

Project Title:

Designing and Implementing Smart Phish Generator

Group Members:

Onat Uzunyayla

Andaç Ünlü

Mehmet Sencer Yaralı

Supervisors:

Orçun Çetin

Onur Varol

Date: 31/10/2021

Sabancı Üniversitesi

1. Abstract

Today, phishing is one of the most popular tools used by cyber-criminals for many malicious purposes, but mostly to steal private information. Phishing is a method that sends messages to victims, which seems like a normal message that is sent from a legitimate institution or person but it is actually a fake message created by cyber-criminals. A phishing message can be sent through different mediums, but it is mostly sent as an E-mail message. A phishing message’s effectiveness is measured with the link’s clicking rate since phishing messages are harmless till the link inside the message is clicked by the victim. To increase the clicking rate of the victims, cyber-criminals use techniques of both psychology and computer science. A very common psychological trick is evoking urgency to make victims click the phishing link as quickly as possible since phishing sites do not last for a long time.

We propose a machine learning based smart phishing component, that bypasses the spam filters of the mailing services and achieves high clicking rate by generating personalized phishing mails. We aim to achieve the high clicking rate by generating emails based on the victim's personal information such as location, gender, interests, shopping behavior, financial information etc. We are planning to use Twitter’s trending topics to generate more interesting emails which will achieve more clicking rate. Creating such a smart phishing component will give a better understanding about the phishing attacks and ways to prevent them also, this tool could be used to test phishing filters which will make the development of better filters possible.

2. Introduction

Phishing is the process where one tries to trick users by representing them with malignant websites, or emails instead of legitimate ones in order to acquire private information of the users. In today's era, phishing is one of the most important threads one can face in the digitized world. Designers must have knowledge about which attack strategies work and why to be able to build systems shielding users from phishing attacks (Dhamija, Tygar, Hearst, 2006). It is especially crucial for this project to understand which techniques and tools attackers use while creating phishing attacks so that the project can design and implement a smart phish generator as efficiently and effectively as possible. Lack of knowledge, visual deception, and bounded attention are the main causes why many users fall for these phishing attacks, and designing phishing content as carefully and accurately as possible increases the success of such attacks (Dhamija et al. 2006). There are different approaches to detect phishing emails, one of them is random forest machine learning algorithm with an objective of improving the phishing emailer classifier with better prediction accuracy and fewer numbers of features (Akinyelu, Adewumi, 2014). Phishing email filters use different techniques to protect users, and compared to all other techniques mentioned, machine learning achieved the best result according to Akinvelu and Adewumi, therefore we will use machine learning algorithms dominantly for this project (2014). Since we will not process sensitive data, we are planning to use centralized machine learning algorithms since they yield generally better accuracy than the decentralized algorithms. Generating phishing messages by using phishing datasets is a complex NLP task, therefore we are planning to use a pipeline which consists of some features of a Variational Auto-encoder architecture and a pre-trained BERT model, which has a better word embedding approach than the previous methods. BERT is an NLP model proposed by Google researchers in 2018 on the paper "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" and it gives fantastic results on NLP tasks, since it creates the embedding vector of a word by also calculating its contextual meaning and index. Pipeline will also receive input from the Twitter API, and other various inputs to generate the phishing Email. Getting to know what's happening inside of a security system could make the systems' flaws more visible. MLES is a machine learning powered Expert System that analyzes the properties of phishing urls by looking up their expiration times, url lengths, url portions such as domain and directory etc (Fitzpatrick, 2021). MLES can also detect Fake News by using the same techniques. What is special about this system is that even though there are already ready-to-use tools to detect phishing attempts like PhisTank and many blocklist with user-submitted data, it is never safe enough. For instance, LIAR dataset for sentimental analysis is utilized for this work in order to target fake news. Fake news are especially important in this case due to some psychological reasons like confirmation bias we have seen during 2020 US Presidential Elections (Fitzpatrick, 2021). These insights from this MLES system is going to be a great knowledge to keep in mind so that the solution the team of Smart phishing will come up with may be more accurate and successful.

3. Proposed Solution and Methods

Thanks to phishing attacks, millions of dollars are being lost by many companies and individuals every year. Our project aims to improve both the systems used to protect users from such attacks by pointing out the weaknesses of these detection systems through creating a smart phishing generator that tricks these systems and inform individuals about the dangers of these phishing attacks by tricking the human factor as well. The scope of this project is to generate phishing emails that can trick both the system and human factors, generating phishing websites is out of the scope of this project, and the goal is to be able to bypass the security systems first, and then find elements and techniques to bypass the human factor also. The implementation of the project starts with being able to generate carefully designed emails that would look legitimate to users, then the next step is to contemplate ways in order to bypass the security systems that protect users from phishing attacks, after accomplishing that the challenge is to trick the human factor into clicking the emails and the links by making them believe the email they received is legitimate, and have the contents of the emails get their interests. The solution of our problem starts with gaining knowledge about techniques and tools to generate these smart phishing emails, bypass the security systems and the human factor, and to solve these complex issues further research and testing is necessary.

3.1 Realistic constraints

Manufacturability of the final product will not be possible due to potential commitments of fraud in the hands of criminals. In order to ensure this the source code for the project will be private to only developers. For testing purposes, the code will most likely run on a server after the finalization of testing.

Development of this project does not require big fundings behind it, considering the fact that the final product is simply a software to achieve a greater purpose. The tools that will be utilized down the road of the development process are not necessarily charging any fee due to the tools being fundamental programming languages only. However, any additional cost that will be provided for the testing subjects in the future will be compensated by the university, which will be organized by the supervisors.

Achieving true smart phishing is a big deal. The results of this project will reveal some certain facts related to the cyber security field and yield a sustainable information that can affect the security applications for phishing detection.

It’s almost impossible to face any health issues and hazardous situations. Other than that, there is no possible safety issue either, however, the subjects who will encounter our harmless phishing attempts might be offensive so it will be a good practice to ensure anonymity. Social responsibilities are severely crucial inside this project’s operational area. Thus, university officials and government officials will be informed about the situation in order to make sure there is no misunderstanding and obviously this is not a criminal activity. This will ensure that this project is not an offensive one, but an academic one. Without any doubt, legal background will be recognized by the team.

4. Risk Management

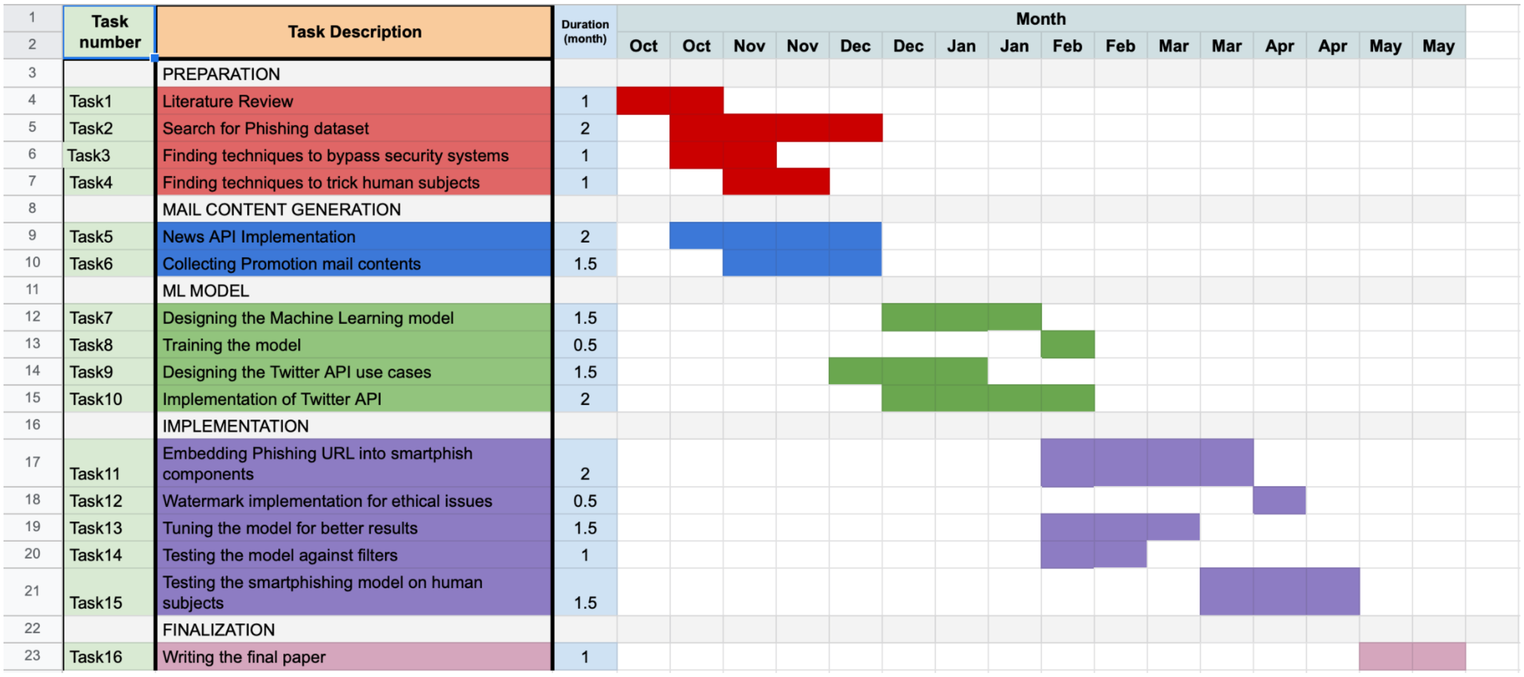
One of the risks can be not being able to access/use the resources planned while creating the content for smart phishing emails. In that situation, we plan to request the necessary access to be able to use the resources or find any other materials we may be able to use. Another risk can be the knowledge we have about creating these smart phishing email materials may not be enough, in that case, we plan to do further research and ask our supervisors to guide us about how to proceed.

5. Project Schedule

As it can be seen in the Figure A, Gantt chart and overall roadmap for the project is ready. Since coding implementation of tasks can be complicated and are likely to contain errors, the team decided on scheduling the project to end 1 month earlier than usual. Therefore any expected delays will be compensated beforehand so that the team will have an additional one month to cover up those incomplete tasks. Recognizing the possible delays for this project will affect the development time fundamentally as it is stated that the project's main purpose is to come up with an end product. However, "teamwork makes the dreamwork" must be the motto of the team which means any task that finishes earlier than it is expected then the free team member can switch to any of the on-going tasks to speed up the development.

Figure A

Gantt chart of the tasks



Task number	Task Description	Duration (month)	Month															
			Oct	Oct	Nov	Nov	Dec	Dec	Jan	Jan	Feb	Feb	Mar	Mar	Apr	Apr	May	May
	PREPARATION																	
Task1	Literature Review	1	🟥	🟥														
Task2	Search for Phishing dataset	2		🟥	🟥	🟥	🟥											
Task3	Finding techniques to bypass security systems	1		🟥	🟥													
Task4	Finding techniques to trick human subjects	1			🟥	🟥												
	MAIL CONTENT GENERATION																	
Task5	News API Implementation	2		🟦	🟦	🟦	🟦											
Task6	Collecting Promotion mail contents	1.5			🟦	🟦	🟦											
	ML MODEL																	
Task7	Designing the Machine Learning model	1.5						🟩	🟩	🟩								
Task8	Training the model	0.5									🟩							
Task9	Designing the Twitter API use cases	1.5					🟩	🟩	🟩									
Task10	Implementation of Twitter API	2						🟩	🟩	🟩	🟩							
	IMPLEMENTATION																	
Task11	Embedding Phishing URL into smartphish components	2									🟪	🟪	🟪	🟪				
Task12	Watermark implementation for ethical issues	0.5													🟪			
Task13	Tuning the model for better results	1.5									🟪	🟪	🟪					
Task14	Testing the model against filters	1									🟪	🟪						
Task15	Testing the smartphishing model on human subjects	1.5												🟪	🟪	🟪		
	FINALIZATION																	
Task16	Writing the final paper	1															▨	▨

Note. The task list might change in the future.

6. Ethical issues

Our project's aim is to generate smart phishing emails that can deceive users in order to improve the security systems used to protect users from such attacks. To be able to test if our phishing emails are successful, we will have a control group that agrees to receive these emails in order to not break any ethical rules. If they are tricked into clicking these smart phishing emails, we will show them a warning that they were a victim of a phishing attack and warn them about such attacks, and not receive any private information. Another ethical issue we may face could be someone else implementing their own design based on our smart phishing generator for malicious purposes, in order to avoid this issue we will have a watermark within the source code, to prove we are not committing such crimes to not have any ethical issues.

7. References

Akinyelu, A. A., & Adewumi, A. O. (2014). Classification of Phishing Email Using Random Forest Machine Learning Technique. *Journal of Applied Mathematics*, *2014*, 1–6. https:// doi.org/10.1155/2014/425731

Fitzpatrick, B., Liang, X., & Straub, J. (2021). Fake News and Phishing Detection Using a Machine Learning Trained Expert System. Retrieved from arxiv-2108.08264

Dhamija, R., Tygar, J. D., & Hearst, M. (2006). Why phishing works. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. https://doi.org/ 10.1145/1124772.1124861