Literature Review of Social Engineering Detection Models

Marcel Teixeira
Department of Computer Science
University of Cape Town
marceltex@gmail.com

ABSTRACT

As the nature of information stored digitally becomes more important and confidential, the security of the systems put in place to protect this information needs to be increased. The human element, however, remains a vulnerability of the system and it is this vulnerability that social engineers attempt to exploit. Various detection models have been proposed to prevent social engineering attacks. Some of these models assist the user to identify whether he or she is the victim of a social engineering attack, while other models use an automated system to detect social engineering attacks. This literature review examines the Social Engineering Attack Detection Model (SEADM), the psychological measure for the SEADM, the Social Engineering Attack Detection Model version 2 (SEADMv2), using neural networks to detect social engineering, the Social Engineering Defense Architecture (SEDA) and language parsing to detect social engineering. It compares these detection models with regards to the back-end model used for the detection model, whether the model requires user interaction or not and which types of social engineering attacks the model can detect. This literature review has found the Social Engineering Attack Detection Model version 2 (SEADMv2) to be the better model, because it can be used to detect both textual and verbal social engineering attacks. It requires human interaction, however, it educates the user in the process and makes the user more vigilant. The SEADMv2 is also modular and can easily be adjusted should the need arise.

CCS Concepts

•**Security and privacy** → *Social aspects of security and privacy;*

Keywords

Natural language processing, Neural network, Psychological measure, Social Engineering, Social Engineering Attack Detection Model, Social Engineering defence Architecture

ACM ISBN 123-4567-24-567/08/06.
DOI: 10.1145/1235

1. INTRODUCTION

Social Engineering refers to various techniques that are utilised to obtain information through the exploitation of human vulnerability in order to bypass security systems [7]. Social engineers exploit the helping and trusting nature that most humans inherently have. Social engineers also prey on the fact that most people never suspect to be a victim of social engineering and are rather careless at times [9].

This literature review focuses on the existing social engineering detection models which have been proposed in various journal articles. It provides a brief description of each, describing how it achieves social engineering detection. The Social Engineering Attack Detection Model (SEADM) [1] has been documented thoroughly. In addition, the original SEADM has been improved to produce a second version of the Social Engineering Attack Detection Model (SEADMv2) [10]. Both the SEADM and the SEADMv2 are described in more detail in this literature review.

Other social engineering models have also been proposed. Some detection models assist the user to identify whether he or she is a victim of a social engineering attack, like the SEADM and the SEADMv2. Others use an automated system to detect social engineering attacks. Making use of a neural network to detect social engineering [13], is a hybrid approach. It requires the user to enter values required by the input nodes and uses a trained neural network to determine whether the user is a victim of a social engineering attack or not. The Social Engineering Defense Architecture (SEDA) [4] and using natural language processing to detect social engineering attacks [14] both use automated systems to detect social engineering attacks. The above mentioned detection models are all described in more detail in this literature review.

This literature review compares each of the detention models described, highlighting each model's advantages and disadvantages. The comparison of the detection models is summarised in Table 1. Table 1 indicates which models require human interaction and the advantages and disadvantages of each model. Lastly, the results are discussed and the better detection model is identified and motivated.

The remainder of this literature review is structured as follows: Section 2 provides a description of social engineering and discusses the financial implications that social engineering can have. Section 3 describes the Social Engineering Attack Detection Model as proposed in [1] and the improvements to this model proposed in [11] and [10]. Section 4 describes alternative social engineering detection models. Section 5 compares all the social engineering detection models

and Section 6 concludes this literature review.

2. SOCIAL ENGINEERING

Social engineering is defined as the techniques used to exploit human vulnerability to bypass security systems in order to gather information [7]. In social engineering the vulnerability of the system is considered to be the human element. The attacker exploits the trusting nature of most humans in order to get the information he or she desires. It is common for attackers to pose as an authoritative figure, such as a manager or IT support, in order to make the receiver of the call more inclined to provide them with the information they desire [4].

Successful social engineering attacks have proven to be extremely expensive. In the UK, for example, it is estimated that identity theft[1] related crimes cost the UK economy around 1.2 billion pounds in 2009 [13]. Losses from phishing[2] were around 23.2 million pounds in 2005. This is almost double the amount loss due to phishing in 2004, which was 12.2 million pounds [13]. In 2004, the US department concluded that one in three people are more likely to become a victim of social engineering in their lifetime [15]. It is therefore essential that a thorough and foolproof detection model be established to save individuals and corporations from losing millions.

3. ITERATIONS OF THE SOCIAL ENGINEERING ATTACK DETECTION MODEL

This section provides a detailed analysis of the Social Engineering Attack Detection Model as proposed in [1] and its improvements proposed in [11] and [10]. The Social Engineering Attack Detection Model has been improved over three iterations, each iteration is described in more detail in the subsections that follow.

3.1 Social Engineering Attack Detection Model (SEADM)

The Social Engineering Attack Detection Model (SEADM) proposed in [1], provides a clear guideline of how an individual can detect whether he or she is the victim of a social engineering attack. It achieves this by proposing a set of binary states in a diagram, illustrated in Figure 1. The user is required to progress through the diagram until they reach an ending state. The ending state will help the user identify whether they should provide the requester with access or elevate the requester's request. [1] provides a thorough description of each state of the model and provides real world scenarios that describe how the model could be used to detect social engineering attacks.

3.2 Cognitive Functioning Psychological Measures for the SEADM

The first state, of the SEADM, *Figure 1*, requires an individual to describe their emotional state and the last state asks an individual to evaluate the level of discomfort they are experiencing. Sometimes it is very difficult, if not impossible, for an individual to describe their own emotional state. Similarly, it could be a challenging task for an individual to evaluate the level of discomfort imposed on them by the attacker.

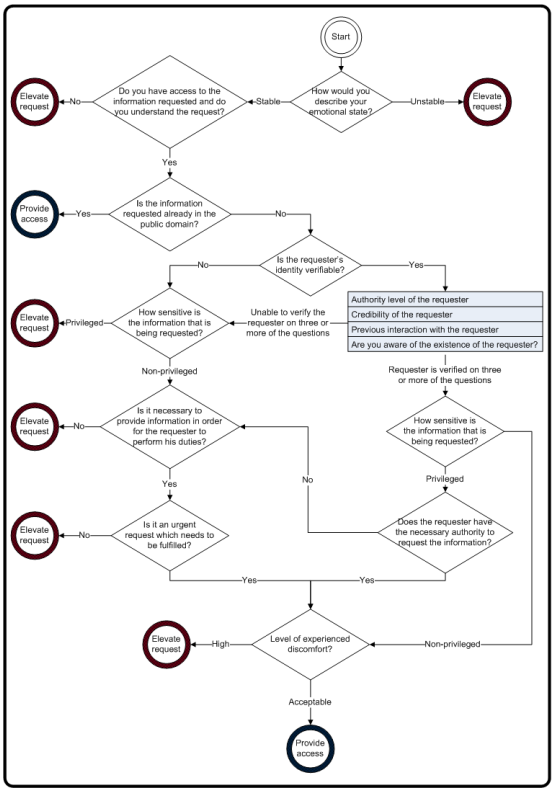

Figure 1: Social Engineering Attack Detection Model

Since one's emotional state is not a quantitative measure, an individual's own interpretation of their emotional state may be open to bias. Therefore, in [11] it is proposed that a psychological measure be used to determine the emotional state of an individual. However, this psychological measure would be rather impractical, if it was based on personality testing [11]. Instead, it has been shown that there is a link between the performance of an individual on certain cognitive functioning based tests and the emotional state of an individual [6], [5].

In [11] three cognitive functioning based tests are identified to be used as psychological measures for the SEADM. The three psychological measures include, the Wisconsin Card Sorting Test [8], Eriksen's Flanker Test [3] and the Dot Judgement Test [2]. All three of these psychological measures are ideal, because they can be taken very briefly by an individual and they return numerical values. These numerical values can then be fed into a feedforward neural network[3], that has already been trained with appropriate training data, to identify an individual's emotional state.

3.3 Social Engineering Attack Detection Model Version 2 (SEADMv2)

[10] describes a second version of the Social Engineering Attack Detection Model (SEADMv2). A diagram of this

[1]The fraudulent acquisition and use of a person's private identifying information, usually for financial gain.

[2]The activity of defrauding an online account holder of financial information by posing as a legitimate company.

[3]A computational model that is biologically inspired by the neurons of the brain [12].

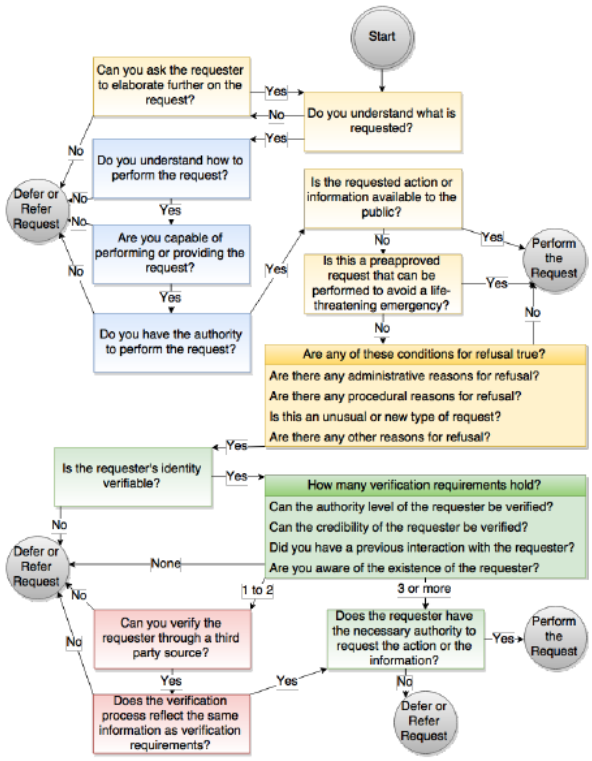

Figure 2: Social Engineering Attack Detection Model version 2

model can be seen in Figure 2. Figure 2 has more states than Figure 1 and the states are also colour coded. The colours used for the states are to differentiate the different types of states supported by the model. Yellow states are request states and they deal with the request itself. Blue states are receiver states and they deal with whether an individual understands what is being requested. The green states deal with the requester and any information that can be determined about the requester and the red states are third party states and refer to whether the requester can be verified using a third party [10].

While the states in the SEADMv2 refer to four different parties, it is important to keep in mind, that the SEADMv2, will still be used by an individual receiving a request. The individual will use this model to determine whether the request can be performed or if it should be deferred or referred to someone else. [10] describes each state in the SEADMv2 thoroughly and also provides three scenarios to clearly illustrate how it can be used in reality. The SEADMv2 can be used to detect either bidirectional, unidirectional or indirect communication. Lastly, it is important to note,that the states that dealt with the receiver determining his or her emotional state and the level of discomfort experienced are omitted from the SEADMv2. These states are to be dealt with by a separate psychological measure, as discussed in the previous subsection.

4. ALTERNATIVE SOCIAL ENGINEERING DETECTION MODELS

The Social Engineering Attack Detection Model (SEADM) is not the only way of detecting social engineering attacks. This section will discuss other detection models that are used to detect or prevent social engineering attacks.

4.1 Social Engineering Detection Using Neural Networks

A neural network is a computational model that is biologically inspired by the neurons of the brain [12]. Neural networks make use of different layers of nodes which are trained with training data in order to produce the right output, given valid input values.

In [13], a feedback neural network with 4 input layer nodes, 2 hidden layer nodes and 1 output layer node is proposed. Only one output layer node is required since the result is binary, it is either a social engineering attack or not. The neural network was trained using training data that consisted of 20 examples. It was then tested using sample examples of both non-malicious requests and social engineering attacks. The neural network identified the attacks really accurately. There is a clear pattern of input data that the neural network used to identify social engineering attacks. However, it should be noted that the neural network has yet to be tested with real-world data.

4.2 Social Engineering Defense Architecture (SEDA)

The Social Engineering Defense Architecture (SEDA), proposed in [4], is a software system used to detect telephonic social engineering attacks. Social engineers often use identities of authoritative figures in order to persuade the person on the other end of the phone to give them the information they require or to do what they want. Typical identities social engineers use include IT support, managers or a trusted third party. The only authentication required is answering a few questions regarding information that only an employee would know. Should the social engineer know the answers to these questions the receiver of the telephone call will assume the social engineer is who they say they are and proceed to follow the attacker's orders.

The SEDA proposes using a voice signature authentication system. The idea is to link the voice signatures to a database of personal information of the employees, such as their name, corporate association, job title as well as all the phone numbers that a particular employee would possibly phone from. The SEDA would be able to identify social engineers even if they are able to answer all security questions. In addition, the SEDA would prevent social engineers from calling multiple times under different aliases. It would detect that the same voice signature is being used for different people and immediately flag the caller as an attacker.

One major advantage of the SEDA is that it does not delay or alter the flow of normal operation. The SEDA runs in the background recording the caller's voice and querying the database, while the receiver is talking to the caller. If the SEDA detects that a social engineer may be on the other end of the line, it immediately alerts the receiver. On the other hand, a major flaw of the SEDA is its inability to deal with voice modulation. Should the social engineer be aware of the SEDA system installed, he or she could make use of voice recordings of the person they are imitating and play it back into the phone. This would make the SEDA believe that the caller is indeed the person they claim to be. However, obtaining these voice recordings, especially of somebody high

up in the company could be somewhat difficult and would require an extremely skilled social engineer.

4.3 Detection of Social Engineering Attacks Through Natural Language Processing

An approach to social engineering detection using natural language processing, is proposed in [14]. It is similar to the SEDA [4], in that it uses a software system to detect social engineering attacks. However, this mechanism only detects textual social engineering attacks, such as phishing emails.

The software system achieves the detection of social engineering attacks through natural language processing by going through a series of steps. The first step involves determining whether the attacker is using a command or a question. A command is when the attacker tells the victim to do something that will most likely bring harm to the victim or the company for which the victim works. A question is when the attack asks the victim to provide him or her with information that the attacker is not authorised to have. To detect whether a sentence is a command or question, the system places the sentence in a parse tree. This parse tree is then examined for patterns which can be used to determine whether a sentence is a question or a command.

Once the system has determined whether a sentence is a question or command, it uses the parse tree to extract the topic of the sentence. The topic of the sentence, in this context, is a pair consisting of the main verb of the sentence and its direct object. The topic is then checked against the topic blacklist and if it is found in the topic blacklist, the victim is alerted that this could be a social engineering attack. The topic blacklist is a list of action-resource pairs that describe operations which can not be performed on certain resources. If the the topic is not found in the topic blacklist, the system does not alert the user and proceeds to parse the following sentence.

5. COMPARISON OF SOCIAL ENGINEERING DETECTION MODELS

The Social Engineering Attack Detection Model (SEADM) as proposed in [1] has both strong points and shortfalls. Advantages of the SEADM include the fact that it is has a very modular design. It will therefore be relatively straight forward to implement it in code, as well as to make additions or remove states later on. A disadvantage of the SEADM is that it requires the user to determine his or her own emotional state as well as the level of discomfort experienced. This is not ideal, since most of the time individuals find it difficult to determine their own emotional state accurately.

The psychological measure described in [11], highlights three cognitive functioning tests that could be used to determine an individual's emotional state. These tests are advantageous in that they are quick to perform and also provide a concrete way of determining an individual's emotional state, rather than expecting an individual to describe his or her own emotional state. A disadvantage is that after performing the tests multiple times a day, the individual could start finding the tests repetitive and not give their full attention when answering the questions. This could make the tests yield inaccurate results.

The Social Engineering Attack Detection Model version 2 (SEADMv2) proposed in [10] makes adjustments to the original Social Engineering Attack Detection Model (SEADM) proposed in [1]. Adjustments that improve the model include adding colour codes to the states to indicate the type of state. It also has more state transitions to improve the accuracy of the model when used for real world scenarios. In addition, it is more modular than the SEADM. A disadvantage of the SEADMv2, is that it has no states that examine the emotional state of an individual. This is because the emotional state of the individual is assumed to be determined by making use of one of the psychological measures mentioned previously.

Using a neural network to detect social engineering is suggested in [13]. The advantage of this approach is that the neural network, if trained well, tends to be accurate at detecting social engineering attacks. However there are multiple disadvantages to this approach. For one, it has not been tested in real-world situations and just been proven to work well with sample data and in sample scenarios. In addition, it could be quite a tedious task for an individual to enter data values into the input nodes of the neural network each time a requester requests information.

The Social Engineering defence Architecture (SEDA) outlined in [4] is a system used to detect telephonic social engineering attacks using voice signatures. This approach of detecting social engineering attacks is advantageous, because it requires minimal extra effort from the receiver. The system runs in the background and detects whether the caller's voice signature matches the person he or she claims to be. Another advantage is that it prevents the same social engineer from calling different employees at the same company, under different aliases. However, this system is not foolproof, should a very skilled social engineer obtain sufficient voice recordings of the person he or she is imitating, they would be able to trick the SEDA into telling the receiver of the call that the attacker is who they say they are. This is essentially a flaw in the system more than it is a disadvantage and will need to be corrected, should the SEDA be implemented.

Using natural language processing to detect social engineering attacks as proposed in [14] is a system that parses textual messages to determine whether they are attacks or not. An advantage of this approach to social engineering detection is that it is accurate and processes text rapidly. According to the results obtained in [14], the system was able to parse 74 sentences in 5.357 seconds. It identified 6 of the 10 malicious sentences and had no false positives[4]. Another advantage of this detection method, is that it runs in the background with no user interaction required. A disadvantage is that it only works for detecting textual based social engineering attacks. It is therefore completely useless for a call centre worker, unless some speech recognition software is used to transcribe the phone conversation.

Table 1, provides a summary of the the different aspects of the social engineering detection models discussed in this literature review.

6. CONCLUSIONS

The social engineering detection models discussed in this literature review are all viable solutions for detecting or preventing social engineering attacks. It is clear from Table 1 that most detection models have more advantages than they

[4]A false positive occurs when the system detects a non-malicious sentence as a malicious one.

have disadvantages. Most of the detection models have also been shown to be accurate in real world situations.

The detection model that proved to be the best, of the detection models compared in this literature review, is the Social Engineering Attack Detection Model version 2 (SEADMv2). The SEADMv2 can be used to detect both textual and verbal social engineering attacks, while the Social Engineering Defense Architecture (SEDA) can only detect verbal attacks and the use of natural language parsing can only detect textual attacks. Unlike the use of neural networks to detect social engineering attacks, the SEADMv2 has actually been proven to work accurately with real world scenarios. The SEADMv2 is more modular than the SEADM and contains more states which help it model real world scenarios more accurately.

The only real downfalls of the SEADMv2 is that it does not examine the user's emotional state and that it requires user interaction. It has been proposed to use psychological measures with a feedforward neural network in order to determine the user's emotional state. This could be incorporated into the SEADMv2 with ease, due to the SEADMv2's modular design. The fact that the user has to interact with the SEADMv2, could be seen as tedious. However, it will also serve to educate the user and make the user more vigilant to social engineering attacks in his or her everyday life.

7. REFERENCES

[1] Bezuidenhout, M., Mouton, F., and Venter, H. S. Social engineering attack detection model: Seadm. In *Information Security for South Africa (ISSA), 2010* (2010), IEEE, pp. 1–8.

[2] Cicchetti, D. V., and Rourke, B. P. *Methodological and biostatistical foundations of clinical neuropsychology and medical and health disciplines*. CRC Press, 2004.

[3] Eriksen, C. W. The flankers task and response competition: A useful tool for investigating a variety of cognitive problems. *Visual Cognition 2*, 2-3 (1995), 101–118.

[4] Hoeschele, M., and Rogers, M. Detecting social engineering. In *Advances in Digital Forensics*. Springer, 2005, pp. 67–77.

[5] MacLeod, C., and Mathews, A. Biased cognitive operations in anxiety: accessibility of information or assignment of processing priorities? *Behaviour research and therapy 29*, 6 (1991), 599–610.

[6] Mathews, A. Why worry? the cognitive function of anxiety. *Behaviour research and therapy 28*, 6 (1990), 455–468.

[7] Mitnick, K. D., and Simon, W. L. *The art of deception: Controlling the human element of security*. John Wiley & Sons, 2011.

[8] Monchi, O., Petrides, M., Petre, V., Worsley, K., and Dagher, A. Wisconsin card sorting revisited: distinct neural circuits participating in different stages of the task identified by event-related functional magnetic resonance imaging. *The Journal of Neuroscience 21*, 19 (2001), 7733–7741.

[9] Mouton, F., Leenen, L., Malan, M. M., and Venter, H. S. Towards an ontological model defining the social engineering domain. In *ICT and Society*. Springer, 2014, pp. 266–279.

[10] Mouton, F., Leenen, L., and Venter, H. S. Social engineering attack detection model: Seadmv2. In *2015 International Conference on Cyberworlds (CW)* (2015), IEEE, pp. 216–223.

[11] Mouton, F., Malan, M. M., and Venter, H. S. Development of cognitive functioning psychological measures for the seadm. In *HAISA* (2012), IEEE, pp. 40–51.

[12] Rao, V. B., and Rao, H. C++ neural networks and fuzzy logic. 599–610.

[13] Sandouka, H., Cullen, A., and Mann, I. Social engineering detection using neural networks. In *CyberWorlds, 2009. CW'09. International Conference on* (2009), IEEE, pp. 273–278.

[14] Sawa, Y., Bhakta, R., Harris, I. G., and Hadnagy, C. Detection of social engineering attacks through natural language processing of conversations. In *2016 IEEE Tenth International Conference on Semantic Computing (ICSC)* (2016), IEEE, pp. 262–265.

[15] Workman, M. Gaining access with social engineering: An empirical study of the threat. *Information Systems Security 16*, 6 (2007), 315–331.

	SEADM	SEADM - Psychological Measure	SEADMv2	Social Engineering Detection using Neural Networks	SEDA	Social Engineering Detection using Natural Language Processing
Detection Model Used:	States	Cognitive functioning measures	States	Neural networks	Voice signatures	Natural language processing
User Interaction Required:	Yes	Yes	Yes	Yes	No	No
Types of Social Engineering Attacks which Model can Detect:	Textual and verbal	Textual and verbal	Textual and verbal	Textual and verbal	Verbal	Textual
Advantages:	Modular design.	Quick to perform tests. Provide a concrete way of determining emotional state.	Colour codes to differentiate types of states. More state transitions than the SEADM. More modular design than the SEADM. Caters for bidirectional, unidirectional and indirect communication.	Accurate at detecting attacks.	No user interaction required. Prevents same social engineer targeting different employees.	No user interaction required. Processes text rapidly. Accurate at detecting attacks.
Disadvantages:	Requires user to determine own emotional state. Only caters for bidirectional communication.	Tests could become repetitive if performed too many times.	No states to examine the emotional state of the user.	Never been tested in a real world scenario. Tedious for the user to enter values into the input nodes.	Social engineer could trick the system by using voice recordings of the person they are imitating. Only works for verbal social engineering attacks.	Only works for textual social engineering attacks.

Table 1: Comparison of Social Engineering Detection Models