The Honors College at the University of Missouri-Kansas City

Day of the Week Twitter Phishing Impact

Zachary Theiss

May 17th, 2019

Written under the direction of Professor Kuhail

The School of Computing and Engineering

A thesis submitted in partial fulfillment of the requirements to graduate as an Honors Scholar from the University of Missouri-Kansas City

Abstract

Phishing has become ever more prevalent in everyday life with new attacks and attempts being made every hour of every day. Twitter has been a major social media player for many years now and continues to deal with phishing in every post. Phishing attempts are harmful to every user and currently most individuals cannot identify a phishing tweet, nor accept appropriately and avoid them in their entirety. Our original hypothesis was that the day of the week would impact the number and frequency of phishing attempts.

We created a Python-based program, in conjunction with the Python Module of Tweepy to catch posts to Twitter over a two-week period of July 2nd to July 15th. The data was then processed through ScrapeBox to identify phishing tweets with Google Safe Browsing API. The results were then identified by date, time, day of the week, and specific post URL. From there, another Python Module called Pandas was used to manage the over 8 billion twitter posts as well as gather statistical information about our data to find a statistically significant aspect. Conclusions were drawn based on the influence of the day of the week which lead us to our conclusion about Twitter phishing attempts throughout the week and including holidays.

Introduction

Wombat Security, a leader in the tracking of phishing attempts, cited that in 2016, "76% of organizations reported being victim of a phishing attack."

[1] Phishing is specifically the act of stealing information from an individual or organization in a misleading or aggressive manner that would trick the individual into believing they are giving out personal information to a lawful and trustworthy entity.

Using Tweepy, ScrapeBox, and Pandas we will dive into the validity of Twitter links occurring Monday-Sunday at each hour to check for patterns or regularities that can assist with machine learning and better identify phishing links. All individuals online run into phishing links and it is continuously a major issue for businesses to prevent employees from gathering viruses and other malware type items from these links along with providing sensitive information about individuals bank accounts and personal information. Providing a basis to identify a day or time when a phishing attempt may come in, can assist many technology branches in preventing phishing scams from hitting employees company-wide.

The research was undertaken to find the occurrence of the phishing attacks in the United States rather than the validity, thus, to better inform the average individual of when an attack is more likely, allowing them to act in a more defensive way on certain days or times or possibly avoid Twitter altogether.

We are specifically looking at the frequency of phishing attempts throughout the days and times and will not get into the real-time ability to predict if a link is safe or not, as those topics have already been investigated as seen in the related work. This is purely to guide an individual about when to be cautious and the frequency of phishing at that time.

Taking the idea of looking at frequency and time, this paper will dive through the related work, then move onto the development of our program and all of the API tools we utilized to gather the Twitter posts, identify phishing attempts, and handle the data analytics. From there, we will discuss the findings and identify a few key aspects of Twitter phishing attacks and identify a correlation that is occurring in the data.

Related Work

Analyzing Social and Stylometric Features to Identify Spear phishing Emails[2]: Focusing on spear phishing (attacking a specific group or individual), the authors dove into LinkedIn and looked at emails to determine if there was a mindset or tactic used to target an individual or group of individuals. The dataset consisted of nearly 5 thousand targeted attack to roughly 2,500 victims based on information publicly available from their LinkedIn profiles. Their machine learning algorithm applied to identify spear phishing emails cam in at a 97.79% accuracy with using a combination of social features and stylometric features. The results showed a slightly better accuracy without social features (98.28%) and this provided the conclusion that the social features from LinkedIn do not help in identifying spear phishing emails.

PhishAri: Automatic Realtime Phishing Detection on Twitter[3]: This thesis identifies phishing attempts in real-time and makes predictions based off their analysis. This general idea has been done before in many papers, but the standoff this thesis holds is looking at creating a browser extension, through Google Chrome, to help individuals determine the risk of a link. The Twitter real-time phishing detection looks at the characteristics such as, length, hashtags, and mentions in the post. PhishAri also uses WHOIS, an online URL determiner that finds all of the information connected to domains and identifies key aspects that can assist in determining if phishing is being used. The tweet steam catches the tweet, then sends it to a blacklist lookup, using PhishTank and Google Safe Browsing API. The results are then identified as "Safe Tweets" or "Phishing Tweets" and that information is sent back to the user. The WHOIS system is a query and response that gives information about ownership, creation date, and updates. Identifying the changes to a website and matching the time to the tweet showed if something was made for a phishing attempt.

Real-Time Detection of Phishing Tweets[4]: This paper reviews the ability to identify phishing by using the tweet id and specific keywords to run a URL check and determine the validity of the actual tweet. This system uses a machine learning algorithm known as random forest classification to determine a basis of what might be a phishing attempt. They also used WHOIS to determine what might standout as something to look deeper into to find a phishing correlation. Using WHOIS, they followed a similar path to the PhishAri paper in using the data retrieved to check the age of a domain compared to how long ago the tweet was. Also how long was taken between the creation of the domain and the Twitter account. Their "Web Framework" takes roughly 0.501 seconds to detect phishing tweets with an accuracy of 94.56%.

Methodology

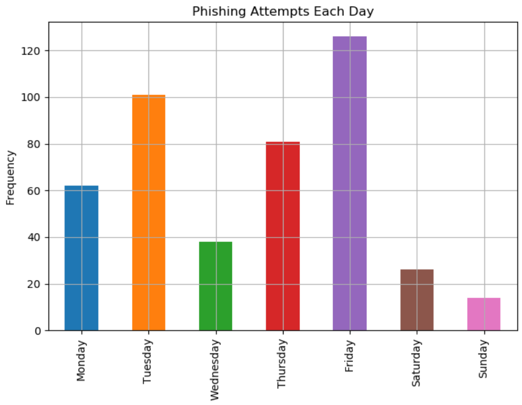
Utilizing Tweepy API in Python to first collect the data, we were able to gather tweets posted between July 2nd and July 15th. Gathering our own data was a choice we made due to wanting clean, un-edited data and a large enough data source that our findings would carry some weight. The total number of tweets collected was 8,044,555 and these were then saved in a JSON format and were searched for the URL and Date/Time to give each a unique identifier. The URLs were then tested with ScrapeBox, a program that structures Google Safe Browsing HTTP Requests to allow for a maximum usage (500 URLs per request). This was key, as without it, Google would shut down the number of requests a user could send as it believed the user to be spamming the server. After successfully running through all the URLs on Google Safe Browsing, it was discovered that 448 instances of phishing were captured. These phishing links, as defined by Google, are "attack[s that] trick users into performing an action that they normally would not if they knew the true identity of the attacker"[5]. Thus, the Safe Browsing API is designed to identify who is lying about their identity and provide the truth to the user. This was what gave us an idea as to the total phishing instances and was used for determining the significance in phishing tweets. Using Pandas and Scipy (stats) correlations and statistical significance was tested with the following results. The P-value for Day of the Week of when phishing attempts occur was statistically significant with an alpha of 0.05 > 0.04467 (P-Value). This was proof of a significance, but the Multiple R value of 0.4 proves a 40% relationship of the day of the week and the amount of phishing attempts that day. Thus, more testing would be required to prove a stronger relationship, but based on just over 8 million tweets, the day of the week has a significant affect on how many phishing tweets could occur. There was also a holiday of July 4th, Independence Day in the United States that was captured and proved to sway the results some as there were only 10 phishing attempts on the holiday while the next Wednesday provided 28. This decrease falls in-line with a hypothesis, that was proven with this data, that the weekend produces less scams while the work week provides some of the most phishing attempts. This can be seen below in the Figure A. Figure B looked at the time of day for the tweet but provided no direct correlation to the number of phishing attempts, meaning the day of the week has an impact to the phishing tweets, but the time of the day, does not.

Tweepy API in Python to gather the Twitter posts.

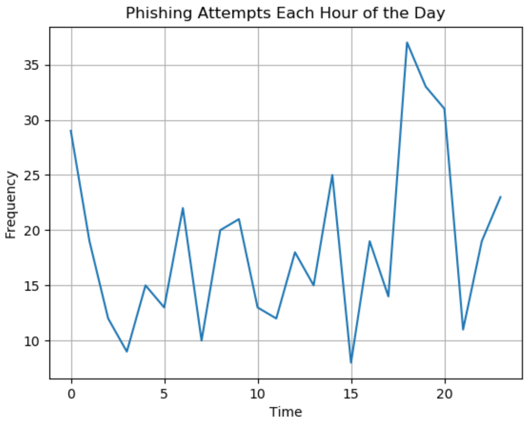
```
import os
import time
import tweepy
import sys
from datetime import datetime

from tweepy import OAuthHandler
from tweepy import Stream
from tweepy import StreamListener

class StreamListener(StreamListener):

    def on_data(self, data):

        try:
            with open("Output_1.json", 'a') as f:
                f.write(data)
                print("Got data")
                f.write('\n')
                return True
        except BaseException as e:
            print("Error on_data: %s" % str(e))
        return True

    def on_error(self, status):
        print(status)
        return True

def authenticate():
    """Use credentials to authenticate user'''
    consumerkey = '$$$$$$$$$$$$$$$$$$$$$$$$$$$$'
    consumersecret = '$$$$$$$$$$$$$$$$$$$$$$$$$$$$'
    accesstoken = ‘$$$$$$$$$$$$$$$$$$$$$$$$$$$$'
    accesssecret = '$$$$$$$$$$$$$$$$$$$$$$$$$$$$'
    auth = OAuthHandler(consumerkey, consumersecret)
    auth.set_access_token(accesstoken, accesssecret)
    api = tweepy.API(auth)
    return auth, api
```

GoogleSafeBrowsing API utilization to identify Phisphing attemps.

```
import pandas as pd
import os
import safebrowsing
import numpy
import numbers

URLList = []

os.chdir('C:…)
f = open("PythonOutput.txt", "a+")

for root, dirs, files in os.walk(‘E:…’):
    x = 0
    for name in files:
        x = 0
        data = pd.read_json(name, lines=True)
        data.shape
        df = pd.DataFrame.from_dict(data, orient='columns')
        source_list = df['source'].tolist()
        final_list = source_list
        y = 0
        while y < len(source_list):
            if type(source_list[y]) == float:
                y += 1
            if type(source_list[y]) == str:
                final_list[y] = source_list[y].replace('href="', '').replace('"', '').split()
                final_list[y] = source_list[y][1]
                y += 1
        del df
        x = 0
        apikey = '$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$'
        sb = safebrowsing.LookupAPI(apikey)
        print ("Starting Google API")
        while x < 1: #len(final_list):
            resp = sb.threat_matches_find(final_list[x])
            print (resp)
            if (len(resp) > 0):
                if (isinstance(final_list[x], float)):
                    print("")
                else:
                    f.write('{0} {1} {2}\n'.format(x, final_list[x], resp))
                    URLList.append(final_list[x])
            x = x + 1
```



A.



B.

Conclusion

Our original hypothesis was accurate that the day of the week would be an impact as to the number and frequency of phishing attempts. Figure A shows that Tuesday and Friday are the largest days for phishing attempts, and it was consistent among both weeks. While Sunday was the lowest overall output, thus from a visual perspective, one can identify that the day does matter, but as explained in the methodology, the statistical significance of the data shows that this statistical model is accurate at least 40% of the time and doubling our sample size would most likely continue to support the claim. Twitter users should be cautious when it comes to seeing links on Tuesday and Friday, as those are the days most likely their information is being stolen. Keeping an extra eye out and staying focused on Twitter links is important for user's protection.

[1] https://www.wombatsecurity.com/state-of-the-phish
[2] P. Dwan, A. Kashyap, P. Kumaraguru, "Analyzing Social and Stylometric Features to Identify Spear phishing Emails", Cybersecurity Education and Research Centre, 2014 (https://arxiv.org/pdf/1406.3692.pdf)
[3] A. Aggarwal, A. Rajadesingan, P. Kumaragugu, "PhishAri: Automatic Realtime Phishing Detection on Twitter", Arizona State University, 2013 (https://arxiv.org/pdf/1301.6899.pdf)
[4] N. Sharma, N. Sharma, V. Tiwari, S. Chahar, S. Masheshwari, "Real-Time Detection of Phishing Tweets", 2014 (https://airccj.org/CSCP/vol4/csit42520.pdf)
[5] Google, "Google Safe Browsing", 2008 (https://safebrowsing.google.com/)