Email Spoofing Detection Using Machine Learning

Janhavi Madiwale[1] Swati Saundale[2] Pooja Mehere[3]
[1,2,3]Student
[1,2,3]Department of Computer Science and Engineering
[1,2,3]Sipna College of Engineering & Technology, Amravati, India

Abstract — Email spoofing is a form of caricature where a scammer creates an email message with a forged sender address in hopes of deceiving the acquirer into thinking the email arise from someone other than the actualized root. Scammers will use email spoofing to help camouflage themselves as a executive program, academician, or financial organization to trick users into activity some type of action. Scammers use this acting of insincere because they know a person is more likely to absorb with the content of the email if they are familiar with who sent the message. We planned an online email spoofing detection application in which we will develop a machine learning model which will be able to classify received email into any one of the two collection spoofed or median. We planned mind tree algorithm to classify inward email.

Keywords: Spoofing, Phishing, Spamming, Domain Forgery, Malware, Cyberattack, Fraudulent

I. Introduction

Email is omnipresent in our company, and it is an primary part of regular abstraction, in particular in the workplace where it is static the most common form of communication but also in every online experience where an account is required. As thoroughbred in (Radiate Group, 2019), in 2019, the total number of enterprise and user emails sent and received per day will exceed 293 billion and is prediction to grow over 347 billion by the end of 2023. Scorn the benefits provided by snail mail communication, it has also create new impostor possibility which can subject the end-user clubby information to stern legal document and privacy threats. In new years the share of unasked email sent incline to steal private information or harm the acquirer device is increasing. Basing on the Spam and phishing report published by Kaspersky Lab 1, the average percentage of spam email in world-wide mail traffic in 2018 and Q1 2019 are comprised between 50% and 60%. The most general spam attacks are scam emails where the cattish user tries through assurance tricks to deceive the person into stealth personal information. One of the forms of scam attack is diagrammatic by the spear phishing, in which the attacker is intended to steal sensitive information from a specific victim often formation the email header so that the message come out to have originated from someone or somewhere other than the actual source. This type of attack can achieve a high degree of occurrent because people are more inclined to open an email when they think a true source has sent it. The nature of the original Simple Mail Transfer Prescript (SMTP) used in electronic mail transmittance (Hoffman, 2002), does not provide an mark mechanism that can verify information about the origin of email pass on. A large number of legal protocols have been proposed to solve the difficulty such as ESMTP (Myers, 1999), SPF (Wong and Schlitt, 2006), DKIM (Allman et al., 2007), DMARC (Kucherawy and Zwicky, 2015). Nevertheless, the freehand SMTP is still more widely used. Therefore, a system of email initiation proof based on the writing style synthesis can be a valid disjunctive to support end-user to determine, with a certain authority degree, whether the email sender is who declares to be. In this paper, we focused on a specific email scam (spear phishing) based on email spoofing attack and we enforced a new support end-user system able to detect such attack analyzing the email content. In the paper is given the description of the email scam attack and, as countermeasures, two different scenarios based on email composition verification are presented: (i) a detection on the server side which can exploits the word-painting of the overall writing style of a sender, and (ii) a detection on the client side that minimum the delineation of a sender only to a specific receiver (End to End writing style). We thoughtful solutions based on machine learning systems research both standard machine learning categorize based on well-known text stylometric features and deep acquisition classifiers defined by an machine-controlled features descent. To reach the best accuracy has been research different preparation conceptualisation, which consider different subset of the dataset used. The best model has been engaged in the realization of a warranted email client utilization for Android as assistant to support the end-user in the detection of shady emails. The paper is formed as follows. In Section 2 the background concepts related to the spear phishing attack and an introduction of the authorship problem are explained. In Section 3, the proposed founding approach is unrepentant, and the details of the possibility applied in two manageable premise, are provided. In Section 4, the feature-based and the deep learning classifiers used and enforced are detailed. Section 5 provides a description of the dataset used and the experiments conducted. In Section 6 the results obtained are bestowed and discussed. Section 7 describes a broad of initiation works analyzed in the profession. In Section 8, the terminal remark and the accomplishable early work are discussed.

II. Methodology

The goal of email spoofing is to trick users into basic cognitive process the email is from someone they know or can trust—in most cases, a co-worker, marketer or brand. Accomplishment that trust, the attacker asks the receiver to divulge substance or take some other activity. As an example of email spoofing, an assailant might create an email that sensing like it comes from PayPal. The communicate tells the user that their revelation will be supported if they don't click a link, manifest into the situation and change the account's password. If the user is successfully gimmick and types in credentials, the attacker now has credentials to manifest into the fair game user's PayPal account, potentially stealing money from the user. More complex attacks target commercial enterprise worker and use social engineering and online stupidity to trick a quarry user into sending large whole number to an attacker's bank account.

In this project we proposed email spoofing detection system using decision tree algorithm. We have implemented the model for outlook emails. We have developed java web application in which user will register by his outlook email id. Our model will fetch inbox messages from outlook and classify them one by one using decision tree algorithm.
We are classifying the mails in three categories

- Normal
- Spoofing
- Phishing

To classify incoming mails we have generated dataset using following features

- Authentication
  If sender name, alias name and reply_to attributes are different, value of authentication will be 0 otherwise 1
- links_present
  if links present in email body, its value is 1 otherwise 0
- scripts_present
  if scripts present in email body, its value is 1 otherwise 0
- iframe
  if iframe tag present in email body, its value is 1 otherwise 0
- on_mouseover
  if on_mouseover event present in email body, its value is 1 otherwise 0
- suspiciouskeyw
  if body contains suspicious keywords greater than 20 percent, its value is 1 otherwise 0
- double_slash_redirecting
  if body contains form or redirection links to send the content to other site, its value is 1 otherwise 0

Following is a dataset

authentication	links_present	scripts_present	iframe	on_mouseover	suspiciouskeyw	double_slash_redirecting	label
0	1	1	0	0	1	0	1
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1
0	0	0	0	0	1	1	1
1	0	0	0	0	1	1	2
1	1	1	1	1	1	1	2
1	1	1	1	0	1	0	2
0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	1

Python code for decision tree is as given below:

```
def decisiontree(lst):
col_names =
['authentication','links_present','scripts_present','iframe','on_mouseover','suspiciouskeyw','double_slash_redirecting','label']
# load dataset
pima = pd.read_csv("dataset.csv", header=0, names=col_names)
#split dataset in features and target variable
feature_cols =
['authentication','links_present','scripts_present','iframe','on_mouseover','suspiciouskeyw','double_slash_redirecting']
X = pima[feature_cols] # Features
y = pima.label # Target variable
# Split dataset into training set and test set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1) # 70% training and 30% test
# Create Decision Tree classifer object
clf = DecisionTreeClassifier()

# Train Decision Tree Classifer
clf = clf.fit(X_train,y_train)
#Z_test=[[0,0,0,1,1,0,1,1,1,0,0,0,0,0,0,1,1]]
Z_test=  [lst]
#Predict the response for test dataset
print(Z_test)

y_pred = clf.predict(Z_test)
print(y_pred)
return y_pred
```



```
# Model Accuracy, how often is the classifier correct?
#print("Accuracy:",metrics.accuracy_score(y_test, y_pred))
def decisiontree1(lst):
col_names = ['floorNo','cityType','housecond','furniture','label']
# load dataset
pima = pd.read_csv("dataset1.csv", header=0, names=col_names)
#split dataset in features and target variable
feature_cols = ['floorNo','cityType','housecond','furniture']
X = pima[feature_cols] # Features
y = pima.label # Target variable
# Split dataset into training set and test set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1) # 70% training and 30% test
# Create Decision Tree classifer object
clf = DecisionTreeClassifier()

# Train Decision Tree Classifer
clf = clf.fit(X_train,y_train)
#Z_test=[[0,0,0,1,1,0,1,1,1,0,0,0,0,0,0,1,1]]
Z_test=  [lst]
#Predict the response for test dataset
print(Z_test)

y_pred = clf.predict(Z_test)
print(y_pred)
return y_pred
```

III. Result

The model is used to information your paper and style the text. All margins, column widths, line spaces, and text fonts are appointed; please do not alter them. You may note specialty. For example, the head boundary in this template measures pro rata more than is customary. This measurement and others are deliberate, using specifications that evaluate your paper as one part of the entire legal proceeding, and not as an independent document. Please do not reorganize any of the contemporary organisation.

IV. Conclusion

In the world of cybersecurity, spoofed email protection is now a essential problem. Early and effective detection of email spoofing is now essential due to the function rise in the occurrent of spoofed email over the past respective years. As a result, many methods were proposed in this tract to address the trouble. With progression in internet discipline and the succeeding modification in online mortal fight, security concerns have grown progressively serious. In this study, multiple email spoofing catching methods are reviewed. It was observed from examination these proficiency that Machine Learning using Decision Tree Algorithm is the most efficient and high-fidelity technique for email spoofing detection. Using this method, the possibility of detecting spoofed emails angularity, the accuracy of detection of spoofed emails is high and thus, users are capable of efficient email spoofing detection.

References

[1] T. Verma, N. S. J. I. J. o. I. T. Gill, and E. Engineering, "Email Spams via Text Mining using Machine Learning Techniques," 9, no. 4, pp. 2535-2539, (2020).
[2] N. Saidani, K. Adi, M. S. J. C. Allili, and Security, "A semantic-based classification approach for an enhanced spam detection," 94, p. 101716, (2020).
[3] H. Taylor, "Making Mass-Spamming Illegal Rises," Harris Interactive (2011).
[4] P. Heymann, et al., "Fighting spam on social web sites: A survey of approaches and future challenges," Internet Computing, IEEE, 11, pp. 36-45, (2007).
[5] Clearbridge. What is the global cost of spam? Available: http://www.mailshine.com/2011/06/whats-the-globalcost-of-spam/(2011).
[6] M. Fossi, et al., "Symantec global internet security threat report," White Paper, Symantec Enterprise Security, 1, (2013).
[7] X. Guo and Z. Xia, "Fighting spam," University of California Berkeley, (2012).
[8] S. Youn and D. McLeod, "A comparative study for email classification," in Advances and Innovations in Systems, Computing Sciences and Software Engineering, ed: Springer, pp. 387-391 (2007).
[9] I. Firdausi, et al., "Analysis of Machine learning Techniques Used in BehaviorBased Malware Detection," in Advances in Computing, Control and

Telecommunication Technologies (ACT), 2010 Second International Conference on, pp. 201-203, (2010).
[10] S. T. Maller, "Email filtering methods and systems," ed: Google Patents, (2006).
[11] D. Cook, et al., "Catching spam before it arrives: domain specific dynamic blacklists," in Proceedings of the 2006 Australasian workshops on Grid computing and e-research- 54, pp. 193-202, (2006).
[12] P. Warkhede, et al., "Fast packet compartmentalisation for two-dimensional conflict-free filters," in INFOCOM. Twentieth Yearly Joint Conference of the IEEE Computer and Communications Social club. Proceedings. IEEE, 2001, pp. 1434-1443 (2001).
[13] P. O. Boykin and V. Roychowdhury, "Personal email networks: An effective antispam tool," arXiv preprint cond-mat/0402143, (2004).
[14] A. W. Moore and D. Zuev, "Internet text classification using theorem analysis skillfulness," in ACM SIGMETRICS Performance Evaluation Review, pp. 50-60, (2010).
[15] N. J. Kawale and S. Y. Sait, "A Review on Various Techniques for Spam Detection," in 2021 World-wide Conference on Artificial Intelligence and Smart Systems (ICAIS), pp. 1771-1775: IEEE, (2021).
[16] [16] O. El Kouari, H. Benaboud, and S. Lazaar, "Using machine learning to deal with Phishing and Spam Detection: An overview," in Proceedings of the 3rd International Conference on Networking, Information Systems & Security, pp. 1-7, (2020).
[17] M. Sahami, et al., "A Bayesian approach to filtering junk e-mail," in Learning for Text Categorization: Papers from the 2008 workshop, pp. 98-105, (2008).
[18] B. Cui, et al., "On effective e-mail classification via neural networks," in Database and Expert Systems Applications, pp. 85-94, (2005).
[19] G. Leroy and T. C. Rindflesch, "Using symbolic knowledge in the UMLS to disambiguate words in small datasets with a naive Bayes classifier," Medinfo, 11, pp. 381-385, (2004).
[20] E. Byvatov, et al., "Comparison of support vector machine and artificial neural network systems for drug/nondrug classification," Journal of Chemical Information and Computer Sciences, 43, pp. 1882-1889, (2009).
[21] A. Lorenz, et al., "Comparison of different neuro-fuzzy classification systems for the detection of prostate cancer in ultrasonic images," in Ultrasonics Symposium, 2005. Proceedings., 2005 IEEE, , pp. 1201-1204 (2005).
[22] N. Widiastuti, "Convolution neural network for text mining and natural language processing," in IOP Conference Series: Materials Science and Engineering, 662, no. 5, p. 052010: IOP Publishing, (2019).

