www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 5 Issue 12 Dec. 2017, Page No. 19832-19841
Index Copernicus Value (2015): 58.10, DOI: 10.18535/ijecs/v5i12.84

Social Engineering Attacks Detection Techniques: Survey Study

Mohamed H. Haggag, Ensaf H. Mohammed, Mariam S. El-Rahmany

Department of Computer Science
Helwan University,
Cairo, Egypt
mohamed.haggag@fci.helwan.edu.eg
Department of Computer Science
Helwan University,
Cairo, Egypt
ensaf_hussein@fci.helwan.edu.eg
Department of Computer Science
Helwan University,
Cairo, Egypt
mariam.sk10@gmail.com

Abstract—Social Engineering Attacks becomes a real threat especially with the emerging of the Online Social Networks (OSN) which provides the attacker with personal information about the victim that facilitates the attack. It becomes more common threat against enterprises and SMBs (Small and Medium Business) like threaten its financial and trust work. The E-mails and OSNs such as Twitter and Facebook are the most common environments used in this kind of attacks. In this paper, we have reviewed the existing techniques for detecting the Social Engineering Attacks mainly on e-mails and OSNs. Mostly focus on the Natural Language Processing (NLP) and machine learning techniques. A comparative study and evaluation of these approaches is presented. This provides an understanding of the problem, its current solution alternatives, and the anticipated future research directions

I. *INTRODUCTION*

In information security context, social engineering attack refers to psychological manipulation of people into performing actions or disclosing confidential information [1]. In other words, it is an attack vector that relies heavily on human interaction and often involves tricking people into breaking normal security procedures[1].

Cyber attackers target the weakest part of a security system which are people that are often more vulnerable than a secure computer. So, it involves some form of psychological manipulation leading the victim to reveal sensitive information, click a malicious link, or open a malicious file. Because social engineering involves a human element, preventing these attacks can be tricky for enterprises. It is a form of confidence scam. A study by Verizon of security breaches in 2013 has shown that 29% of all security breaches involve social engineering to extract information for use primarily for phishing, bribery, and extortion [25]. These attacks have many ways primarily via email but also in-person, via phone, SMS, websites.
In Fig 1 we present the main categories of social engineering attacks:

1http://searchsecurity.techtarget.com/definition/social-engineering.

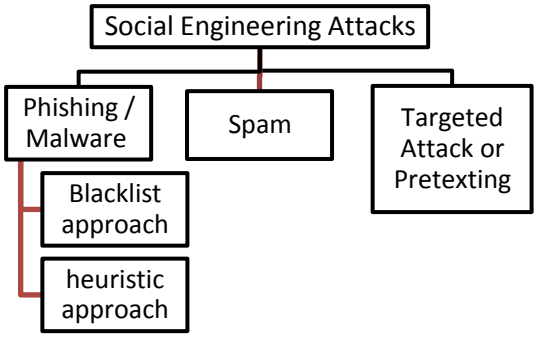

Fig. 1. Categories of Social Engineering Attacks

These attacks can be done through e-mails or Online Social Networks (OSNs). Such as Facebook which considered as the second most visited site on the internet, and it has high growth rate 3% per week. One of the important services in Facebook is finding new friends. This service is a way for the attacker to build a high level trusted relationship with the victim and start his attack by sending a message. This can happen due to the large amount of information available on these social networks.
The common pattern associated with a social engineering attack is the cycle in Fig 2. Where the attacker establishes contact with the target, and sends some initial request to start the attack.

DOI: 10.18535/ijecs/v5i12.84

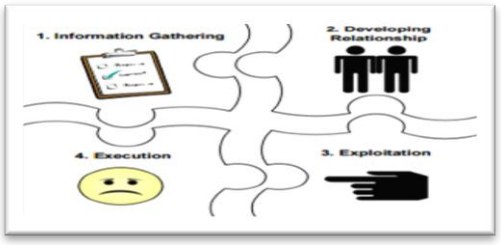

Fig. 2. Common pattern associated with a Social Engineering attack [20]

In this survey, we present the current models and applications for social engineering attacks detection. We focus on detecting social engineering attacks at emails and messages level on Online Social Networks (OSN) such as Twitter and Facebook.

This survey begins by defining the Social Engineering attacks and presenting the main three categories which are: Phishing and Malware, Spam campaign, and targeted attacks or pretexting.

In section II, we define related works that present the different techniques for phishing attacks detection. In section III, we present the different techniques for spam campaign detection. In section IV, we present the different techniques for targeted attacks detection. The conclusion is drawn in section V.

I. PHISHING / MALWARE ATTACKS

Phishing is the most common type used for attacking victims. It is when a malicious party sends a fraudulent email or message masked as a legitimate. The message goal is to trick the recipient into sharing personal or financial information or clicking on a link that installs malware or leads to websites designed to impersonate real systems to capture sensitive data. For example, a message might come from a bank or other well-known institution with the need to "verify" your login information. Fig. 3 shows an example of Life cycle of phishing email [27].

Fig. 3. The life cycle of phishing E-mail [27]

Spear phishing is like phishing, but tailored for a specific individual or organization.

Malware Attacks: malicious attachment that gathers e-mail addresses and spreads by copying itself. Also, it can be sent to a victim's "friends list" in Facebook for example with links to infected servers.

Phishing attacks detection techniques/approaches can be categorized as heuristic and blacklists approaches. Where heuristic approaches use HTML or content signatures to identify phish.

Since the important motive behind phishing emails is tricking the users into disclosing confidential information. This can be achieved using the following contexts [7]: (i) invoking a sense of false urgency – a user asked to provide his credentials in order to validate his account during limited time in a masqueraded website. (ii) Invoking a sense of threat – a user asked to disclose his credentials to avoid account cancellation. (iii) Invoking a sense of concern – the user asked to change his password as false security concern in fake website similar to the official one. So, they can capture the correct password. (iv) Invoking a sense of opportunity/reward – the user asked to provide his information to transfer money to his account.

A. *Phishing or Malware detection approaches*

Yue and H. Wang [2] propose client-side anti-phishing tool named BogusBiter. They use an offensive technique which is injecting the phishing website with large number of bogus credentials. For example, users who ignore the warning message, this tool will generate bogus credentials to hide the real credential. The defensive technique is when a phisher spend time trying to filter the bogus credentials to find the real one. They use simple substitution rule to meet both the correlation and indiscernibility (inability to be observed) requirements. The rule is designed to have S <= 10. (With BogusBiter equipped at each web browser, the real-to-all ratio will be determined by two factors. The first is the set size S, i.e., the number of credentials submitted by BogusBiter for each phishing site visit. The second is the cheat-to-click ratio, which is the ratio between the number of victims who reveal their credentials and the total number of users who visit the phishing site.)

P. Prakash and et al. [3] provide anti-phishing tool called "PhishNet" with two components: 1- URL prediction component that generates new URLs as child from predefined phishing URLs as parent by using various heuristics. Then test the new URLs if they are malicious. 2- Approximate URL matching component that finds if new URL match with predefined blacklist.

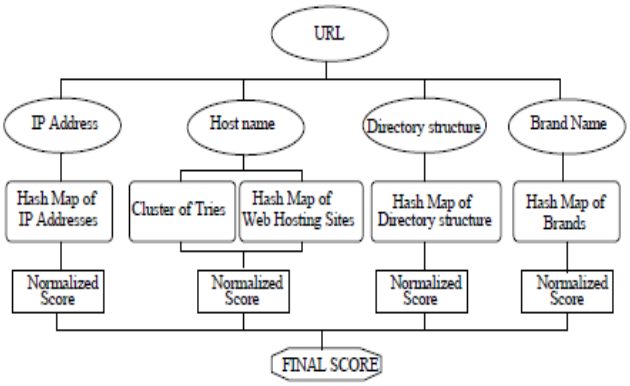

Fig. 4. Computing the score of a new URL in PhishNet. If the score is above the threshold, they flag it as phishing site [3].

Y. Cao et al. [4] provide anti-phishing approach named Automated Individual white-list (AIWL). This approach aims to prepare a list of legitimate websites or trusted Login User Interface (LUI). This list contains features describing trusted LUI where the user submitted his credentials and

successfully logged in for sufficient amount of times. The features vector of LUI is compared with features vector in the white list. If all features of current page not matched with all features of any LUI in white list, so this page is assumed to be untrusted and warnings are shown to end user. They use Naïve Bayesian classifier to construct a model to generate probabilities for each successful login attempts. This probability compared to predefined threshold and if it exceeds the threshold, then the login considered as successful login.

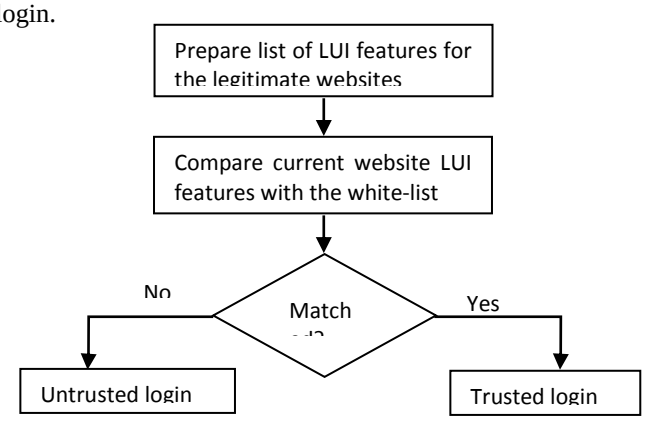

Fig. 5. Automated Individual white-list (AIWL) approach.

Jain et al. [5] implemented a prototype for web browser using C#.NET which can be used as an agent that processes each email for phishing attacks. While most phishing emails are sent asking the user to click on a hyperlink, they categorized hyperlinks to: hyperlink doesn't link to the apparent location, hyperlink contains DNS name or IP address, hyperlink has similar DNS name to the DNS that phishers trying to attack, encoded/long hyperlinks or a hyperlink asking sensitive data. Their method includes extracting the features of the hyperlinks like visible, invisible and un-matching that have numerical values. If they found the value for "invisible_links" and "unmatching_urls" to be nonzero then they consider the given email as a possible phishing attack. In case an attack is detected the user is notified of the forged email suspicion and advised to delete the email. The advantage in their method is the categorization of hyperlinks in order to ease the detection process

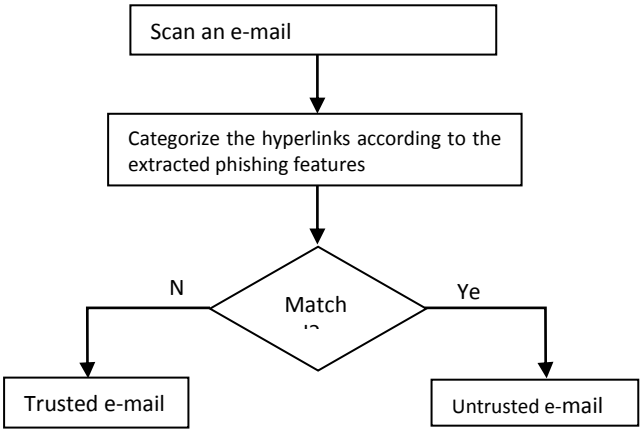

Fig. 6. A proposed approach for web browser by Jain et al.

Kirda et al. [6] present an Anti-Phishing tool called "Anti-Phish". This tool works as web browser extension. Depends on user request, it captures, encrypts and save the user's sensitive data and tracks where these data are sent.

Chandrasekaran et al. [7] present a technique for phishing email detection by analyzing unique structural features of an email. Their model captures the characteristics of phishing emails such as sense of threat, concern, or urgency. These features used with One-Class SVM (Support Vector Machine) to classify phishing emails.

They start by selecting accurate features and eliminate the weak features (noise classification) using heuristic search based algorithms and apply simulated annealing [8] to locate the global optimum in large search space. Then they rank the selected features using IG (Information Gain) concept as feature ranking metric. The SVM is widely used in text classification applications and especially security field like spam detection. The extracted features are weighted according to their relevance to the dataset that's consists of 400 e-mail includes 200 phishing emails. The classification of features is done using SVMlight [9] which minimizes the generalization error. The classification depends on set of invariant characteristics of emails like language, layout and structure. This can help in capturing the different contexts (and their associated keywords) of phishing emails. They used phishing attacks vector: URL and host name obfuscation attacks, embedded e-mail attachment (embedded HTML forms), browser vulnerabilities, Cross-site scripting (XSS), and Session hijacking attacks.

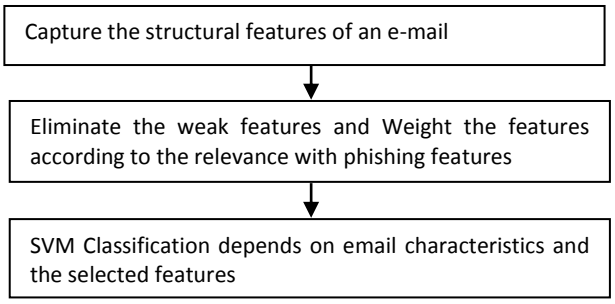

Fig. 7. Proposed phishing e-mail detection approach by Chandrasekaran et al.

Ying et al. [10] present a phishing detector as a plug-in for a web browser that is consists of: Identity extractor and page classifier. (i) The identity extractor, the website considered as set of words and the identity defined by objects or properties like organization name in the webpage. (ii) The page classifier depends on structural features that are identity related W3C Dom objects in a web page like URI of a link or structural features that are HTTP transactions like a domain name. Then a SVM classifier is used to classify a website as legitimate or phishing website. The SVM is trained by large dataset of identities and features from both phishing and legitimate websites. Example on W3C Dom objects and properties: URI of links, title of elements, action of a form, text of body.

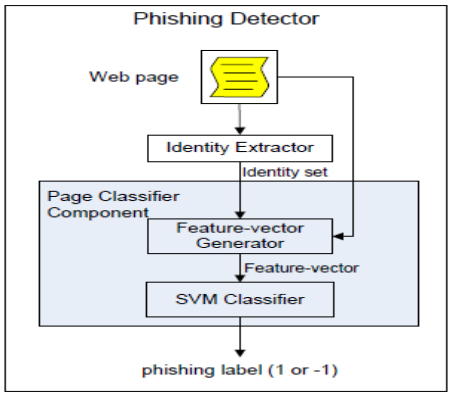

Fig. 8. The architecture of phishing detector [10].

Ian Fette et al. [11] present an approach to detect phishing e-mails called "PILFER". They extract structural features from the emails like ip-based URLs, Age of linked-to domain names (by performing WHOIS query to get the registration date of the domain then compare it with the sent date of the e-mail), "Here" links to non-modal domain, HTML emails, number of links, number of domains, number of dots, Non-matching URLs, and contains javascript.

These features can be used on both e-mail and webpage levels but they focus on e-mail. Another set of features can be used on the browser such as site in browser history, redirected site, and tf-idf (term frequency-inverse document frequency)

Then they use all information from the email itself and the extracted features as input to machine learning classifier named Random forest classifier that creates a number of decision trees. For testing they used a "the ham corpora" dataset from the SpamAssassin project and the publicly available phishing corpus. Their approach achieves 0.0013 as false positive rate. Also, they indicate to sender authentication technologies that can improve the classification process like Sender ID Framework (SIDF) [12] and Domain Keys [13].

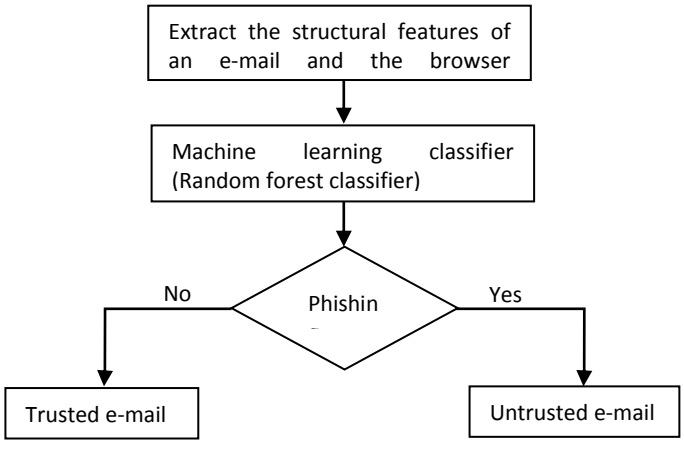

Fig. 9. Phishing e-mail detector "PILFER"

Ponnurangam [14] defines three types of the security attacks which are physical, syntactic and semantic. Ponnurangam focused on the semantic attacks that depend on user interaction and proposed anti-phishing training system for training of military personnel on new Future Combat System (FCS). Users find the training e-mails like phishing e-mails on their inbox. These phishing e-mails urging users to click on a link or entering their credentials. If users didn't discover that it is a phishing e-mail, an alert message will show up to explain that it is an attack and offers some tips a user can follow to protect himself. This immediate feedback during their working duties will enhance learning and influence the user behavior. So it is a continuous training include the emerging types of attacks also, it is a practical training that allow users to interact with real phishing cases. The training applies instructional design, conceptual knowledge and procedural knowledge using the following principles: learning by doing (i.e. training messages appears only when a user clicking on phishing e-mail), immediate feedback, conceptual procedure (i.e. they present phishing definition as conceptual knowledge and ways to protect as procedural knowledge), contiguity, personalization and story based agent environment. They use Signal Detection Theory (SDT) where the sensitivity and the Criterion. The sensitivity to measure the ability of the user to distinguish between the legitimate and phishing websites. The criterion or user's decision tendency to measure if a user is cautious, neutral or liberal.

1. Spam campaign attacks

Spam Attacks: it is spam or spam campaigns that embed phishing advertisements in an email or in a post on facebook.

2. Spam detection approaches

Gao et al. [15] supposed that (i) spam wall posts on Facebook generated using templates, (ii) Posts of same template are similar and (iii) an account is "malicious" if it has made at least one malicious wall post. So, they group posts with "similar" textual description using probabilistic fingerprint. (iv) Same destination of the URL must come from same spam campaign. So, they group all posts with same destination + hidden URLs (i.e. www dot hack dot com).

Kandasamy et al. [16] determine the types of spammers on twitter as Phishers - Malware propagators – Marketers - Adult content propagators. They focus on the following features for each type: The spammers have certain characteristics, phishers may use more number of URLs, marketers may use more number of hash tags and so on. Number of # tags (marketers use many # tags to promote different things), Number of unique # tags (user promote a single product with unique # tag many times-particular topic), Number of URLs (more number of URLs used), Number of unique URLs (a phisher may use same URL many times).

Their method depends on NLP and Machine learning techniques. They use NLP to remove stop words, stemming only keywords extracted, compare Stemmed keywords with the set of identified spam words. If found, regarded as spam else use the machine learning approach. In this technique, they use test set and training set of form (a1, a2,…,an, L) a-> attribute , L -> label and they use naïve-bayes (probabilistic classifier) and SVM (support vector machine).

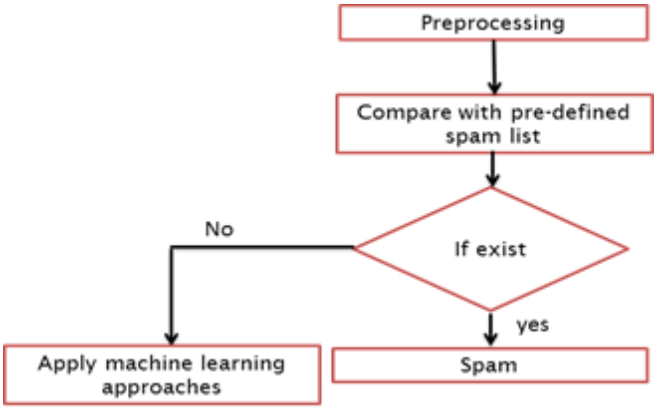

Fig. 10. An approach for spam classification on twitter

Fuad et al. [17] present a trainable spam detection system. This system uses fuzzy and inference engine. (1) Features extraction: They extracted the main email features and the spam features by extracting words from a corpus of spam emails [21] and legitimate emails.

Using tokenization, they prepare a list of all words and their frequencies. Then calculate the weight of each word that indicates how many times they appear in spam or legitimate mail. (2) fuzzy classification system: pass values of a fuzzy set to the fuzzification layer. The fuzzification stage determines the degree to which this input belongs to the respective fuzzy set. The fuzzy classification rules have the form of IF-Then rules. They compare the crisp output to a predefined threshold value and predict the message as Spam or Non-Spam.

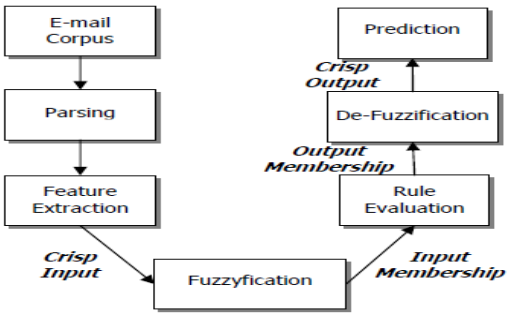

Fig. 11. Trainable spam detection system [17]

TARGETED ATTACKS OR PRETEXTING

Targeted attack or Pretexting: in this type the attackers focus on creating a good pretext, or a fabricated scenario, that they can use to gain their victims' sensitive and non-sensitive information. Also, manipulate their victims into performing an action that enables them to exploit the structural weaknesses of an organization or company. A good example of this would be an attacker who impersonates an external IT services auditor and manipulates a company's physical security staff into letting them into the building. They rely on building a false sense of trust with the victim. This requires the attacker to build a credible story[2].

B. Targeted attacks or Pretexting detection approaches

Bhakta et al [18] proposed semantic analysis of dialogs to detect social engineering attacks. Their method includes defining a topic for each line in the discussion text then comparing it with pre-defined topic blacklist (TBL).
This topic blacklist prepared using security policy document. They check for appropriateness for each Topic where a statement is inappropriate (if requests secure information or requests to perform a secure operation). Also, they analyze the values of features and correlations between feature values such as inclusion of a company logo at a website whose URL is not related to the company. Each topic is composed of 2 elements: action (an operation which the subject may wish to perform), resource (the resource inside the system has restricted access). This take a form generic topics ([action], [resource]): such as {"tell","social security number"} .

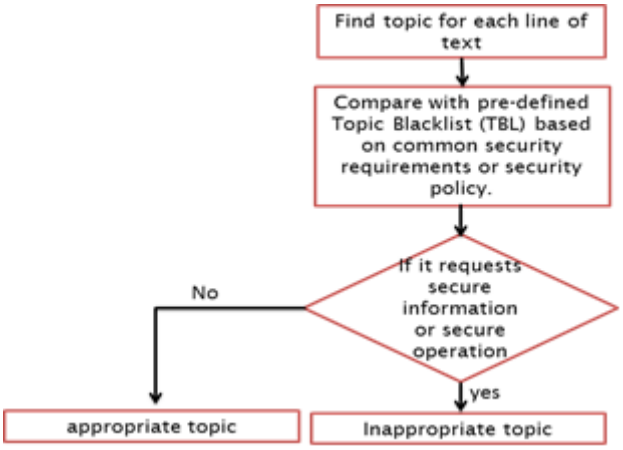

Fig. 12. Semantic analysis of dialog

Dewan et al. [19] focus on spear phishing attack as one of targeted social engineering attacks where it is more powerful than normal phishing, it focusses on contextual information about the victim. Online social media services can be an example for a website that gathers vital information about an individual. In this paper, they identify and characterize a true positive dataset of spear phishing, spam, and normal phishing emails from emails (Symantec's enterprise email scanning service) and LinkedIn profiles. Also, they use a model to spear phishing detection using social features extracted from LinkedIn as social network and stylometric features extracted from email subjects, bodies, and attachments. They found that using only social features extracted from LinkedIn are not powerful for detecting spear phishing. They use various machine learning algorithms on their dataset.
Yuki et al. [24] present an approach to detect social engineering attacks in dialog text. They use natural language processing techniques to detect questions and commands. Each sentence is parsed by Stanford parser and a parse tree is created. They search it for patterns which are indicative of questions and commands depending on the imperative and interrogative clauses syntactic forms. Then determine the topic of each sentence depending on the noun and the verb in the sentence. After that they compare the extracted topic with blacklist topics that are prepared manually from common security requirements or from a specific security policy. An example of a sentence expressing a soft imperative is, "you should shut down the router". They identifying a verb with a preceding modal verb and the pronoun "you" before that

DOI: 10.18535/ijecs/v5i12.84

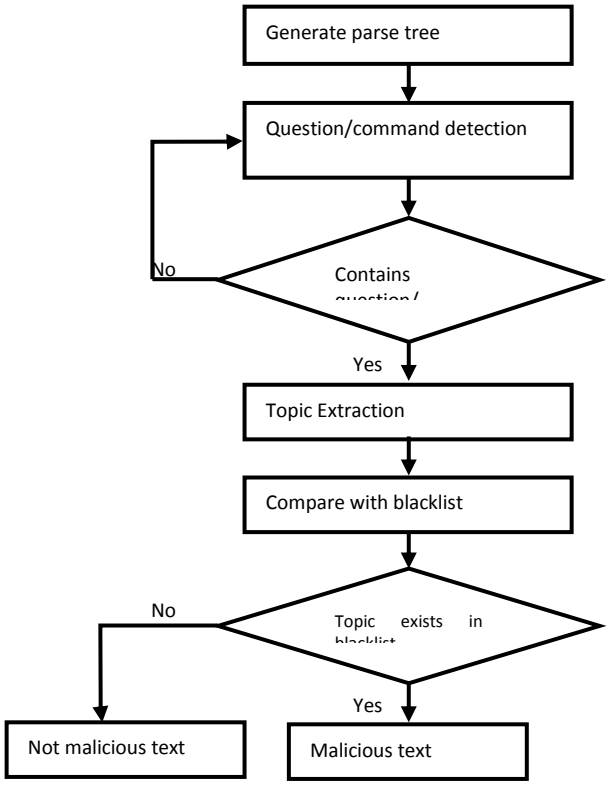

Fig. 13. Proposed spear phishing detection approach

APPENDIX

TABLE I

Summary of Social Engineering Attacks detection approaches

Author	**Attack type**	**Model/ Contribution**	**Weakness**	**Features**
C. Yue and H. Wang [2]	Phishing	BogusBiter (Client-side Anti-Phishing tool) With Offensive /defensive technique	The offensive technique consumes time and memory.	It integrates the phishing detection feature in a web browser like Firefox. Once a website is detected as phishing their tool starts. It depends on injecting the phishing website with large number of bogus credential as offensive technique.

P. Prakash, M. Kumar, R. R. Kompella, and M. Gupta [3]	Phishing	PhishNet tool	Needs continuous online updates for blacklist.	It uses predefined Blacklist (as parent) then predict new malicious URLs (as child) using some heuristics. When a URL is not exact matching a URL in blacklist, they use approximate matching algorithm and scoring to discover if a URL is a potential phishing site.
Y. Cao, W. Han, and Y. Le [4]	Phishing	Anti-phishing approach named (AIWL) Automated Individual White-list	White list is saved on user local machine so, it is controllable -Data can be lost – a trusted website could change some of its features values so, it will give wrong warning – time consuming – high false positive	It provides a white-list that contains the features vector of the trusted sites that the user is frequently uses. Features of the website compared with the features vector to determine if it is trusted website or not. They use the Naiive Bayesian classifier to train AIWL to identify the successful login process according to predefined threshold.
Jain, Aanchal, and Vineet Richariya [5]	Phishing	Web browser uses Features extraction	More feature consumes time and memory	It uses link related features and categorizes hyperlinks. It is user input-based and guarantees that sensitive information will not be transferred to a web site that is untrusted.
Kirda, Engin, and Christopher Kruegel [6]	Phishing	Anti-Phish (Mozilla Browser extension) Depends on user request	Used with only one browser	It uses Mozilla XML UI language (XUL) and JavaScript. It captures the user ID and asking him to enter master password which is used to decrypt the stored sensitive information. Then compare the inputs with the trusted stored information. If the website of this login info is not the stored trusted one, an alert will show up.
Chandrasekaran, Madhusudhanan, Krishnan Narayanan, and Shambhu Upadhyaya [7]	Phishing	Phishing e-mail detection	Not formal features – small dataset for test	It captures the e-mail structural features and eliminates the weak ones. Then it ranks the extracted features. A SVM classifier is used to classify the emails as phishing or legitimate.
Pan, Ying, and Xuhua Ding [10]	Phishing	Web Browser plug-in	High false positive rate compared to the other approaches	It extracts the identity (properties or objects) of a webpage like organization name then it uses the SVM classifier to classify the page according to the features of the webpage.
Ian Fette, Norman Sadeh, and Anthony Tomasic [11]	Phishing	Phishing E-mail detector (PILFER)	Sizeable number of phishing and ham emails was not well classified.	It extracts the structural features of the e-mail by performing WHOIS query, features of the e-mail itself, and the features of the browser like tf-idf. All extracted features are used as input to machine learning classifier (Random forest classifier).
Ponnurangam Kumaraguru [14]	Phishing	Anti-phishing training system (Phishguru)	It depends on user mentality and behavior	It sends training e-mails to the employees. This training system sends e-mails semantically urge the users to click on link or enter their credentials for example. It trains users to discover if an email is

				phishing or legitimate. If a user failed to discover the phishing e-mail, an alert message will show up with explanation of the attack and tips to protect himself. It measures the ability of the user to distinguish the phishing e-mail and measures the type of the user.
Gao, Hongyu, Jun Hu, Christo Wilson, Zhichun Li, Yan Chen, and Ben Y. Zhao. [15]	Social Spam campaigns	Facebook posts analysis using probabilistic fingerprint and Semantic analysis	It needs continuous blacklist update	It finds similar textual description using probabilistic fingerprint. And classify malicious URL according to set of well-known keywords (Blacklisting) then it finds the users that generate them.
Kandasamy, Kamalanathan, and Preethi Koroth [16]	Spam	Works on Twitter using Natural language processing and machine learning techniques	Consume time	It uses NLP techniques to remove stop words, stemming only keywords extracted. Then compare Stemmed keywords with the set of identified spam words. If found, regarded as spam else use the machine learning approach (Use naïve-bayes (probabilistic classifier) and SVM (support vector machine)).
Fuad, M. Muztaba, Debzani Deb, and M. Shahriar Hossain [17]	Spam	Spam detection using Fuzzy	Limited to only text posts	It extracts email features (and weights each of them) and spam features. It is not just depending on spam words or phrases but also check the frequency of each of them to be in legitimate or spam email text. Then it uses set of fuzzy classification rules to classify the email as spam or legitimate.
Bhakta, Ram, and Ian G. Harris [18]	Social engineering attacks	Semantic analysis for discussions	It doesn't semantically accurate. for example, if same words play different roles in 2 sentences but it considers them same.	It finds a topic for each line of the discussion text Then compares it with pre-defined topic blacklist (TBL) Which is prepared using security policy document.
Dewan, Prasun, Arti Kashyap, and Ponnurangam Kumaraguru [19]	Spear phishing as targeted social engineering attack	Works on E-mails using Features extraction and machine learning algorithms	They focus on collecting data from only one online social media which is LinkedIn.	It focuses on contextual information about the victim. It uses a true positive dataset of spear phishing, spam, and normal phishing emails. It uses a model with social features extracted from linkedIn and stylometric features extracted from email details. They use various machine learning algorithms on their dataset.
Sawa, Yuki, Ram Bhakta, Ian G. Harris, and Christopher Hadnagy [24]	Social engineering attacks	Semantic detection	Applied on text messages only.	It focuses on semantic detection. It uses the Stanford parser and the penn Treebank Tagset as parser's grammar to create a parse tree. It uses Tregex tool [26] to match patterns in the parse tree.

Conclusion

In this survey, we focus on detection approaches for detecting malicious URLs and malicious messages that may appear in the e-mails or the Online Social Networks (OSN). Approaches given in the literature still have much limitation on accuracy or performance, especially with targeted/ pretexting attack. Most approaches are based on blacklists where it contains a list of all malicious words. Also, most approaches extract the characteristics and the features (structural, browser, etc) then they use them to classify content as malicious or not. Most classifiers used to identify phishing email are based on: supervised learning so, they must learn before they can be used to detect a new attack; unsupervised learning, which is faster, but has a low level of accuracy. Also, some literatures present a training method for the users so, it depends on the user's ability to distinguish the attack message.

References

[1] Christopher Hadnagy: "Social Engineering: The Art of Human Hacking", John Wiley & Sons, Nov 29, 2010.

[2] C. Yue and H. Wang, "Anti-Phishing in Offense and Defense", Computer Security Applications Conference, 2008. ACSAC 2008. Annual, Anaheim, CA, 2008, pp. 345-354.

[3] P. Prakash, M. Kumar, R. R. Kompella, and M. Gupta, "Phishnet: predictive blacklisting to detect phishing attacks", in INFOCOM'10: Proceedings of the 29th conference on Information communications. Piscataway, NJ, USA: IEEE Press, 2010, pp. 346–350.

[4] Y. Cao, W. Han, and Y. Le, "Anti-phishing based on automated individual white-list," in DIM '08: Proceedings of the 4th ACM workshop on Digital identity management. New York, NY, USA: ACM, 2008, pp. 51–60.

[5] Jain, Aanchal, and Vineet Richariya. "Implementing a web browser with phishing detection techniques." arXiv preprint arXiv:1110.0360 (2011).

[6] Kirda, Engin, and Christopher Kruegel. "Protecting users against phishing attacks." The Computer Journal 49.5 (2006): 554-561.

[7] Chandrasekaran, Madhusudhanan, Krishnan Narayanan, and Shambhu Upadhyaya. "Phishing email detection based on structural properties." NYS Cyber Security Conference. 2006.

[8] Debuse, Justin CW, and Victor J. Rayward-Smith. "Feature subset selection within a simulated annealing data mining algorithm." Journal of Intelligent Information Systems 9.1 (1997): 57-81.

[9] Joachims, Thorsten. "Text categorization with support vector machines: Learning with many relevant features." European conference on machine learning. Springer Berlin Heidelberg, 1998.

[10] Pan, Ying, and Xuhua Ding. "Anomaly Based Web Phishing Page Detection."Acsac. Vol. 6. 2006.

[11] Ian Fette, Norman Sadeh, and Anthony Tomasic. "Learning to detect phishing emails." Proceedings of the 16th international conference on World Wide Web. ACM, 2007.

[12] Microsoft. Sender ID framework, 2006.
http://www.microsoft.com/senderid.

[13] Yahoo. Domainkeys, 2006.
http://antispam.yahoo.com/domainkeys.

[14] Ponnurangam Kumaraguru. "Phishguru: a system for educating users about semantic attacks". ProQuest, 2009.

[15] Gao, Hongyu, et al. "Detecting and characterizing social spam campaigns."Proceedings of the 10th ACM SIGCOMM conference on Internet measurement. ACM, 2010.

[16] Kandasamy, Kamalanathan, and Preethi Koroth. "An integrated approach to spam classification on Twitter using URL analysis, natural language processing and machine learning techniques." Electrical, Electronics and Computer Science (SCEECS), 2014 IEEE Students' Conference on. IEEE, 2014.

[17] Fuad, M. Muztaba, Debzani Deb, and M. Shahriar Hossain. "A trainable fuzzy spam detection system." Proc. of the 7th Int. Conf. on Computer and Information Technology. 2004.

[18] Bhakta, Ram, and Ian G. Harris. "Semantic analysis of dialogs to detect social engineering attacks." Semantic Computing (ICSC), 2015 IEEE International Conference on. IEEE, 2015.

[19] Dewan, Prasun, Arti Kashyap, and Ponnurangam Kumaraguru: "Analyzing social and stylometric features to identify spear phishing emails." Electronic Crime Research (eCrime), 2014 APWG Symposium on. IEEE, 2014.

[20] Allan, Ant, Kristen Noakes-Fry, and Rich Mogull. "Management Update: How Businesses Can Defend Against Social Engineering Attacks." InSide Gartner(2005).

[21] The Great Spam Archive, www.annexia.org/spam/, 2004.

[22] M. Wu, R. C. Miller, and G. Little. Web Wallet: preventing phishing attacks by revealing user intentions. In Proceedings of the SOUPS, pages 102–113, 2006.

[23] Y. Zhang, J. Hong, and L. Cranor. CANTINA: A content-based approach to detecting phishing web sites. In Proceedings of the WWW, pages 639–648, 2007.

[24] Sawa, Yuki, et al. "Detection of Social Engineering Attacks Through Natural Language Processing of Conversations." 2016 IEEE Tenth International Conference on Semantic Computing (ICSC). IEEE, 2016.

[25] 2013 Data Breach Investigations Report. Verizon, 2013. [Online].
Available: http://books.google.com/books?id=YXi0nQEACAAJ

[26] R. Levy and G. Andrew, "Tregex and tsurgeon: tools for querying and manipulating tree data structures," in International Conference on Language Resources and Evaluation, 2006.

Almomani, Ammar, et al. "A survey of phishing email filtering techniques." IEEE communications surveys & tutorials 15.4 (2013): 2070-2090