Proceedings of the Eleventh International Symposium on
Human Aspects of Information Security & Assurance (HAISA 2017)

Mapping the Anatomy of Social Engineering Attacks to the Systems Engineering Life Cycle

J. van de Merwe[1] and F. Mouton[2]

[1]Defence Peace Safety & Security, South Africa
[2]Council for Scientific and Industrial Research, Pretoria, South Africa
e-mail: JvdMerwe3@csir.co.za; moutonf@gmail.com

Abstract

Social engineering attacks present a material threat to the security of information systems. To date security professionals only manage the potential effects of a social engineering attack. Security professionals consider such attacks as external threats to the overall information system and so far preventative measures are mostly focused around asking people to be aware and guard against becoming victims through tailored cyber-awareness campaigns. The social engineering attack framework (SEAF) presents a way to think about social engineering pro-actively. Furthermore, systems engineering is about coping with complexity. Systems engineering helps to avoid omissions and invalid assumptions. It also helps to manage real world changing issues, and produce the most efficient, economic and robust solution. Within the systems engineering discipline extensive techniques have been developed to support its underlying principles and processes. By aligning the SEAF to systems engineering life cycle, access to those techniques are granted allowing for a security professional to cope with the complexities of social engineering attacks in a defined and quantitative manner. This gives the opportunity to explore applying the various techniques to assist in handling social engineering attacks as part of system security, including people, processes and technology, not to mention it links the efforts to a budget. The latter is especially relevant when justifying the means to cope with social engineering attacks, for example to establish and drive an awareness campaign. Before all this can happen, we first need to establish the link between the SEAF and systems engineering, which is what this paper is aimed at. The benefit of this link is that it will allow for a direct translation of our premised scenario to the tools used in the systems engineering space. These include a context diagram, functional modelling, holistic requirements modelling, matrix diagrams, stakeholder maps and a viewpoint analysis.

Keywords

Bidirectional Communication, Indirect Communication, Mitnick's Attack Cycle, Social Engineering Attack Detection, Social Engineering Attack Framework, Social Engineering Ontology, Systems Engineering Life Cycle, Unidirectional Communication, Information Security.

1. Introduction

The field of information security is a fast growing discipline. The protection of information is of vital importance to organisations and governments, and the development of countermeasures against illegal access to information is an area that receives increasing attention. Organisations and governments have a vested interest in securing sensitive information and thus securing the trust of clients and citizens.

Technology on its own is not a sufficient safeguard against information theft; staff is often the weak link in an information security system. Staff members can be influenced to divulge sensitive information which subsequently allow unauthorised individuals to access protected systems. Social engineering poses a material threat to security of passwords, i.e. authentication, and security of networks and therefore has the ability to degrade two of the three objectives within the information assurance triad (confidentiality, integrity and availability) directly, information availability (networks) and confidentiality (achieved through authentication).

Perhaps a good starting point is the perspective of social engineering from an information security perspective as adopted from the Certified Information Systems Security Professional material (Stewart, J.M, Chapple, M., Gibson, 2013). Social engineering refers to various techniques that are utilised to obtain information through the exploitation of human vulnerability in order to bypass security systems (Mitnick and Simon, 2002, 2005). Social engineers exploit the helping and trusting nature that most humans inherently have. Social engineers also prey on the fact that most people never expect to be a victim of social engineering and are rather careless at times (Mouton, Leenen, *et al.*, 2014).

Successful social engineering attacks have proven to be extremely expensive. In the UK, for example, it is estimated that identity theft related crimes cost the UK economy around 1.2 billion pounds in 2009 (Sandouka, Cullen and Mann, 2009). While all of this cannot be attributed to social engineering attacks, it is reasonable to expect that a significant proportion of these losses will be related to such attacks.

The authors have previously proposed both an ontological model and a social engineering attack framework (SEAF). The ontological model includes components of a social engineering attack and divides the attack into different classes and subclasses. The two classes of a social engineering attack are: Direct communication and indirect communication. The direct communication class is further divided into two subclasses: Bidirectional communication and unidirectional communication. A social engineering attack is then further explained to contain the following components: one Social Engineer; one Target; one or more Compliance Principles; one or more Techniques; one Medium; and one Goal (Mouton, Leenen, *et al.*, 2014).

This paper focuses on the systems engineering perspective in developing a methodology which is aligned to the SEAF in assisting an information security professional. The purpose of this paper is to lay the groundwork for developing a methodology that will aid the information security professional to build systems that are resistant to social engineering attacks. This paper provides the link between the systems engineering life cycle and the process that is followed by a social engineer that performs a social engineering attack. Having this link between the two processes, allows one to further develop a process to aid in countering social engineering attacks. This will ultimately assist an information security professional in implementing people, processes and technology that will intrinsically offer resistance to social engineering as a specific attack vector.

So why align to systems engineering? Systems engineering is about coping with complexity. It helps to avoid omissions and invalid assumptions, it helps to manage real world changing issues, and produce the most efficient, economic and robust solution (Smith and Brown, 2014). Further, within the systems engineering discipline extensive techniques have been developed to support its underlying principles and processes. By aligning the SEAF to systems engineering life cycle, access to those techniques are granted allowing for a security professional to cope with the complexities of social engineering attacks in a defined and quantitative manner.

The rest of the paper is structured as follows: Section 2 provides the definition of social engineering and social engineering attacks. Furthermore, it provides the social engineering attack framework. Section 3 discusses the systems engineering life cycle and provides the mapping between the social engineering attack framework and the systems engineering life cycle. Section 4 elaborates on each phase of the systems engineering life cycle and how it is mapped to the social engineering attack framework. Section 5 concludes the paper with a brief summary of the linkage and the benefits it provides. The section also provides an elaborate discussion on how the methodical mapping will aid future research.

2. Background

There are many models and taxonomies concerning social engineering attacks which are explored and analysed in the author's previous paper (Mouton, Leenen, *et al.*, 2014) such as (Harley, 1998; Laribee, 2006; Ivaturi and Janczewski, 2011; Mohd Foozy *et al.*, 2011; Tetri and Vuorinen, 2013). The most commonly known model is Kevin Mitnick's social engineering attack cycle as described in his book, *The art of deception: controlling the human element of security* (Mitnick and Simon, 2002). Mitnick's attack model has four phases: research, developing rapport and trust, exploiting trust and utilising information.

The picture below is a representation of Mitnick's attack cycle created by the authors. Figure 1 depicts the four phases and the flow between each of the phases. Each of these phases are briefly discussed below as explained in Mitnick's book.

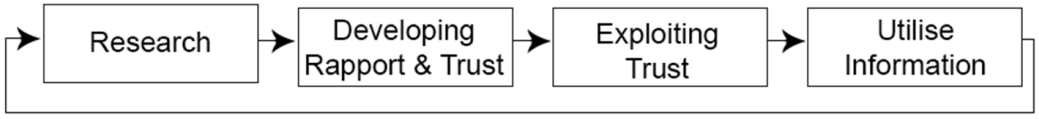

Figure 1: Kevin Mitnick's Social Engineering Attack Cycle (Mouton, Leenen, *et al.*, 2014)

Research is an information gathering process where information about the target is retrieved. The attacker should know as much as possible about the target before starting the attack.

The next phase is the **Development of the rapport and trust** with the target. A target is more likely to divulge requested information to an attacker if he trusts the attacker. According to (Mitnick and Simon, 2002), rapport and trust development can be done by using insider information, misrepresenting an identity, citing those known to the victim, showing a need for assistance, or occupying an authoritative role.

When a target appears to trust an attacker, the attacker **Exploits the trust** to elicit information from the target: this can either take the form of a request for information, a request for a specified action from the victim or, alternatively, to manipulate the victim into asking the attacker for help (Mitnick and Simon, 2002). This phase is where the previously established relationship is abused to get the initially desired information or action.

Finally, the outcome of the previous phase is **Utilised** to reach the goal of the attack or to move on to further steps which may be required to reach the goal.

A trivial example is when an attacker supposedly needs to connect to an organisation's network. As a result of his research the attacker finds out that a help-desk staff member knows the password to the organisation's wireless network. In addition, the attacker found personal information regarding the staff member who has been identified as the target. The attacker initiates a conversation with the target, using the acquired information to establish trust; in this case the attacker misrepresents himself as an old school acquaintance of the target. The attacker subsequently exploits the established trust by asking permission to use the company's wireless network facility to send an e-mail. The help-desk attendant is willing to supply the required password to the attacker due to the misrepresentation, and is able to gain access to the organisation's network and achieve his objective.

The authors' ontological model defines that a social engineering attack "employs either direct communication or indirect communication, and has a social engineer, a target, a medium, a goal, one or more compliance principles and one or more techniques" (Mouton, Leenen, *et al.*, 2014). The attack can be split into more than one attack phase, each phase handled as a new attack according to the model. The model is depicted in Figure 2.

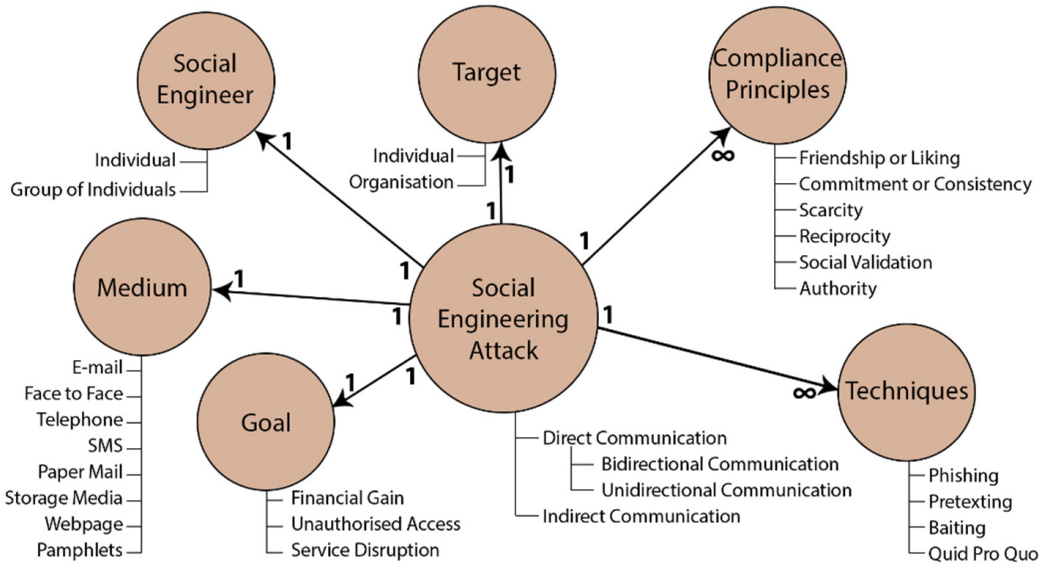

Figure 2: An Ontological Model of a Social Engineering attack (Mouton, Leenen, *et al.*, 2014)

Direct communication, where two or more people communicating directly with each other, is sub-divided into b*idirectional communication* and *unidirectional communication*. Bidirectional communication occurs when both parties participate in the conversation. For example, an e-mail is sent from the attacker to the target and the target replies to the attacker. Unidirectional communication occurs when the conversation is one-way only: from the attacker to the target. For example, if the attacker sends a message through paper mail without a return address, the target cannot reply to the message. Phishing attacks are also a popular type of attack in this category.

Indirect communication is when there is no actual interaction between the target and the attacker; communication occurs through some *third party medium*. An example of this type of communication is when the attacker infects a flash drive and leaves it somewhere to be found by some target. The target is curious to find out what is on the flash drive for personal gain or, motivated by ethical consideration, to attempt to find the owner of the flash drive. The target inserts the flash drive into their computer, and the infection on the flash drive is activated.

The ontological model also contains components such as a goal, a medium, a social engineer, a target, compliance principles and techniques. The goal of an attack can be financial gain, unauthorised access or service disruption. The medium is a way of communication such as e-mail, face-to-face contact, a telephone call, etc. The social engineer can be either an individual or a group of individuals. The target can either be an individual or an organisation. Compliance principles refer to the reasons why a target complies with the attacker's request, and techniques include those used to perform social engineering attacks. Examples of techniques include phishing,

pretexting, baiting and quid pro quo (Mouton, Leenen and Venter, 2016). Examples of compliance principles include (Cialdini, 1987, 2007):

- *Friendship or liking:* People are more willing to comply with requests from friends or people they like.
- *Commitment or consistency:* Once committed to something, people are more willing to comply with requests consistent with this position.
- *Scarcity:* People are more willing to comply with requests that are scarce or decreasing in availability.
- *Reciprocity:* People are more willing to comply with a request if the requester has treated them favourably in the past.
- *Social Validation*: People are more willing to comply with a request if it is seen as the socially correct thing to do.
- *Authority:* People comply easily to requests given by people with more authority than they have.

Once the compliance principles, techniques and medium have been selected, the attack vector can be set up and the social engineer can continue with the actual attacking phase. The social engineering attack framework can be used to depict the planning and flow of the full attack. Figure 3 depicts the social engineering attack framework (Mouton, Malan, *et al.*, 2014).

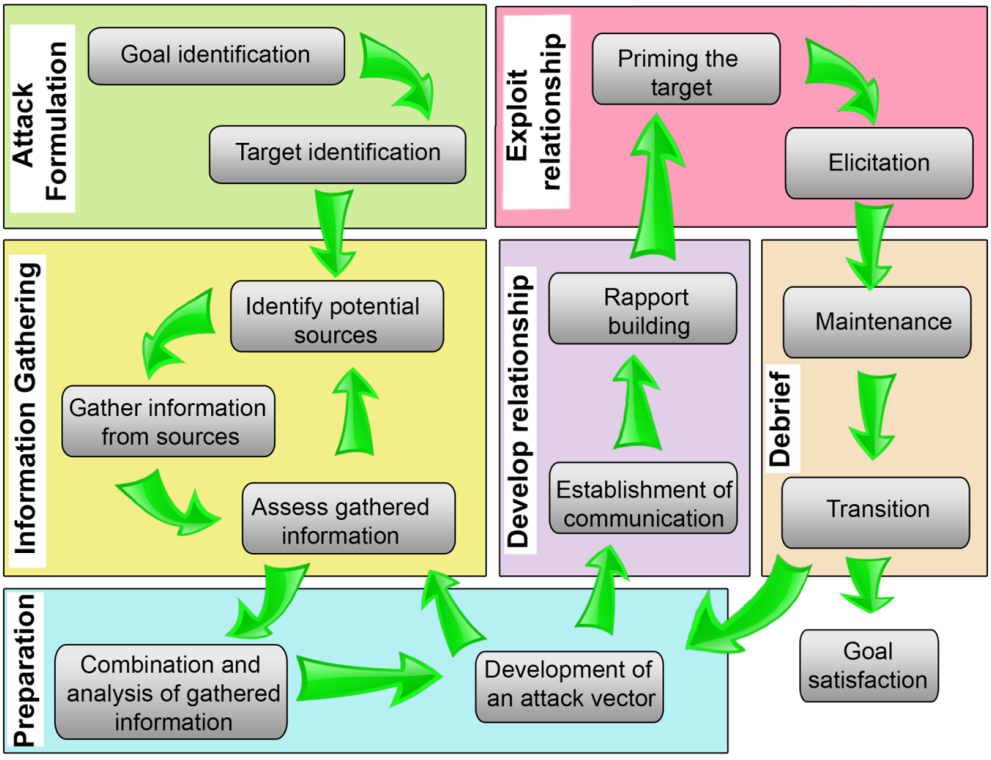

Figure 3: Social Engineering Attack Framework (Mouton, Malan, *et al.*, 2014)

The social engineering attack framework has six core phases, namely attack formulation, information gathering, preparation, develop relationship, exploit relationship and debrief (Mouton, Malan, *et al.*, 2014).

The *attack formulation* phase is used to identify both the goal and the target of the specific attack. The *information gathering* phase is used to identify all sources of information on both the goal and the target, as well as to gather information from the identified sources. In the *preparation* phase, all the gathered information is combined and the social engineering attack vector is developed. It is during the *preparation* phase that all the elements in the social engineering ontological model can be identified. The *develop relationship* phase is where the attacker establishes communication with the target and attempts to build a trust relationship with the target. The *exploit relationship* phase is used to prime the target and to elicit the target to perform the request or action. The final phase is the *debrief* phase, in which the target is brought out of a primed state during the *maintenance* step, and the *transition* step tests whether the goal has been satisfied.

The next section briefly discusses the systems engineering life cycle after which it maps the social engineering attack framework to the systems engineering life cycle.

3. Mapping Social Engineering to Systems Engineering Principles

According to (Honour, 1998), Systems engineering is an engineering discipline whose responsibility it is to create and execute interdisciplinary processes to ensure the customer and stakeholders' needs are satisfied in a high quality, trustworthy, cost-efficient and schedule compliant manner. An example is such a generic life cycle process for a system that addresses the need and results in a satisfactory outcome is presented in (Jacobs, 2015). There are slight nuances depending if one looks at the product system life cycle, product design process, the generic system life, and considering a complex system, software system and then also the development model being followed, agile, waterfall, etc. Without delving into specifics the generic systems engineering life cycle process for the purposes of this paper will suffice (Honour, 1998). This process is depicted in Figure 4.

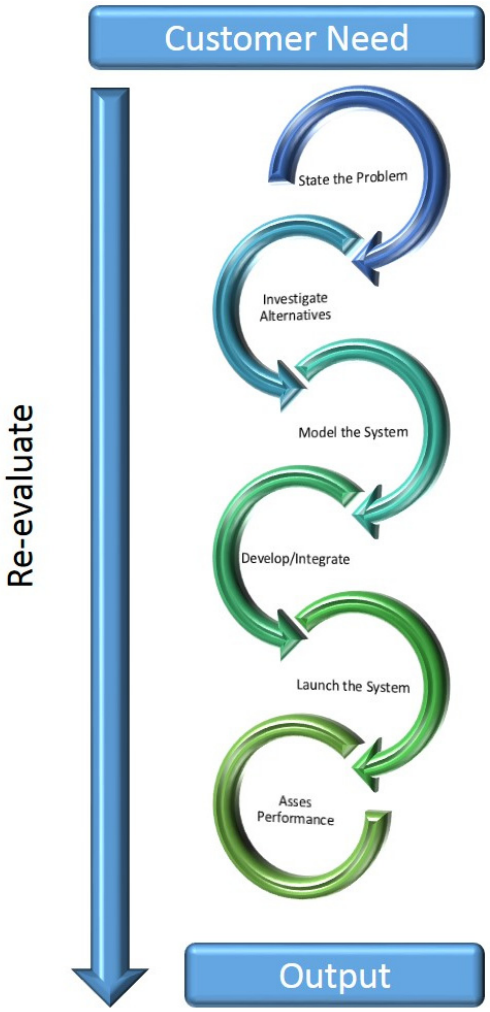

Figure 4: Systems Engineering Life Cycle Process

During the *State the Problem phase* we consider what we want to achieve and define the exploit including performance measurements. *Investigate alternatives* and *Model the system* occur iteratively to do exactly that, investigate the various alternatives of how we would like to address the problem, informed by measures of performance. These measures are defined during the *investigate* phase to ensure a metric once the completed "system" has been designed and implemented and we need to ensure, did we do the right thing. During the later *Asses Performance* phase, the technical performance measures will be used to ensure we did it right. The *Development/Integrate* and *Launching* phases respectively complete to detailed planning and design of the solution and integrate it into the final solution which is then released upon the world. The *Re-evaluation* activity occurs throughout all the phases and serves a reference to ensure one is addressing the stated problem in the relevant way, it runs concurrently and not just as a once-off activity.

As a security professional, you are entrusted with ensuring the confidentiality, integrity and availability of the information to the organisation. It is thus important to understand social engineering techniques in order to prevent attacks that stem from this attack vector. It is the authors' hope that systems engineering principles can aid in this objective. Due to the nature of social engineering attacks it is by far the most difficult type of information attack to protect against. Its unpredictability and variability make it a major security risk to any information system with humans in the loop. A social engineering attack is a type of attack, which is based on building trust with someone, or tricking him or her in an attempt to gain information the person wouldn't normally reveal, or performing an action the person wouldn't normally perform. As indicated, the ultimate goal is to develop a methodology that would assist in the prevention of social engineering attacks. For the purposes of this paper and to also first lay the groundwork, view it from the attacker's perspective, i.e. with relation to the systems engineering life cycle, the customer is the attacker that has the need to perform a successful social engineering attack employing the social engineering attack framework as proposed by (Mouton, Malan, *et al.*, 2014).

Having both the social engineering attack framework, as proposed by (Mouton, Malan, *et al.*, 2014), and the generic systems engineering life cycle, as proposed by (Honour, 1998), the authors now explore how these two models can be aligned. The starting point of this alignment is still the "Need", however, from a social engineering point of view it might be more apt to consider it a "want", a so-called desire to achieve a socially engineered exploit. This is in slight contrast to a customer's need to solve a problem. Disregarding the ethical implications of this statement, the attacker has a problem, which is to perform an exploit and obtain information. In the systems engineering case, one assumes complete "good" in the desire to solve a problem. This discussion is not for the scope of this paper and it is implied that the attacker has no ethical considerations and has the intent to cause harm (Mouton, Malan and Venter, 2013; Mouton *et al.*, 2015).

In this case the starting point is that in the same way a systems engineer has the need to solve a problem, and then embarks on the systems engineering process, so too does an attacker when planning to solve his or her need to run an exploit. There is reason for this in that it will allow the social engineering attack framework to be considered a process, or methodology aligned in the manner presented. This in turn then creates the opportunity to develop an approach to counter the process used by the social engineer in a manner which enables frameworks with appropriate controls to be developed that makes it particularly hard for the attacker to be successful. A good example of using systems engineering to develop such a counter-process is by using a context diagram from systems engineering which is a good tool to assist with classifying and understanding the type of interaction external entities have on the proposed system (Kossiakoff *et al.*, 2011). One can use it in one of two ways. Either during the second phase, *investigate alternatives*, of the systems engineering life cycle, as it provides the information security professional with a tool to indicate entities or actors, as well as the relationship it has with the systems. Alternatively or additionally, the context diagram can be used as part of continuous *re-evaluation* to ensure the system as a whole remains true to its intent. Being able to define the

boundary between the system, or part thereof and the environment it interacts with, i.e. the purpose of a context diagram, ironically seems a better fit for use in the social engineering context than the systems engineering context it was designed for. For example, the information specialist could use this tool to understand the entities and actors interacting with the systems and classify the approaches of attackers, thereby facilitating a better designed system to cope with specific attack vectors.

This paper does not seek to develop that counter-process, but lays the groundwork necessary to start with its development by establishing the link between the SEAF and the systems engineering life cycle. Figure 5 depicts the mapping between the social engineering attack framework and the systems engineering life cycle.

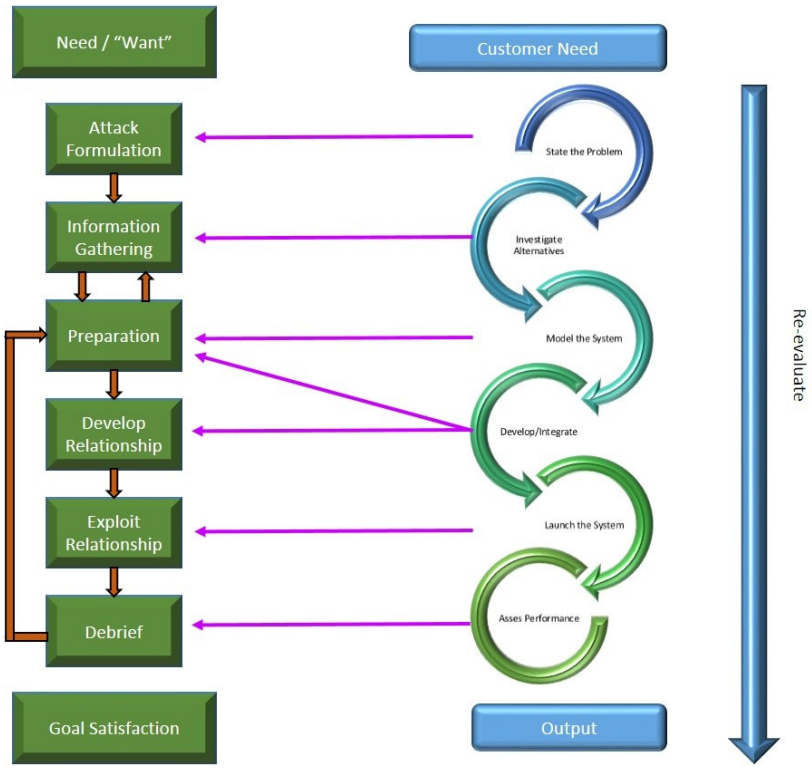

Figure 5: Aligning SEAF phases to Systems Engineering Life cycle phases

From Figure 5, one notes a straightforward mapping. The benefit of this mapping is that it now allows for a direct translation of our premised scenario to the tools used in the systems engineering space. These include, a context diagram, functional modelling, holistic requirements modelling, matrix diagrams, stakeholder maps and a viewpoint analysis (Burge Hughes Walsh Limited, 2017). This will allow for a deeper understanding of the social engineering problem, an understanding an information security specialist will use to further our research in developing a counter process and methodology to better guard against social engineering attacks.

Using this as the premise, Section 4 is dedicated to discussing the mapping between the social engineering attack framework and the systems engineering life cycle.

4. Presenting the need: A Systems engineering perspective

4.1. State the Problem

The focus here is to describe "what" must be done, not the "how". It takes the customer need and translates it into functional or behavioural terms. It may include a description of the deficiency that must be removed. Performance measures, known as measures of effectiveness are defined to ensure the designed solution's effectiveness at addressing the problem are stated (Johns Hopkins Whiting School of Engineering, 2013). Given the attack framework from Figure 3, the problem as indicated here is simply to perform a successful social engineering attack by identifying the goal, and then identifying the target individual.

The goal could simply be to gain access to an IT infrastructure or physical facility. Occasionally, the goal is to install malware, which then assists the attacker on gaining the information through the malware, like a key logger, to gain them the access they require.

There remains a question on the "why" the attack is executed, which could be for financial gain of some sort, pure malice or just to be a nuisance. Irrespective, the driver behind the attack will not determine the process taken to attack, although it may motivate the attacker to keep trying and increase his or her resourcefulness until success is achieved.

4.2. Investigate Alternatives and Model the System

It makes sense to consider these two processes together as it entails investigating various designs and evaluating predicted compliance to performance and functionality. This is a good place to make use of techniques that assist the decision making process. For example, in control theory one would model the system, verify the model through simulation, perform hardware-in-the-loop testing, implement the design, evaluate various design parameter combinations, simulate the effect all whilst verifying the model to a real-life system as you go. The measures of effectiveness (MoE) are expressed as technical performance measures (TPM) (Johns Hopkins Whiting School of Engineering, 2013). Here the attacker would identify potential sources of the necessary access information, then gather this information through a variety of techniques and then assess whether gathered information is sufficient to obtain desired level of access. A good measure of effectiveness, would thus be the assessment of this gathered information, whereas the technical performance measures would include specific pieces of information that would be specific to the attack being run. A typical example of a MoE and TPM for a social engineering attack would be as follows:

- TPM – gain subject's network password.
- MoE^1 – Asses whether gathered password gives access with necessary privileges to perform elevation of privilege and gain network autonomous access, or at least access to desired information.
- MoE^2 – Developed attack vector based on assessed information.

The premise of the attack often leverages human traits such as basic human trust, desire to provide assistance or correct someone if they get something wrong, propensity to show off, being distracted, following orders and fearing reprimands. The attacker may use impersonation, masquerading as a legitimate entity seeking information, or perhaps resort to shoulder surfing to obtain information which is involuntarily offered. In terms of some techniques used to source the information may include various types of phishing, i.e spear fishing, whaling and vishing as well as pretexting or baiting (Stewart, J.M, Chapple, M., Gibson, 2013).

4.3. Develop/Integrate

Once a satisfactory solution has been engineered, one needs to develop the solution for deployment, which entails the successful integration of people, processes and technology with the systems and sub-systems. For the case of the social engineer this would imply developing the attack vector and establishing initial communication. After the attack vector has been developed and ready to execute, communication is established with the identified individual and rapport building commences to facilitate information sourcing. A very useful technique to consider in rapport building and re-enforcing effective information exploitation in the next phase is profiling.

4.4. Launch the System

This phase is concerned with putting the developed solution into action, also considered the commissioning of the solution. For the social engineer it would be exploiting the established relationship and eliciting the information. This entails using effective profiling techniques to assist in rapport building to facilitate information elicitation without compromising the relationship. A useful reference for this is presented by (Korem, 2012) where techniques are presented which, if at the very least, will enable effective communication. It is this communication, which will enable the social engineer to effectively communicate "speaking the language" of the identified source. In (Korem, 2012), it demonstrates how profiling accuracy improved to above 75% from an initial 25% through basic training in the techniques presented.

The interest for the authors in this lies in the fact that through a concerted effort the effectiveness of a social engineering attack can be reduced through altering the mental schema, a process that is influenced by the "type" of person you are (Bezuidenhout, Mouton and Venter, 2010; Mouton, Malan and Venter, 2012; Mouton, Leenen and Venter, 2015). Profiling relies on communication, either direct or indirect communication, similar to social engineering attacks.

4.5. Asses Performance

A critical step is measuring the performance of the implemented solution and evaluating the TPM in support of achieving the MoE, which in turn represents the effectiveness of the developed system commissioned and whether it is accomplishing what it is or was supposed to accomplish.

Unlike conventional systems that have been implemented of which their performance is now assessed to ensure alignment to the intended goal, the social engineering attack ends with the information obtained in terms of Measure of Performance. Consider a conversation where once you have discovered what you wanted to, you simply walk away. This phase is thus focussed on maintaining socially accepted constructs and transitioning into an accepted exit point from the conversation, i.e. this phase focussed on a smooth transition from the point of getting what you want to walking away. Debriefing the target is of utmost importance, to ensure the target does not suspect that they were compromised and thus not feel that they need to perform corrective actions.

4.6. Re-evaluation

It is necessary to note that throughout the life cycle re-evaluation of the life cycle phases, as well as the validation of the final system, occurs with multiple feedback points. This occurs concurrently with the life cycle phases. This concurrent "sanity check" has a baselining effect which ensures the right problem is being solved in the right way. For the SEAF, this same activity occurs from the *Debrief* to *Preparation* phase. Unlike the systems approach where the re-evaluation occurs concurrently, the evaluation cycle with the SEAF is a little extended in that it iterates the *Preparation*, *Develop Relationship*, *Exploit Relationship* and *Debrief* phases. This actually aligns very succinctly with the systems engineering approach in one of two ways:

a) This cycle can be viewed as investigate alternatives and model the solution in itself, prior to launching the exploit, OR
b) It happens as indicated with the Preparation to Debrief phases occurring, but then cycling back and iterated until the outcome is satisfactory.

The first option is good representation of a designed-to-work-first-time approach, rather than a try-it-and-see approach.

5. Conclusion

Social engineering is a very pervasive and material threat to information security systems. There are methods for "patching" humans to be more aware of social engineering attacks and therefore more resilient in thwarting such attacks. That said, systems engineering approaches offer a more methodic approach to developing systems, processes, technologies, and products which solve customer needs. The need to obtain information via social engineering attacks can be viewed as a need to be solved by a process resulting in obtaining said information. This paper has

indicated that one can map the social engineering attack framework to the systems engineering life cycle.

Systems engineering is about coping with complexity and assist with avoiding omissions and invalid assumptions, helping to manage real world changing issues, like a social engineering scenario, and produce the most efficient, economic and robust solutions that are understood and repeatable. The alignment of the SEAF to the systems engineering life cycle puts us in a position to exploit the extensive techniques, within the systems engineering domain, that have been developed to support its underlying principles and processes. By aligning the SEAF to systems engineering life cycle, access to those techniques are granted allowing for a security professional to cope with the complexities of social engineering attacks in a defined and quantitative manner.

As a first step toward that goal, this paper establishes the link between the social engineering attack framework and information security using systems engineering principles. The premise of an attacker representing the customer with a goal of obtaining information to gain access to a system is considered as a need to be addressed using the systems engineering life cycle.

The link between the social engineering attack framework and the systems engineering life cycle allows one to have a more methodical view of the social engineering attack. In addition, this now gives access to creative ways of using systems engineering techniques in facilitating of further development of counter processes to aid information security professionals in achieving information assurance. The mapping also provides the opportunity for introducing any process, being it to obtain information as discussed here, or the protection of information as is the final intent, into an enterprise using a best practice framework like Cobit5. It essentially packages the social engineering attack into a palatable format easily introduced into enterprise architecture design and information assurance systems design.

This now lays the groundwork for the same process to be followed to develop a counter process or system. Attempts have been made at the development of attack detection models and other thwarting techniques for social engineering attacks (Hoeschele and Rogers, 2005; Sandouka, Cullen and Mann, 2009; Bezuidenhout, Mouton and Venter, 2010; Bhakta and Harris, 2015; Mouton, Leenen and Venter, 2015). The most prominent of these models is the social engineering attack detection model (SEADM) (Bezuidenhout, Mouton and Venter, 2010; Mouton, Leenen and Venter, 2015; Mouton *et al.*, 2017). The purpose of introducing the SEADM to the design process is due to the fact that the SEADM allows the human to alter their mental schema in order to better their decision making capability. Breaking the mental schema allows the human to utilise cognitive processing in order to make decisions and no longer a mental schema which automates the decision making process. This has been proven to have a material impact on the success rate of social engineering attacks (Mouton, Teixeira and Meyer, 2017). One can now use the link between the social engineering attack framework and the systems engineering life

cycle to further improve on the SEADM by addressing at which process each step of the SEADM can be applied, and how to implement it.

6. References

Bezuidenhout, M., Mouton, F. and Venter, H. S. (2010) 'Social engineering attack detection model: SEADM', in *Information Security for South Africa*. Johannesburg, South Africa, pp. 1–8. doi: 10.1109/ISSA.2010.5588500.

Bhakta, R. and Harris, I. G. (2015) 'Semantic analysis of dialogs to detect social engineering attacks', in *Semantic Computing (ICSC), 2015 IEEE International Conference on*, pp. 424–427. doi: 10.1109/ICOSC.2015.7050843.

Burge Hughes Walsh Limited (2017) *Systems Engineering Tools & Techniques*, *Electronic*. Available at: http://www.burgehugheswalsh.co.uk/Uploaded/1/Documents/Tool-Map-Banner.pdf (Accessed: 23 August 2017).

Cialdini, R. (2007) *Influence: The Psychology of Persuasion*. Edited by R. Cialdini. HarperCollins Publishers.

Cialdini, R. B. (1987) 'Compliance principles of compliance professionals: Psychologists of necessity', in *Social Influence: The Ontario Symposium*, pp. 165–184.

Harley, D. (1998) 'Re-Floating the Titanic: Dealing with Social Engineering Attacks', in *European Institute for Computer Antivirus Research*, pp. 4–29.

Hoeschele, M. D. and Rogers, M. K. (2005) 'Detecting Social Engineering', in Pollitt, M. and Shenoi, S. (eds) *Advances in Digital Forensics*. Springer US (IFIP: The International Federation for Information Processing), pp. 67–77. doi: 10.1007/0-387-31163-7_6.

Honour, E. C. (1998) 'INCOSE: History of the International Council on Systems Engineering', *Systems Engineering*, 1(1), pp. 4–13. doi: 10.1002/(SICI)1520-6858(1998)1:1<4::AID-SYS2>3.0.CO;2-M.

Ivaturi, K. and Janczewski, L. (2011) 'A Taxonomy for Social Engineering attacks', in Grant, G. (ed.) *International Conference on Information Resources Management*, pp. 1–12.

Jacobs, S. (2015) *Engineering Information Security: The Application of Systems Engineering Concepts to Achieve Information Security Assurance*. 2nd edn. Hoboken, NJ: John Wiley & Sons.

Johns Hopkins Whiting School of Engineering (2013) *Explaining KPPs, KSAs, MOEs, and MOPs*. Available at: https://ep.jhu.edu/about-us/news-and-media/explaining-kpps-ksas-moes-and-mops (Accessed: 21 August 2017).

Korem, D. (2012) *The Art of Profiling: Reading People Right the First Time*. International Focus Press. Available at: https://books.google.co.za/books?id=A6CjMAEACAAJ.

Kossiakoff, W. *et al.* (2011) *Systems Engineering: Principles and Practice*. 2nd Editio. New Jersey: John Wiley & Sons.

Laribee, L. (2006) *Development of methodical social engineering taxonomy project*. Naval Postgraduate School.

Mitnick, K. D. and Simon, W. L. (2002) *The art of deception: controlling the human element of security*. Edited by W. Publishing. Indianapolis: Wiley Publishing.

Mitnick, K. D. and Simon, W. L. (2005) *The art of intrusion: the real stories behind the exploits of hackers, intruders and deceivers.* Edited by W. Publishing. Indianapolis: Wiley Publishing.

Mohd Foozy, F. *et al.* (2011) 'Generic Taxonomy of Social Engineering Attack', in *Malaysian Technical Universities International Conference on Engineering & Technology*. Batu Pahat, Johor, pp. 1–7.

Mouton, F., Malan, M. M., *et al.* (2014) 'Social engineering attack framework', in *Information Security for South Africa*. Johannesburg, South Africa, pp. 1–9. doi: 10.1109/ISSA.2014.6950510.

Mouton, F., Leenen, L., *et al.* (2014) 'Towards an Ontological Model Defining the Social Engineering Domain', in Kimppa, K. et al. (eds) *ICT and Society*. Springer Berlin Heidelberg (IFIP Advances in Information and Communication Technology), pp. 266–279. doi: 10.1007/978-3-662-44208-1_22.

Mouton, F. *et al.* (2015) 'Necessity for ethics in social engineering research', *Computers & Security*, 55, pp. 114–127. doi: http://dx.doi.org/10.1016/j.cose.2015.09.001.

Mouton, F. *et al.* (2017) 'Underlying Finite State Machine for the Social Engineering Attack Detection Model', in *Information Security for South Africa*. Johannesburg, South Africa, pp. 98–105.

Mouton, F., Leenen, L. and Venter, H. S. (2015) 'Social Engineering Attack Detection Model: SEADMv2', in *International Conference on Cyberworlds (CW)*. Visby, Sweden, pp. 216–223. doi: 10.1109/CW.2015.52.

Mouton, F., Leenen, L. and Venter, H. S. (2016) 'Social engineering attack examples, templates and scenarios', *Computers & Security*, 59, pp. 186–209. doi: http://dx.doi.org/10.1016/j.cose.2016.03.004.

Mouton, F., Malan, M. M. and Venter, H. S. (2012) 'Development of cognitive functioning psychological measures for the SEADM', in *Human Aspects of Information Security & Assurance*. Crete, Greece, pp. 40–51.

Mouton, F., Malan, M. M. and Venter, H. S. (2013) 'Social engineering from a normative ethics perspective', in *Information Security for South Africa*. Johannesburg, South Africa, pp. 1–8. doi: 10.1109/ISSA.2013.6641064.

Mouton, F., Teixeira, M. and Meyer, T. (2017) 'Benchmarking a Mobile Implementation of the Social Engineering Prevention Training Tool', in *Information Security for South Africa*. Johannesburg, South Africa, pp. 106–116.

Sandouka, H., Cullen, A. J. and Mann, I. (2009) 'Social Engineering Detection Using Neural Networks', in *CyberWorlds, 2009. CW '09. International Conference on*, pp. 273–278. doi: 10.1109/CW.2009.59.

Smith, S. and Brown, D. (2014) 'Why do Systems Engineering? - SE101', *INCOSE Transportation*, p. 2.

Stewart, J.M, Chapple, M., Gibson, D. (2013) *CISSP*, *ISC2 Official Study Guide*. doi: 10.1017/CBO9781107415324.004.

Tetri, P. and Vuorinen, J. (2013) 'Dissecting social engineering', *Behaviour & Information Technology*, 32(10), pp. 1014–1023.