Improving User Awareness of Social Engineering

M.Newbould and S.M.Furnell

Centre for Information Security and Network Research,
University of Plymouth, Plymouth, United Kingdom
e-mail: info@cscan.org

Abstract

With online social engineering currently a major problem, it is vital to educate the potential victims of these attacks – everyday Internet users. This paper looks specifically at how a board game was developed and tested to try and educate the user in a more interactive way, with results suggesting that this solution does increase awareness of social engineering with nobody scoring under 55% whilst playing the game, and 86% of users feeling they had improved their knowledge on the subject.

Keywords

Social Engineering, Phishing, Advance Fee Fraud, Spam, Game Development

1 Introduction

This paper summarises a project that was undertaken to improve user awareness of online based social engineering attacks. With this main aim, a range of potential solutions were examined, with research taking place to see what educational solutions already existed. It was found that whilst a lot of literature was available on the Internet, there were fewer solutions that were aimed at users with less technical ability and that involved a higher level of interaction as well as being educational.

It was decided to pursue this path further and ultimately produce a website containing literature and a board game where the user could refer to the website literature when required, thus providing a more interesting way to learn. Rather than aim to produce a finished article which would most likely be rushed in the small development window it was decided that a testable prototype would be produced and with the aid of user feedback, could be developed further at a later date

The focus of the coming sections consists of background information on the topic area and discussion on the game prototype itself, including the reasoning behind the decisions that were made, as well as explaining the defined requirements. The implementation of the solution will follow, as well as how the game was tested on the potential audience and the results that were obtained from this.

2 Background

A social engineering attack in relation to IT is based upon attacking the person rather than hacking them. The attacker uses psychology to typically trick the user into

believing something which will usually result in them handing over money or information that can be used against them. Whilst social engineering attacks did not originate from IT, the frequency of social engineering attacks has increased significantly. With the majority of people now able to access the Internet, a goldmine has developed for attackers with millions of potential victims a single mouse click away.

There are a number of different types of social engineering attacks, and the remainder of this section will give an overview of these attacks, including looking at how they have evolved with new technology.

The most frequent type of social engineering attack is phishing (Microsoft, 2007). This typically involves an email from the attacker imitating an organisation such as banks in an attempt to trick the user into following a link and entering their details. According to research carried out by Gartner (2007), between August 2006 and August 2007 3.6 million users lost money to phishing scams in the US, resulting in a combine loss of $3.2million which shows the scale of the ever increasing problem.

Certain spam can be classed as a social engineering attacks as many offer enticements such as free images, a movie or friendship – provoking intrigue and interest from the user however the attachment is typically a Trojan, Worm or a Virus. The result of this is that a range of things could happen to the users computer such as damage to critical files, a program that steals information such as passwords or a key logger so that a hacker could keep track of everything the victim types on their keyboard.

Another form of social engineering attack is advance fee fraud, also known as the 419 scam which is where the attacker usually exploits human greed or sympathy. When exploiting greed, the attacker suggests to the victim that they will get a large amount of money for sending a small amount of money first, usually explained as a release fee, bribe or legal fee. Other forms of this attack can consist of the attacker imitating a victim of a recent natural disaster – trying to exploit the reader's sympathy. Although most recipients of the emails do not respond there is also a long list of those that do.

The perpetrators of these attacks are highly versatile when it comes to making use of new opportunities. The increase in the popularity of social networking sites has meant that targeted 'spear phishing' attacks are easier to carry out, with attackers making use of information that is freely available on these sites and targeting potential victims specifically via email, including some of the user's personal information to make the attempt look more convincing and authentic.

In August 2009 Sky News reported of attackers hacking users Facebook accounts, imitating the victim and emailing their friends, claiming that they have been the victim of a mugging and requesting some financial aid from their friends. This shows how Web 2.0 websites have opened the gates for perpetrators to move away from the traditional form of social engineering attacks into newer areas that make the attacks more difficult to identify. With attacks such as this rendering email spam filters

useless in protecting the end user, it is not viable for 100% reliance on technical protection measures – therefore education is essential.

3 Board Game Prototype

Once it was decided that an educational game was going to be the main deliverable, a decision had to be made as to what type of game this would be. There were a number of initial considerations regarding this, with early ideas consisting of a role playing game (RPG) where the user takes control of a character and navigates through an area in order to achieve the main goal which would have been centred on answering questions and gaining knowledge. Ultimately, it was decided that with a fairly small development window a more intricate game such as this would pose a greater risk to the project in terms of completing it on time.

Instead, it was decided that a board game would be developed. This was to have accompanying website literature that could be referred to should the user be unable to answer a question. This allowed for an interactive way to learn that was also a much more realistic option given the allotted time frame.

3.1. Required Technologies

The decision was made to develop the solution using an object oriented language, making use of the advantages of developing such a system in an OO language which among other things, allowed for a modular structure to the program, with a reduction in the amount of code needed to update and improve the system. This was seen as essential with the system being a prototype it was plausible for it to be further developed at a later stage. So having a program with clearly defined classes that represent real world objects, each with their own responsibilities, made the system as a whole easier for present and potentially future developers to understand and modify.

When looking at potential ways to develop the solution, it was ultimately decided to use Flash CS3 and therefore Actionscript as the development language of choice. Whilst it was accepted that other IDE's such as Adobe Flex and other languages such as Java would have allowed for a similar implementation, it was felt the advantage of being able to develop and test within the Flash IDE would be beneficial. In addition, there was a lot of learning material in books and online in terms of developing games, whereas literature for other options was less common. This was important as the author was new to web based game development.

A UML approach was chosen for the process of getting the game from the requirements phase, through analysis and design, to a point where it could be confidently implemented. UML helps the software engineer specify, visualize, and document models of software systems, including their structure and design. It consists mainly of a graphical language to represent concepts that are required in the development of an object oriented system (Bennett, 2002).

3.2. Defining the Solution

Having decided on the technologies to be used, it was necessary to define what would be required in the game. It was important that the game was fairly simplistic in terms of how it would be played as the aim of this was to educate users, not to require them to read pages of instructions informing them of how to play the game. A board game ensured that the solution would be easy to understand and therefore maximise time to educate the user whilst still having an interactive interface.

In terms of how the board would be laid out, it was opted for a generic square board, being a prototype it was important to implement a working solution, with aesthetically pleasing additions added at a later stage if time allowed.

The board was to consist of 32 squares with each one to be assigned one of the four categories upon the game starting. The knowledge gained from carrying out background research was used in deciding how to categorize the questions. Time was taken to look at each type of social engineering attack and the most common attacks were used as categories for the game. This encompassed phishing, advance fee fraud and spam, with the fourth and final category labelled ‘other’ which consisted of the less common attacks. Each one of these categories represented a colour on the game board, for example phishing was red and advance fee fraud was yellow. The number of times each colour appears on the board is pre-determined. For example, as phishing was identified as one of the most damaging form of social engineering the red square appears more frequently than the advance fee fraud questions.

In terms of the format of the questions it was decided to use a multiple choice format. This format would provide an effective way to ask the questions and allow easy answer selection compared to asking the user to manually enter the answer which would yield many problems in terms of verifying their answer. A range of other formats were considered but it was felt that there was insufficient time to implement a range of formats and it would instead be included in the testing phase as an option for improvement and therefore ascertain its popularity with the test audience.

A player's score would be tracked throughout the game. This was added to provide an extra level of motivation to the user and thus increase their knowledge and education. This was also used at the testing phase to determine the solutions success and was the first step towards a high score board for the game.

Rather than being a straight forward ‘roll and move forward’ type of game, the option to move back was added. Upon rolling, all squares were to dim, with the available squares staying lit. Whilst this offered nothing other than a minor addition to gameplay, it was felt it was adding something that could be more useful in a later development. For example, if a ‘bonus’ square was added in a future development, the option to move back may allow for the user to get there quicker. This reasoning would also be the same if there were targets to achieve, for example if they were given the task of answering a certain number of phishing questions within a set time.

3.3. Implementing the Solution

As previously stated, Actionscript was used in the Flash CS3 IDE to develop the game and implement the class structure defined by the UML diagrams. The UML analysis and design work carried out meant that the class structure and the communication between them had already been identified and this was vital in getting the solution completed on time.

Actionscript revolves around events when things such as user actions take place. When an event takes place, e.g. clicking on a board square, an event is dispatched by the system and the event listener calls an appropriate function to run. For example, when clicking a square, an event is triggering upon the user click. The event listener was registered in the Game Manager class, which was the main game function, and when the user clicks on a square, the event is triggered by the BoardCollection class and caught by the Game Manager class and the function set to call upon the event trigger is run – in this case the question manager is called. This method was implemented for all game actions, and essentially controlled the game

3.4. Game Flow

The following steps show the general game flow – how the game works from the users' perspective:

1. Upon opening the game the user is greeted with the main menu. From here they can view the instructions and click 'start game' or just click 'start game' without reading viewing the instructions.
2. The user enters their name, selects their piece, and clicks 'submit'.
3. Upon submitting their name and piece, the game begins (see figure 1). The user clicks 'start' to begin the dice animation, upon clicking 'stop' the selector stops.
4. With a number selected, all squares are dimmed, with the two possible movable squares staying lit.
5. Upon choosing a square the user's piece is moved to the chosen location and a question is asked (see figure 2)
6. The user then selects an answer and feedback is immediately provided.
7. Depending on whether the user gets the answer right or wrong, the square deactivates and contains a tick or a cross.
8. Upon answering all questions, or clicking the 'quit' button, the user is given their score and can either close the game completely, or click 'finish' which will take them back to step 1 – the main menu.

3.5. Testing

It was attempted to recruit users who had a varying existing knowledge in the area. Although the solution assumed very little knowledge in the subject area, ideally it would be beneficial to the majority of users and so it was vital to ensure that the test participants represented a range of user ability. A total of 35 requests for user testing were sent, with 21 users subsequently carrying out the test (14 male and 7 female). In

relation to age, 2 were 18 or under, 12 were 19-25, 3 were 26-35, 1 was 36-45 and 3 were aged 45+.

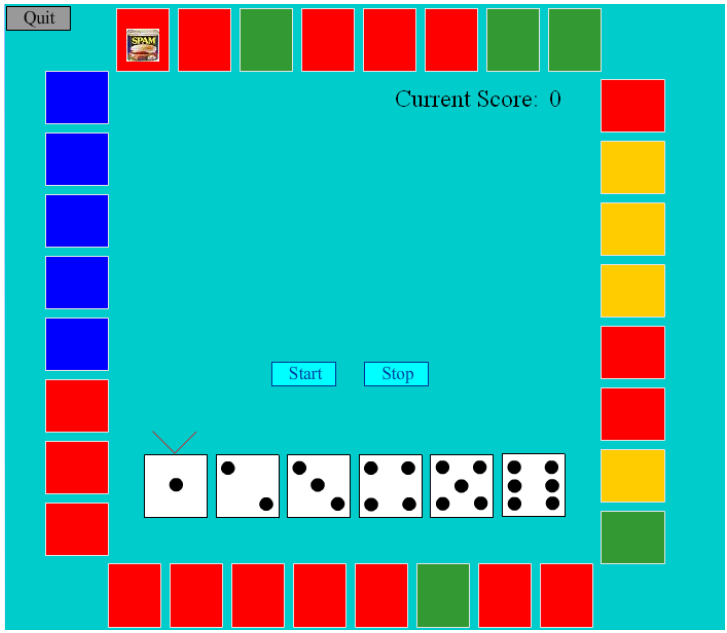

Figure 1: The main game board

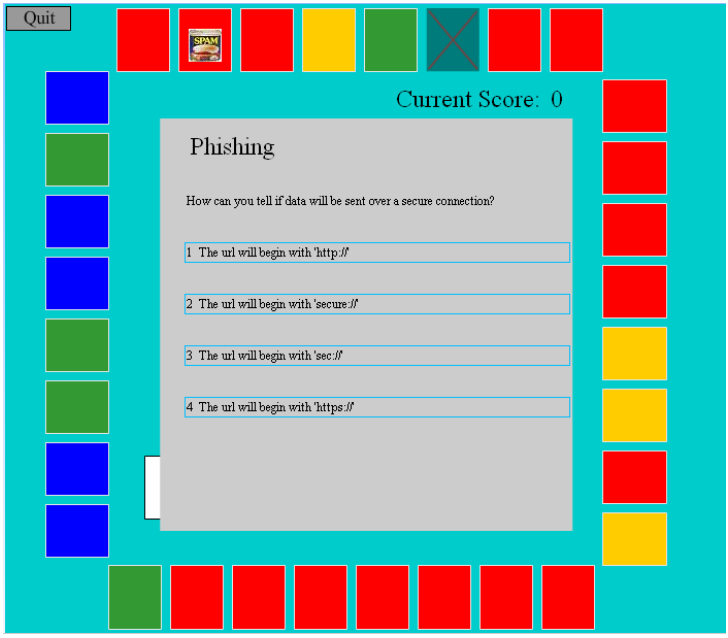

Figure 2: Question Interface

The solution was tested by asking the users three preliminary questions to find the length of time they spend on the Internet every week and their current knowledge in the subject area. They were then asked read through the online material once, and then play through the game whilst referring to the literature if necessary. They then resumed the questionnaire and were asked to provide their score, overall feelings on the game and literature and their recommendations for future development which was vital with this being a prototype.

Figure 3 shows the relation between time spent online and the achieved score. It would be expected that the longer someone spends on the Internet, the better score they would achieve. As seen below this seemed to be generally the case. As the solution is aimed at every day Internet users, the users at the lower end of the Internet usage were not expected to get poor scores as it was ensured that the material was aimed for a range of users. This seems to be shown below, with the lowest score achieved being 450 which is approximately 56% from someone who spends very little time on the Internet

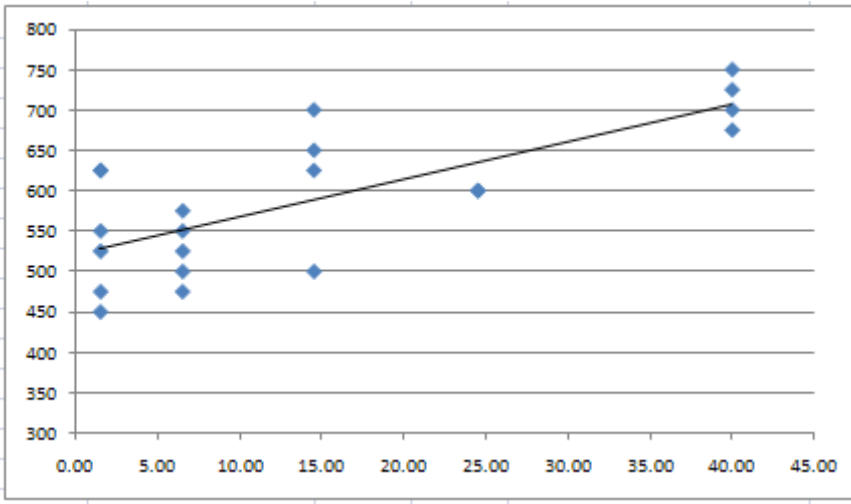

Figure 3: The relationship between time spent on the Internet and score

4 Conclusion and Future Development

The testing results suggested that the project had been a success. The overall aim was to improve user awareness of social engineering and the results detailed in section 3.5 suggest that had in fact happened. In addition to this, 17 out of 21 of the test participants found the level of website literature suitable, with 18 of them satisfied with the question levels in the game. Feedback regarding the questions and literature was very positive. Figure 4 shows the user responses when asked what they would like to see in future developments.

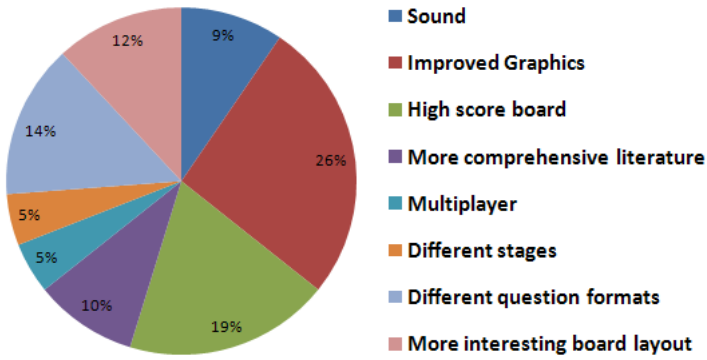

Figure 4: Features desired in future developments

Improved graphics was the most popular option. There is a range of ways this could be developed in the future including a more aesthetically pleasing board layout rather than the coloured squares in the prototype.

The second most popular feature was a high score board, which as well as adding a user requested element to the game, may also be a way of retaining the users attention, with a goal for themselves of getting on the high score board they are less likely to become bored. It may also tempt the user to replay the game.

Different question formats was the third most popular feature. This could include things such as displaying four emails to the user, and have them select the email that is a social engineering attack. An alternative to this could be displaying a single email and have the user select the 'hotspots' that would suggest that it is a social engineering attack.

It can be seen that with the prototype showing potential, if the above developments are implemented in a future development, participation should increase and ideally so will the awareness levels of those who play.

5 References

Bennet et al. *Object Oriented Analysis and Design*. 2nd Ed. London. McGraw-Hill

Gartner. 2007. *Gartner Survey Shows Phishing Attacks Escalated in 2007* [online] Available: http://www.gartner.com/it/page.jsp?id=565125 Date Accessed: 14/01/09

Microsoft. 2007. *What Is Social Engineering?* [online] Available: http://www.microsoft.com/protect/yourself/phishing/engineering.mspx Date Accessed: 14/01/09

Sky. 2009. *Facebook Scam* http://news.sky.com/skynews/home/technology/ facebook-Scam-Ive-Been-Mugged-In-London/Article/200908315363182 Date Accessed: 21/08/09