EAI Endorsed Transactions
on Scalable Information Systems
Research Article EAI.EU

A Game Theoretical Model for Anticipating Email Spear-Phishing Strategies

Franklin Tchakounte[1,3,*], Virgile Simé Nyassi[1], Duplex Elvis Houpa Danga[1,3], Kalum Priyanath Udagepola[2], Marcellin Atemkeng[4]

[1]Department of Mathematics and Computer Science, Faculty of Science, University of Ngaoundéré, Cameroon.
[2]Research Development Institute of Technology, Australia
[3]Laboratory of Mathematics and Applications (LAMAP), University of Ngaoundéré, Cameroon
[4]Department of Mathematics, Rhodes University, 6140 Grahamstown, South Africa

Abstract

A solution to help victims against phishing is anticipating and leveraging impacts related to phisher actions. In this regard, this work reshapes game theoretical logic between Intrusion Detection System (IDS) agents and insiders to email spear-phishing interactions. The email spear-phishing attack is designed as a non-cooperative and repeated game between opponents. Additionally, this work relies on Quantal Response Equilibrium (QRE) to build a game theoretical approach to predict the phisher's future intent based on past actions of both players. This approach is coupled with a recommendation strategy of appropriate allocation of resources to invest to strengthen user protection. Simulations on spear-phishing scenarios demonstrate the ability of the final system to intuitively guess the most likely phisher decisions. This work provides intelligence to spear-phishing detectors and humans such that they can anticipate next phisher actions.

Received on 11 May 2020; accepted on 09 September 2020; published on 18 September 2020

Keywords: Attack, game theory, non-cooperative game, email spear-phishing, QRE

doi:10.4108/eai.26-5-2020.166354

1. Introduction

Phishing is a fraudulent practice using a medium of communication (such as email, social networks, website) to lure users to provide sensitive information (such as username, password, credit card number, etc.) for malicious purposes [1, 2]. This cybercrime's number of victims is perpetually increasing as reported by the Anti-Phishing Working Group (APWG)[3, 4]. The phisher is the malicious object (human, robot, etc.) initiating social engineering activities to the potential victim whereas the victim is the phisher's target or an intermediary to the phisher's target[5]. There are several forms of phishing [6–8]; email phishing in which the attacker sends a fake email to trick users to send sensitive information [9, 10], Uniform Resource Locator (URL) phishing aims at building fake copies of legitimate websites. Vishing in which malicious people initiate voice calls, put the target victim in alert or build a fake environment related to the victim with collected information. The finality of vishing is to scam money. Spear-phishing refers to phishing attacks which target identified (groups of) people within an organization. Spear-phishing is considered as one of the most prevalent forms of Advanced Persistent Threats (APTs) [11] and its perpetrators are hard to catch. In fact, the phisher disguises as a colleague of the target and creates contextually persuasive emails with urgency [12–14]. Email is the attack vector most exploited in spear-phishing [15]. This work focuses on email spear-phishing due to its popularity[7]. The terms phishing, spear-phishing, and email spear-phishing mean interchangeably the same in this paper. The terms such as attacker, phisher and defender also mean interchangeably the same. Research provides different categories of approaches to tackling spear-phishing[6–8]. The first category includes preventive educational approaches on online platforms and gaming solutions[16–19]. The second category includes client-side tools that users install

*Corresponding author. Email: tchafros@gmail.com

either on the browser [20–23] or in the operating system (OS)[24, 25]. The third category relies on artificial intelligence to profile spear-phishing activities [26–31]. The fourth category contributes to list-based filtering of URL based on predefined database of annotated URLs (phished and benign) and rules [32, 33]. The last category provides visual similarity based approaches to identify variants of a website [33, 34]. These various different approaches all help mitigate spear-phishing to some degree. They are, however, inefficient at delimiting different interactive activities inside the attack for final diagnoses.

According to Shiva [35], game theory is able to model interactions involving different actors, looking for rewards resulting from optimal actions. Exploiting game theory is relevant to circumscribe strategic interactions between a defender (i.e. the potential victim) and an attacker during different stages of communication. At the end, game theory provides defender strategies to improve to deceive attackers. Indeed, the potential victim is targeted by the attacks trying to lure him in different stages. The attack can succeed once the user receives the first mail (in case of fragile victim) or the attacker varies techniques and sends multiple messages tricking the user to fall (in case the user suspects messages). Adapting defense tactics and varying attack techniques during exchanges between both players are modeled with game theory and can be combined to reinforce existing anti-phishing tools. If we take the user u who is targeted by the attacker A motivating scenario can be the one illustrated in the TR-FR attack, available in the paper [36]. During this scenario, the attacker changes its strategies over time while the victim behaves differently based on his level of knowledge. Game theory is therefore penetrating into the cybersecurity sphere and is exploited to fight against phishing ([37–40]). However, these proposals lack to consider repetitive interactions between players, iteration after iteration. As a result, they are not able to efficiently predict the phisher's future intent. Some researches based on game theory include this flaw. Nonetheless, they model interactions between attackers and Intrusion Detection System (IDS). Based on these observations, this work deals with the following research questions:

- How to adapt IDS game theory models to email spear-phishing scenarios?
- What game-theory models are suitable to model repetitive spear-phishing interactions to anticipate phisher intentions?
- What appropriate defensive measures should be invested based on previous attack activities?

Quantal Response Equilibrium (QRE) is adopted in this work due to the fact that individuals have beliefs supported in equilibrium by the strategies that players choose, considering that players make systematic mistakes or deviations in their choices [41]. As a consequence, payoffs of players are influenced by social preference. This paper designs spear-phishing as a non-cooperative and repeated game between opponents. It proposes a QRE game theoretical model which learns defensive measures to anticipate an upcoming phishing strategy based on historical exchanges between opponents. To the best of our knowledge, this paper is the first work to focus on exploring QRE-based strategies for email spear-phishing. In summary, this paper presents three main contributions:

- Designing an extensive form of one-stage email phishing game;
- Modelling a non-cooperative and repeated game between the attacker and the defender
- Modelling and simulating the repeated game to predict the future behavior of the phisher.

The rest of this paper is organized as follows. Section 2 characterizes and conceptualizes the game. Section 3 is dedicated to formulate a one-shot phishing game before extending it to a repeated game. Further, a method of calculating QRE-based strategies is dissected. Section 4 concerns modelling and implementing a prediction mechanism for phishing tricks based on QRE. Section 5 includes experiments on real spear-phishing scenarios coupled to preventive recommendations and discusses results. Related works are surveyed in section 6 to reveal contributions of this work. Conclusion and perspectives are provided at the end of the document.

2. Game description and game scenarios

2.1. Game description

Players. Interactions during a phishing email attack are modelled as a game between two actors: the phisher A with bad intentions and the defender D who is an employee targeted by phisher strategies.

Scope and assumptions. This work falls within the context of protecting company infrastructures. In this environment, email spear-phishing is the main exploit since it relies on the psychological state of employees. Additionally, email is the vector from which URL phishing, vishing and ransomware infiltration can be triggered. Organizations evolve in size over the time. Even if they put in place training programs, new recruited employees can still be unaware of phishing for a period and ignorant of detection measures. We assume that a new employee has no defensive measures. Email accounts are safe and are not supported with

encryption protocols such as Pretty Good Privacy (PGP). Nowadays, phisher initiates exchanges to a target and adapts them during interactions until success. We consider this scenario where an employee receives emails from a single phisher who adapts the contents over the time. The players are both rational, meaning that they seek to maximize their gain and thus minimize their losses. The network protections in the organization have no effective and updated anti-phishing tools.

Characteristics.

Incomplete information game Phishing is modelled in this work as a game with **incomplete information** because the defender is supposed to be ignorant of phishing techniques. The defender does not *a priory* recognize a fake email and is not aware of the attack strategy and rewards (or payoffs or utilities) expected by the attacker. In other words, the defender is not aware of the attacker's strategies as well as both possible rewards.

Imperfect information game Our study focuses on a game with **imperfect information** because the defender is not (*a priori*) aware of historical actions taken by the attacker, before acting during the current iteration.

Non-cooperative game Email phishing game is a **non-cooperative game** because:

- Both players do not communicate with each other before making a decision;
- opponents maintain conflictual interactions and seek contrary goals;
- Both players have strictly opposite preferences – the phisher prefers a successful attack whereas the defender wants a failed attack.

Sequential game The phisher starts the game with the following three actions.

- Inquiring about the victim;
- Choosing the attack strategy;
- Building a fake email to send to the user.

The user has the opportunity to take an action in the game, only after receiving the email sent by the phisher.

Non-zero sum game Indeed, for each outcome of the game, the sum of rewards of the players is always different from zero.

2.2. Attack scenarios

The attacker can either attack based on building fake e-mails or based on infiltrating malicious scripts.

Phishing based on mimicking . This category is based on techniques of social engineering, which mimic email from legitimate entities. Three strategies are used to succeed in such an attack.

Embed an answer email address Here, the malicious email contains only text and an email address to which the user could respond to provide the desired personal information to the attacker. The email content is carefully adapted to convince the recipient through emergency words or expressions. The aim is to incentive D to respond. For example, the email content could state that D's online account will be disabled in case the email is ignored. In addition, the hacker can also spoof the email address of a credible sender to request sensitive information. This strategy is denoted as S_1.

Embed a phone number At this level, A encourages D to continue the conversation via phone calls using a phone number delicately introduced into the email. This process leads to **vishing attacks**. This strategy is denoted as S_2.

Embed a fake link This strategy also called **URL phishing** inserts a fake link in the email. The latter redirects D to a counterfeit site specially designed to malintentionally gather sensitive data. Information from D is automatically redirected to A. This strategy is represented by S_3.

Phishing based on infiltration. The second attack's technique consists of infiltrating malicious codes into an email attachment. Once activated (attachment downloading), the code runs and two scenarios can occur.

- The code encrypts the host machine's files claiming for ransom: this is called **ransomware**;
- The code masquerades on the host computer to
  - Spy and collect sensitive information at an appropriate moment and transfer them to the phisher;
  - Turn the host computer as a bot.

This strategy is represented by S_4.

2.3. Scenarios for possible user responses

There are four response scenarios delivered by D depending on the attack strategy.

Scenario 1. The phisher sends a fake email according to the strategy S_1; then, the user can

- Be lured to send sensitive information by email to the phisher's email address (D_{11});
- Suspect a hoax and ignore the email (D_{12});
- Rely on anti-phishing training [1] but falls victim (D_{13});
- Avoid the trap thanks to an anti-phishing training adapted to the type of attack S_1 (D_{14}).

Scenario 2. The phisher sends a forged email related to S_2. The user can make the following reactions.

- Be lured to send sensitive information by phone to the number contained in the email (D_{21});
- Suspect a hoax and ignore the email (D_{22});
- Rely on anti-phishing training but falls victim (D_{23});
- Avoid the trap thanks to an anti-phishing training adapted to the type of attack S_2 (D_{24}).

Scenario 3. In this scenario, the phisher chooses to send a fraudulent email based on strategy S_3. The defender can accordingly,

- Be lured to click on a fake link, to be redirected to a counterfeit website (D_{31});
- Suspect a hoax and ignore the email (D_{32});
- Fall victim despite being assisted by an anti-phishing tool [2] (D_{33});
- Avoid the trap thanks to the anti-phishing toolbar (D_{34}).

Scenario 4. The phisher sends a fraudulent email based on strategy S_4. The defender can,

- Be lured to download malicious scripts (hidden behind the attachment) designed to redirect sensitive information to the phisher (D_{41});
- Suspect a hoax and ignore the email (D_{42});
- Use an antivirus and still bite the hook (D_{43});
- Avoid the trap by setting up a powerful and up-to-date antivirus (D_{44}).

Figure 1 summarizes the aforementioned scenarios involved in the spear-phishing game.

[1] In the rest of the document, User Education is denoted by U_e to meaning anti-phishing training

[2] This is an anti-phishing toolbar to define a blacklist of phished websites. In the rest of this paper, such a software is denoted by **Bt** for ***Blacklisting Tool***.

According to Figure 1, the game has 16 possible outcomes listed in Table 1.

3. Formulation and construction of the model

This section is dedicated to the formulation of the phishing game and its construction.

3.1. Model formulation

Model definition. The interaction between both players is defined as the ***4-tuple*** $\mathbb{G} = (\mathcal{N}, \mathcal{A}, \mathcal{U}, \preceq)$ where:

- $\mathcal{N} = \{1, 2\}$ is the set of players, with:
  - 1 to denote the attacker;
  - and 2 to denote the user.
- $\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2$, the set of possible outcome's game where the symbol $\times$ denotes the Cartesian product; $\mathcal{A}_1 = \{S_1, S_2, S_3, S_4\}$, the set of possible phisher's moves in the game; $\mathcal{A}_2 = \{D_{11}, D_{12}, D_{13}, D_{14}, D_{21}, D_{22}, D_{23}, D_{24}, D_{31}, D_{32}, D_{33}, D_{34}, D_{41}, D_{42}, D_{43}, D_{44}\}$ the set of possible user's actions.
- $\preceq$ denotes the preference's relation having three variants:

  $\prec_1$ the preference's relation over different outcomes of the game from the phisher point of view; $\prec_2$ the preference's relation over different outcomes of the game from the user point of view; And $\sim_i$ the indifference of the player i.

  Moreover, $\forall(x, y) \in \mathcal{A}$ and $i \in \{1, 2\}$,

  $x \prec_i y \Rightarrow$ the player i prefers the outcome y to the outcome x; $x \sim_i y \Rightarrow$ the player i prefers the outcome x as much as the outcome y.
- $\mathcal{U} = \{\mathcal{U}_1, \mathcal{U}_2\}$, with $\mathcal{U}_1 : \mathcal{A} \mapsto \mathbb{R}$ the phisher's utility function; $\mathcal{U}_2 : \mathcal{A} \mapsto \mathbb{R}$ the defender's utility function.

Players' preferences and utility functions. To quantify the different outcomes of the game, the player's preferences are firstly specified on these outcomes. Then, the Von Neumann-Morgenstern utility function [42] is applied to assign numbers that reflect these preferences.

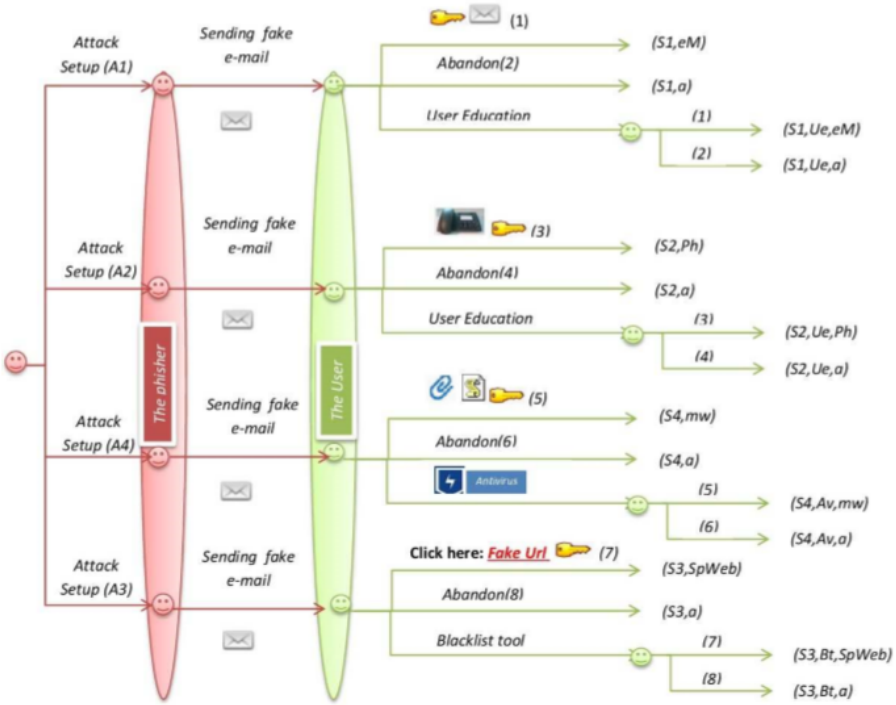


Figure 1. Interactions' scenarios between the attacker and the user.

Table 1. Game's outcomes

Outcomes	Meanings
(S_1, D_{11}) or (S_1, eM)	the trick S_1 succeeds in luring the user
(S_1, D_{12}) or (S_1, a)	the user suspects a subterfuge and ignores the email
(S_1, D_{13}) or (S_1, Ue, eM)	the trick S_1 succeeds despite the user's education
(S_1, D_{14}) or (S_1, Ue, a)	the trick S_1 fails thanks to an anti-phishing training
(S_2, D_{21}) or (S_2, eM)	the trick S_2 succeeds in luring the user
(S_2, D_{22}) or (S_2, a)	the user suspects a subterfuge and ignores the email
(S_2, D_{23}) or (S_2, Ue, eM)	the trick S_2 succeeds despite the user's education
(S_2, D_{24}) or (S_2, Ue, a)	the trick S_2 fails thanks to an anti-phishing training
(S_3, D_{31}) or $(S_3, SpWeb)$	the trick S_3 succeeds in luring the user via a fake site
(S_3, D_{32}) or (S_3, a)	the user suspects a subterfuge on the link and ignores the email without clicking on it
(S_3, D_{33}) or $(S_3, Bt, SpWeb)$	the trick S_3 succeeds despite the use of Bt (an anti-phishing toolbar)
(S_3, D_{34}) or (S_3, Bt, a)	the trick S_3 fails using Bt
(S_4, D_{41}) or (S_4, mw)	the trick S_4 succeeds in luring the user via the downloaded spying code
(S_4, D_{42}) or (S_4, a)	the user suspects a subterfuge related to the attachment and ignores the email
(S_4, D_{43}) or (S_4, Av, mw)	the trick S_4 succeeds despite the use of an antivirus Av
(S_4, D_{44}) or (S_4, Av, a)	the trick S_4 fails thanks to antivirus which detected malware (mw)

Phisher's preferences Ranking outcomes according to phisher's preferences involves the ability to identified outcomes of the game, which leads to a best situation for the phisher or not.

Outcomes giving the highest preference for the hacker are those that result in a successful attack. They are $(S_1, eM), (S_2, Ph), (S_3, SpWeb)$ and (S_4, mw). It is adopted that, from the phisher's point of view, preparing an attack with the user's personal information gathering by email (eM) or phone (Ph) is less complex than the following ones: designing an attack where the phisher harvests via a counterfeit website (SpWeb), setting up spyware [3] (mw), embedding a malicious attached file. Equation (1) is therefore obtained.

$$(S_4, mw) \prec_1 (S_3, SpWeb) \prec_1 (S_1, eM) \sim_1 (S_2, Ph). \quad (1)$$

[3] Requires advanced malware enforcement.

The ranking continues with outcomes that lead to a successful attack despite the presence of defensive measures (Ue, Av or Bt). It is less preferable to the ranking in (1) (from the phisher's point of view) because attacking an unprotected system increases its chances of success (and decreases the probability of being unmasked). This scenario confers the phisher the opportunity to target the same victim or other potential victims of the same victim's network.

Despite existing anti-phishing tools, the last decision belongs to the user. A well-performed education of the defender could be seen as the most feared mechanism by the phisher. In addition, the hacker dreads Av (Antivirus) more than Bt (Blacklist tool). The defender would prefer Bt to Av because it is harder to bypass Av than Bt. Antiviruses are increasingly coupled with artificial intelligence to be able to anticipate and quickly update signatures. Blacklists are less reliable for filtering because attackers can build other variants of websites with existing technologies. Equation (3) summarises previous preferences.

$$(S_1, Ue, eM) \sim_1 (S_2, Ue, Ph) \prec_1 (S_4, Av, mw) \quad (2)$$
$$\prec_1 (S_3, Bt, SpWeb) \quad (3)$$

Subsequently, outcomes representing failures of the phisher without any defense system are derived. From the attacker's point of view, it is preferable to fail after dedicating less effort (S_3, a) on attack than failing after dedicating much (S_4, a) and failing anyway.

$$(S_4, a) \prec_1 (S_3, a) \prec_1 (S_1, a) \sim_1 (S_2, a). \quad (4)$$

Equation (5) includes the worst outcomes for the phisher. The strategies involved in this equation fail because of defensive measures.

$$(S_1, Ue, a) \sim_1 (S_2, Ue, a) \prec_1 (S_4, Av, a) \prec_1 (S_3, Bt, a). \quad (5)$$

The overall phisher preference's ranking is given in (6):

$$(5) \prec_1 (4) \prec_1 (3) \prec_1 (1). \quad (6)$$

Defender's preferences There are four main possible outcomes from the user's point of view:

D falls victim without being assisted (by Ue, Bt or Av); D falls victim despite some countermeasures (Ue, Bt, Av). D avoids scamming with the help of defensive measures; D avoids scamming without being assisted (by Ue, Bt or Av).

In view of these four outcomes, the user's preference ranking is established as follows:

$$ⓑ \prec_2 ⓐ \prec_2 ⓒ \prec_2 ⓓ. \quad (7)$$

The user prefers avoiding the attack as much as possible without any expense in the acquisition of Ue, Bt or Av; The user prefers losing without any expense in the acquisition of a countermeasure than to lose having made such an expense.

In addition, it is assumed that in terms of defensive measures, the user establishes the order of preference in (8):

$$Bt \prec_2 Av \prec_2 Ue. \quad (8)$$

Security turns around the user no matter which tool is used. Therefore education is of huge significance; Despite the strategy (S_4), Av also protects the defender's computer against other computer threats (malware or denial of service).

Equations (7) and (8) establish the ranking in (9), from the user's point of view:

$$(S_1, Ue, eM) \quad (9)$$
$$\sim_2 (S_2, Ue, Ph) \quad (10)$$
$$\prec_2 (S_4, Av, mw) \quad (11)$$
$$\prec_2 (S_3, Bt, SpWeb) \quad (12)$$
$$\prec_2 (S_4, mw) \prec_2 (S_3, SpWeb) \quad (13)$$
$$\sim_2 (S_1, eM) \sim_2 (S_2, Ph) \quad (14)$$
$$\prec_2 (S_3, Bt, a) \prec_2 (S_4, Av, a) \quad (15)$$
$$\prec_2 (S_1, Ue, a) \sim_2 (S_2, Ue, a) \quad (16)$$
$$\prec_2 (S_4, a) \sim_2 (S_3, a) \sim_2 (S_1, a) \sim_2 (S_2, a) \quad (17)$$

Construction of utility functions of players Constructing utility function consists to compute $\mathcal{U}_i(outcome)$, $\forall\ outcome \in \mathcal{A}$ and $i \in \{1, 2\}$. To achieve this, the Binmore's method is exploited [42] and implemented using a Matlab script[4]. The Binmore's method builds the Von Neumann-Morgenstern utility functions by assigning a number that reflects preferences established in (6) and (9) so that:
$\forall\ Oc1, Oc2 \in \mathcal{A}$ and $i \in \{1, 2\}$,

$$\mathcal{U}_i(Oc1) \leq \mathcal{U}_i(Oc2) \Longleftrightarrow Oc1 \preceq_i Oc2. \quad (18)$$

The values of this function are called payoffs [43].

3.2. Model construction

Firstly, the model is built using the open source software Gambit to solve the game under NE. NE is a solution concept [5] which describes a steady state condition of the game [44]. Given that players agreed on the NE's set of strategies, a player who deviates

[4]It is available at https://github.com/virgilo/PhishingGame/blob/master/Binmore.m.

[5]A solution concept is a systematic description of how the game will be played by employing the best possible strategies and what the outcomes might be

Table 2. Utility function for **phisher**

Outcome	U_1(Outcome)	$16 \times U_1$(Outcome)
(S_1, Ue, a)	0	0
(S_2, Ue, a)	0	0
(S_4, Av, a)	1/8	2
(S_3, Bt, a)	1/4	4
(S_4, a)	3/8	6
(S_3, a)	7/16	7
(S_2, a)	1/2	8
(S_1, a)	1/2	8
(S_1, Ue, eM)	5/8	10
(S_2, Ue, Ph)	5/8	10
(S_4, Av, mw)	11/16	11
$(S_3, Bt, SpWeb)$	3/4	12
(S_4, mw)	7/8	14
$(S_3, SpWeb)$	15/16	15
(S_1, eM)	1	16
(S_2, Ph)	1	16

Table 3. Utility function for **defender**

Outcome	U_2(Outcome)	$8 \times U_2$(Outcome)
(S_1, Ue, eM)	0	0
(S_2, Ue, Ph)	0	0
(S_4, Av, mw)	1/8	1
$(S_3, Bt, SpWeb)$	1/4	2
(S_4, mw)	3/8	3
$(S_3, SpWeb)$	1/2	4
(S_1, eM)	1/2	4
(S_2, Ph)	1/2	4
(S_3, Bt, a)	5/8	5
(S_4, Av, a)	3/4	6
(S_1, Ue, a)	7/8	7
(S_2, Ue, a)	7/8	7
(S_1, a)	1	8
(S_2, a)	1	8
(S_3, a)	1	8
(S_4, a)	1	8

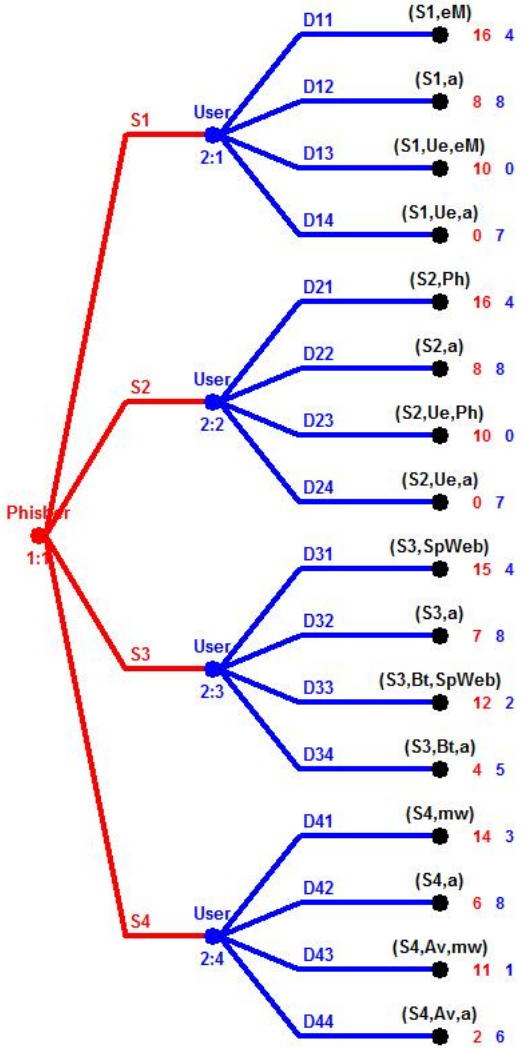


Figure 2. Extensive form of the phishing game under Gambit

from this agreement would reduce related payoffs. This solution concept specifies only the steady state and not how that steady state is reached in the game. Secondly, Matlab scripts are implemented to build the repeated game and predict the future behavior of the phisher.

Gambit. Gambit [45] is a software library of game theory that provides tools necessary for the construction and analysis of games in normal or extensive forms. Gambit is adopted because it is only dedicated to non-cooperative games. It fits therefore to the game characteristics (see Section 2.1). Figure 2 shows the model in an extensive form generated using Gambit version 16.0.1. The defender has four possible reactions (D_{ij}) for each strategy (S_i) developed by the phisher. Each reaction (D_{ij}) is associated with a utility obtained from [42]. D_{21} is associated with a utility of 4 for the defender and a utility of 16 for the attacker.

Players' successful attack probabilities and players' losses. Three elements are formulated based on the model. They include

The calculation of the probability for an attack to succeed as $\mathbf{Proba}_{attackSucceed}$ established in (20), The probability's calculation that the defensive measure succeed $\mathbf{Proba}_{defenseSucceed}$ established in (23), Losses related to each player respectively established in (24) and (25).

Utility functions such as $\mathcal{U}_1 : \mathcal{A} \mapsto \mathbb{R}$ and $\mathcal{U}_2 : \mathcal{A} \mapsto \mathbb{R}$ are assimilated to probability distributions, $\mathcal{U}_1^{'} : \mathcal{A} \mapsto [0;1]$ and $\mathcal{U}_2^{'} : \mathcal{A} \mapsto [0;1]$, according to Equation (19). $\forall\ player \in \{1,2\}$,

$$\mathcal{U}_{player}^{'}(S_{i_k}, D_{i_k j_k}) = \frac{\mathcal{U}_{player}(S_{i_k}, D_{i_k j_k})}{\sum_{i,j} \mathcal{U}_{player}(S_i, D_{ij})}. \quad (19)$$

Table 4 records processing values from Equation (19). Let's consider the fourth line to illustrate how to compute output values, based on Equation (19).

- Phisher: As shown in Figure 2, $\mathcal{U}_{phisher}((S_3, D_{34})) = 4$ and $\sum_{i,j} \mathcal{U}_{phisher}(S_i, D_{ij}) = 16+8+10+0+16+8+10+0+15+7+12+4+14+6+11+2 = 139$. Then $\mathcal{U}_{phisher}^{'}(S_3, D_{34}) = \frac{4}{139}$

- Defender: As shown in Figure 2, $\mathcal{U}_{defender}((S_3, D_{34})) = 5$ and $\sum_{i,j} \mathcal{U}_{defender}(S_i, D_{ij}) = 4+8+0+7+4+8+0+7+4+8+2+5+3+8+1+6 = 75$. Then $\mathcal{U}_{defender}^{'}(S_3, D_{34}) = \frac{5}{75} = \frac{1}{15}$

Table 4. Probability weights associated to each outcome

Outcome	$\mathcal{U}_1^{'}(Outcome)$	$\mathcal{U}_2^{'}(Outcome)$
(S_1, Ue, a)	0	7/15
(S_2, Ue, a)	0	7/15
(S_4, Av, a)	14/139	1/25
(S_3, Bt, a)	4/139	1/15
(S_4, a)	6/139	8/75
(S_3, a)	7/139	8/75
(S_2, a)	8/139	8/75
(S_1, a)	8/139	8/75
(S_1, Ue, eM)	10/139	0
(S_2, Ue, Ph)	10/139	0
(S_4, Av, mw)	11/139	1/75
$(S_3, Bt, SpWeb)$	12/139	2/75
(S_4, mw)	14/139	1/25
$(S_3, SpWeb)$	15/139	4/75
(S_1, eM)	16/139	4/75
(S_2, Ph)	16/139	4/75

Probability of a successful attack The probability $\textbf{Proba}_{attackSucceed}$ is formulated as follows:

$$\begin{aligned}\textbf{Proba}_{attackSucceed} = {} & P(S_1) \times [\mathcal{U}_1^{'}(S_1, eM) \\ & + (1 - Eff_{Ue}) \times \mathcal{U}_1^{'}(S_1, Ue, eM)] \\ & + P(S_2) \times [\mathcal{U}_1^{'}(S_2, Ph) \\ & + (1 - Eff_{Ue}) \times \mathcal{U}_1^{'}(S_2, Ue, Ph)] \\ & + P(S_3) \times [\mathcal{U}_1^{'}(S_3, SpWeb) \\ & + (1 - P(Bt)) \times \mathcal{U}_1^{'}(S_3, Bt, SpWeb)] \\ & + P(S_4) \times [\mathcal{U}_1^{'}(S_4, mw) \\ & + (1 - P(Av)) \times \mathcal{U}_1^{'}(S_4, Av, mw)]\end{aligned} \quad (20)$$

where

- $P(S_i)$ is the probability that strategy S_i is chosen by attacker;
- Eff_{Ue} refers to the effectiveness of the anti-phishing training;
- $P(Bt)$ is the probability that the anti-phishing tool detects a fake link ;
- $P(Av)$ is the probability that the antivirus detects the malicious attachment.

For the attack to succeed, the attacker either uses the trick S_1, S_2, S_3 or S_4. If the trick S_1 is used, the attack is successful in two cases:

The user is lured to send sensitive information by email (S_1, eM). The user falls into the phisher's traps (S_1, Ue, eM) despite the anti-phishing training Ue.

It is also noted that equation (20) is formulated in such a way that the satisfaction of the phisher is zero i.e. the defensive measure is effective at 100%:

$$Eff_{Ue} = P(Bt) = P(Av) = 1 \quad (21)$$

$$\Rightarrow \begin{cases}(1 - Eff_{Ue}) \times \mathcal{U}_1^{'}(S_1, Ue, eM) = 0 \\ (1 - Eff_{Ue}) \times \mathcal{U}_1^{'}(S_2, Ue, eM) = 0 \\ (1 - P(Bt)) \times \mathcal{U}_1^{'}(S_3, Bt, SpWeb) = 0 \\ (1 - P(Av)) \times \mathcal{U}_1^{'}(S_4, Av, mw) = 0\end{cases} \quad (22)$$

Probability that the defensive measure succeeds Since the event "*the success of the defensive measure*" is the opposite event of "*the success of the attack*", Equation 23 is obtained as follows.

$$\textbf{Proba}_{defenseSucceed} = 1 - \textbf{Proba}_{attackSucceed} \quad (23)$$

Attacker and defender losses Phisher and user losses can be estimated through Equations (24) and (25); The calculation of these losses requires a thorough

evaluation of costs (financial, intellectual and time) of the development of each attack as well as costs (financial, intellectual and time) related to the acquisition of the defense measure. The proposed formulas (24) and (25) assume that cost variables C_{S_i} are known:

$$\begin{aligned}\textbf{Losses}_{phisher} = \\ P(S_1)\times C_{S_1} + P(S_2)\times C_{S_2} \\ +P(S_3)\times C_{S_3} + P(S_4)\times C_{S_4}\end{aligned} \tag{24}$$

$$\begin{aligned}\textbf{Losses}_{defender} = \\ [P(S_1)+P(S_2)]\times(1-Eff_{Ue})\times C_{Ue} \\ +P(S_3)\times(1-P(Bt))\times C_{Bt} \\ +P(S_4)\times(1-P(Av))\times C_{Av},\end{aligned} \tag{25}$$

where

- C_{S_i} refers to costs related to the development of the attack S_i;
- $1-P(Bt)$ is the probability that the blacklisting tool is faulty;
- $1-P(Av)$ is the probability that the antivirus is faulty.

Prediction of the phisher's future behavior.

Repetition of the stage game We have so far modelled situations where the interaction between the attacker and the user takes place only once (**one-shot game** also called **stage game**). Actually, interactions described in the **stage game** (shown in Figure 2) are not performed just once. The game $\mathbb{G}$ is played more than once. These interactions should therefore be modelled as a repeated game. According to Shen *et al.* [46], a repeated game is a particular style of an extensive game in which each stage is a repetition of the same strategic game. The number of instances in a repeated game may be finite or infinite. If the game never ends (i.e. players interact forever) or players do not know when the game ends, it is called an infinitely repeated game. This paper considers this type of game. Indeed, during an email phishing attack, the user and the attacker interact without knowing the end.

Players' utility function in the repeated game This phase consists of determining rewards of both players during the game. This paper considers this type of game, i.e. taking into account the dynamicity of the game. Both players strive to maximize the expected gains, iteration after iteration.

Inspired by the work of Shen *et al.* [46], the total utility for the phisher, after the t^{th} iteration of the game, is given by:

$$\mathcal{U}^t_{phisher}(S_i) = \sum_{y=0}^{t} \delta^y \mathcal{U}^y_{phisher}(S_i). \tag{26}$$

Here $\delta \in [0;1[$ is a discount factor.

The term $\mathcal{U}^t_{phisher}(S_{i_k})$ is defined as the total utility that the phisher expects to obtain by choosing the strategy S_{i_k} at the t^{th} iteration of the game. Equation (26) is improved to equation 27 by better emphasizing impact of each outcome obtained during iterations $0\ldots(t-1)$:

$$\begin{aligned}\mathcal{U}^t_{phisher}(S_{i_k}) = \sum_{h=0}^{t-1} \delta^h \times \mathcal{U}_{phisher}(S^h_{i_h}, D^h_{i_h j_h}) \\ + \delta^t \times \mathcal{U}_{phisher}(S_{i_k}),\end{aligned} \tag{27}$$

where

- $U^t_{phisher}(S_{i_k})$ is the total utility of the phisher during repeated game, choosing the strategy S_{i_k} at the t^{th} iteration before the user's reaction during this iteration.
- $S^h_{i_h}, D^h_{i_h j_h}$ is the outcome of the game obtained at the ith iteration.
- $U_{phisher}(S_{i_k})$ is the *a priori* phisher's utility by choosing the strategy S_{i_k} during the stage game without any reaction from the user.

***A priori* utility $\mathcal{U}_{phisher}(S_{i_k})$** The prediction proposed by the model in (40) is intrinsically related to $\mathcal{U}^t_{phisher}(S_{i_k})$ which depends itself on the *a priori* utility $\mathcal{U}_{phisher}(S_{i_k})$. Similarly Shen *et al.* [46] have used a method based on QRE to predict future attacks, but without explicitly describing how to obtain such *a priori* utility. That is, how to evaluate the gain of a player who has chosen an action without the other player's choice made? Table 2 provides the value of $U_{phisher}(S_1, D_{11})$ [6] but lacks to give the value of $\mathcal{U}_{phisher}(S_1)$.

The relation (28) is proposed to overcome this issue.

$$\begin{aligned}\mathcal{U}_{phisher}(S_{i_k}) = \sum_{j=1}^{4} Proba(D_{i_k j}|S_{i_k}) \\ \times \mathcal{U}_{phisher}(S_{i_k}, D_{i_k j}),\end{aligned} \tag{28}$$

where

[6] $U_{phisher}(S_1, D_{11}) = \mathcal{U}_{phisher}(S_1, eM) = 16$

- $Proba(D_{i_kj}|S_{i_k})$ is the probability for the user to choose $D_{i_kj_k}$ knowing that the phisher has chosen S_{i_k}.

- $U_{phisher}(S_{i_k}, D_{i_kj})$ is the phisher's gain when the outcome (S_{i_k}, D_{i_kj}) is realized.

Moreover,

$$Proba(D_{i_kj_k}|S_{i_k}) = \frac{Proba(S_{i_k}, D_{i_kj_k})}{Proba(S_{i_k})} \tag{29}$$

with,

$$Proba(S_{i_k}, D_{i_kj_k}) = 0.5 \times \frac{\mathcal{U}_{phisher}(S_{i_k}, D_{i_kj_k})}{\sum_{i,j=1}^{4} \mathcal{U}_{phisher}(S_i, D_{ij})} + 0.5 \times \frac{\mathcal{U}_{User}(S_{i_k}, D_{i_kj_k})}{\sum_{i,j=1}^{4} \mathcal{U}_{User}(S_i, D_{ij})} \tag{30}$$

such that:

$$\sum_{i,j=1}^{4} Proba(S_i, D_{ij}) = 1. \tag{31}$$

Furthermore, the probability $Proba(S_{i_k})$ is computed in Algorithm 1 within two constraints:

it depends on the history of the game $History = \{\{S_i^h, D_{ij}^h\}\}_{h=0,1,\ldots,(t-1)}$, it increases $(sign = -1)$[7] or decreases $(sign = 1)$ proportionally to the satisfaction that the hacker would have obtained by opting during previous iterations $(h = 0, \ldots, t-1)$.

Algorithm 1: Adjustment of probabilities according to the game history

1

Require: δ and $\mathcal{U}'_{phisher}$
Ensure: Adjusted probabilities $Proba(S_i)\ \forall i \in \{1, 2, 3, 4\}$
Begin
1- **For** $i \longleftarrow 1$ **to** 4 **do**
2- $Proba(S_i) \longleftarrow \frac{1}{4}$;
3- **End For**
4- $\beta_2 \longleftarrow 2 + \varepsilon$; $Proba_{max} \longleftarrow 0.65$;
$//\varepsilon = 2.2204 * 10^{-16}$.
5- $k \longleftarrow \frac{(\beta_2 - 1) \times Proba_{max}}{(1 - Proba_{max})}$;
6- **For** $h \longleftarrow t - 1$ **to** 0 **do**
7- $up \longleftarrow 1 + \delta^h \times \mathcal{U}'_{phisher}(S_{i_h}^h, D_{i_hj_h}^h)$;
8- $up \longleftarrow min(k + 1 - \varepsilon, up)$;
9- sign$\longleftarrow +1$;
10- **If** $(j_h == 2$ **Or** $j_h == 4)$ **Then**
11- sign$\longleftarrow -1$;
12- **End If**
13- $\tau \longleftarrow sign \times (\frac{1-up}{k}) \times Proba(S_{i_h})$;
14- $Proba(S_{i_h}) \longleftarrow Proba(S_{i_h}) + \tau$;
15- **For** $i \longleftarrow 1$ **to** 4 **do**
16- **If** $(i \neq i_h)$ **Then**
17- $Proba(S_i) \longleftarrow Proba(S_i) - \frac{\tau}{3}$;
18- **End If**
19- **End For**
20- **End For**
21- **Return** (Proba);
End

Note: $Proba_{max} = 0.65$ means that, during the probabilistic adjustment phase, the largest value of $Proba(S_{i_h})$, $\forall S_{i_h} \in \mathcal{A}_1$, is 0.65; however, it does not necessarily imply that, during the prediction phase (after including the QRE), the probabilities obtained will also be increased by $Proba_{max}$. Then, the operation $Proba(S_{i_h}) \longleftarrow Proba(S_{i_h}) + \tau$ in statement 14 must be checked, so that $Proba(S_{i_h})$ remains a probability [8] despite the progressive additions of τ during the game iterations. It is therefore a question of ensuring that $\forall \tau$:

$$\begin{cases} Proba(S_{i_h}) + \tau \leq 1 \\ and \\ Proba(S_{i_h}) + \tau > 0 \end{cases} \tag{32}$$

[7] $\frac{(1-up)}{k} < 0$, since $up > 1$ and $k > 0$

[8] $Proba(S_{i_h})$ must always be between 0 and 1

Let,

$$\begin{cases} up = 1 + \delta^h \times \mathcal{U}'_{phisher}(S^h_{i_h}, D^h_{i_h j_h}) \\ \tau = sign \times (\frac{1-up}{k}) \times Proba(S_{i_h}) \\ Proba_{max} = \max\{Proba(S_{i_h}), S_{i_h} \in \mathcal{A}_1\} \\ \text{with } Proba_{max} = 0.65. \end{cases} \tag{33}$$

The following aims to look for $k \in \mathbb{R}^*_+$ such that $0 < Proba(S_{i_h})\tau \leq 1$.

On the one hand,

$$\text{Proba}(S_{i_h}) + \tau \leq 1 \Leftrightarrow$$

$$\begin{cases} Proba(S_{i_h}) + (\frac{1-up}{k}) \times Proba(S_{i_h}) \leq 1 \\ Proba(S_{i_h}) - (\frac{1-up}{k}) \times Proba(S_{i_h}) \leq 1 \end{cases}$$

$$\Leftrightarrow \begin{cases} 1 + (\frac{1-up}{k}) \leq \frac{1}{Proba(S_{i_h})}, & \text{with } Proba(S_{i_h}) > 0. \\ 1 - (\frac{1-up}{k}) \leq \frac{1}{Proba(S_{i_h})}, & \text{with } Proba(S_{i_h}) > 0. \end{cases}$$

$$\Leftrightarrow \begin{cases} up \geq k + 1 - \frac{k}{Proba(S_{i_h})} \\ up \leq -k + 1 + \frac{k}{Proba(S_{i_h})} \end{cases}$$

$$\Leftarrow \begin{cases} up \geq \max\{k + 1 - \frac{k}{Proba(S_{i_h})}, S_{i_h} \in \mathcal{A}_1\} \\ up \leq \min\{-k + 1 + \frac{k}{Proba(S_{i_h})}, S_{i_h} \in \mathcal{A}_1\} \end{cases}$$

$$\Leftrightarrow \begin{cases} up \geq k + 1 - \frac{k}{Proba_{max}} \\ up \leq -k + 1 + \frac{k}{Proba_{max}} \end{cases}$$

$$\Leftarrow \begin{cases} 1 \geq k + 1 - \frac{k}{Proba_{max}}, & \text{because } up \geq 1. \\ 2 \leq -k + 1 + \frac{k}{Proba_{max}}, & \text{because } up \leq 2 \text{ since } \delta^h \text{ and} \\ & \mathcal{U}'_{phisher}(S^h_{i_h}, D^h_{i_h j_h}) \in [0;1], \forall\, h. \end{cases}$$

$$\Leftrightarrow \begin{cases} k + 1 - \frac{k}{Proba_{max}} = \beta_1, & \text{with } 1 \geq \beta_1. \\ -k + 1 + \frac{k}{Proba_{max}} = \beta_2, & \text{with } 2 \leq \beta_2. \end{cases}$$

$$\Leftrightarrow \begin{cases} k = \frac{(\beta_1 - 1) \times Proba_{max}}{Proba_{max} - 1}, & \text{with } 1 \geq \beta_1. \\ k = \frac{(\beta_2 - 1) \times Proba_{max}}{1 - Proba_{max}}, & \text{with } 2 \leq \beta_2. \end{cases}$$

$$\Leftrightarrow \begin{cases} k = \frac{(\beta_2 - 1) \times Proba_{max}}{1 - Proba_{max}} \\ \frac{(\beta_1 - 1) \times Proba_{max}}{Proba_{max} - 1} = \frac{(\beta_2 - 1) \times Proba_{max}}{1 - Proba_{max}}, & \text{with } 1 \geq \beta_1 \text{ and } 2 \leq \beta_2. \end{cases}$$

$$\Leftrightarrow \begin{cases} k = \frac{(\beta_2 - 1) \times Proba_{max}}{1 - Proba_{max}} \\ (\beta_1 - 1) = -(\beta_2 - 1), & \text{with } 1 \geq \beta_1 \text{ and } 2 \leq \beta_2. \end{cases}$$

$$\Leftrightarrow \begin{cases} k = \frac{(\beta_2 - 1) \times Proba_{max}}{1 - Proba_{max}} \\ \beta_1 + \beta_2 = 2, & \text{with } 1 \geq \beta_1 \text{ and } 2 \leq \beta_2. \end{cases}$$

It is further noted that

$$[(\beta_1 = 2 - \beta_2) \text{ and } (2 \leq \beta_2)] \Rightarrow (\beta_1 \leq 0 \leq 1). \tag{34}$$

Thus, to have $Proba(S_{i_h}) + \tau \leq 1, \forall\ S_{i_h} \in \mathcal{A}_1$, it is enough to have

$$[(k = \frac{(\beta_2 - 1) \times Proba_{max}}{1 - Proba_{max}}), (\beta_1 = 2 - \beta_2) \text{ and } (2 \leq \beta_2)]. \tag{35}$$

On the other hand, while assuming that

$$(k = \frac{(\beta_2 - 1) \times Proba_{max}}{1 - Proba_{max}}), (2 \leq \beta_2) \quad \text{and } (\beta_1 = 2 - \beta_2), \tag{36}$$

In the following, we look for a sufficient condition to have $Proba(S_{i_h}) + \tau > 0, \forall\ S_{i_h} \in \mathcal{A}_1$.
We have,

$$\begin{aligned} Proba(S_{i_h}) + \tau > 0 \quad &\Leftrightarrow \quad \begin{cases} Proba(S_{i_h}) + (\frac{1-up}{k}) \times Proba(S_{i_h}) > 0 \\ Proba(S_{i_h}) - (\frac{1-up}{k}) \times Proba(S_{i_h}) > 0 \end{cases} \\ &\Leftarrow \quad Proba(S_{i_h}) + (\frac{1-up}{k}) \times Proba(S_{i_h}) > 0, \text{ because } \frac{1-up}{k} < 0 \\ &\Leftarrow \quad 1 + \frac{1-up}{k} > 0, \text{ because } Proba(S_{i_h}) > 0 \\ &\Leftrightarrow \quad k + (1-up) > 0, \text{ because } k > 0 \\ &\Leftrightarrow \quad up < k + 1 \end{aligned}$$

Therefore

$$[up = min(k + 1 - \varepsilon, 1 + \delta^h \times \mathcal{U}'_{phisher}(S^h_{i_h}, D^h_{i_h j_h}))] \tag{37}$$

$$\Rightarrow [Proba(S_{i_h}) + \tau > 0, \forall\ S_{i_h} \in \mathcal{A}_1\ ] \tag{38}$$

To ensure that $(Proba(S_{i_h}) + \tau \in]0; 1], \forall\ S_{i_h} \in \mathcal{A}_1$, it is sufficient to have the following conditions

$$\begin{cases} k = \frac{(\beta_2 - 1) \times Proba_{max}}{1 - Proba_{max}} \\ up = min(k + 1 - \varepsilon,\ 1 + \delta^h \times \mathcal{U}'_{phisher}(S^h_{i_h}, D^h_{i_h j_h})) \\ (2 \leq \beta_2) \text{ and } (\beta_1 = 2 - \beta_2) \end{cases} \tag{39}$$

Lines 4, 5, 7 and 8 in Algorithm 1 are exploited to demonstrate that the condition (39) is respected. As a conclusion, during the execution of this algorithm, the sum $[Proba(S_{i_h}) + \tau]$ remains in the interval $]0; 1]$.

Probability model of choosing a future attack strategy This step relies on QRE, initially defined by McKelvey and Palfrey [47]. It is a concept of equilibrium, which besides being more realistic contrary to the equilibrium of Nash, captures the limited rationality of each player and predicts the future behavior of the attacker as claimed by Kantzavelou and Katsikas [48]. In this regard, Shen *et al.* [46] proposed a model for calculating the probability of a future attack. Their model is adapted to fit our context through Equation (40).

$$Proba^t_\lambda(S_{i_k}) = \frac{exp(\lambda \times u^t_{phisher}(S_{i_k}))}{\sum_{i=1}^{4} exp(\lambda \times u^t_{phisher}(S_i))} \tag{40}$$

Where $Proba^t_\lambda(S_{i_k})$ is the probability that, given (t-1) iteration (s) already performed in the game, the phisher decides to attack via the S_{i_k} strategy during the next iteration, with $i_k \in \{1, 2, 3, 4\}$. QRE introduces a decision parameter λ which represents the rationality of the player. When $\lambda = 0$, the phisher is completely irrational which corresponds to a random choice. Moreover, the phisher's next action is gradually influenced by the expected reward as λ increases to ∞ which reflects a fully rational attacker acting to maximize gains.

Based on the prediction (Equation 40), the defender is recommended to invest up to $Mean(\{Proba^t_\lambda(S_{i_k})\}_{\lambda=0...\lambda_{max}})$% of its resources and budget (as computed in Equation 41). Such recommendations support countermeasures against attack and are considered as appropriate responses to avoid being lured in the t^{th} iteration. Gambit sets its threshold to $\lambda_{max} = 1000000$ [48]. The threshold has rather been set to $\lambda_{max} = 100$ since Matlab simulations revealed that, starting from $\lambda = 100$ the phisher's intent no longer fluctuates according to the strategies.

$$\begin{aligned} &Mean(\{Proba^t_\lambda(S_{i_k})\}_{\lambda=0...\lambda_{max}}) \\ &= \frac{1}{(1 + \lambda_{max})} \sum_{\lambda=0}^{\lambda_{max}} Proba^t_\lambda(S_{i_k}) \end{aligned} \tag{41}$$

4. Implementation and complexity

Here, the components of the final system are presented and its temporal complexity is analyzed.

4.1. Implementation phases

The implementation of the proposed approach requires four phases as shown in Figure 3). The first phase initializes the game. The second phase formalizes

historical interactions. The third phase concerns how to adjust probabilities from history. The last phase predicts the future phisher's intent. The game phases have been implemented with MATLAB and the Github page with all the artifacts are included in the GitHub project page[9]. Some elements have been considered to ease the development phase. The first one is that the number of attackers is one, meaning that we are in a situation where an attacker develops strategies in different stages to get rewards in the game. The second is that the number of instances in a repeated game is finite, meaning that the game ends after some time. However, in practice, various attackers can target the same victim and an attacker can target multiple victims simultaneously as well as players should not know when the game is closed. The proposed development is an attempt of real cases.

Phase 1: Stage game construction The phisher's utility function $\mathcal{U}_1$ and the user's utility function $\mathcal{U}_2$ are built.
Next, the method described in (Equation 19) is implemented for obtaining probability weights $\mathcal{U}_1^{'}$ and $\mathcal{U}_2^{'}$.

Phase 2: History definition This phase is devoted to define
$History = \{\{S_i^h, D_{ij}^h\}\}_{h=0,1,...,(t-1)}$ as the set of previous outcomes that occurred during the interaction between both players. The script [10] provides to the user the ability to input these previous outcomes or generate them randomly for simulation purposes.

Phase 3: Probabilities adjustment based on the game history This phase aims to implement the algorithm described in Algorithm 1. In so doing, the probability that the phisher chooses a given trick S_i, depends on the reward related to this trick in the past interactions with the user.

Phase 4: Prediction and recommendations This phase consists of the implementation of Equation (27) [11] to evaluate the reward of the phisher, obtained during the history, according to the probabilities adjusted in phase N° 3.

The concept of QRE (Equation (40)) is subsequently implemented to predict the future phisher's intent. Appropriate defensive measures are recommended to the defender using Equation 41.

4.2. Analysis of complexity

The first phase does not depend on the history of the game and it is realized once. Other phases, using the game data, can be performed several times to simulate different "**history**" cases.

Fundamental operations for determining temporal complexity belong to phases N° 2, N° 3 and N° 4.

The construction of the stage game in phase N° 1 is excluded because it is not executed when the number of iterations during the repeated game increases. The worst scenario in the second phase concerns the generation of the game's history. t successive assignments are required to generate t outcomes $\{\{S_i^h, D_{ij}^h\}\}_{h=0,1,...,(t-1)}$ earlier in the history.
Let be:

$$N_2 = t \text{ operations.} \tag{42}$$

In the third phase, operations $up \longleftarrow \delta^h \times \mathcal{U}'_{phisher}(S_{i_h}^h, D_{i_h j_h}^h)$, $Proba(S_{i_h}) \longleftarrow Proba(S_{i_h}) + \tau$ and $Proba(S_i) \longleftarrow Proba(S_i) - \frac{\tau}{3}$ are considered as fundamental. Then, phase N° 3 requires N_3 assignments defined as follows

$$N_3 = 5 \times t \text{ operations.} \tag{43}$$

Assignments required in equations (27)[12], (28), (29), (30) and (41)[13] are considered as fundamental operations in phase N° 4.

First, the calculation of $Proba(S_{i_k}, D_{i_k j_k})$ in Equation (30), requires 16 assignments to obtain $\sum_{i,j=1}^{4} u_{phisher}(S_i, D_{ij})$, 16 assignments to obtain $\sum_{i,j=1}^{4} u_{User}(S_i, D_{ij})$, and one addition. It makes a total of N_{4_1} assignments.

$$N_{4_1} = 16 + 16 + 1 = 33 \text{ assignments} \tag{44}$$

The calculation of $Proba(D_{i_k j_k} | S_{i_k})$ in (Equation 29) requires therefore N_{4_2} assignments defined in Equation 45.

$$N_{4_2} = N_{4_1} + 1 = 34 \text{ assignments.} \tag{45}$$

The *a priori* utility $\mathcal{U}^t_{phisher}(S_{i_k})$ (Equation (28)) costs about N_{4_3} assignments as defined as in Equation 46.

$$N_{4_3} = 4 * N_{4_2} = 136 \text{ assignments.} \tag{46}$$

So, for any trick S_{i_k}, the calculation of $\mathcal{U}^t_{phisher}(S_{i_k})$ from from equation (27) is computed in N_{4_4} as defined in Equation 47.

$$N_{4_4} = t + N_{4_3} = t + 136 \text{ assignments,} \tag{47}$$

[9] https://github.com/virgilo/PhishingGame
[10] available at https://github.com/virgilo/PhishingGame/blob/master/CreateStory.m.
[11] The script is available at https://github.com/virgilo/PhishingGame/blob/master/UphisherHistory.m
[12] For predictions.
[13] For recommandations.

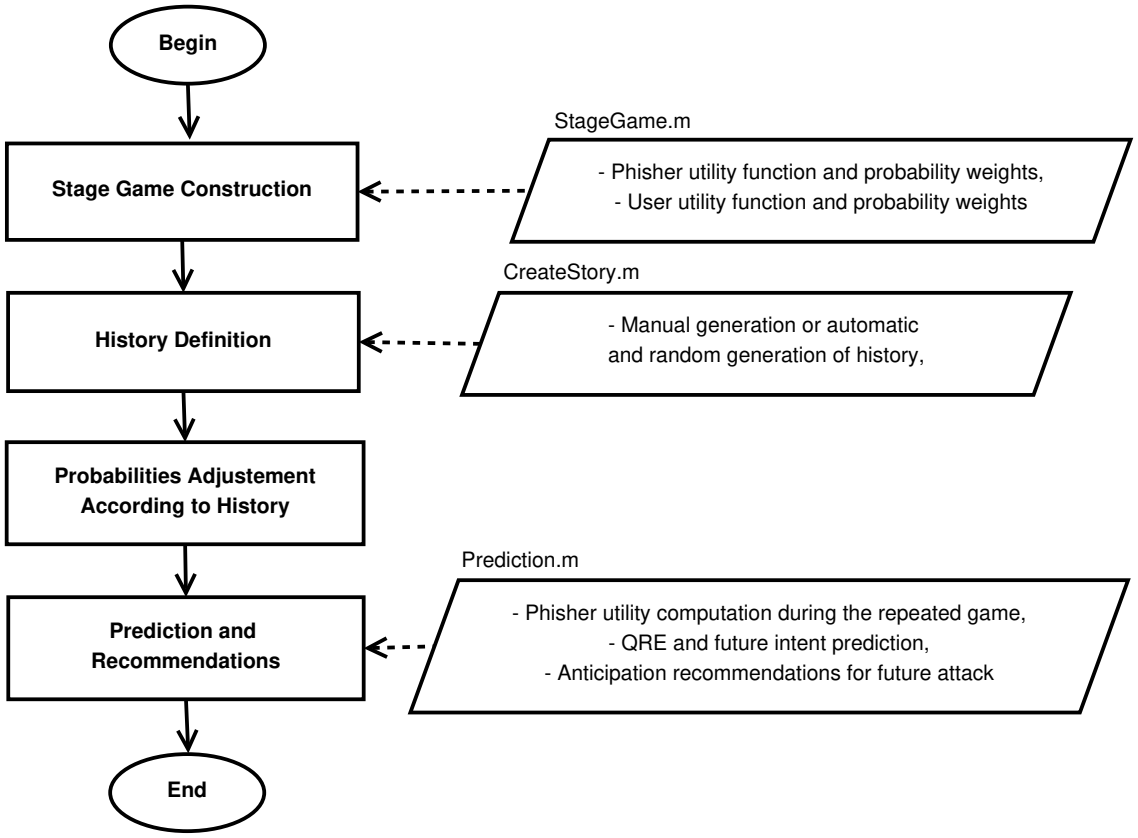


Figure 3. Implementation phases

Hence, the prediction in Equation 40 takes N_{4_5} assignments.

$$N_{4_5} = 4 \times N_{4_4} + 1 = 4 \times t + 545 \text{ assignments.} \quad (48)$$

Recommendations related to the four tricks require N_{4_6} assignments. N_{4_6} is evaluated based on Equation (41).

$$\begin{aligned} N_{4_6} &= 4 \times [(\lambda_{max} + 1) \times N_{4_5}] \quad (49) \\ &= 16 \times (\lambda_{max} + 1) \times t + 2180 \times (\lambda_{max} + 1) \end{aligned}$$

The prediction approach requires the total number of fundamental operations N for each simulation. N is obtained as follows:

$$\begin{aligned} N &= N_2 + N_3 + N_{4_6} \quad (50) \\ &= t + 5 \times t + 16 \times (\lambda_{max} + 1) \times t \\ &\quad + 2180 \times (\lambda_{max} + 1) \\ &= [16 \times (\lambda_{max} + 1) + 6] \times t \\ &\quad + 2180 \times (\lambda_{max} + 1) \end{aligned}$$

In sum, the temporal complexity is linear in t. This complexity grows with the number of instances in one stage. And it grows much higher when the number of stages increases.

5. Simulations and interpretations

We simulated the model to validate its intelligence to predict reasonably future phisher decisions. This section has two orientations. The first orientation builds the one-shot game model using Gambit to obtain NE. It is realized using the integrated package for calculating **NE**. The second orientation determines probabilities of a successful attack and the phisher attack anticipation based on **QRE** through the repeated game. Gambit does not have an embedded library to model the repeated game.

5.1. Nash Equilibrium of the model

Result. The Nash Equilibrium $\mathbb{G}$ presented in Figure 2 is illustrated with more details in Figure 4. Figure 4 describes the behavior of both players at the equilibrium.

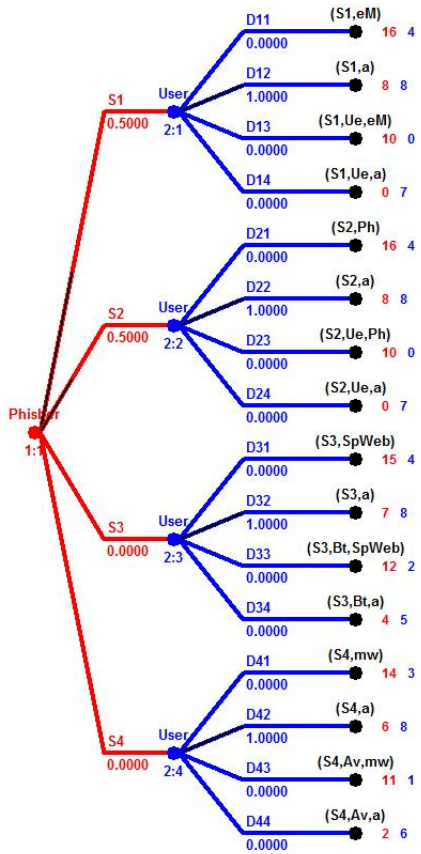


Figure 4. NE of the model

Interpretation. The equilibrium presents that the phisher is likely to opt for strategies S_1 and S_2 with the same probability of 0.5. In both cases, the defender would better abandon suspected e-mails with a probability of 1. This reaction, *a priori* plausible during an attack, confirms the consistency of the proposed model. However, since it is a repeated game, the total number of strategies at the *t*th stage is a multiple of the number of history strategies at all stages *0*, *1* ... *t–1*. This number grows with the number of iterations. Consequently, the time to predict the future behavior of a player at the NE state in a stage becomes higher and higher and the one for the whole game explodes accordingly.

5.2. Simulations and interpretations in case of an attack

A script [14] in Matlab has been written to determine utility functions of players for a not repeated game and simulates Equation (20). Figure 5 presents probability results of both opponents during an attack. The experiment is made under the following hypotheses.

Simulations are made under NE; $Eff_{Ue} = 0.8$, $P(Bt) = 0.7$, and $P(Av) = 0.7$.

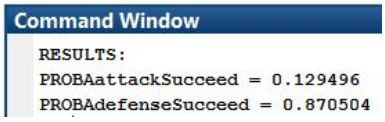
- 

Figure 5. Probability for an attack to succeed during Nash Equilibrium

The attack has less chance of success because

$$\textbf{Proba}_{attackSucceed} = 0.1295$$

at **NE** whereas the defensive measure is more likely to protect the user because

$$\textbf{Proba}_{defenseSucceed} = 0.8705.$$

5.3. Simulations and interpretations of attack predictions

The purpose of this section is to predict the future behavior of the hacker at the t^{th} iteration based on previous $(t-1)$ iterations defined as // $History = \{\{S_i^h, D_{ij}^h\}\}_{h=0,1,\ldots,(t-1)}$). For this purpose, three general cases are simulated and discussed.

Case Nº 1.

Inputs

The number of iterations already traversed is $(t-1) = 2$, $History = \{(S_4; D_{44}), (S_1; D_{13})\}$.

The history is explained as follows: the phisher starts the attack using an email concealing a malicious attachment (S_4); Fortunately, the user with the help of antivirus foils this attack (D_{44}). Subsequently, the phisher continues the attack with a suitable text including an email address (S_1) to direct related responses; The user falls into the trap despite the anti-phishing training (Ue). The attack succeeds.

[14] available at https://github.com/virgilo/PhishingGame/blob/master/StageGame.m

Results and interpretations Figure 6 illustrates recommendations to be made by the defender at the third iteration. Figure 7 shows the phisher intent predictions during the third iteration. Firstly, Figure 7 reveals that S_1 is more likely to be used by the phisher for the next attack. Indeed, the most recent attack of the phisher [15] succeeded despite the countermeasure of the user (D_{13}); Thus, the attacker during the third iteration, seeks to replicate previous success. It is done by betting on the trick S_1, to thwart the protection implemented by the user (D_{13}).

Secondly, Figure 7 also reveals that the trick S_4 is less likely to be chosen at the next stage of the game. Indeed, the game's history indicates that the user has already taken a defensive action against S_4, which allows to thwart the trick S_4.

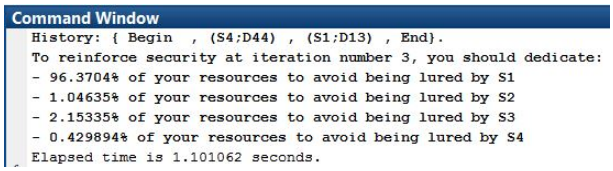


Figure 6. Case Nº 1: recommendations

To reinforce security during the third iteration, the proposed model therefore recommends the user to dedicate:

> 96.3704% of its resources to avoid being lured by the attack's trick S_1; 1.04635% of its resources to avoid being lured by the attack's trick S_2; 2.15335% of its resources to avoid being lured by the attack's trick S_3; 0.429894% of its resources to avoid being lured by the attack's trick S_4.

Case Nº 2.

Inputs There are two inputs in this case.

> The number of iterations already spent is $(t-1) = 3$, $History = \{(S_1; D_{13}), (S_1; D_{12}), (S_3; D_{32})\}$.

The phisher starts with an adapted text attack that contains an email address (S_1) to direct related responses; the attack succeeds despite the defensive measure of the user (D_{13}). Subsequently, the phisher insists on the same strategy which results in a failure (D_{12}). Finally, the phisher decides to change the strategy and opts for a malicious link attack (S_3); The user is careful and ignores the mail (D_{32}).

[15] The one occured at the second iteration (S_1).

Results and interpretations Figure 8 outlines the recommendations on resources to allocate to reinforce defensive measures. Figure 9 presents the phisher intent predictions at the fourth iteration.

Our model predicts that the phisher's intent for the next attack will be S_2. Indeed, the most recent trick (S_3) failed; According to the game's history, the strategy S_1 is the only successful strategy to lure the user despite the defender's training.

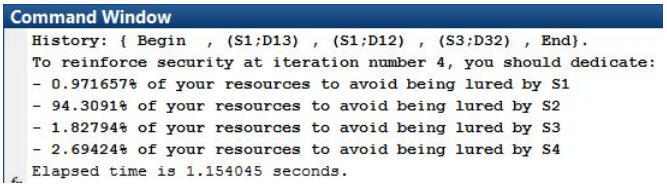


Figure 8. Case Nº 2: recommendations

However, the outcome $(S_1; D_{13})$ followed by $(S_1; D_{12})$ shows that the user has been trained and knows how to recognize the strategy S_1. Since the phisher's preferences on S_1 and S_2 are almost similar [16], the phisher's strategy is changed to S_2. The aim is to hopefully deceive the user who has already been lured by a similar ruse in the past.

Furthermore, S_1 and S_3 have a very low probability of appearing once again because they have been foiled during previous game's iterations. However, S_3 has the lowest probability [17] because it is the most recent one and it has never been beneficial for the phisher during the game's history.

During the fourth iteration, the model therefore recommends the user to dedicate

> 0.971657% of its resources to avoid being lured by the attack's trick S_1; 94.3091% of its resources to avoid being lured by the attack's trick S_2; 1.82794% of its resources to avoid being lured by the attack's trick S_3; 2.69424% of its resources to avoid being lured by the attack's trick S_4.

Case Nº 3.

Inputs

> The number of iterations already traversed, $(t-1) = 4$, $History = \{(S_4; D_{44}), (S_4; D_{44}), (S_4; D_{44}), (S_4; D_{44})\}$.

The phisher succeeds four (04) attacks based on the attached file concealing a spy code, and the user foils this attack each time, with an antivirus.

[16] See relation (1).
[17] The lowest curve in the Figure 9.

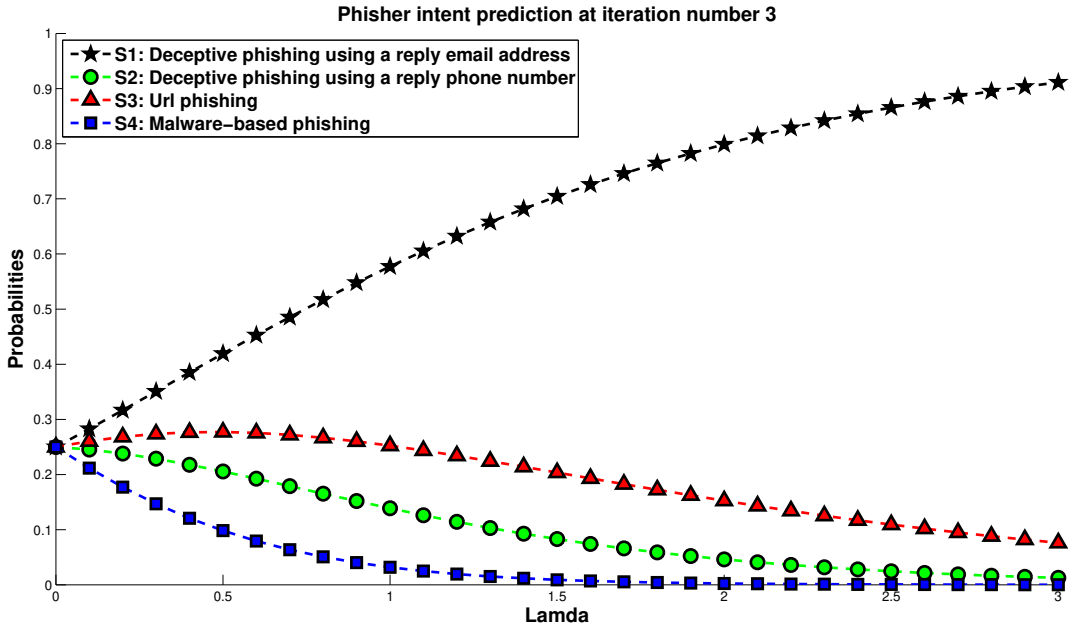


Figure 7. Case Nº 1: predictions

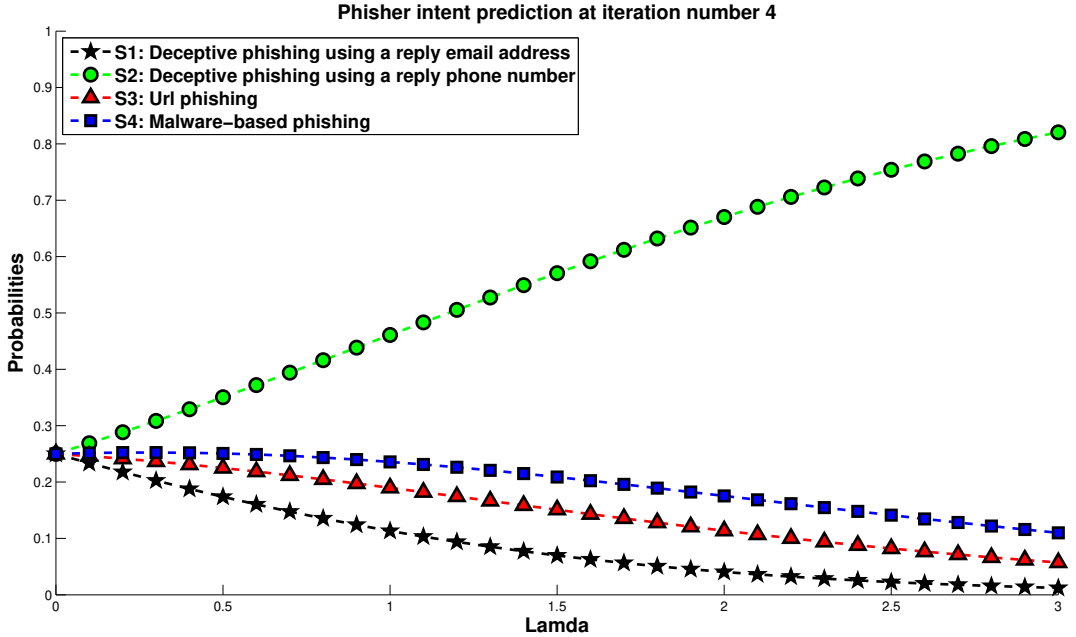


Figure 9. Case Nº 2: predictions

Results and interpretations Figure 10 presents recommendations obtained to anticipate the fifth attack. Figure 11 presents the phisher intent predictions at the fifth iteration.

The model predicts that the phisher, after having tried to lure the user four consecutive times via S_4, will abandon this strategy to bet primarily on an attack based on a forged URL (S_3) as shown in Figure 11. The model predicts an equiprobability in the hacker's

choice between these two strategies, based on the indifference in the phisher's preferences between S_1 and S_2 (Equation (1) in Section 3.1).
The model advocates therefore the user to devote

> 3.92921% of its resources to avoid being lured by the attack's trick S_1; 3.92921% of its resources to avoid being lured by the attack's trick S_2; 91.7156% of its resources to avoid being lured by the attack's trick S_3; 0.426001% of its resources to avoid being lured by the attack's trick S_4.

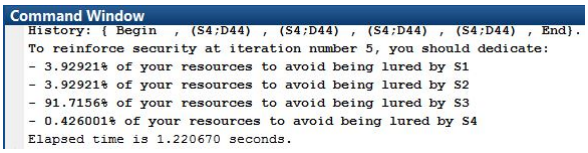


Figure 10. Case N° 3: recommendations

6. Related Works

This section describes solutions for spear-phishing. The first part deals with main approaches and the second part presents research works exploiting game theory for phishing and intrusion detection.

6.1. Prevention and mitigation approaches

Companies acquire network protection solutions (IDS, firewalls, honeypots etc..) to mitigate spear-phishing intrusion[24, 25]. At the employee level, they opt for antiviruses [49, 50] or filters based on black and white lists installed on browsers [20–23]. Training sessions with tools simulating real attacks are planned and educational games [16–19] set up for this purpose are used in short or long term. Employees can also voluntarily take ownership of educational tools such as TORPEDO [51] to prevent suspicious emails. Literature provides more technical solutions. They rely on artificial intelligence including automatic or deep learning to generate intelligence necessary to characterize spear-phishing activities [26–31, 49] based on an annotated sample of emails or URLs [52]. Other orientations seek to determine signatures to characterize variants of Web pages or emails to recognize similarities and to deduce malicious characters[34, 53, 54].

Limitations Existing solutions aim to identify the nature of email or URL as phished or genuine, and educate people to recognize this nature. They specifically rely on static features extracted from emails, URLs, or other vectors. They hardly take into account the whole interaction. Such types of detectors require a minimal knowledge for learning and a minimal expertise for exploitation. Game theory is a powerful tool to learn and represent knowledge related to opponent interactions.

6.2. Exploitation of game theory

Authors are interested to investigate how game theory can improve research towards the phishing detection area.

Game theory for spear-phishing. Yu *et al.* [37] models phishing through Stochastic Game Nets (SGN). Their work determines the probability of a successful attack and the average time for a successful attack. Figueroa *et al.* [38] combine classification techniques and signaling games. This association aims to develop a computer tool allowing the administrator of a network to classify an email. Zhao *et al.* [40] investigate email filtering tools diagnoses while modelling sequential spear-phishing attacks as Stackelberg game model with one and multiple credentials. They propose optimization in decision making of opponents based on the veracity weight of such first line defence. Pawlick and Zhu [39] apply Poisson Signaling Game (PSG) to capture phishing assets in the Internet of Things (IoT). Their approach captures situations with multiple receivers and gives the receivers abilities to detect deception with probabilities. Zu and Rass [55] propose to design a game-theoretical model to capture player interactions in each phase of general advanced persistent threats (APT). For instance, they provide a model for phase 1 – initial penetration and establishment, phase 2 – learning and propagation, and phase 3 – damage.

Game theory for intrusion detection. Kantzavelou and Katsikas [48] applies game theory to model interactions between insiders and IDS. The game outcomes are quantified by specifying preferences of players.
The von Neumann-Morgenstern utility function is then used to assign numbers that reflect these preferences. They extend Nash Equilibrium (NE) to QRE to capture bounded rationality of players and model behavior of insiders. These authors used QRE to determine how an insider will interact in the future, and how an IDS will react to protect the system. Shen *et al.* [46] formulate, similarly, a stage Intrusion Detection Game (IDG), where they thoroughly consider preferences of players. They assign payoffs of players based on Binmore's method, to describe interactions between the attacker and IDS agents. Authors define the corresponding payoffs by extending the stage IDG to a repeated IDG, to reflect the reality of continuous interactions. They further propose a method of calculating QRE-based strategies that predict the attacker's future behavior.

Limitations Researches based on game theory models for phishing are limited by the following aspects.

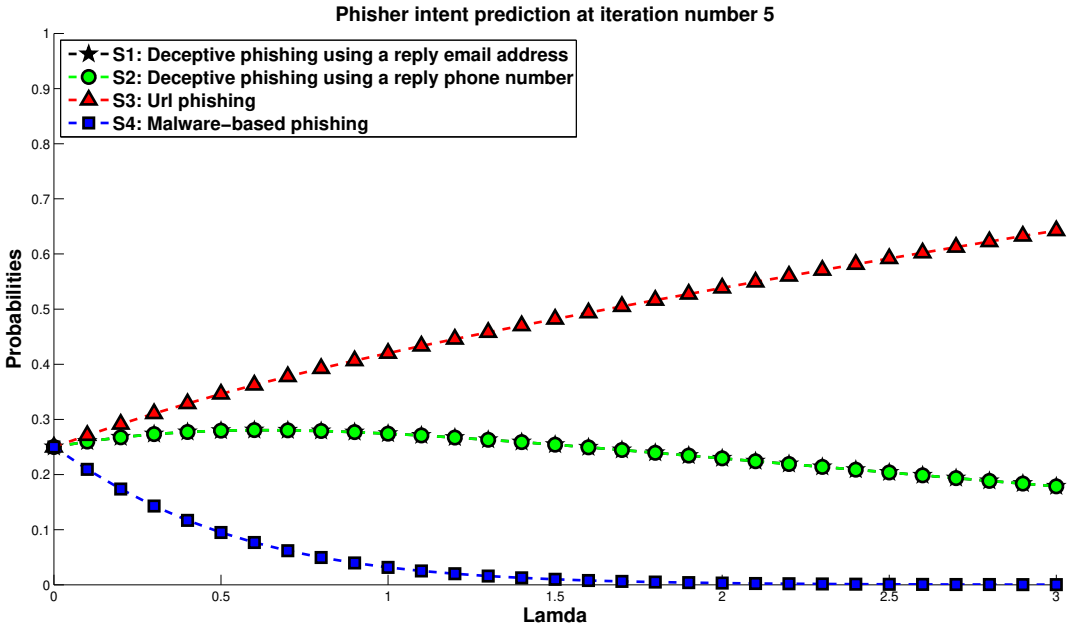


Figure 11. Case Nº 3: predictions

- Interactions can take place several times, that means phisher and defender continuously interact;
- The next attack's anticipation and prediction are not emphasized.

Kantzavelou and Katsikas [48] and Shen *et al.* [46], integrate these two aspects, but their solution does not deal with spear-phishing attacks.

Contribution This work proposed therefore to adapt approaches dealing with game theory for IDS to spear-phishing, where the defender receives from the attacker one fake message repeatedly until success.

Table 5 provides a comparison between the proposed approach and researches that applied game theory to phishing detection. One can note that research, although dealing with phishing, differs from their objectives. Authors adopt certain types of games based on their objectives even if they are mainly designed as sequential. Existing works aimed at predicting whether an incoming e-mail is fraudulent or not. Based on that, they take into consideration in their models, that the defender is likely to misclassify emails. Compared to our work, works lack to represent the phishing game with the specific strategies on both opponents. They consider in general that the attacker sends malicious objects and the receiver tries to recognize it as such or just one attack scenario. However, it is relevant to consider specific actions exploited by the attacker to infiltrate and to lure the defender. Our work rather intended to derive as precise as possible knowledge based on historical interactions that users rely on to anticipate future phisher actions. We therefore required to design as generic as possible the game with possible opponent strategies. We have designed our approach strictly to one-to-one deception games but in reality, the game can involve several attackers. This case is effective for example in case of Distributed Denial of Service (DDoS) where multiple bots redirect requests to the target. Several defenders can also be involved in case the attacker targets a group of people in a company. These two facts have been considered in other works although in other directions. We should extend our work while integrating them. The fact that one supposes that defenders have a certain defensive knowledge is most verified in developed countries. Nonetheless, we propose to take into consideration worst cases when users ignore security concerns and even adequate defensive measures. Contextually, there are also companies without any phishing filters. One positive fact is that these works can be exploited in association from the prediction to the detection of fake messages.

Table 5. Comparison with related works

	Objective	Game model	Game characteristics	Type of phishing	Defender assumptions	Results
Yu *et al.* (2013)[56]	Modelling phishing based on stochastic game	Stochastic game	one-to-one deception	URL phishing	No assumption	A Stochastic model to analyze phishing
Zhao *et al.* (2016) [40]	Investigating the problem of setting personalized email filtering thresholds against sequential spear phishing attacks.	Stackelberg game	- one attacker - one defender with one/multiple credentials- one-to-one deception	Spear phishing	They assume that there is a network filter in the defender network, which makes a first line defence. The defender controls the probability that malicious emails will pass the filter and the probability that normal emails are filtered.	Two proposals to optimize the defender utilities according to filtering threshold (when it is considered one credential and several credentials)
Pawlick and Zhu (2017) [39]	Modelling one-to-many deception via game theory by combining Signalling game with Poisson game	Signalling and Poisson game	Phisher vs. many receiversvariable number of receivers; receivers of multiple types-Deception is strategic - Asymmetric information, Dynamic &one-to-many& perfect game, incomplete information	Phishing in general	Receivers are of different types: receivers able to detect deception with active defence (resp. no active defence) and receivers not able to detect with no defence.	A Poisson Signalling Game capturing situations with multiple receivers with different level of expertise.

Table 5 continued from previous page

	Objective	Game model	Game characteristics	Type of phishing	Defender assumptions	Results
Figueroa *et al.* (2017) [38]	Modelling the interaction between classifiers and opponents to detect fake e-mails (adversarial classification)	Signalling games	- Players: classifiers, opponents- Dynamic game -Incomplete information- many-to-many deception-QRE-based algorithms (S-QRE, P-QRE, and AAO-SVM)	Email phishing	The classifier can make false predictions (false positive and false negative)	A Classifier which changes the margin error dynamically as the game evolves, including an embedded awareness of the adversarial environment.
Zhu and Rass (2018) [41]	Modelling each of the three major temporal phases of general advanced persistent threats (APT) as a game and connecting the games at the transition points between the phases.	Overall sequential game: Phase 1: repeated gamePhase 2 : sequential gamePhase 3: repeated game	many-to-many two-player game, with each player physically being a whole team of actors	Advanced persistent threats in general. A case study for phishing in general	Players can differ in each phase of the APT game. Authors assume that there is a filter tool in each phase.	Authors propose and connect different game-theoretical models as well as optimal defences of each APT stage.
Our proposal	Modelling spear-phishing based on QRE game theory-based approach to predict phisher's future actions.	Sequential and non-zero and non-cooperative and repeated game	one-to-one deception; incomplete information; imperfect information	Spear-phishing	The defender is an employee with no phishing defences. The network servers have no effective and updated anti-phishing tools. Email accounts do not have filtering tools embedded.	Game theoretical models to acquire knowledge from interactions between the phisher and the victims, to predict the phisher's next actions according to knowledge from the past interactions and to recommend defensive actions.

7. Conclusion and future works

Attackers exploit spear-phishing attacks to infiltrate cyber systems through employees to gain sensitive information from companies. Researchers try to develop approaches to make these attacks unsuccessful. The approach proposed in this work consists of acquiring knowledge from interactions between the phisher and the victims, to predict the phisher's next actions according to knowledge from the past interactions and to recommend actions on the victim side. In this regard, this work adapted a game between IDS agents and insiders to propose a QRE game theory-based approach to predict the phisher's future intent according to the past actions of both players. A repeated and extensive game has been modelled to represent as many as possible strategies developed by opponents. The Nash Equilibrium provided that the phisher refer to spoof address emails and incentive victims to pursue conversations via phone calls. NE provides that, in this case, the potential victims renounced to avoid any risks. This situation reveals that the proposed model is reliable. The simulation of the game model, on Matlab, has been exploited to predict the future behavior of the phisher based on previous interactions. Three case studies have been drawn. For instance, let us take the case with the two past iterations {phisher: using fake attachment – victim: using antivirus}, {phisher: disguising email contents – victim: anti-phishing training}. Based on that two-historical interactions,

- The model found that the phisher will more likely continue by disguising the mail contents since it was successful despites countermeasures.
- The model suggests to dedicate respectively (96.37%, 1.04%, 2.15%, 0.42%) of its resources to mitigate each of four attack strategies.

The model has also been able to predict the further phisher's actions concerning any other experimented cases. The prediction has been coupled to a recommendation scheme of appropriate allocation of resources to invest to strengthen user protection. The complexity related to the construction of the game with the calculation of prediction probabilities strongly linearly depends on the number of assignments in the historical interactions. The implementation of the model has a linear temporal complexity. Future works will consist of three axes. The first axe consists of identifying and estimating significant parameters required to evaluate the attacker and the defender's losses during an email phishing attack and to integrate the results obtained to the different recommendations proposed by the model. The second axe consists of proposing an approach dealing with a method of profiling phishing attack strategies to combine with the model developed in this work. The third axe consists of extending modelling of interactions between one defender and several attackers.

References

[1] Jang-Jaccard J, Nepal S. A Survey of Emerging Threats in Cybersecurity. Journal of Computer and System Sciences. 2014 aug;80(5):973–993.

[2] Lohani S. Social Engineering: Hacking into Humans. International Journal of Advanced Studies of Scientific Research. 2019 feb;4(1):10.

[3] APWG. Phishing Activity Trends Report 4th Quarter 2018. Anti-Phishing Working Group; 2019.

[4] Goel D, Jain AK. Mobile Phishing Attacks and Defence Mechanisms: State of Art and Open Research Challenges. Computers & Security. 2018;73:519–544.

[5] Salahdine F, Kaabouch N. Social Engineering Attacks: A Survey. Future Internet. 2019 apr;11(4):89.

[6] Aleroud A, Zhou L. Phishing Environments, Techniques, and Countermeasures: A Survey. Computers & Security. 2017 jul;68:160–196.

[7] Chiew KL, Yong KSC, Tan CL. A Survey of Phishing Attacks: Their Types, Vectors and Technical Approaches. Expert Systems with Applications. 2018 sep;106:1–20.

[8] Gupta BB, Tewari A, Jain AK, Agrawal DP. Fighting Against Phishing Attacks: State of the Art and Future Challenges. Neural Computing and Applications. 2017 dec;28(12):3629–3654. Available from: http://link.springer.com/10.1007/s00521-016-2275-y.

[9] Butavicius M, Parsons K, Pattinson M, McCormac A. Breaching the Human Firewall: Social engineering in Phishing and Spear-Phishing Emails. 2016 may;Available from: http://arxiv.org/abs/1606.00887.

[10] Hong J. The State of Phishing Attacks. Communications of the ACM. 2012 jan;55(1):74–81.

[11] Mookjung K, Sangjin L. A Study on the Interrelationship between Types of DISC Personality and Cyber Security Threats: Focused on Spear Phishing Attack Cases. Journal of the Korea Information Security Society. 2019 Feb;29(1):215–223.

[12] Han Y, Shen Y. Accurate Spear Phishing Campaign Attribution and Early Detection. In: Proceedings of the 31st Annual ACM Symposium on Applied Computing. SAC '16. ACM; 2016. p. 2079–2086.

[13] Rajivan P, Gonzalez C. Creative Persuasion: A Study on Adversarial Behaviors and Strategies in Phishing Attacks. Frontiers in Psychology. 2018 feb;9.

[14] Nicho M, Fakhry H, Egbue U. When Spear Phishers Craft Contextually Convincing Emails. In: The 17th International Conferences WWW/Internet 2018 and Applied Computing 2018; 2018. .

[15] Stembert N, Padmos A, Bargh MS, Choenni S, Jansen F. A Study of Preventing Email (Spear) Phishing by Enabling Human Intelligence. In: the 2015 European Intelligence and Security Informatics Conference (EISIC'15). IEEE; 2015. p. 113–120.

[16] Caporarello L, Magni M, Pennarola F. One Game Does not Fit All. Gamification and Learning: Overview and Future Directions. In: In: Lazazzara A., Nacamulli R., Rossignoli C., Za S. (eds) Organizing for Digital

Innovation. Lecture Notes in Information Systems and Organisation. vol. 27. Springer, Cham; 2019. p. 179–188. Available from: http://link.springer.com/10.1007/978-3-319-90500-6{_}14.

[17] CJ G, Pandit S, Vaddepalli S, Tupsamudre H, Banahatti V, Lodha S. PHISHY - A Serious Game to Train Enterprise Users on Phishing Awareness. In: 2018 Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts, CHI PLAY '18 Extended Abstracts. ACM; 2018. p. 169–181.

[18] Landers RN, Auer EM, Collmus AB, Armstrong MB. Gamification Science, Its History and Future: Definitions and a Research Agenda. Simulation & Gaming. 2018 jun;49(3):315–337.

[19] Misra G, Arachchilage NAG, Berkovsky S. Phish Phinder: A Game Design Approach to Enhance User Confidence in Mitigating Phishing Attacks. In: Steven Furnell NLC, editor. Eleventh International Symposium on Human Aspects of Information Security & Assurance, HAISA 2017; 2017. p. 41–51.

[20] Amran A, Zaaba ZF, Singh MM, Marashdih AW. Usable Security: Revealing End-Users Comprehensions on Security Warnings. Procedia Computer Science. 2017 jan;124:624–631.

[21] Jamil A, Asif K, Ghulam Z, Nazir MK, Mudassar Alam S, Ashraf R. MPMPA: A Mitigation and Prevention Model for Social Engineering Based Phishing attacks on Facebook. In: 2018 IEEE International Conference on Big Data (Big Data). vol. 1. IEEE; 2018. p. 5040–5048.

[22] Tsalis N, Mylonas A, Nisioti A, Gritzalis D, Katos V. Exploring the Protection of Private Browsing in Desktop Browsers. Computers & Security. 2017 jun;67:181–197.

[23] Virvilis N, Mylonas A, Tsalis N, Gritzalis D. Security Busters: Web Browser Security vs. Rogue Sites. Computers & Security. 2015 jul;52:90–105.

[24] Santos L, Rabadao C, Goncalves R. Intrusion Detection Systems in Internet of Things: A Literature Review. In: 2018 13th Iberian Conference on Information Systems and Technologies (CISTI). IEEE; 2018. p. 1–7.

[25] Singh R, Kumar H, Singla RK, Ketti RR. Internet Attacks and Intrusion Detection System: A Review of the Literature. Online Information Review. 2017 apr;41(2):171–184.

[26] El-Alfy ESM. Detection of Phishing Websites Based on Probabilistic Neural Networks and K-Medoids Clustering. The Computer Journal. 2017 dec;60(12):1745–1759.

[27] Jain AK, Gupta BB. Towards Detection of Phishing Websites on Client-Side using Machine Learning based Approach. Telecommunication Systems. 2018 aug;68(4):687–700.

[28] Mahdavifar S, Ghorbani AA. Application of Deep Learning to Cybersecurity: A Survey. Neurocomputing. 2019 jun;347:149–176.

[29] McCluskey L, Thabtah F, Mohammad RM. Intelligent Rule-based Phishing Websites Classification. IET Information Security. 2014 may;8(3):153–160.

[30] Sahoo D, Liu C, Hoi SCH. Malicious URL Detection using Machine Learning: A Survey. 2017 jan;Available from: http://arxiv.org/abs/1701.07179.

[31] Shibahara T, Yamanishi K, Takata Y, Chiba D, Akiyama M, Yagi T, et al. Malicious URL Sequence Detection using Event De-noising Convolutional Neural Network. In: 2017 IEEE International Conference on Communications (ICC). IEEE; 2017. p. 1–7.

[32] Jain AK, Gupta BB. A Novel Approach to Protect Against Phishing Attacks at Client Side using Auto-updated White-List. EURASIP J Inf Secur. 2016 dec;2016(1):9. Available from: https://jis-eurasipjournals.springeropen.com/articles/10.1186/s13635-016-0034-3.

[33] Sonowal G, Kuppusamy KS. PhiDMA – A Phishing Detection Model with Multi-Filter Approach. Journal of King Saud University - Computer and Information Sciences. 2017;p. 1–14.

[34] Jain AK, Gupta BB. Phishing Detection: Analysis of Visual Similarity Based Approaches. Security and Communication Networks. 2017 jan;2017:1–20.

[35] Shiva S, Dasgupta D, Wu Q. Game Theoretic Approaches to Protect Cyberspace. University of Memphis, Department of Computer Science; 2010. Available from: https://apps.dtic.mil/dtic/tr/fulltext/u2/a519126.pdf.

[36] Tchakounté F, Nyassi VS, Udagepola KP. True Request–Fake Response: A New Trend of Spear Phishing Attack. Journal of Network Security. 2019;7(3):1–17.

[37] Yu M, Liu C, Qiu X, Zhao S. Modelling and Analysis of Phishing Attack using Stochastic Game Nets. In: International Conference on Cyberspace Technology (CCT 2013). Institution of Engineering and Technology; 2013. p. 300–305.

[38] Figueroa N, L'Huillier G, Weber R. Adversarial Classification using Signaling Games with an Application to Phishing Detection. Data Mining and Knowledge Discovery. 2017 jan;31(1):92–133.

[39] Pawlick J, Zhu Q. Phishing for Phools in the Internet of Things: Modeling One-to-Many Deception using Poisson Signaling Games. 2017 mar;Available from: http://arxiv.org/abs/1703.05234.

[40] Zhao M, An B, Kiekintveld C. Optimizing Personalized Email Filtering Thresholds to Mitigate Sequential Spear Phishing Attacks. In: Thirtieth AAAI Conference on Artificial Intelligence. AAAI Press; 2016. p. 658–664.

[41] McCubbins MD, Turner MB, Weller N. Testing the Foundations of Quantal Response Equilibrium. In: International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction. SSRN; 2013. p. 10p.

[42] Binmore K. Playing for Real : a Text on Game Theory. Oxford University Press; 2007.

[43] Osborne MJ, Rubinstein A, Osborne M, Rubinstein A. A Course in Game Theory. vol. 1. MIT Press; 1994.

[44] Roy S, Ellis C, Shiva S, Dasgupta D, Shandilya V, Wu Q. A Survey of Game Theory as Applied to Network Security. In: the 2010 43rd Hawaii International Conference on System Sciences. IEEE; 2010. p. 1–10.

[45] Richard D M, Andrew M, Theodore T. Gambit: Software Tools for Game Theory, Version 16.0.1. Gambit Project; 2019. Available from: https://buildmedia.readthedocs.org/media/pdf/gambitproject/latest/gambitproject.pdf.

[46] Shen S, Hu K, Huang L, Li H, Han R, Cao Q. Quantal Response Equilibrium-Based Strategies for Intrusion

Detection in WSNs. Mobile Information Systems. 2015 aug;2015:1–10.
[47] Mckelvey RD, Palfrey TR. Quantal Response Equilibria for Extensive Form Games. Experimental Economics. 1998;1(1):9–41.
[48] Kantzavelou I, Katsikas S. A Game-based Intrusion Detection Mechanism to Confront Internal Attackers. Computers & Security. 2010 nov;29(8):859–874.
[49] Chin T, Xiong K, Hu C. Phishlimiter: A Phishing Detection and Mitigation Approach Using Software-Defined Networking. IEEE Access. 2018;6:42516–42531.
[50] Qamar A, Karim A, Chang V. Mobile Malware Attacks: Review, Taxonomy & Future Directions. Futur Gener Comput Syst. 2019;97:887–909.
[51] Volkamer M, Renaud K, Reinheimer B, Kunz A. User experiences of TORPEDO: TOoltip-poweRed Phishing Email DetectiOn. Computers & Security. 2017 nov;71:100–113.
[52] OpenPhish. Timely. Accurate. Relevant Threat Intelligence.;. Available from: https://www.openphish.com/.
[53] Gupta S, Sachdeva S. Invitation or Bait? Detecting Malicious URLs in Facebook Events. In: 2018 Eleventh International Conference on Contemporary Computing (IC3). IEEE; 2018. p. 1–6.
[54] Shirazi H, Bezawada B, Ray I. "Kn0w Thy Doma1n Name": Unbiased Phishing Detection Using Domain Name Based Features. In: The 23nd ACM on Symposium on Access Control Models and Technologies, SACMAT '18. ACM; 2018. p. 69–75.
[55] Zhu Q, Rass S. On Multi-Phase and Multi-Stage Game-Theoretic Modeling of Advanced Persistent Threats. IEEE Access. 2018 Mar;6:13958–13971.
[56] Yu M, Liu C, Qiu X, Zhao S. Modelling and Analysis of Phishing Attack using Stochastic Game Nets. In: International Conference on Cyberspace Technology (CCT 2013). Institution of Engineering and Technology; 2013. p. 300–305.