Phishing Ontology for Malicious Emails

Yamkela Xaka
The University of the Western Cape,
Robert Sobukwe Road, Bellville,
Cape Town
3538718@myuwc.ac.za

ABSTRACT

This report aims to address the dangers of social engineering, to be more specific phishing email attacks, by presenting the design of an ontology-oriented approach to detect if an email is a normal email or a phishing email. This project will deal with the issue of phishing by presenting an ontology to build the knowledge base of phishing email attacks. Building an ontology on phishing emails is a new and exciting approach after seeing many people do not realize that they are being targeted or attacked daily. With the development of IT technology, in modern society, IT security has become an important reliance on information security and privacy [1]. Although there are several security approaches, such as firewalls and intrusion detection systems, which can be used to protect the machines from being attacked, there is a lack of widely accepted mechanisms to prevent machine users from fraud [2]. For example, companies lose huge amounts of money due to phishing email attacks. The author choose ontology instead of machine learning because the ability to work with unstructured, semi-structured, or structured data formats means ontologies can connect and qualify data without any need for standardization. Ontology streamlines the process of identifying core concepts, improving classification results to collate critical information [1]. As a result, data can be found and analyzed faster using Machine learning. The reason stems from a lack of understanding about how to utilize large volumes of data. ML projects frequently fail because of problems with the architecture of the information rather than the quantity. Ontologies can make a big difference.

KEYWORDS

Cybersecurity, ontology engineering, semantic web, social engineering, OWL

1 Introduction

With many gadgets, new powerful computers, and unsecured business machines, phishing became more popular because of the attacks that are happening frequently [1]. Phishing emails mainly rely on social engineering. Social engineering is described as the art of manipulating people, so they give up confidential information [3]. According to Dudley, Tonia [5], if companies or victims act slowly against these attacks, that will lead to a profound effect on their personal lives and their companies, as the companies will lose huge amounts of money due to attacks [9]. Once companies get attacked, they lose trust in internet banking and e-commerce transactions by falling prey to disclosing confidential account details to the phisher's devious tricks [8]. The consequences of such phishing attacks will have a disastrous impact on the corporate and banking sectors and businesses that rely heavily on the internet, as the e-clients would lose their trust in the services owing to their vulnerabilities [6].

The project aims to address the issues of phishing emails and come up will solutions to prevent it by building an ontology. In philosophy, ontology refers to the study of being. In information science, ontology describes a set of concepts and categories in a subject area or domain that shows their properties and the relationships between them. An ontology is a part of Artificial Intelligence (AI) that uses knowledge of the domain and logic to identify if an email is likely to be a phishing email or not [2]. The ontology will provide the difference between normal email and phishing emails so that the users will know when they are dealing with phishing email attacks.

2. Background

2.1 Social engineering

Social engineering is the art of manipulating people, so they give up confidential information about their company or personal information [9]. The types of information these criminals are seeking can vary, but when individuals are targeted, the criminals are usually trying to trick you into giving them your passwords or bank information or access your companies' systems to secretly install malicious software–that will give them access to the passwords of the company or passwords of personal information or sometimes bank information as well as giving them control over your computer [6]. Social engineering attacks include physical, social, and technical aspects, which are used in different stages of the actual attack [4]. In this section, the aim is to list the different types of social engineering attacks that the attackers use to get unauthorized access to private information. Below are the types of social engineering attacks:

- Physical approaches
- Social approaches

- Reverse social engineering
- Technical approaches

Socio-technical approaches

2.2 Phishing

Phishing is a cyber-criminal technique exploiting social engineering to lure target users providing sensitive information such as bank accounts [3]. The report focuses on phishing emails characterized by the fact that phishers design a fake email targeting a specific group of individuals. Phishers mainly follow four steps [9].

- Designing and dissemination of fake emails during which attackers design fake emails and flood them through messaging means to the targeted users.
- Visiting malicious websites during which the victim is redirected to the phisher website.
- Releasing sensitive information during which the user is persuaded to disclose confidential information.
- Gathering of sensitive information during which a user's confidential information is sent to attackers.

The phisher begins with a sharp investigation of the target. It can be done either through social media or by collecting personal traces left on the Web. Once the collection is sufficient, the phisher prepares the phishing attack by minutely designing mail contents and falsifying the mail header. In Figure 2[4], a more specific and complete phishing process is presented. It includes collecting information on the target, representation of elements of incitement, and phisher exploits.

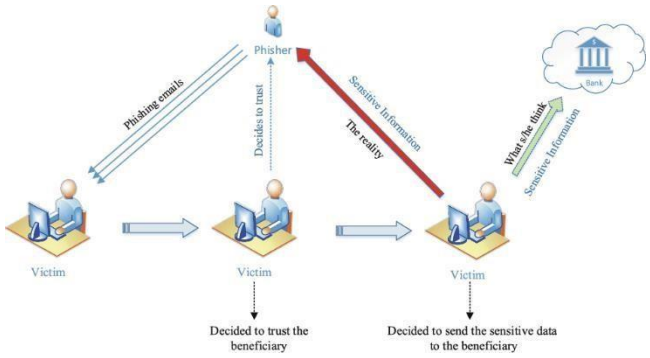

Figure 1. The scenario of email phishing [6].

2.3 Semantic Technology

The word "semantic" refers to meaning in language. The goal of semantic technology is to help machines understand data. To enable the encoding of semantics with the data, well-known technologies are RDF and OWL. These technologies formally represent the meaning involved in information. Semantics gives meaning to entities and the relationships between them. It provides algorithms to compute results that will combine Syntax, Semantics, Queries, and Reasoning to develop an ontology [6]. Encodes meanings separately from Metadata, content files, documents, web resources Services and it is mostly controlled by vocabulary [14]. Semantics enables machines as well as people to understand, share, and reason with each other.

2.3.1 Reasoning

Humans process knowledge by reasoning so that they reach conclusions [12]. Analogously, a computer processes the knowledge stored in the knowledge base (KB) by concluding it, i.e., by deriving new statements that follow from the given ones. Reasoners check for logical contradictions and consistency of the ontology model [11]. A reasoner can invalidate ontologies in different ways:

- An ontology can be detected as inconsistent meaning that there is no possible interpretation of the ontology.
- An ontology is unsatisfiable when there is a possible interpretation of that ontology.

2.4 Ontologies

The notion of ontology was first mentioned by John McCarthy in the field of artificial intelligence (AI) [9]. An ontology is simply the set of concepts, relations, attributes, and hierarchies existing in a domain [5][9]. Ontologies have the following benefits:

- It provides a common understanding of the information structure between people and software manufacturers
- It renders interoperability between systems
- It facilitates the exchange of knowledge between systems

It facilitates the reuse of knowledge on a domain by creating and maintaining a reusable knowledge base.

2.5 Description logic

Ontologies represent knowledge using conceptual knowledge specified by a language such as Description Logic, in an exploitable form by an information system [5]. Description logics extend frame-based systems by expressing definitions of classes and relations [15]. Several description logic languages exist and differ in language expressiveness. DL languages provide formal semantics and can, therefore, represent the knowledge of an application domain structurally and formally. DL is used in this research because of several reasons. These reasons include:

- Description logics have become a major knowledge representation paradigm, for use within the semantic Web. It can be applied in cybersecurity

- DL is decidable, i.e., given a concept, it is possible to determine if this definition is consistent with others. Also given an instance definition, it can be decided which is the concept definition that most fits it.
- DL has sound and complete reasoning mechanisms which guarantee the accuracy and reliability of the results.
- A wide range of logic has been developed till now, from very simple (no disjunction, no full negation) to very expressive, so logic satisfying research needs could be selected in a minimum computational complexity.
- Modern DL reasoning engines are quite efficient when providing results.

DLs rely on three notions such as concept, role, and individual [16]. Concepts correspond to classes of individuals, roles are relationships between these individuals, and individuals correspond to individual concepts. In a descriptive logic knowledge base, there are two components:
TBox and a ABox. The TBox contains all the axioms defining the concepts of the domain. ABox contains assertions about individuals, specifying their classes and their attributes.

3 Related works

This section describes various orientations provided in literature to deal with phishing. This category includes various approaches. Authors rely on content and metadata to profile phishing traces. Similarity measures are designed to compare normal web pages and malicious pages [12]. Machine learning approaches were used to the classification of phishing emails compared with those of the ontology [13]. Knowledge representation-based solutions Some authors oriented their research towards formalizing objects, entities, and their relationships in the cybersecurity area. Ellison et al[1]. formalize description logics to represent and reason knowledge in digital forensics and digital security. Scarpato et al[1]. couple description logics to Web Ontology Language (OWL) and SPARQL queryable to represent the information needed to generate the Reachability Matrix within the Open Systems Interconnection (OSI) protocols. They deduce ontology for cybersecurity.

Lwakatare mentioned in research that social engineering is an attack that comes as a smooth communicating with the victim to reveal valuable information to bypass the security perimeter in front of the information-related resources. This thesis mentioned that the novel taxonomy of social engineering attacks was proposed to understand the concept of social engineering and gain insight into the representative social engineering attacks by applying the taxonomy to them. Furthermore, a multi-layer social engineering defense model is proposed to deal with the threats brought by social engineering attacks. In each layer, different mechanisms are proposed respectively to facilitate the defense against various social engineering techniques to effectively protect information-related resources and guarantee IT security [1].

Also, Mouton et al. [17] proposed an ontological model to define the social engineering domain and another recent novel taxonomy of SE attack was proposed by Heartfield and Loukas [18]. It adopts three distinct control stages orchestration, exploitation, and execution, as the basic categories of the taxonomy. For each stage, it poses questions that can help develop the technical protection mechanisms. The answers to these questions compose the corresponding categories, which consequently establish the whole taxonomy. The orchestration consists of target type (target of choice or opportunity), attack mode (manual or automated), and attack approach (software, hardware without software, or hardware with software). The exploitation includes the deception vector (cosmetic, behavior, or hybrid) and the manipulation interface (user interface or programmatic interface). The execution is comprised of execution steps (single or multiple) and attack persistence (one-off or continual). Furthermore, this taxonomy depicts several mutual-exclusive subcategories whose characteristics should be considered for developing technical protection mechanisms. The taxonomy is not exhaustive and can be expanded based on the three main categories. Also, it is evaluated by being applied to 30 different attacks observed in the wild, which is aimed to help develop the technical protection mechanisms. However, the taxonomy adopts the definition of the three distinct control stages of orchestration, exploitation, and execution as suggested by CESG [18], which aims to describe common cyber-attacks instead of social engineering attacks. Hence, the categories of this taxonomy are more related to common cyber-attacks than to social engineering attacks, which has some specific concerns that should be considered.

4. Methodology

4.1 Existing Methodologies

This section contains an overview of existing methodologies for ontology development. Many ontologies have been developed in different ways. Most common are older Methontology, On-ToKnowledge, NeON, Melting Point methodologies, OntoSpec, DiDOn, and TDDonto, and the older “Ontology Development 101” (OD101) [2]. All these ontology methodologies are not simply interchangeable each of these uses its techniques. Besides that, some are older, but they can be distinguished in the core approach, being between Micro-level ontology authoring against a Macrolevel systems-view of ontology development. Isolated, single, stand-alone, ontology development against collaborative development of

ontologies and ontology networks [16]. Micro-level methodologies focus on the viewpoint of the details emphasizing formalization aspects, which goes into ontology authoring, for it is about writing down the actual axioms and design choices that may even be driven by the language [8].

For the project, the researcher will be using NeON ontology development because it provides a variety of pathways for developing ontologies. NeON methodology provides

- glossary processes and activities involved in developing an ontology
- it provides life-cycle models
- it provides guidelines for different processes and activities which are described it provides functionality in terms of goals, inputs, outputs, and relevant constraints
- it provides procedurally, utilizing workflow specifications.

The proposed model consists of high-level steps that can be summarized as follows:

- identify purpose
- uses and users for the ontology to be developed
- identify the set of requirements that the ontology to be developed should fulfill.

The purpose of the email characteristics is to enable us to run the reasoner so that it can put entities in all the classes where they belong. When the new email is received the reasoner will run it and then put it on either normal email subclass or phishing email subclass. Instead of using machine learning to detect email, the proposed project will use semantic technology.

4.2 Development

Phishing emails in previous research were identified using Machine learning [3]. Machine learning is good for many applications, but it is not doing very well in terms of classifying phishing emails attacks, researchers are still trying to improve machine learning for it to be accurate in classifying phishing emails [3]. This honors project is a new and complex approach that has never been implemented before, the author is still researching how to detect phishing emails without using machine learning. We saw that ML cannot provide a reason when an email has been classified but an ontology you can.

4.3 Development of rules

The rules in the ontology can be used by an automated reasoner to make inferences. The rules will be developed using the Semantic Web Rule Language (SWRL) in the Protégé editor [11]. They will be defined using emails. The rules are written using Boolean connectives, quantifiers, objects, and predicates [3].

- *Boolean connective*: is a word or symbol used to connect two or more sentences grammatically. Examples of quantifiers are listed below:

  And (intersection): ∩
  Or (union): ∪
  Not (negation): ¬
  Only if: →

- *Objects*: In this case, objects will be a concept or individuals.
- *Quantifier:* an expression that indicates the scope of a symbol or word attached to it.
  All:∀
  Some (Existential): ∃

- *Predicate*: something which is affirmed or denied concerning an argument of a proposition e.g. categories or relationships.

SWRL

A Semantic Web Rule Language based on a combination of the OWL DL and OWL Lite sublanguages of the OWL Web Ontology Language with the Unary/Binary Datalog RuleML sublanguages of the Rule Markup Language. It thus enables Horn-like rules to be combined with an OWL knowledge base[2].

Examples of SWRL rules are as follows:

Constraint: Email that contains a grammatic error may indicate that it is a phishing email.

- $\exists x(\text{email}(x) \cap \text{contain (grammatic error)}) \rightarrow$ phishing email(x)

Constraint: Some emails don't contain a grammatical error.

- $\exists x(\text{email}(x) \cap (\neg\text{contain}(x), \text{grammatic}))) \rightarrow \text{email}$

The first rule above can be used when the attacker sends an email with an anagrammatic error, the data set of an email will be loaded into the ontology, and it will be checked if it contains one of the rules defined. The automated reasoner will use the defined rules to do the classification. The rules created describe classes and subclasses. The second rule can be used when the sender sends an email without any grammatical errors. The rules are the main component in ontology development because the rules are used by the reasoner to make an inference. Using the rules, the author will be able to

differentiate when the email is normal or when it is a phishing email.

Below are features that will be built into the ontology.

Phishing email features

- An email that has an attachment with an executable file is likely to be a phishing email.
- An email with a suspicious domain name is likely to be a phishing email.
- An email that contains spelling errors/ bad grammar is likely to be a phishing email.
- An email with a hyperlink's URL that doesn't seem correct or doesn't match the context of the email is likely to be a phishing email.
- Request to click on a link to confirm personal details.
- Generic greeting (hi dear, dear customer)
- An unknown sender addresses you with the first name

Normal email

- If the email header is legitimate
- Sender's email address listed at the top of your email client
- The link sent is legitimate

Rules to assess the features are checking the following:

- Domain name
- The attachment for executable files, etc.
- The email header
- The email address of the email sender
- Multiple spelling errors
- Generic greetings
- Sense of agency
- Hyperlink URLs seem correct or not
- Asks for personal details

To ensure that the reasoner classifies the emails sent there are rules that must be created by the author. These rules are based on the emails sent to the victim. Below are the pictures associated with the rules created

Figure 2. An email with information on the falsified header

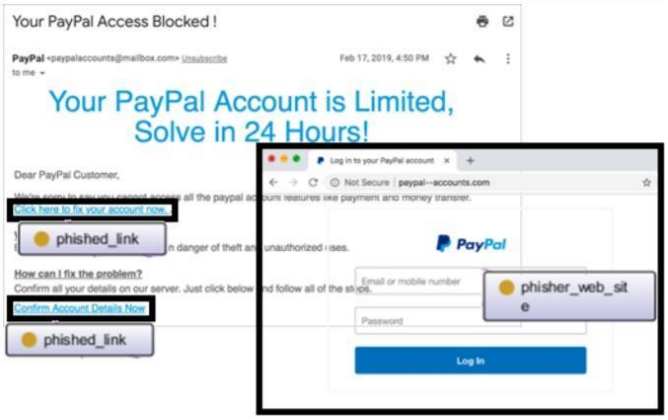

Figure 3. Email with adapted text and phished link

Below is the table that shows the hierarchy of the email ontology, it shows how it is formatted and how the classes and subclasses are created[6].

Table 2: Concepts hierarchy

Main classes	Sub-classes
Thing → Email	• Normal Email • Phishing Email • Low risk • Medium risk • High risk
Content/Email Body	• Normal Spelling • Spelling Error/Grammar
Domain Name	• Genuine Domain Name • Fake Domain Name
Link	• Genuine Link • Phished Link • Phisher's site o Running of malicious script o Collection of personal information
Attachment	• PDF • ZIP • DOCX • EXECUTABLE FILE
Email Address	• Genuine Address • Fake Address
EmailSenderAddress	• Sender

Since most of the normal emails have attachments that are fine the author's rules will include combinations of features as either low probability, medium probability, or high probability of risk, all these will be subclasses of the phishing email. Properties of an email such as the use of bad grammar, a fake domain, sense of agency, etc. will be fed as an input to the ontology.

The ontology is intended to be yet developed as part of a larger system in which analysis will be done to identify the properties of an email instance.
The reasoner is applied to the knowledge base and makes instances. They will allow the ontology to detect if the email sent is phishing or normal email.

Figure 7 illustrates the model for phishing emails:

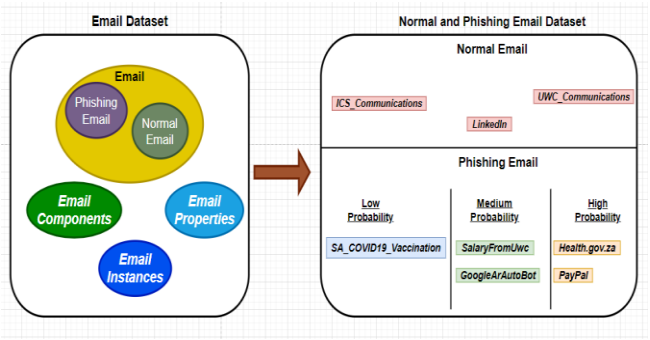

Figure 7: Phishing email model Implementation

The figure above shows the email dataset and the email dataset the reasoner has been executed. The author added instances of email which will be used in the classification field to detect if the email is phishing or not and if it's a phishing email in which category. The author will use an ontology editor the uses a plug-in reasoner to detect if the email is normal or harmful.

4.4 Implementation

Protégé 5.5.0 is used to build the ontology through Ontology Web Language (OWL) [13]. The project adopts OWL DL because, firstly, the OWL DL makes it possible to express multiple cardinalities and, on the other hand, the other languages are unsatisfactory or more complex. The ontology will be developed using the following phases:

- The creation of classes and subclasses
- The creation of properties.

- The creation of individuals

The ontology model consists of:

Main class

- Owl thing

Classes

- Email
  - NormalEmail
  - PhishingEmail
    - Low Probability
    - Medium Probability
    - High Probability
- EmailAttachments
  - Exe
  - PDF
  - Audio
  - Image
  - Image
  - Zip
  - xlsxFile
- Email Components
  - EmailBody
  - EmailDomain
  - EmailSenderAddress
  - EmailSubject
- Email Properties
  - Grammar
  - Link
- Email Instances
  - PayPal
  - GooglrArAutoBot
  - SA_COVID19Vaccine
  - SalarFromUWC
  - Uwc_Communications
  - ICSCommunications
  - Health.gov.za

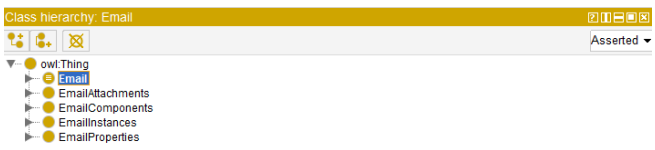

Figure 9: Description of Classes

Object properties

- hasEmailAddress
- hasAttachment
- hasSubject
- hasEmailAddress
- hasEmailBody
- hasEmailSenderAddress
- hasLink
- hasGrammer

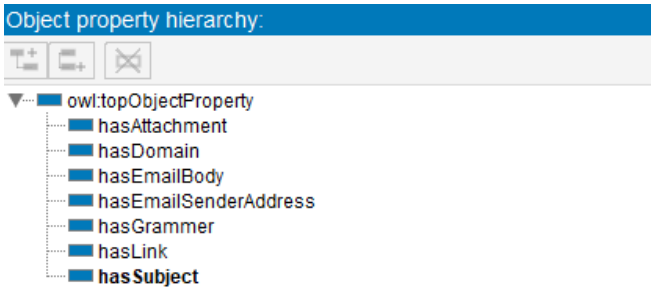

Figure 10: Description of Object Properties

The membership rules of the class “email” are shown in the figure below (Fig. 11). An example of primitive class and defined class.

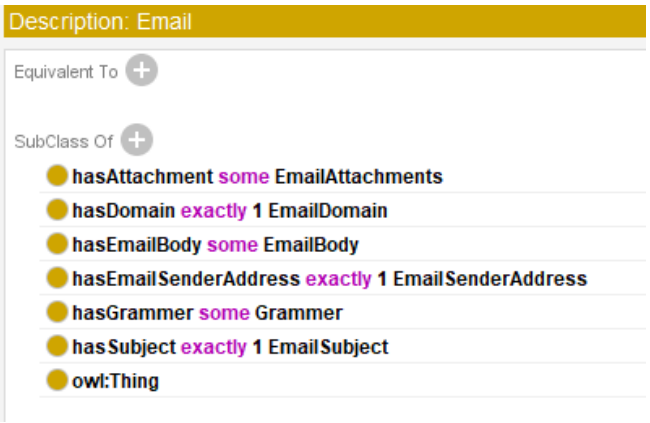

Figure 11: Description of Primitive classes

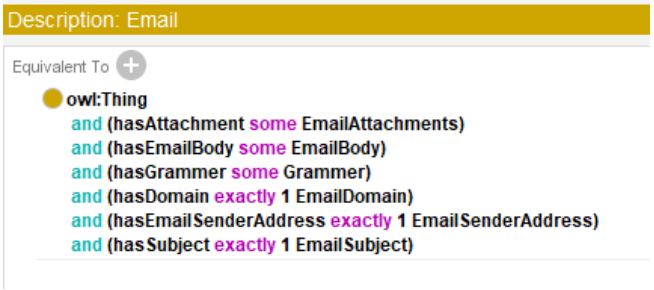

Figure 11: Description of Defined classes

Before the reasoner runs, we see the asserted class hierarchy and after the reasoner is executed, we can change

from asserted to the inferred class hierarchy so that we can see the results. The inferred class hierarchy shows the results of what is been developed. In the case of this project, the reasoner classifies all the phishing emails to the phishing class and the normal emails to the normal class. Once the reasoner classifies normal email and phishing emails, it then goes further to classify them according to their harmfulness to the victim. For example, phishing classes have the subclass of low probability, medium probability, and high probability. In this case, the reasoner decides according to the rules that were set by the author to say that where does the email fall.

The class hierarchy of both asserted and inferred are shown in **Figure 12** and **Figure 13** below:

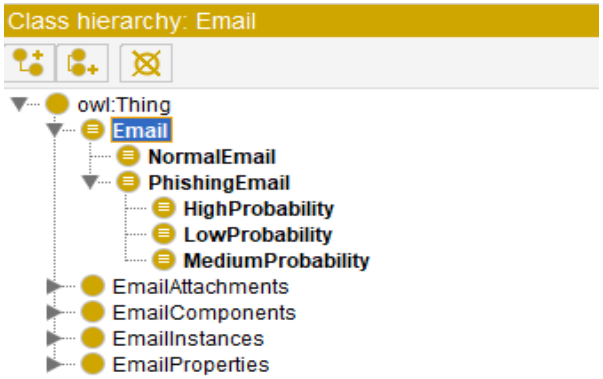

Figure 12: Asserted Email class hierarchy.

Initially, phishing email and the normal email contain no subclasses (see the figure above), but when the reasoner has been executed the phishing email and normal email have subclass. On the phishing email class, the reasoner places each email dataset on either low, medium, or high probability depending on the rules that were set.

The figure below shows the results of the inferred class hierarchy.

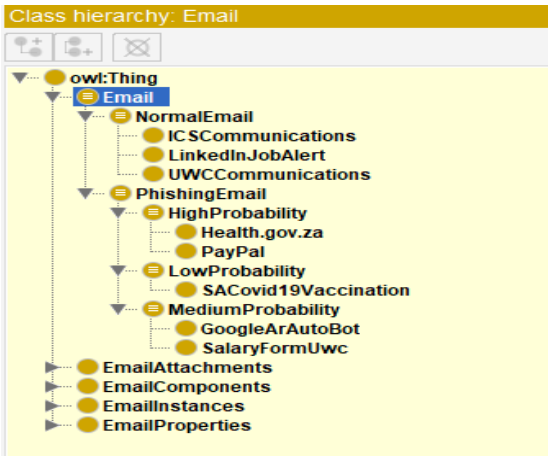

Figure 13: Inferred Email class hierarchy.

On the inferred, the reasoner is running to classify the emails, accordingly, looking at the figure above the reasoner classified all the emails sent accordingly to the classes that they belong to.

Below are the results of phishing emails and Normal emails generated by the DL query

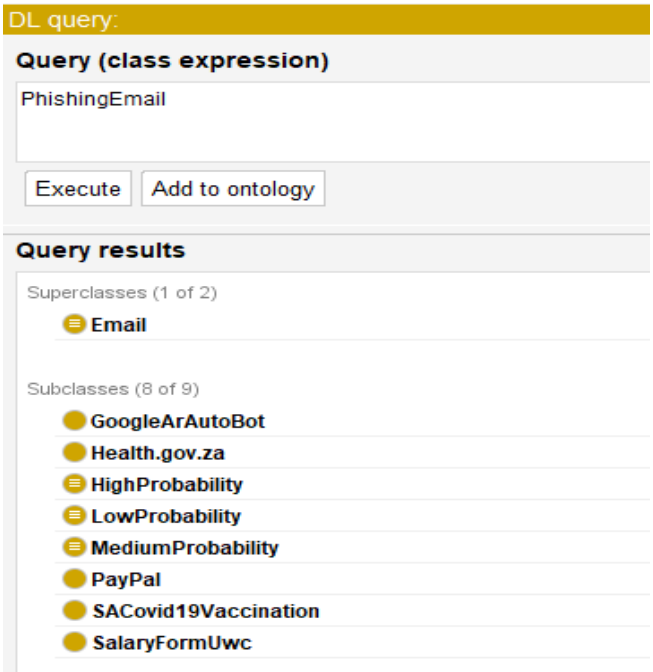

Figure 14: DL queries for Phishing Emails

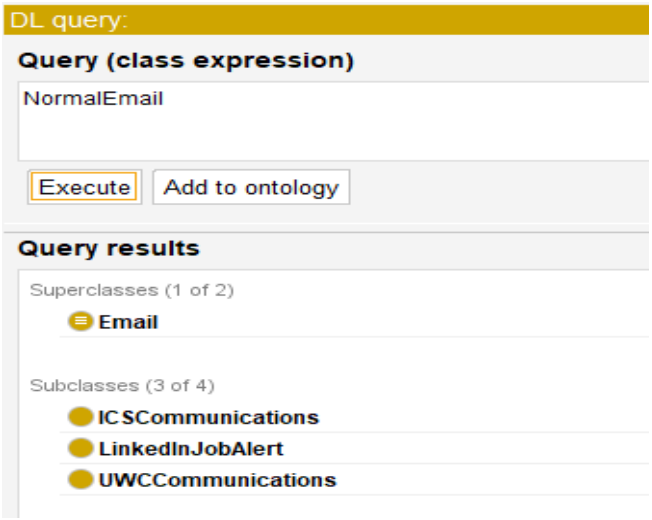

Figure 15: DL queries for Normal Emails

Conclusion

In this paper, I defined social engineering attacks on people and businesses. I believe that a detailed understanding of the attack vectors is required to develop efficient countermeasures and protect businesses and people from social engineering attacks more specifically in phishing email attacks. To facilitate this, I introduced an ontology that will be able to solve phishing email attacks that have become uncontrollable in our society. The ontology aims to detect if the email sent is a normal email or phishing email. This is to feed the ontology with a sample of emails for the reasoner to detect. In this work, I exploited description logics as the main support to design ontology to represent phishing knowledge. The knowledge base includes representing key elements to describe a generic email phishing process. An ontology scheme is proposed based on attacks made by the attackers.

References

[1] A. Vedeshin, “Contributions Of Understanding And Defending Against Social Engineering Attacks,” 2016.

[2] “The 5 most common types of a phishing attack - IT Governance Blog En.” [Online]. Available: https://www.itgovernance.eu/blog/en/the-5mostcommon-types-of-phishing-attack. [Accessed: 06Jun-2020].

[3] “When Phishing Starts from the Inside -.” [Online]. Available: https://blog.trendmicro.com/phishingstarts-inside/. [Accessed: 06-Jun-2020].

[4] F. Mouton, L. Leenen, M. M. Malan, and H. S. Venter, “Towards an Ontological Model Defining,” *IFIP Int. Conf. Hum. Choice Comput.*, pp. 266–279, 2014.

[5] “Semantic Web Technologies – jobontology.” [Online]. Available: https://www.jobontology.com/semanticwebtechnologies/. [Accessed: 26-Feb-2020].

[6] C. M. Keet, “An introduction to ontology,” *Choice Rev. Online*, vol. 51, no. 04, pp. 51-2000-51–2000, 2013, doi: 10.5860/choice.51-2000.

[7] A. Vedeshin, “Contributions Of Understanding And Defending Against Social Engineering Attacks,” 2016.

[8] K. Krombholz, H. Hobel, M. Huber, and E. Weippl, “Advanced social engineering attacks,” *J. Inf. Secure. Appl.*, 2015, doi: 10.1016/j.jisa.2014.09.005.

[9] “What is Cyber Security? Definition of Cyber Security, Cyber Security Meaning - The Economic Times.” [Online]. Available: https://economictimes.indiatimes.com/definition/cyber-security. [Accessed: 26-Feb-2020].

[10] *Ramzan, Zulfikar (2010). "Phishing attacks and countermeasures". In Stamp, Mark; Stavroulakis, Peter (eds.). Handbook of Information and Communication Security. Springer. ISBN 9783642-04117-4.*

[11] Van der Merwe, A J, Loock, M, Dabrowski, M. (2005), Characteristics and Responsibilities involved in a Phishing Attack, Winter International Symposium on Information and Communication Technologies, Cape Town, January 2005.

[12] Jump up to [a] [b] *"Landing another blow against email phishing (Google Online Security Blog)". Retrieved June 21, 2012.*

[13] *Dudley, Tonia. "Stop That Phish". SANS.org. Retrieved 6 November 2019.*

[14] *"What is Phishing?". 2016-08-14.*

[15] Jump up to [a] [b] *"Safe Browsing (Google Online Security Blog)". Retrieved June 21, 2012.*

[16] *Jøsang, Audun; et al. (2007). "Security Usability Principles for Vulnerability*

[17] Francois Mouton, Louise Leenen, and H.S. Venter. Social engineering attack examples, templates, and scenarios. Computers & Security, 59:186 – 209, 2016.

[18] Ryan Heartfield and George Loukas. A taxonomy of attacks and a survey of defense mechanisms for semantic social engineering attacks. ACM Comput. Surv., 48(3):37:1–37:39, December 2015.