

Gaming for Phishing Training: Teaching users to be safe through an interactive  
roleplaying game

A Thesis

Submitted to the Graduate Faculty of the  
University of New Orleans  
in partial fulfillment of the  
requirements for the degree of

Master of Science  
in  
Computer Science

by

Saroj Duwal

B.S. University of New Orleans, 2019

May, 2022

# Table of Contents

	Page
List of Figures .....	iv
List of Tables .....	vi
Abstract .....	viii
1. Introduction .....	1
1.1 What is phishing? .....	1
1.2 Current Mitigations .....	3
1.3 Literature Review .....	5
1.3.1 Serious games .....	5
1.3.2 Board and card games .....	7
1.3.3 Phishing Link (URL) training.....	7
1.3.4 Role playing game .....	8
1.4 Objective.....	10
2. System Description .....	12
2.1 Game Description .....	12
2.1.1 Game Story .....	12
2.2 Mechanics .....	13
2.2.1 Components.....	13
2.2.1.1 Attacker .....	13
2.2.1.2 Marketplace.....	15
2.2.1.3 Emails.....	18
2.2.2 Email efficiency .....	25
2.2.3 Previous Iteration .....	28
2.2.4 Weekly Goals .....	29
3. Evaluation.....	31
3.1 Test Design .....	31
3.2 Methodology .....	32
3.3 Results .....	33
3.3.1 Pre-Survey .....	33
3.3.2 Game .....	34
3.3.3 Post-Survey .....	36
3.4 Pre vs Post Phishing Knowledge .....	38

3.4.1	Evaluating emails in post survey.....	39
3.5	Game Time vs performance .....	40
3.6	Questionnaire .....	42
3.7	Participant Feedback .....	44
4.	Discussion .....	45
4.1	Insights .....	45
4.1.1	Mailing Domains and Subdomains .....	45
4.1.2	Issues with email content .....	46
4.2	Limitations and Future works .....	46
4.3	Conclusion.....	47
References	.....	48
Vita	.....	50

# List of Figures

FIGURE	Page
1.1 Phishing email sent to John Podesta .....	3
1.2 Browser cues on links .....	4
1.3 Garfield's Count Me In .....	6
1.4 Killer Flu .....	6
1.5 What.Hack gameplay .....	9
2.1 Screenshot of initial state of the game .....	12
2.2 Screenshot of the attacker module on week 4 .....	14
2.3 Screenshot of marketplace .....	15
2.4 Top 10 top level domains present Tranco list .....	16
2.5 Different stages of the marketplace .....	19
2.6 Emails generated after training vs before training on passive skills .....	20
2.7 Example of a targeted email generated by the system.....	21
2.8 Examples of hiding the actual link behind text or button .....	22
2.9 Example of a URL shortener option in game .....	23
2.10 Example of a email generated with link confusion .....	25
2.11 Fake email sender .....	26
2.12 Spoofing option in game .....	27
2.13 Initial version of the game .....	29
3.1 The Gmail clone used during evaluation with list of custom emails .....	32
3.2 Participants performance on the phishing emails in the evaluation .....	40

3.3	Comparison of f1-score before and after playing the game with time (Maybe answers are included).....	41
3.4	Responses to the first 7 questions .....	43
3.5	Responses to the 2nd set of questions .....	44
4.1	An email sent by Lyft .....	46

# List of Tables

TABLE		Page
1.1	Different type of training games and their main objectives .....	10
2.1	Different skills and their effect in the game.....	14
2.2	Different second level domain and their similarity with "paypal" .....	18
2.3	URL shortener examples.....	23
2.4	Efficiency of each option .....	27
2.5	Similarity of spoofed email domain and points assigned .....	28
2.6	Different week with their corresponding skills and goals .....	30
3.1	Average performance in pre-survey (I).....	34
3.2	Average performance in pre-survey (II).....	34
3.3	Average time user took for each week (in minutes) .....	35
3.4	Different domains purchased in the marketplace in game .....	36
3.5	Some spoofing emails used by participants .....	37
3.6	Average performance in post-survey (I).....	37
3.7	Average performance in post-survey (II).....	38
3.8	Change in average score in pre-survey vs post-survey .....	39
3.9	Likert scale questions relating opinion of the game .....	42

# Abstract

Phishing attacks are challenging to detect and can have severe consequences. In 2020 alone, phishing attacks cost organizations more than \$1.8 billion. In addition, attacks can have effects other than money, as shown by the infamous case of John Podesta during the 2016 US presidential election, in which staffs were tricked into sharing passwords by fake Google security emails, granting access to confidential information. Vulnerabilities such as these are partly due to insufficient and tiresome user training. We can minimize such vulnerabilities with better and more engaging training against phishing. To address this, we have designed and developed an interactive game to teach users phishing concepts by placing the player as an attacker. Our user study shows that our game was engaging, and after playing the game, participants had a better understanding of phishing and recognizing phishing emails.

**Keywords:** Anti-Phishing, Serious games, Cyber Security, Training

# 1. Introduction

The rapid internet adoption in everyday life and the workplace has presented us with new security challenges. Many users are more active on the internet, giving attackers more opportunities to attack these unsuspecting victims. There are various technical security measures such as firewall, encryption, threat hunting software, and engaging automation to mitigate these challenges. However, studies have shown that the human layer is the weakest link in the security chain [1] and attackers usually start by targeting the most vulnerable link before performing other detrimental attacks. These attacks with human interaction are generally known as "Social Engineering Attacks."

Prevalent social engineering attacks such as phishing, pretexting (inventing a scenario to convince victims to divulge information they should not divulge), baiting (promising an item or good to trick the victim), quid pro quo (promising something in exchange for information), and tailgating (someone without the proper authentication follows an authenticated employee into a restricted area) use psychological manipulation to trick users into making security mistakes or giving away sensitive information. This thesis will focus on phishing and different detection techniques through our role-playing gameplay.

## 1.1 What is phishing?

Phishing is one of the most prevalent social engineering attacks in which attackers target users by contacting them through email, telephone, or text message by posing as a legitimate entity [2, 3] to gain trust. Phishers try to obtain personal (potentially sensitive) data, including login credentials and credit card numbers. They can use this information to create fake accounts, ruin credit, steal money, or identity.

Unfortunately, these attacks are challenging to detect because attackers use the computing infrastructure to trick the victim into doing something while the computing system is working as intended. Due to this, even users with a high-end security system can be victims. An example of



such is the infamous case of John Podesta [4], Hilary Clinton's campaign chairman for the 2016 presidential election. The "googlemail.com" in the domain successfully tricked John Podesta and the Clinton campaign's computer help desk to trust the email (See fig:1.1).

Phishing attacks are constantly evolving with different tricks. For example, Podesta's email shows it was initially generated from "googlemail.com," making it seem like it might be from Google, but it was not true. Attackers can use different spoofing techniques to hide the sender's identity. Another common trick attackers use (also present in Podesta's email) is to confuse the user with links hidden behind some text/button or confuse the user with redirecting links (example: TinyURL). As a result, the displayed text/link might not be the final destination. Podesta's team's failure to deal with this phishing email led to leaks of more than 11,000 emails which included private conversations with 2016 presidential nominee Hillary Clinton [5].

Successful phishing attacks are expensive to organizations. In 2020 alone, phishing attacks cost US businesses more than \$1.8 billion, up from \$1.7 billion in 2019 [6]. These attacks can lead to credential/account compromise, giving the attacker access to sensitive information, et cetera. Attackers may try to use these data for extortion. For example: In 2014, an attack was successful on an invasion of celebrity iCloud accounts, leading to the leaking of nude photos. The leak was initially considered due to a breach on Apple services, but it was a phishing attack. The attackers pretended to be Apple and Google and asked users to change their password [7, 8].

Phishing attacks are continuously rising and have doubled since early 2020. In July 2021 alone, the Anti-Phishing Working Group (APWG), an international consortium that attempts to eliminate fraud and identity theft caused by phishing, saw 260,642 phishing attacks [3]. Additionally, Proofpoint, a security enterprise that provides cybersecurity products relating to emails and digital information, found that more than 75% of organizations faced phishing attacks in 2021 [9]. These uprising trends in phishing attacks have shown some serious need for mitigations.

```

> *From:* Google <no-reply@accounts.googlemail.com>
> *Date:* March 19, 2016 at 4:34:30 AM EDT
> *To:* [REDACTED]ta@gmail.com
> *Subject:* *Someone has your password*
>
> Someone has your password
> Hi John
>
> Someone just used your password to try to sign in to your Google Account
> [REDACTED]ta@gmail.com.
>
> Details:
> Saturday, 19 March, 8:34:30 UTC
> IP Address: 134.249.139.239
> Location: Ukraine
>
> Google stopped this sign-in attempt. You should change your password
> immediately.
>
> CHANGE PASSWORD <https://bit.ly/1PibSU0>
>
> Best,
> The Gmail Team
> You received this mandatory email service announcement to update you about
> important changes to your Google product or account.
>

```

Figure 1.1: Phishing email sent to John Podesta

## 1.2 Current Mitigations

The prevention of phishing attacks can be divided into three steps [10]—the first step is preventing the attack from reaching the end-user. We have seen multiple studies on phishing prevention with the help of the machine learning models [11, 12]. Machine learning approaches such as K-nearest, XGBoost, CNN, RCNN, Random forest, et cetera. are commonly used to detect patterns and generalize phishing attacks. Some of the models have shown promises with more than 90% accuracy—however, a study conducted by What.Hack has shown that only one of the ten anti-phishing tools tested could correctly identify over 90% of phishing websites. That tool also incorrectly identified 42% of legitimate websites as fraudulent [13]. Moreover, attackers are

always looking for the best way to bypass these automated systems and develop new techniques. The evolving nature of phishing attacks calls for an additional layer of security on top of the prevention layer.

If the attacks reach the user, the next step to secure the user is by warning them. Most modern web browsers and email clients warn users of any suspicious activities they detect. For example, the browser actively warns users with pop up for probable phishing sites. In addition, browsers provide passive hints to understand URLs better. Browsers use different shades of white to inform the user about a "fully qualified domain name (FQDN)" (also called absolute domain name), the complete domain name for a specific host on the internet. Figure 1.2 shows a use case for such a hint. Attackers will intentionally have a confusing link to trick users into clicking the link. For example, although "help.google.com.bubble.com/changepassword" seems like an email from Google, the actual domain is bubble.com. The domain owner can add any subdomain to domains they own, such as help.google.com.bubble.com, which can potentially be used in phishing attacks. Modern email clients provide similar hints for spam emails and notify the users if they can not verify the sender. Active warnings are more effective than passive signs [10]. Still, attackers can easily bypass these warnings by creating new sites and context-aware websites or emails every time they are flagged.

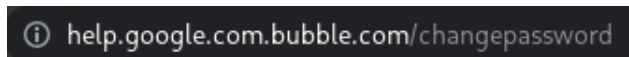


Figure 1.2: Browsers uses different shades to indicate the primary link.

The final step to avoid phishing emails is user training. A study done by Proofpoint shows that 34% of US respondents believe emails with familiar logos are safe [9]. The study indicates a general lack of awareness about phishing campaigns among the general population. As emails are the most frequent phishing attacks, many phishing training focuses on training users to detect phishing emails. Our gameplay will focus on emails to train the users against phishing attacks.

One of the most common tools to train users is cyber security videos and reading materials. However, Kumaraguru et al. saw that users seldom seek these materials and tend to ignore emails directing them to these materials [14]. In addition, they noticed that most users do not spend much time reading security-related tutorials. This calls for an interactive training program to keep the user focused and engaged during training.

We have seen new and existing training materials incorporating gaming techniques. Gamification has been gaining rapid popularity over the past decade[15]. It increases engagement by incentivizing learners to pay attention and complete activities. We can observe existing training videos incorporating gaming techniques, such as letting users choose the correct option in the middle of training videos (a mini quiz game) and giving badges after completion. Newer training videos take gamification further and let learners play through various scenarios, make choices and see the rewards or consequences of their decision. For example, "Infosec's Choose Your Own Adventure Security Awareness" Game[16] has interactive storytelling to keep the user focused till the end of the video.

Gamification has improved the interactivity with the user, but existing training videos fail to cover the technical details that are commonly found in phishing emails. Our gameplay covers various technical aspects commonly found in phishing emails, such as domains, spoofing, and link hiding techniques attackers use to trick users.

## **1.3 Literature Review**

### **1.3.1 Serious games**

Gaming approaches in education have been used for over a decade[17]. There is a dedicated genre of games (typically online applications), termed serious games. These games communicate specific information that helps introduce relevant concepts and apply those concepts to solve problems. The primary purpose of these games is to promote learning alongside entertainment. With the help of different game design techniques (rewards, story progression, feedback systems), users are more engaged and immersed while learning. In addition, the virtual world also provides users

with a safe space to experiment without real-life consequences.

Serious games are used in many fields such as education, healthcare, and training. For example, "Garfield's Count Me In" [18] helps children in (special education) primary school practice their arithmetic skills. This math game contains different exercises or 'brick,' which form the foundation for a new layer of exercises. The game design help students master the first layer of exercises before moving to the next layer (basic to advanced).

"Killer Flu" [19] (one of many games by "Persuasive Games") is another example of a serious game that attempts to explain how flu mutates and spreads and how challenging it can be for a deadly strain to affect a large population geographically. The game helps spread awareness by making the player take the role of the flu itself, trying to mutate and then spread it in various conditions. Serious games (such as Killer Flu) can place the user as any character in the game to get the idea across. We use a similar concept in our game by placing the player as the attacker (rather than the victim as many existing training materials do.)



Figure 1.3: Garfield's Count Me In

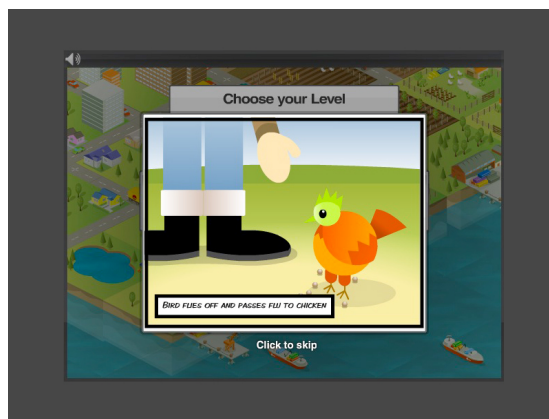


Figure 1.4: Killer Flu

There have been various studies about using games as a practical phishing training module. Hendrix et al. [20] compared the effectiveness of cyber security training tools with some popular games designed for cyber security training. They found that existing games, in general, had a positive impact on learning, with positive education experience.

We can broadly classify current training games into three main categories. We discuss each of these categories and provide some examples of each type in the following sections.

### **1.3.2 Board and card games**

There have been studies based on non-computer-based games. For example, Control-Alt-Hack [21] and "smells phishy" [22] are card games aimed to train users against phishing. Both the games show promise in their approaches and teach the user what to be aware of (such as spelling mistakes, phishing links) through their gameplay. After playing the game, users reported higher efficiency and ability to detect phishing emails.

Although both the games show promise in their approach, non-computer-based games have some inherent limitations. The games have a barrier of entry as it requires pre-setup (with the need for the cards and boards). Furthermore, once the games are deployed, they are permanent, limiting their ability to train and evolve against new phishing attacks. Finally, board and card games fail to communicate the context of the attack and lack examples of where they might be used. For example, although the game might have a "hiding links" card, the users lack knowledge on how and when it might be used. The limited skills these games provide may not be best suited as an individual training module.

### **1.3.3 Phishing Link (URL) training**

There have been numerous computer games about phishing. However, many studies focus on one common category: training users to verify phishing links. Anti-Phishing Phil [23] is one of the pioneers in this field. Their gameplay puts the user as a fish. The goal of the fish is to grow larger by eating the good bugs (non-phishing links) and avoiding the bait (phishing links). The game has four different levels, with each round focusing on another type of deceptive URL. Players move to the next level after correctly identifying six out of eight URLs.

There are other similar games to Anti-Phishing Phil. For example, Phish Phinder [24] builds upon the Anti-Phishing Phil storyline and game design. However, it differentiates itself from Anti-Phishing Phil by integrating self-efficacy in the game design to enhance phishing avoidance

motivation and behavior among users. They were able to positively impact self-efficacy by adding conceptual knowledge (through challenges) and procedural knowledge (through increasing levels and repetition of conceptual knowledge).

"Building Confidence not to be Phished"[25] by Baral et al. developed a game prototype aimed to enhance an individual's self-efficacy in phishing threat avoidance. They developed a balloon shooter game where the main character has to shoot balloons with legitimate URLs (similar to previous games). Although the links and hints are displayed in a different context, this game tries to achieve the same goal as previous games.

All these games have one thing in common: they teach users how to identify legitimate URLs through their gameplay. As URLs are one of the most critical factors while detecting phishing emails, gameplay dedicated to recognizing phishing URLs serves as a suitable training module. Anti-Phishing Phil results show that their game has a good impact compared to existing training material to differentiate legitimate links from phishing links.

However, these games might not be suitable as standalone training against phishing. Although URL training games are a sound training module, these games fail to train users on some common tricks seen in phishing attacks. For example, one of the most significant limitations of these games is the lack of context on where the link might appear. As such, attackers can use different link hiding techniques to trick the user into clicking the link. Moreover, attackers use psychological manipulation to trick users into clicking the link by creating a sense of urgency, fake giveaways, or making it seem like an email from somebody individuals know to trick people into clicking the link.

#### **1.3.4 Role playing game**

"What.Hack" [13] saw the shortcomings of the link-based game and developed gameplay that train the user on links as well as email context. It puts the user as a bank employee required to process emails to acquire contracts and protect their network from cybercriminals. The game approaches the training by having the user role play as a victim and looking at different techniques found in actual attacks. The game's goal is to block phishing attempts and allow legitimate emails.

It simulates the harmful effects of phishing by "firing" the employee in-game if they allow too many phishing emails, take too long, or misclassify a significant number of legitimate emails as phishing emails.

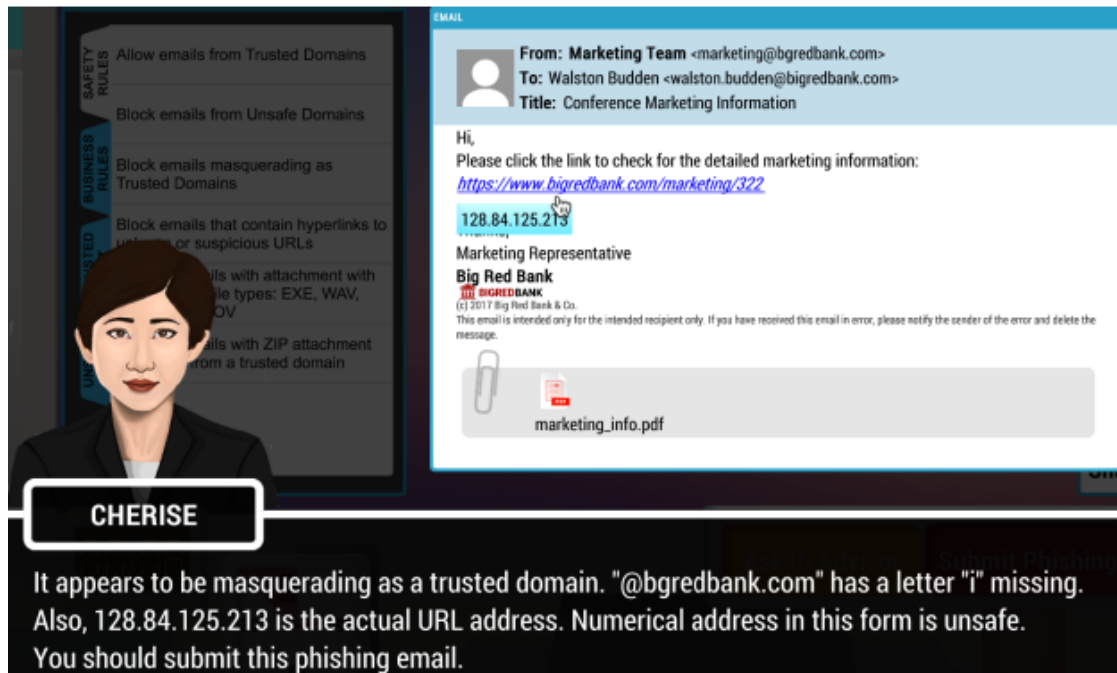


Figure 1.5: What.Hack gameplay

"What.Hack" gameplay incorporates previous studies' different techniques and builds upon them. In addition, its context-based email training module solves one of the drawbacks of link-based training games. Since players are looking at full emails (both phishing and legitimate) obtained from the real world, users can better recognize the context of the email and what to look out for in emails to stay protected. Moreover, the emails generated in this game successfully incorporate the objective of primary link-based game goals.

The result from "What.Hack" shows clear improvement regarding link-based games. In comparison to Anti-Phishing Phil, "What.Hack" improved players correctness in identifying phishing emails by 36.7% [13].



## 1.4 Objective

"What.Hack" clearly demonstrated that role-playing games with contextual emails were more effective than existing gameplays. Unfortunately, we could not find any other significant study that tried to build upon this finding. Therefore, we have developed gameplay inspired by "What.Hack" but approached the role-playing aspect as an attacker instead of a victim.

Current phishing training programs focus on training the user as a victim. Our game tries to approach training from a new direction by placing the player as an attacker. This approach will let the users see what the phisher might concentrate on while creating a phishing email and, in turn, use that knowledge to detect phishing emails. The training objective of our game is similar to existing games and tries to build upon it. Table 1.1 compares the main training objective of our game with existing games.

Game Type	Description	URL	Spear	Spoof
<b>Link Based</b>	Teach users to differentiate phishing links from non-phishing links	✓		
<b>Board Game</b>	Teach users high- level security concepts	✓	✓	✓
<b>What.Hack</b>	Teach players to defend against phishing attempts in realistic simulation game (playing as a victim)	✓	✓	
<b>Our Game</b>	Teach players to recognize and defend against phishing attempts by role playinig as a victim	✓	✓	✓

Table 1.1: Different type of training games and their main objectives

Our goals for the study can be summarized as:

- We designed and developed a phishing training game that attempts to incorporate the objectives of the current existing games and build upon them. Our game includes link hiding training and provides context to those links.

- We compare the participants' performance before and after playing the game. We evaluate users' results and try to understand user thought processes behind each decision.
- Finally, we list some of the insights found during the survey and list potential mitigation and training.

## 2. System Description

This chapter will discuss our game design, the story, mechanics, and tools used, and briefly discuss the previous iteration.

### 2.1 Game Description

The game is developed on React<sup>1</sup> with Chakra UI<sup>2</sup>. We use Supabase<sup>3</sup> to keep logs of user interaction with the game. Figure 2.1 shows the initial state of the game.

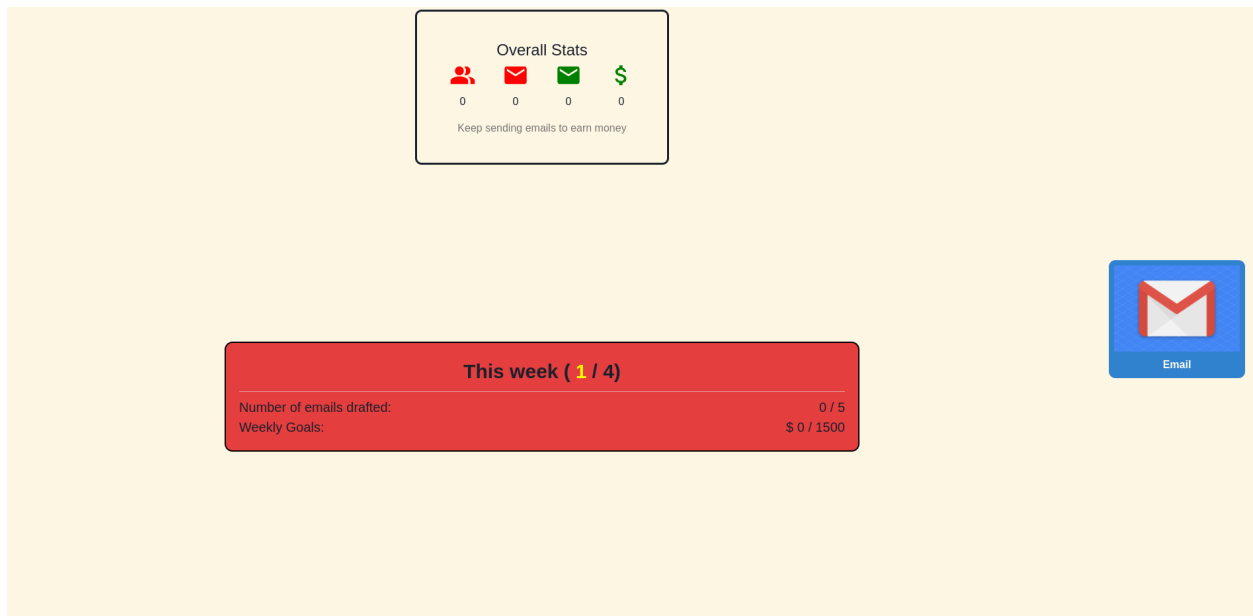


Figure 2.1: Screenshot of initial state of the game

#### 2.1.1 Game Story

The game's main character has taken a large loan from a loan shark. The goal of the game is to pay off the loan in time. However, as the loan is substantial, he cannot earn enough money

---

<sup>1</sup><https://reactjs.org/>

<sup>2</sup><https://chakra-ui.com>

<sup>3</sup><https://supabase.com/>

through hard work and uses phishing tricks to scam people. However, the main character does not have enough skills, so he hires a helper to create phishing emails. Together, they have decided to impersonate PayPal. The player has four weeks to earn enough money to pay back.

## **2.2 Mechanics**

One of our main objectives while developing the game was to streamline the player experience. We divided the game into four weeks (parts) to achieve this goal. Each week's progression will unlock specific skills users' can use to create an email. We will discuss specific week progression and how it ties the game together in later sections.

### **2.2.1 Components**

Before we deep dive into the game's flow, we have to discuss individual components. At the top level, we can divide the game into three components: attacker, marketplace, and email generation.

#### *2.2.1.1 Attacker*

The attacker module handles training the helper with available skills. There are six different skills that the player can train the helper on, namely spelling, grammar, styling, links, spoofing, and research. We divide these skills as language skills (spelling and grammar) and technical skills (styling, links, spoofing, research). Language skills are passive skills in the game, whereas technical skills, except for styling, are active skills.

Training on passive skills will improve the quality of the email generated by the helper without any additional input from the player. In contrast, active skills will give the player more options while generating the email. For example, after the player trains their helper on spelling, the attacker will stop making spelling errors without additional feedback from the user. Training the helper on links will give the user different options to hide the links while creating the email. Table 2.1 lists the skills and their effect in the game in brief. We will talk about the different properties activated by each skill in the email generation section.

We chose the skills in the game to replicate the training objective of existing training modules and common properties found in phishing emails. Each skill in the game has a training cost asso-

Skills	Active/ Passive	Cost	Effect
Spelling	Passive	1,000	Creates emails without spelling errors
Grammar	Passive	1,000	Create emails without grammar errors
Styling	Passive	2,000	Create stylized emails with better header, footer, and images
Links	Active	3,000	Unlocks different techniques to hide the link while sending email
Research	Active	3,000	Gives the user option to generated targeted emails
Spoofing	Active	4,000	Gives the user ability to spoof the email

Table 2.1: Different skills and their effect in the game

ciated with it. We associated some costs with skills to represent training requires some resources. The cost is kept at a minimum to let the user unlock it as soon as possible but scaled such that more efficient skills have a higher price than general skills. Keeping the cost low allowed the user to focus more on using those skills to generate emails rather than earning money to train attackers.

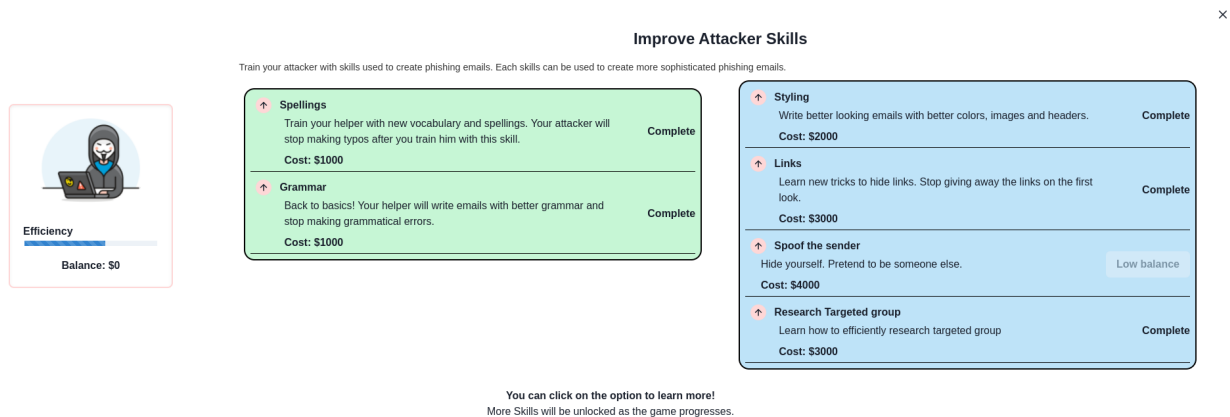


Figure 2.2: Screenshot of the attacker module on week 4

Although we tried to include all the common properties found in phishing emails, we could not itemize some general properties such as sense of urgency, generic greetings, too good to be true emails, et cetera. Instead of itemizing and adding more passive skills, we decided to limit the number of skills and let users actively concentrate on those. On the technical side, the limited number of properties to look at while generating emails made email generation much more manageable and allowed us to generate a broader range of emails. Current emails generated by the system still include the properties of phishing emails that we could not itemize. Exposing players to more phishing emails may help players recognize similar emails and patterns later.

### 2.2.1.2 Marketplace

The marketplace allows the player to purchase domains to use while generating emails (See figure 2.3). Existing training modules train the players to recognize phishing links (which are generated by the system) but do not allow players to try custom domains. In our gameplay, the attacker "link" skill teaches how phishing emails hide links to trick victims into clicking the link.

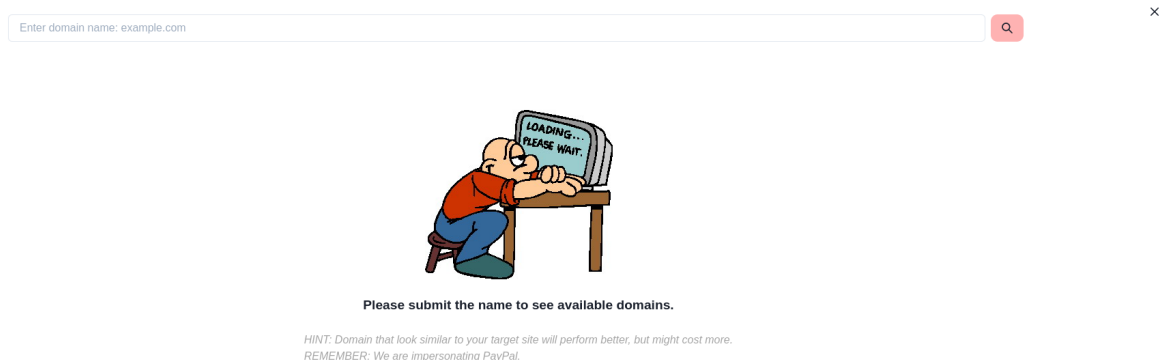


Figure 2.3: The marketplace accepts any valid domain name

The first step when letting the user purchase a domain is to check if the domain is valid. A

valid domain is a second-level domain followed by a top-level domain in our game. A top-level domain refers to the last segment of a domain name, and a second-level domain refers to the name just to the left of the top-level domain. For example, "test.com" is a valid domain where "test" is a second-level domain and "com" is the top-level domain. We validate the second-level domain with the help of the following regex code:

```
1  if (userLink.includes(" ") || !/^[a-zA-Z0-9-]*$/ .test(userLink))  
2  return;
```

We do not allow special characters (Fada Accent) in the domain name for simplicity.

There are 1,500 top-level domain [26]. For simplicity, we only allow users to choose from a predefined list filtered from Tranco list [27]. Tranco list provides us with the most popular one million domains, similar to Alexa top site. We filtered top-level domains that occurred a hundred times from Tranco's list and got 262 top-level domains. This limited number of top-level domains allowed us to incorporate commonly used domains while maintaining the game's simplicity.

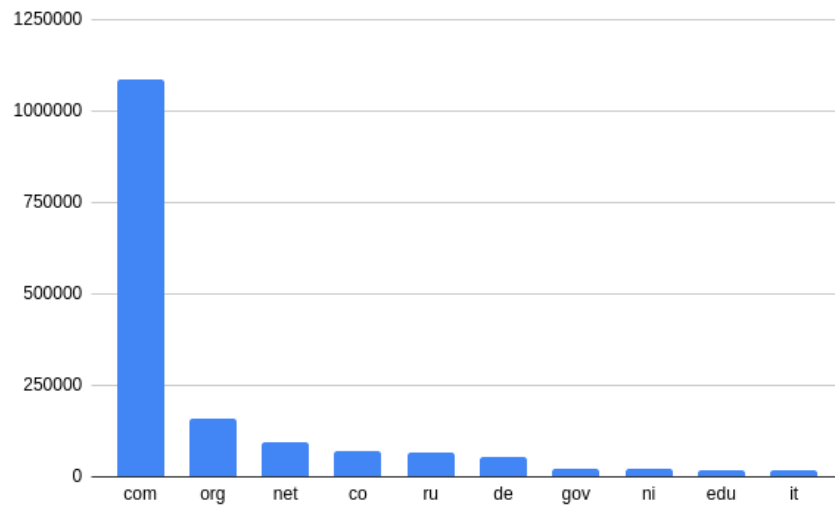


Figure 2.4: Top 10 top level domains present Tranco list

Players can choose any combination of valid characters for the second-level domain (validated by the regex pattern shown above). For example, "123test.com" is a valid domain. However,

"%ssda1.com" is not a valid domain. We maintain the top 1000 domains in the Tranco list to prevent purchasing popular and already existing domains. This list consists of popular sites such as "google.com," "facebook.com," "netflix.com," including "paypal.com" (which is used by our game to train the user). We had to trim the number of domains to a thousand as processing a million domains required significant processing time, impacting the gameplay.

Our game uses PayPal to trick the victims, so domains closer to PayPal will perform better. Similar to the real world, purchasing a new domain requires money. Domains similar to popular services (determined in the top 1000 domains) will have a higher cost. Since the user can purchase any domain and domain closer to the top thousand domains are more expensive, the cost of a domain purchased does not directly correlate to higher efficiency in the game.

We determine the closeness between two domains based on string similarity. We use Sørensen-Dice coefficient to compute the similarity between two strings. Mathematically, given two sets, X and Y, we can define Dice coefficient as:

$$DSC = \frac{2|X \cap Y|}{|X| + |Y|}$$

It produces a value between zero and one, making the cost calculation of domains much easier. Table 2.2 shows examples of some custom domains along with their similarity. We can see that a domain similar to "paypal" has a higher similarity. Therefore, we treat domains with higher similarity score as more efficient for our game. We do not use the top-level domain for cost calculation, as most players usually choose ".com," which impacts the string similarity.

The cost of the domain does not depend solely on similarity to "paypal". While calculating the cost, we get the maximum similarity with any of the domains in the top-1000 list. If the similarity with the existing domains is below 0.6, we assign a base price of 500 for the domain; else, the general cost of the domain is calculated as:

$$cost = 500 + (similarity * 100)^2 * 0.56$$

If the player tries to buy existing domain, the game suggests domains ending with alternate top level domains (See figure 2.5a). For example, if the player tries to buy "paypal.com", the game



Custom Domain	Similarity with "paypal"
paypale	0.90
palpay	0.80
paypl	0.66
appl	0
test	0

Table 2.2: Different second level domain and their similarity with "paypal"

suggests top 10 alternate top level domains such "paypal.org" or "paypal.net". The cost of such domain is not based on similarity. We use the frequency of top level domains in Tranco list and compute the cost as follows:

$$cost = (50 * \sqrt{10 - index}) * 25$$

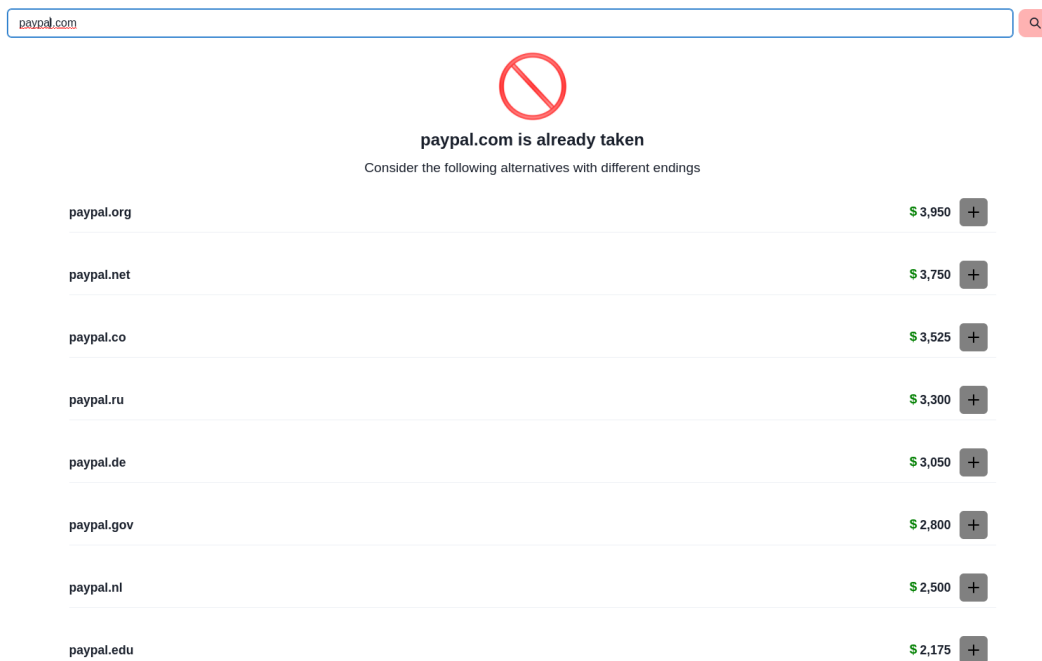
where index is the ranking of the top level domain based on frequency in Tranco list.

These formulas to calculate cost are not standard and were achieved through trial and error. Although not an actual scale, we set the cost to show that domains closer to real-world domains will have a higher cost.

### 2.2.1.3 Emails

The emails component ties up the game by generating and sending emails based on attacker skills and the current domain. The system randomly chooses an email from 30 available emails based on the user input (active and passive skills). These emails were handpicked to replicate real-life phishing attempts and include the most commonly found phishing emails. As a result, emails in the system include common phishing tricks used by attackers and different email models such as log-in, welcome emails, limited account emails, et cetera.

Before discussing sending email and efficiency, let us discuss how each skill impacts the email generation process. As mentioned in the attacker component section, passive skill does not require additional input from the player and improves the generated email after training them. Spelling,



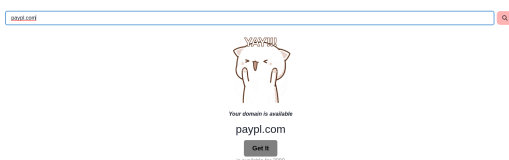
paypal.com

**paypal.com is already taken**

Consider the following alternatives with different endings

paypal.org	\$ 3,950	+
paypal.net	\$ 3,750	+
paypal.co	\$ 3,525	+
paypal.ru	\$ 3,300	+
paypal.de	\$ 3,050	+
paypal.gov	\$ 2,800	+
paypal.nl	\$ 2,500	+
paypal.edu	\$ 2,175	+

(a) Marketplace unavailable

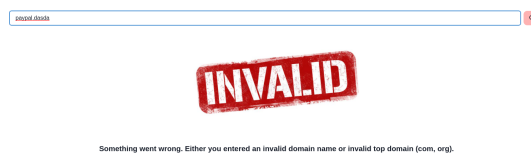


paypl.com

Your domain is available

Get it

(b) Marketplace available



INVALID

Something went wrong. Either you entered an invalid domain name or invalid top domain (.com, .org).

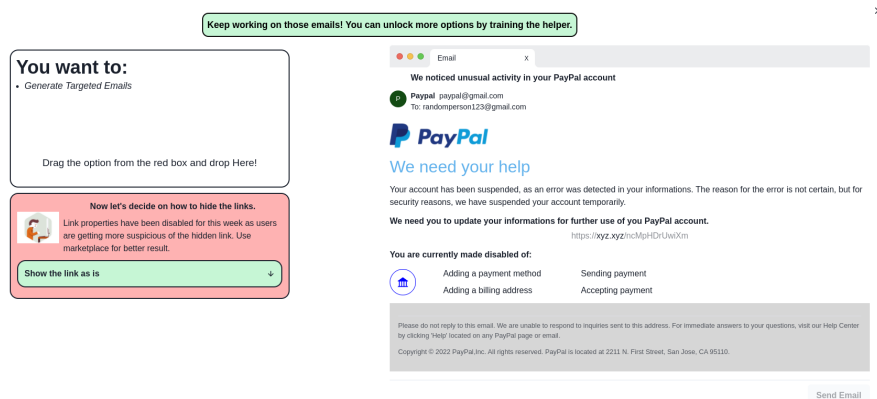
(c) Marketplace invalid

Figure 2.5: Different stages of the marketplace

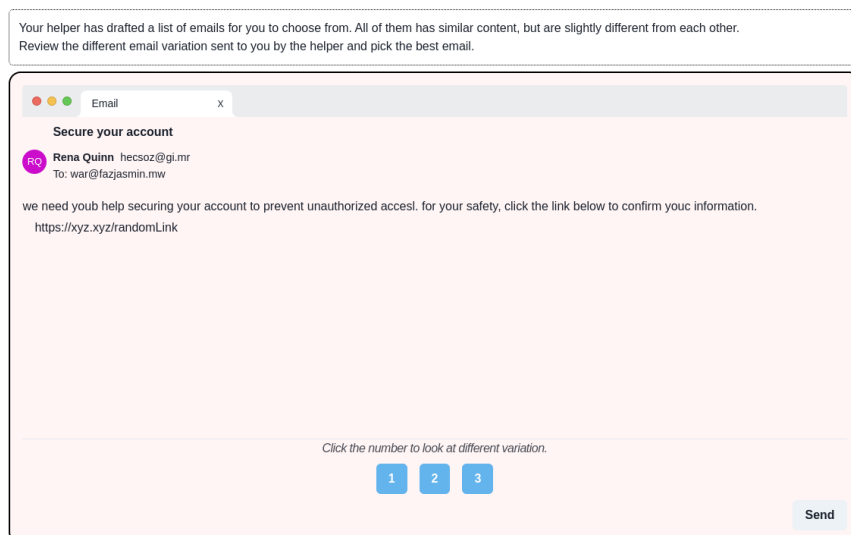
grammar, and styling in our game fall under passive skills. Before players train on spelling and grammar, they will be required to recognize spelling and grammar errors. We wanted to point out spelling and grammar errors as they are commonly found in poorly worded phishing emails and recognized as one of the common ways to differentiate phishing emails from legitimate emails. However, we do not want users to spend all their time finding language errors, so we generate emails with proper grammar and spelling after training.

Styling increases the visual appeal of the email with the help of images, header, footer, and better styles. Emails sent by an organization generally contain images and styling. Attackers use this fact to trick victims by including company logos and images. As stated before, users generally

trust familiar logos and 34% of users believe emails with familiar logos are safe [9].



(a) Emails generated after training on passive skills generate emails with proper grammar and spelling with styling



(b) Emails generated before training on passive skills contains spelling and grammar error with no styling (contains text only)

Figure 2.6: Emails generated after training vs before training on passive skills

Active skills give the user more options to fine-tune the generated email. Our game's active skills are links, research, and spoofing. Each option allows the user to modify a part of the generated email. We discuss each of these skills in detail below.

Research skill allows the player to generate targeted emails. Our game approaches the concept of spear-phishing with targeted email. Before training on research, the helper only generates a generic email, and players do not get an option to send targeted emails. Generic emails target a larger audience and do not contain any specific user details. On the other hand, targeted emails contain user-specific information such as address and name.

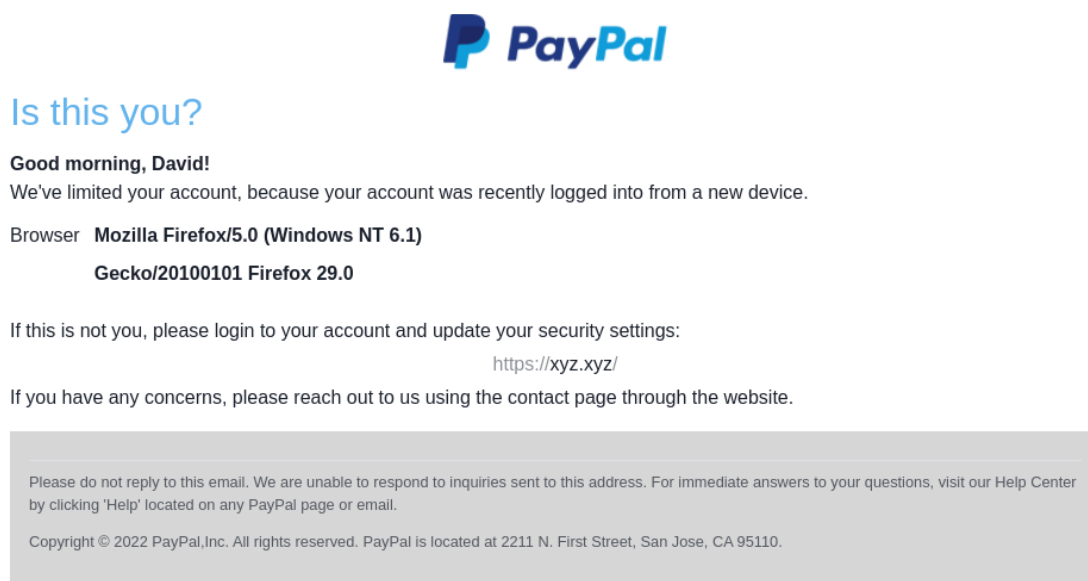


Figure 2.7: Example of a targeted email generated by the system

All the emails on the system are pre-labeled to either generic or targeted. When the user chooses an option, the system will filter out emails (based on user option) and randomly choose an email from the filtered list.

Our second active skill, links, attempts to cover URL/link training many current phishing training games covers. Training helper on link unlocks different ways to display the link when generating an email. We chose these options after reviewing real-world phishing emails and current training materials.

The game allows the user four different ways to display the link:

## 1. Hide under button or text

One common phishing trick is to hide the actual link behind some text or button. We replicate this behavior with our "Hide link" option, which displays some text or buttons as links without displaying the actual link. To familiarize users with different forms of text messages and buttons, we randomly choose a text message( Example figure 2.8b) or button (Example figure 2.8a) from predefined set of input. Our input contains familiar texts such as "Click here," "Go to PayPal," "www.paypal.com/help," and other similar alternatives. Players can hover the text or button to visualize the link (similar to real email clients).

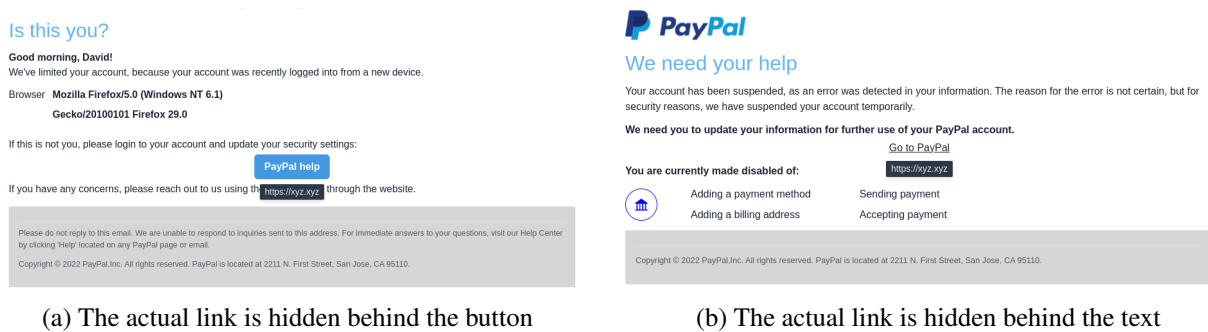


Figure 2.8: Examples of hiding the actual link behind text or button

The primary goal of this option is to encourage the user to check for the actual destination links and not to trust the display message.

## 2. Link shortener

During our survey of phishing emails, we noticed attackers use URL shorteners to confuse the end-users. A URL shortener is a tool that creates a short, unique URL that will redirect to the specific website of the user choosing. There are multiple free URL shortening services that shorten the URL with a button click. TinyURL, Bitly, Short.io, BL.INK are some popular examples of shortening services. Table 2.3 shows an example of a different URL shortener. The shortened links do not expose the actual domain it redirects to. Phishers use

this fact by hiding the actual domain with the help of shortening services.

Service	Shortener
Original URL	<a href="https://www.uno.edu/academics/colaehd/ehd/elcf/educational-leadership-graduate-programs/masters">https://www.uno.edu/academics/colaehd/ehd/elcf/educational-leadership-graduate-programs/masters</a>
TinyURL	<a href="https://tinyurl.com/5n6ehd6k">https://tinyurl.com/5n6ehd6k</a>
bitly	<a href="https://bit.ly/3CGFfBC">https://bit.ly/3CGFfBC</a>
is.gd	<a href="https://is.gd/MKZdLO">https://is.gd/MKZdLO</a>
Tiny	<a href="http://tiny.cc/unjpuz">http://tiny.cc/unjpuz</a>
RB.GY	<a href="https://rb.gy/nrwbqb">https://rb.gy/nrwbqb</a>

Table 2.3: Example of different URL shortener and their corresponding shortened links

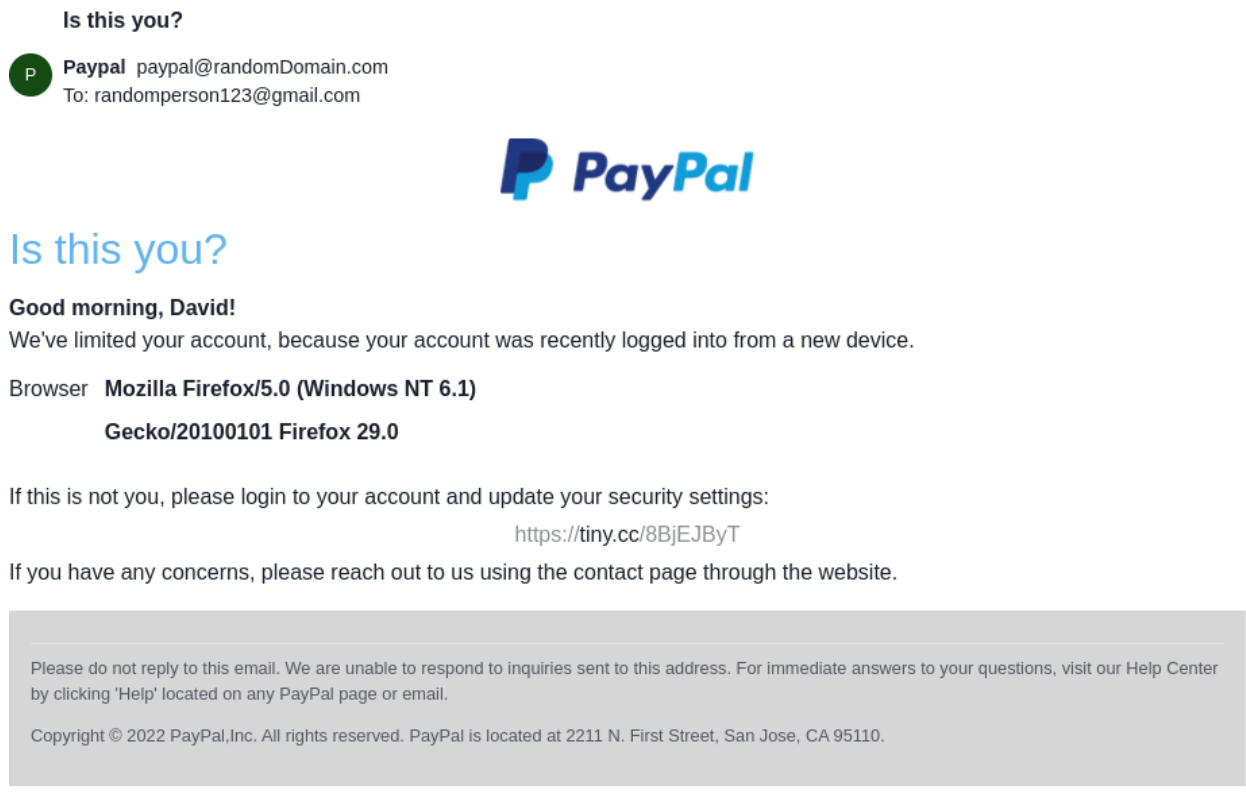


Figure 2.9: Example of a URL shortener option in game

Figure 2.9 shows an example of an email generated with the shortener option. The primary goal of this option is to familiarize players with different URL shortening services and how they can be used to hide actual links. In addition to just knowing how to hide links with shorteners, we want the user to know about different shortening services. Hence, every time the user chooses to hide the link with the shortening service, we randomly choose one of the shortening services and attach a nano id <sup>4</sup> at the end. Table 2.3 shows different link shortener services included in the game with an example.

### 3. Confusion

The confusion option teaches users to be careful about familiar links that might look familiar. We focus on subdomains for this option as they are free, can be anything (including existing organization names), and can be added to any existing domains. Phishing links attempt to confuse the users by including the organization name as a subdomain. We try to show the player this by adding "paypal" to the current domain in the game. For example, "paypal.xyz.xyz" may look like a PayPal domain but is a page in xyz.xyz.

### 4. Display link as is

The "display link as is" option allows players to see the actual link without modification. This option is useful when the domain purchased by the player is very similar to PayPal. For example, "paypai.com" (with i) looks similar to paypal.com. This domain can easily trick the victims into clicking the link if they are not closely paying attention. We use this option to train users on a similar-looking domain bought from the marketplace.

Player select the link hiding method with an interactive drag and drop approach. We want the player to have immediate feedback on their action. Hence, when the player chooses an option, we immediately change the email. This visualization allows the players to see how the links are used in context to the email.

The final active skill in the game is spoofing. Existing games do not focus on training users

---

<sup>4</sup><https://github.com/ai/nanoid>

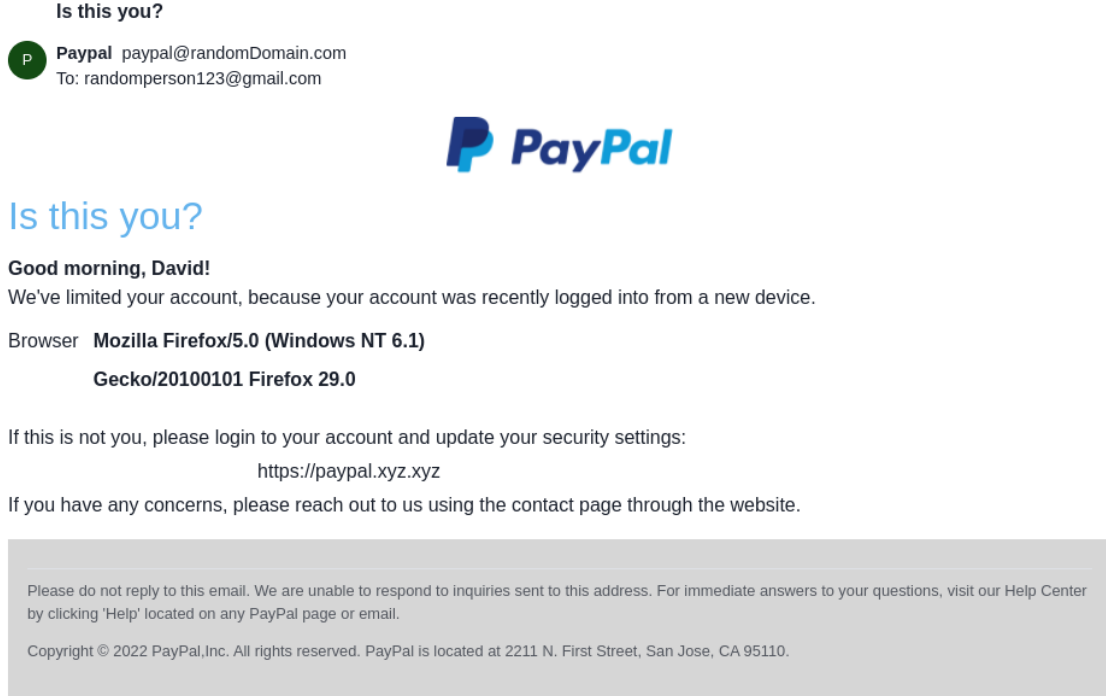


Figure 2.10: Example of a email generated with link confusion

on spoofing. However, users can easily get tricked into giving sensitive information if they receive emails from a familiar source. Various free services (Example: figure 2.11) send emails with custom header (custom to, reply-to, subject fields in the email) without additional verification.

After the player unlocks the "spoof" skill, we allow the player to change the sender's email address to any valid email address. The primary goal is to show the user that the sender can be anyone, and the user has to pay attention to other details of the emails, such as context and links.

### 2.2.2 Email efficiency

The efficiency of the email generated by the system depends on the options chosen by the user. Each new skill can improve the efficiency of the email. We calculate the efficiency of generated email as:

$$E = \frac{\text{Sum of trained passive skill points} + \text{Sum of skill point of active skills chosen by the user}}{\text{Total Available skill points}} \times 100\%$$



[Home](#)
[Send fake mail](#)
[FAQ](#)
[Do it yourself](#)
[Contact](#)

### Send a fake email

Use this page to send an email to whoever you want. You can make it look like it's coming from anyone you like. Just fill in the form below and press send.

Also make sure that the From address you choose contains a real internet domain name. For instance, don't choose bush@the.government, choose bush@whitehouse.gov. If you choose a domain that hasn't been registered, the mail may not be delivered.

There are other reasons why mail may not be delivered. It's hard to be perfect with this sort of thing! Don't forget to check the [FAQ](#) for more information and try [sending it from your PC](#) if this doesn't do what you want.

We will never ever send you junk email, or give your email address away to anyone. We hate spam at least as much as you do - maybe more (and that's why this page can't be used by spammers to send bulk email or any other funny stuff).

To:

From:

Subject:

Message:

Font

Arial

Size

1

Security:

Please enter the code in the box to the left ([why?](#))

Want to surf completely anonymously? [Try it now!](#)

Figure 2.11: Example of a fake email sender. User can send the email to any email address as any sender.


Table 2.4 shows the max efficiency point for each skill in the game. The efficiency of passive skills is either 0 (if absent) or max efficiency points (if present), whereas active skills efficiency is calculated based on user input. Active skills options are scaled to show the efficiency of each option. For example, generating targeted (spear phishing) is more efficient as it contains personal information in emails. We add 20 points to the efficiency of the generated email if targeted to replicate this, 0 otherwise.

Different link hiding skills have different efficiency, although close to each other. Hiding the link behind the text gives 18 points, shortening the link gives 20 points, and using confusion gives 25 points. When the player decides to display the link as is, we calculate the string similarity of the user domain with "paypal.com." If the similarity is greater than 80%, we add 20 points to the efficiency. Else, we add 3 points to efficiency.

**You want to:**

- *Generate Targeted Emails*
- *Linked Displayed using: Confusion*

Drag the option from the red box and drop Here!



**Great! You can pretend to be someone else for better performance. Fill the email below and drag the option to the basket!**

spoofo@newemail.com

Enter valid email address, then drag the option. HINT: Business usually have business email with their domain.

Spoof email address ↓

No Spoof email address ↓

Figure 2.12: Spoofing option in game

Skill	Max Efficiency Point
Spelling	5
Grammar	5
Styling	10
Research	20
Links	25
Spoofing	25

Table 2.4: Efficiency of each option

Similarly, for spoofing, we want to encourage players to notice that they can pretend to send the email as anyone. We compare the domain (the part after @) in the spoofed email chosen by the user with "paypal.com." Depending on similarity the similarity score, we assign points as shown in table 2.5.

We considered different keywords seen in real emails sent by organizations and wanted the players to try these keywords. For example, if the name included keywords "contact," "help,"

Similarity	Point
90%	20
80%	18
60%	7
Below 60%	0

Table 2.5: Similarity of spoofed email domain and points assigned

"info," "no-reply," or "noreply," we add 5 points to the efficiency. Similarly, if the email contains "paypal" in the name but was sent from a domain with a low similarity score, we add 10 points.

We calculate the efficiency based on these criteria by adding the player skill points and dividing it by the sum of all available points.

### 2.2.3 Previous Iteration

The initial version of the game was an open system where users could train with any skill at any given point (given they had enough amount to train). We used time to incentivize users to explore different options and generate efficient emails. The game had the same goals, options, and components but required players to avoid running out of time.

We noticed a couple of drawbacks during the testing of the initial version. First, we realized that players were constantly worried about running out of time and were not reading the emails and paying attention to all the options. This limitation challenged us to balance the game such that users had enough time to read all the emails but could not brute force (send unlimited emails to achieve the goal) through it. However, different players played the game at different paces, which made us realize that time might not be the best way to incentivize the players to complete the game.

Second, we wanted to ensure that all the players had a similar experience and explored all possible training options. Unfortunately, the open system prevented us from confirming that all the players explored the training module in the same order. It also allowed the player to train on multiple modules simultaneously. Due to this, some combination of training orders allowed

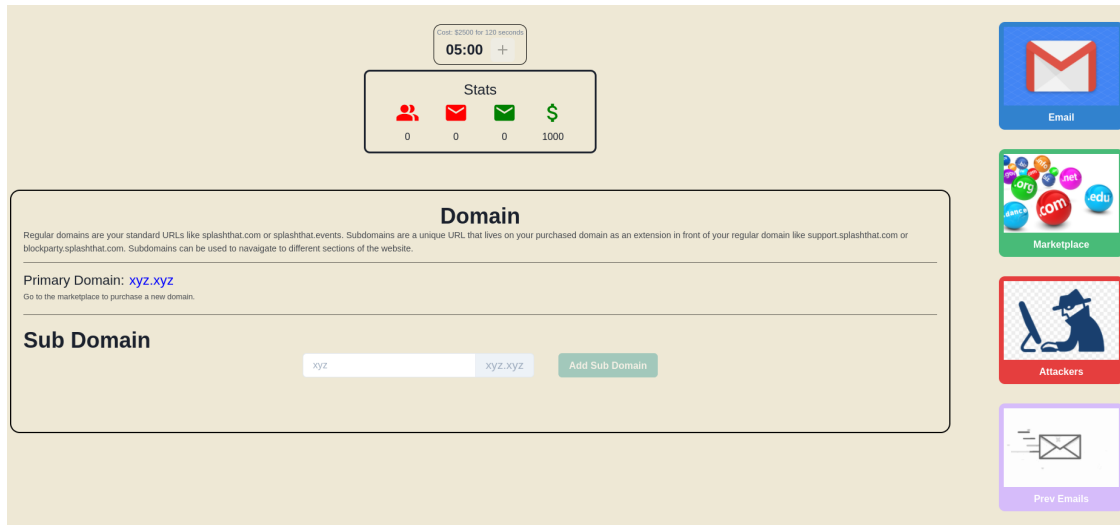


Figure 2.13: Initial version of the game included timer to incentivize the user to complete the game.

players to skip some training modules.

In order to ensure players were exploring all training modules and had the same experience, we developed a new system that lets the player focus on a particular skill at a time. The current design unlocks each skill in part and ensures users are familiar with current techniques before moving on to a new technique.

## 2.2.4 Weekly Goals

The game's current design divides the game into four parts (week). We let the player generate a limited number of emails each week. Each week unlocks new skills that the user must train on to achieve the goal. Table 2.6 shows skills unlocked each week along with the weekly goals and number of emails they can send each week. The weekly goals are adjusted based on the efficiency of the emails for the current week. As discussed above, the user's current skills determine the efficiency of the email. We determined the weekly goal through the process of trial and error. The weekly goals increase each week as we unlock new skills, which leads to more efficient emails.

Week 1 does not contain any trainable skill and solely focuses on language in the email. We want the player to know that low-tier phishing emails may contain spelling and grammar problems,

<b>Week</b>	<b>Trainable Skills</b>	<b>Weekly Goal</b>	<b>No. of Emails</b>
1	None	1,500	5
2	Spelling, Grammar, Links	15,000	10
3	Marketplace, Styling, Research	38,000	10
4	Spoofing	80,000	10

Table 2.6: Different week with their corresponding skills and goals

whereas official/legitimate emails are usually proofread and do not contain these issues.

Week 2 lets the user train on spelling, grammar, and links. Players can remove spelling and grammar errors by training language skills and entirely focus on different link hiding techniques. We let the players play with the link skill by giving a higher number of emails for the week.

Week 3 opens the marketplace along with styling and research. At this point, users have explored different ways to hide the link, and we want to focus on links that might look similar to trick the user. To force users to explore different domains, we disable all link hiding techniques and force users to show the link as is. This forces the player to utilize the marketplace and explore multiple domains.

Finally, week 4 disables the marketplace and unlocks spoofing. We unlock the link hiding skill and let the user play around with all the options. Our initial survey showed players only required half the available emails to figure out spoofing. However, we want the player to play around with all possible combinations for the final few emails, due to which we have a higher number of emails to send than required.

### 3. Evaluation

The goal was to confirm whether our game could improve the correctness of identifying phishing emails with a statistically significant result and understand common patterns found among users who fall for phishing attacks.

#### 3.1 Test Design

Since we want to understand the improvement after the user plays the game, we present each user with pre-survey and post-survey questions. Both surveys contain the same email users have to classify as phishing, legitimate, or need more information (maybe phishing). In addition, we ask the user to provide an optional field to provide feedback on the choice they made.

We curated a list of 12 emails with eight phishing emails and four legitimate emails. We replicated emails from Netflix, a common service used by many individuals. In addition to emails from Netflix, we also had an email from a "co-worker." The context of the email was pre narrated in the survey question. Using emails from a domain different than PayPal allowed us to verify that our game works across different domains.

Emails for our evaluation were hand-picked from the most common phishing emails we found during our research (commonly listed in different articles, multiple occurrences in Google search). The emails include suspicious account warnings such as payment failure, login attempts, suspicious logins, account cancellation, et cetera, and some common too good to be true emails such as free Netflix. We present these emails to the user in a Gmail clone <sup>1</sup>. Figure 3.1 shows the Gmail clone site with some emails. Although the site looks visually similar to Gmail, it has limited functionality and only allows users to click on "Inbox."

To evaluate engagement, we ask the participants various 5-point Likert scale ration questions in the post survey. These questions were derived from "Smells Phishy?"[22].

---

<sup>1</sup><https://github.com/codermother/Gmail-Clone>

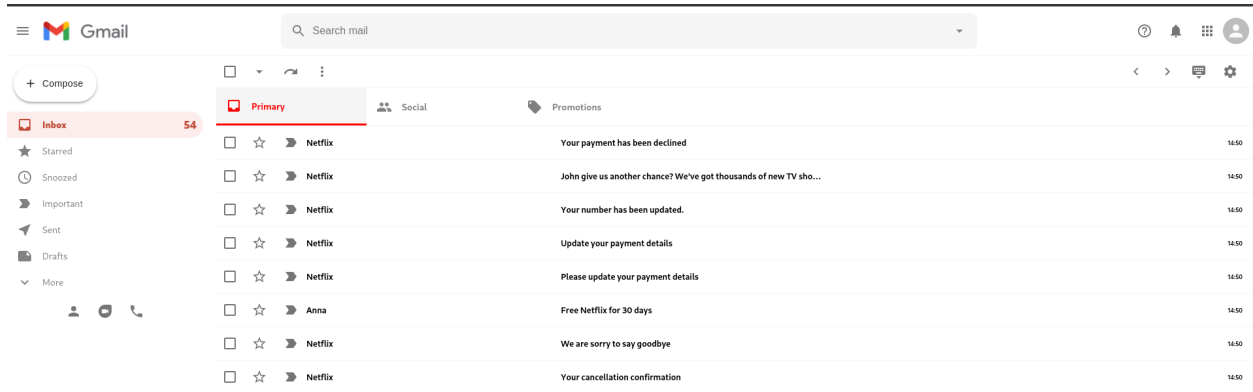


Figure 3.1: The Gmail clone used during evaluation with list of custom emails

## 3.2 Methodology

**Participants:** We distributed the evaluation (along with the game) primarily through email. In addition, we had a group meeting where players completed the survey in person. In total, 13 people attempted the game. However, only 11 participants completed both the presurvey and post-survey. Therefore, we will not consider the data for the two people that did not complete the post-survey. There were four female and seven male participants. All the participants were between 20 and 40, with seven participants in the range 20-30 and 4 participants in the range 30-40. Most of our participants were currently university students, and except for one, all were either enrolled or had a previous background in computer science. The one participant that did not have a Computer Science background had a degree in Biology.

**Sessions :** All the participants had a single flow for the study.

1. Complete the pre-survey and obtain the passcode for the game. The pre-survey contains some general demographic questions and email classification questions.
2. Play the game. We provide the post-survey link after the player completes the game. Players can restart the game from the current week if they fail to complete it in a single try.
3. Complete the post-survey. The post-survey contains the same set of email classification questions. In addition, it contains questions regarding game engagement.

**Data Collection:** In addition to the data collected through pre-survey and post-survey, we keep track of all activity of players in the game. This log helps us verify that players played the game in the order we want and can be used to improve future iterations of the game.

### 3.3 Results

#### 3.3.1 Pre-Survey

The pre-survey responses provided the participant's pre-existing knowledge of phishing scams. Participants chose one of three options (as discussed above in test design): Phishing, Legitimate, and Need more information (maybe phishing). To understand user thought behind their selection, we provide an optional text area to explain the user choice.

Each user option is classified as below:

1. **True Positive:** The user correctly classified the email as phishing.
2. **True Negative:** The user correctly classified the email as legitimate.
3. **False Positive:** The user incorrectly classified legitimate emails as phishing emails.
4. **False Negative:** The user incorrectly classified phishing emails as legitimate emails.

Since participants could mark emails, they are not confident with as "Maybe," we analyze our data in two separate ways, first by neglecting emails marked as "Maybe," and second by including all the results. On average, participants marked 17.41% of questions as "Maybe" in our presurvey. Since marking an email as "Maybe" shows caution and does not engage the user, we treat these results as "Marked phishing." Hence, phishing emails marked as "Maybe" are classified as "True Positive," and legitimate emails marked as "Maybe" are classified as "False Positive."

Table 3.1 shows the average performance (in percent) of each user when we ignore questions marked as "Maybe" and Table 3.2 show the average performance (in percent) when we include all the result and treat "Maybe" as phishing.

Based on these result (Table 3.1 and 3.2), we assume participants have some beforehand knowledge about phishing emails. However, we can also see some confusion with legitimate vs. phishing emails based on the number of emails marked as "Maybe" (17.41%)



Actual	User response	
	<b>Phishing</b>	<b>Legitimate</b>
<b>Phishing</b>	63%	37%
<b>Legitimate</b>	7%	93%

Table 3.1: Average performance in pre-survey when we ignore emails marked as "Maybe"

Actual	User response	
	<b>Phishing</b>	<b>Legitimate</b>
<b>Phishing</b>	74%	26%
<b>Legitimate</b>	23%	77%

Table 3.2: Average performance in pre-survey when we include all the results and treat "Maybe" as marked phishing

We manually reviewed the user text responses to understand why they made each decision. We noticed similar thought processes in many of the participants:

1. Participants who missed phishing emails were more suspicious of the content of the email. For example, some participants thought organizations do not send emails regarding common account information/errors such as subscription cancellation, number update, etc. Account update emails are commonly sent to the account holder by existing popular services.
2. Only a couple of participants actively looked at technical details such as the link and the sender. Since participants focused more on the context of the email, any email that asked or put some personal context in the email was marked as potentially phishing or phishing.

### 3.3.2 Game

Users start the game after the pre-survey. Since we distributed the game through email, we need to verify that the user completed the survey before attempting the game. Therefore, we lock

the game with a passcode, provided at the end of the pre-survey.

Participants took a little over 25 minutes to complete the game. The fastest recorded time was 18 minutes, and the longest was 43 minutes. We noticed that participants took the longest in Week 3, where they had to play around with different domains. Table 3.3 shows the average time user took for each week.

<b>Week</b>	<b>Average Time</b>	<b>Fastest Time</b>	<b>Slowest Time</b>
1	2.27	1.77	2.80
2	5.18	4.18	6.73
3	10.54	2.11	18.18
4	8.54	4.00	23.43

Table 3.3: Average time user took for each week (in minutes)

Participants quickly figured out spelling and grammar errors in Week 1 and did not have trouble understanding different link hiding techniques in Week 2. However, many participants had to redo Week 3 as they could not find good domains to try. Some participants tried domains that had no connection with PayPal, such as "starbucks.com," "fredmeyer.com," and "youtube.org." However, participants successfully figured out that purchasing similar domains provided better efficiency. Table 3.4 shows different domains used in the game.

Although most participants chose domains with the alternate ending as suggested by the game (when they enter paypal.com), some users were creative and figured out purchasing paypal.com (with i) would make it look visually similar and be more efficient. We did not consider visual similarity in this iteration, and it can be something to consider in future iterations.

A few participants had a problem with spoofing, but generally, participants quickly understood spoofing and used efficient email addresses. Table 3.5 shows different emails used by participants during spoofing. Participants used efficient domains (paypal.com) with commonly seen names such as accounts, contact, support, etc.

<b>Domains</b>	<b>Number of used</b>
paypal.org	8
paypai.com	8
paypal.co	4
paypal.nl	1
info-paypal.com	1
youtube.org	1
paypalsupport.com	1
paypayl-info-gmail.com	1
paypal-paypal.com	1
billingpal.com	1
paypay-hotmail.com	1
paypal.gov	1
paypal.cm	1
startbucks.com	1
fredmeyer.com	1

Table 3.4: Different domains purchased in the marketplace in game

Overall, we believe participants were able to play the game smoothly and complete all the game objectives without any hiccups. Furthermore, based on the game logs, players explored all the goals in the game (even though most players had to play the game at least a couple of times).

### 3.3.3 Post-Survey

The post-survey responses give us an idea of our game’s effectiveness by allowing us to compare the before and after game scores. In addition to classifying the emails, we ask the participants 5-point Likert scale questions.

Like pre-evaluation, we evaluate the user performance on two different scales, one without emails marked as "Maybe" and one with all the emails. We repeat the same process for the second

<b>Emails used by participants</b>
accounts@paypal.com
admin@paypal.com
alert@paypal.com
ben@paypal.com
contact@paypal.com
customerservice@paypal.com
paypal@gmail.com
paypalsecurity@pp.com
paypalsupport@paypal.com
security@paypal.com
support@paypal.com

Table 3.5: Some spoofing emails used by participants

part (with maybe) as done in the presurvey. Emails marked as "Maybe" are treated as "Phishing."

After playing the game, we noticed participants were more confident in their answers, and on average, participants marked only 10.5% emails as "Maybe" (7% improvement from pre-survey).

Table 3.6 and 3.7 shows participant performance on classifying email after playing the game.

Actual	User response	
	<b>Phishing</b>	<b>Legitimate</b>
<b>Phishing</b>	80%	20%
<b>Legitimate</b>	35%	65%

Table 3.6: Average performance in post-survey when we include all the results and treat "Maybe" as marked phishing

	Actual	User response	
		Phishing	Legitimate
Phishing		83%	17%
Legitimate		36%	64%

Table 3.7: Average performance in post-survey when we include all the results and treat "Maybe" as marked phishing

We manually reviewed user text responses on why they made certain choices (similar to what we did in the pre-survey). We noticed a few common thoughts:

1. Participants were paying closer attention to the sender of the emails. However, many participants did not trust emails generated from a subdomain such as mailer.netflix.com. Instead, users believed that companies generally send emails from a primary domain like "netflix.com."
2. Most of the participants noticed the links hidden under the text. Participants actively avoided links that were phishing, such as "netflixmovies.com."
3. Few participants were still untrusting of the content of the email, especially contents with private details such as phone number and credit card info.

### 3.4 Pre vs Post Phishing Knowledge

Table 3.8 shows the change in the average score of the participants before and after the game. Although we can not confidently claim that our game improved performance given the sample size and change in score, a few things stand out.

First, participants were more confident in their email classification and marked their email either phishing or non-phishing. Emails marked as "Maybe" decreased from 17% to 10%. For example, one of the participants went from six "Maybe Phishing" to two. Only two out of eleven participants marked more emails as maybe phishing (+1 compared to pre-survey).

Answer	A. Change in average score	B. Change in average score
True Positive	17%	9%
True Negative	2%	-14%
False Positive	28%	14%
False Negative	-17%	-9%

Table 3.8: Change in average score in pre-survey vs post-survey. Column A. represents change in score when we ignore emails marked as "Maybe" and Column B. represents all the emails when we consider emails marked "Maybe" as phishing.

Participants were correctly identifying more phishing emails. We see, on average, participants correctly identified 17% more phishing emails when we ignore emails where participants could not be sure. In addition, participants' performance improved by 9% even when we treated emails they could not confidently identify as "Phishing."

Our results show that participants were also more skeptical of legitimate emails after playing the game. Participants marked 28% more legitimate emails as phishing (when we don't consider maybe) and 14% more legitimate emails as "Phishing" when we consider all the emails. We saw six participants out of eleven marked at least one of the legitimate emails as phishing.

Overall, we can see that the game positively impacted the participants. Participants were better at identifying phishing emails and all the participants had at least a partial knowledge about protective actions after playing the game. In addition, based on participants' comments, we can see they had at least some knowledge about different objectives of the game: spoofing, link hiding, and language skills.

### 3.4.1 Evaluating emails in post survey

Out of the eight phishing emails, there were only two emails (Column 1 and Column 5 in figure 3.2) where the user performed significantly worse than other emails. Both the emails had a common theme. All the details looked legitimate, but we hid the actual link under a button. We used "netfliix.com," which is visually similar to "netflix.com," to trick the users.

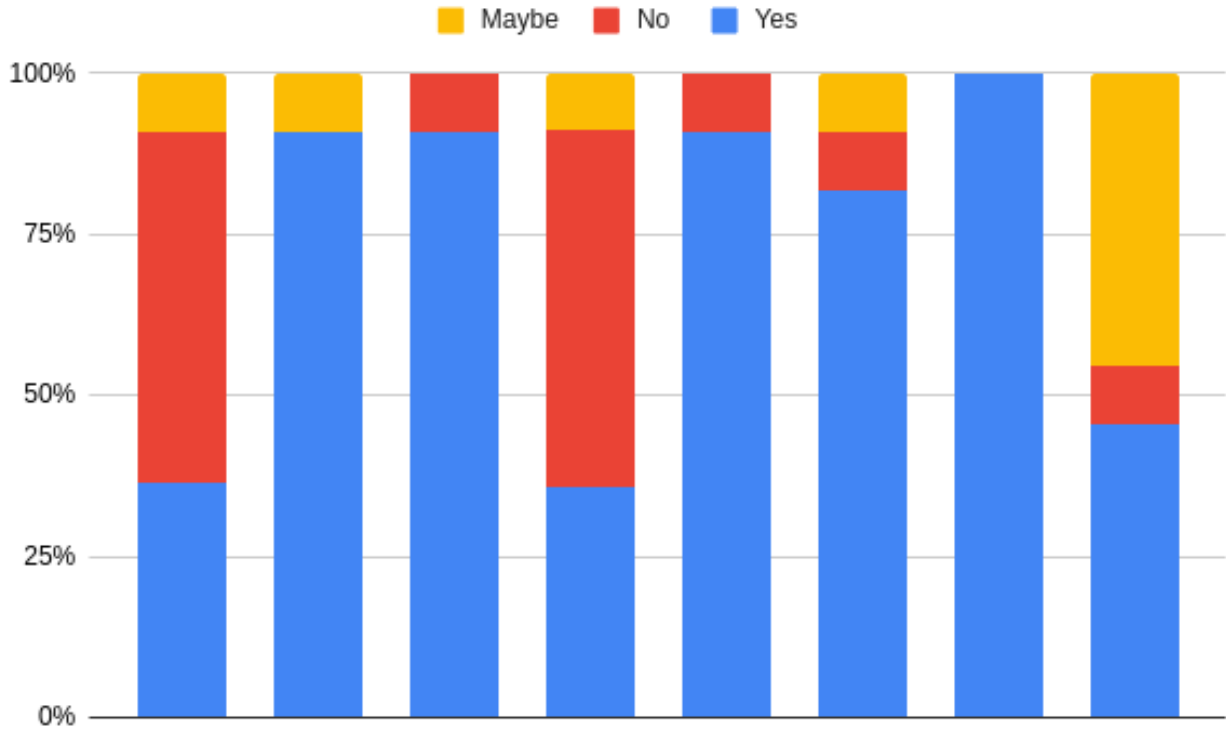


Figure 3.2: Participants performance on the phishing emails in the evaluation

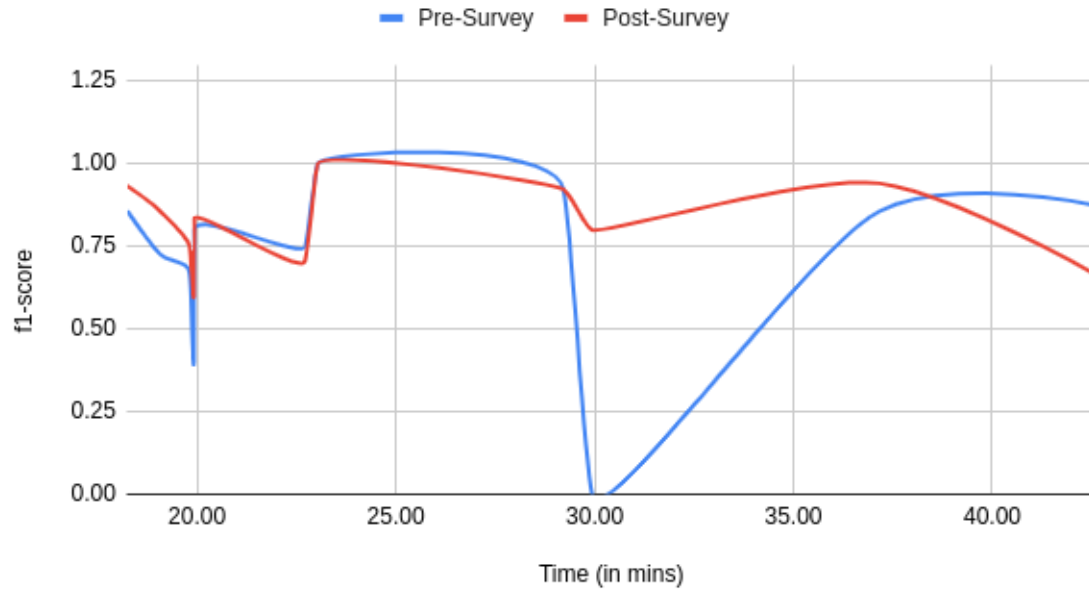
Participants performed well when the sender is not netflix domain (netflix.com). Obvious fake domains such as fakenetflix.com, netflix.com, tinyurl.is and phishing.com were caught by the user (Column 6, Column 7, Column 8).

### 3.5 Game Time vs performance

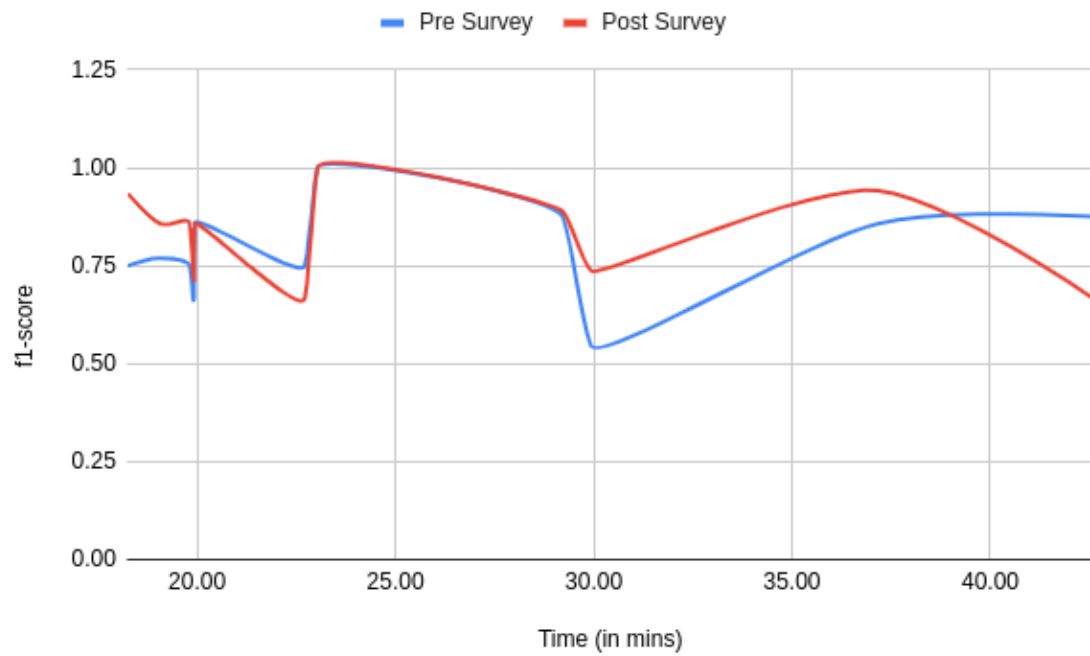
We wanted to check if participants spending more time to complete the game performs better. To compare participants' performance before and after the game, we compute the f1-score<sup>2</sup> of each participant separately.

Figure 3.3 show f1-score of each participants with game time. Based on the chart, we can not conclude that higher game time results in better performance. However, we believe that with subtle hints and more emails, player playing longer will have better performance.

<sup>2</sup><https://en.wikipedia.org/wiki/F-score>



(a) Ignoring "Maybe"



(b) All emails

Figure 3.3: Comparison of f1-score before and after playing the game with time (Maybe answers are included)



### 3.6 Questionnaire

Table 3.9 shows different opinion-based questions asked in the post-survey. The focus of the questionnaire was to assess participants' perceptions and opinions of the game and collect additional feedback about potential improvements. Likert-scale questions used a 5-point scale ranging from 1 = strongly disagree to 5 = strongly agree.

	Question	Average Rating
1.	The game showed phishing tricks.	4.90
2.	I better understand phishing scams after playing the game.	4.54
3.	I will use the techniques mentioned in the game to avoid phishing.	3.9
4.	I understand spoofing better.	4.18
5.	The game taught me how to protect myself from phishing.	4.27
6.	I learned something new.	4.72
7.	I know more about different link hiding techniques and what to look out for.	4.27
8.	I would like to play the game again.	2.9
9.	The game is complicated.	2.63
10.	I prefer reading an educational document to playing a game to learn about phishing.	1.81
11.	Education games are important to understand security.	4.45

Table 3.9: Likert scale questions relating opinion of the game; Higher score means more positive response

Questions 1-7 (in table 3.9) focus on the educational features of the game and user feelings towards our approach. Based on the average score, we can see that participants welcomed the

game and found it helpful. Figure 3.4 summarizes the distribution of responses for each questions relating to phishing. We had positive responses for the game and techniques displayed in the game. For example, almost all participants agreed that the game showed phishing tricks and a majority of participants strongly agree that the game helped them understand phishing better.

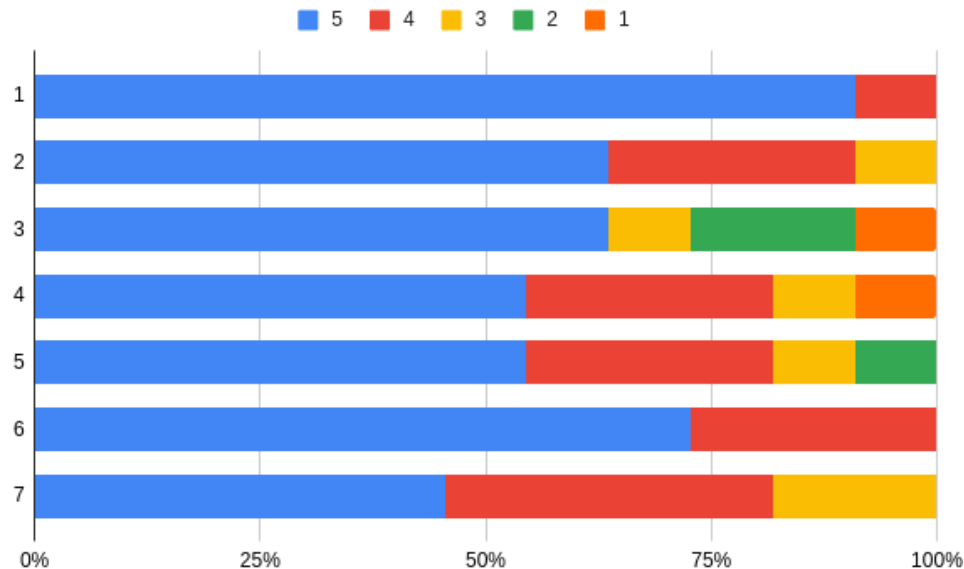


Figure 3.4: Participants response to the first 7 questions. The numbers correspond to the question numbers in table 3.9.

The second set of questions (8-11 in table 3.9) is related to participants' general opinions of the game and game-based learning. Figure 3.5 illustrates the distribution of responses for question through 8-11. More than half of the participants mentioned that they would not play the game again. We had a chance to talk to many of the participants and got common feedback on the gameplay. They were eager to play it the first time but would not like to play it again.

The majority of the participants did not find the game hard and could complete the game objective. However, we believe the game can be further simplified with better UI and hints to the player.

Finally, we asked participants their opinion on game-based learning. We can see that most

of the participants prefer educational games over reading materials. Only one participant favored reading books over educational games for training purposes.

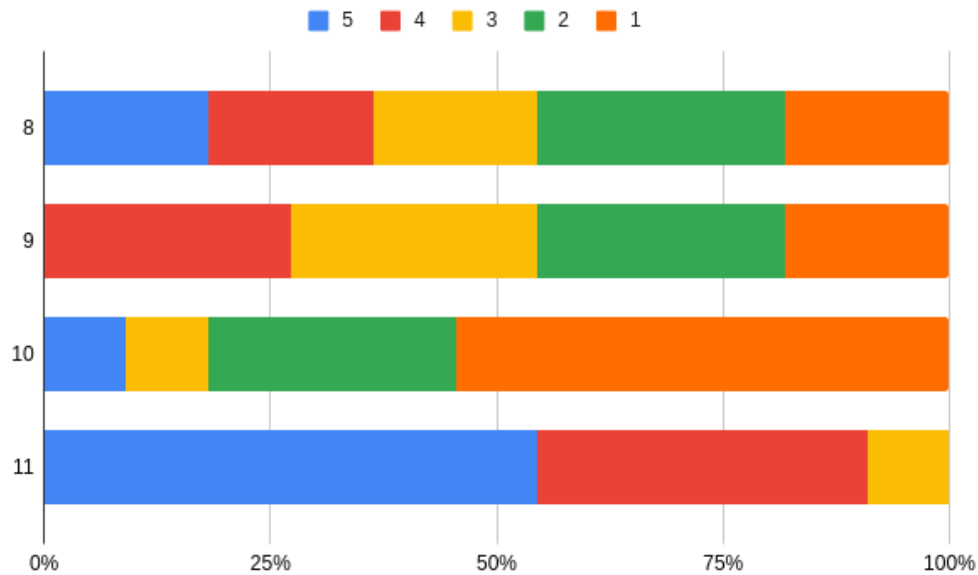


Figure 3.5: Participants response to questions 8-11. The numbers correspond to the question numbers in table 3.9.

### 3.7 Participant Feedback

At the end of the post-survey, we allow participants to express any comments or improvements to the game. The majority of the comments were positive about the game and pointed out it was a good experience. However, the majority of participants wanted a concise version of the game. One of the participants said he would ask his family to try out the game if it was short.

## 4. Discussion

We designed and implemented a role-playing game that puts the player as an attacker and shows different phishing techniques attackers can use. Overall, we are pleased with the result of this first iteration of the game and its ability to convey educational materials.

### 4.1 Insights

We noticed a few common patterns during our study, and we believe additional training materials on these would help strengthen users against phishing attackers.

#### 4.1.1 Mailing Domains and Subdomains

Legitimate emails used in our survey used "info@mailer.netflix.com" as the sender, which is the email used by Netflix to send updates about user accounts. We noticed that participants marked these legitimate emails "Maybe phishing" or "phishing" although they were satisfied with other contents of the email.

We believe the confusion is mainly due to the following two reasons:

1. Users are expecting the email to be from netflix.com. However, confusion arises when users see emails originate from "mailer.netflix.com." We can see a similar example in figure 4.1. Lyft used "noreply@lyftmail.com" to send the email. This pattern can easily confuse users, and attackers can potentially use similar patterns for other organizations to trick the victims.
2. The issue mentioned in (1) also highlights another problem with current phishing training modules: subdomains. Although our game touches it briefly (and many games discussed in the literature review briefly cover it), we haven't found games that fully cover subdomains.

We can mitigate this problem if organizations (such as Netflix) warn the users about the domains they use or stick with a similar pattern of domains when emailing the user. In addition, new

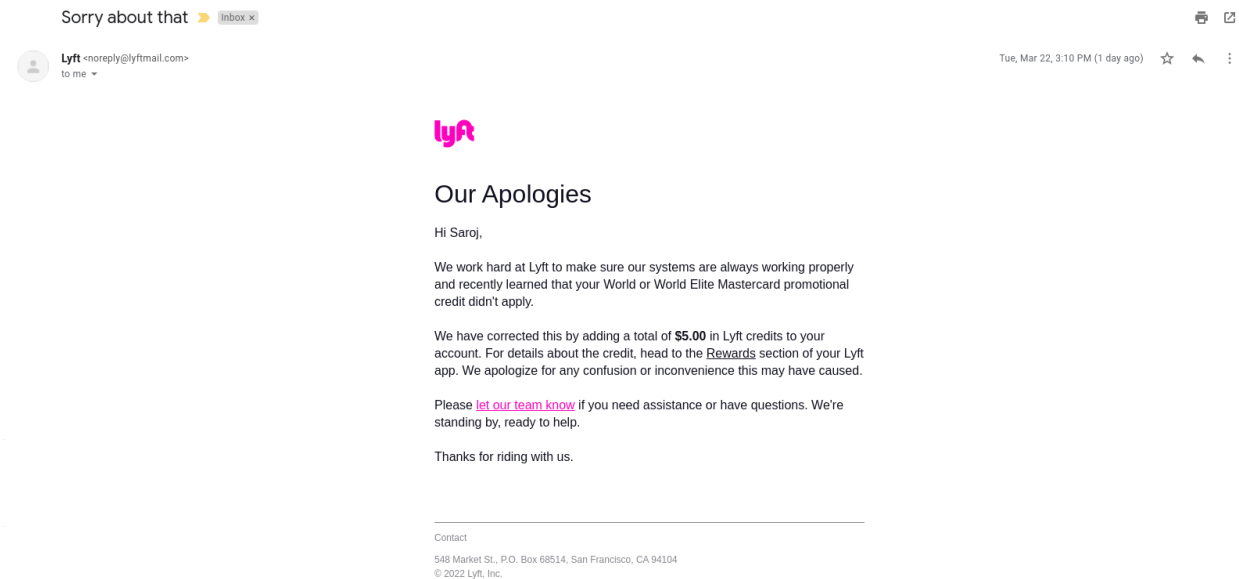


Figure 4.1: An email sent by Lyft. The sender of the email is *noreply@lyftmail.com*

games should focus on developing games that include subdomains as a separate training module instead of incorporating it inside link training.

#### 4.1.2 Issues with email content

Our pre-survey (and in some part post-survey) showed participants did not like when they got personal information (such as phone number and credit card number) in their email. We believe users would be more comfortable with such emails if they only notified the user of changes through emails and gave the details in their platform.

### 4.2 Limitations and Future works

We conducted a user study with 11 participants, which provided important insight into the game's viability, but further studies are needed to confirm that the results hold in other settings. In addition, most of our participants had some previous knowledge about phishing. Therefore, we need to further study with diverse demographics to verify the works even when participants have no prior knowledge. Furthermore, the interpretation of players' comments may have been biased by our intimate knowledge of the game, although we made every effort to remain objective.

The overall experience of our gameplay takes around 25 minutes. We can incorporate larger demographics by creating a compact version of the game. Based on the feedback, participants were willing to share the game with friends and family if it required a shorter time commitment.

The current iteration of the game does not focus on subdomains. However, our results show that subdomain is an area that requires more focus. Future improvements can be made by adding subdomains as a separate training module.

### **4.3 Conclusion**

We designed and developed a role-playing game that lets the player play as an attacker and explore different phishing techniques attackers can use. The main goal of our games is to teach players anti-phishing techniques by showing the players different tricks attackers. We conducted our user study with 11 participants and presented our results demonstrating our game positively impacted players' confidence to detect phishing emails. In addition, we listed a few common patterns that we found during our study. Based on the result, we found some areas such as subdomain and organization emails that require more focus.

Overall, we are pleased with the result of this first iteration of the game and its ability to convey educational materials. This game can serve as a good starting point for future work.

## References

- [1] D. Jampen, G. Gür, T. Sutter, and B. Tellenbach, “Don’t click: towards an effective anti-phishing training. a comparative literature review,” *Human-centric Computing and Information Sciences*, vol. 10, no. 1, 2020.
- [2] KnowBe4, “What is phishing?.”
- [3] A.-P. W. Group, “Phishing activity trends report 3rd quarter 2021,” tech. rep., Anti-Phishing Working Group, 2021.
- [4] “The phishing email that hacked the account of john podesta,” Oct 2016.
- [5] M. Anderson, “Wikileaks releases more purported emails, bringing total to more than 11,000,” Oct 2016.
- [6] “Cybercrime statistics: Top threats and costliest scams of 2020.”
- [7] A. Duke, “5 things to know about the celebrity nude photo hacking scandal,” Oct 2014.
- [8] “Nude celebrity picture leak looks like phishing or email account hack,” Sep 2014.
- [9] “2021 state of the phish report,” 2021.
- [10] I. Vayansky and S. Kumar, “Phishing-challenges and solutions,” *Computer Fraud & Security*, vol. 2018, no. 1, p. 15–20, 2018.
- [11] R. Yang, K. Zheng, B. Wu, C. Wu, and X. Wang, “Phishing website detection based on deep convolutional neural network and random forest ensemble learning,” *Sensors*, vol. 21, no. 24, p. 8281, 2021.
- [12] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, “Machine learning based phishing detection from urls,” *Expert Systems with Applications*, vol. 117, pp. 345–357, 2019.
- [13] Z. A. Wen, Z. Lin, R. Chen, and E. Andersen, “What.hack,” *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019.

- [14] P. Kumaraguru, S. Sheng, A. Acquisti, L. F. Cranor, and J. Hong, “Teaching johnny not to fall for phish,” *ACM Transactions on Internet Technology*, vol. 10, no. 2, p. 1–31, 2010.
- [15] T. Schultz, “Gamification - cybersecurity’s turn to play,” Dec 2021.
- [16] “Choose your own adventure security awareness games,” Jan 2022.
- [17] L. C. Almeida, “The effect of an educational computer game for the achievement of factual and simple conceptual knowledge acquisition,” *Education Research International*, vol. 2012, p. 1–5, 2012.
- [18] “Garfields count me in,” Sep 2021.
- [19] “Killer flu.”
- [20] M. Hendrix, A. Al-Sherbaz, and V. Bloom, “Game based cyber security training: Are serious games suitable for cyber security training?,” *International Journal of Serious Games*, vol. 3, no. 1, 2016.
- [21] T. Denning, A. Lerner, A. Shostack, and T. Kohno, “Control-alt-hack,” *Proceedings of the 2013 ACM SIGSAC conference on Computer; communications security - CCS ’13*, 2013.
- [22] M. Baslyman and S. Chiasson, ““smells phishy?”: An educational game about online phishing scams,” in *2016 APWG Symposium on Electronic Crime Research (eCrime)*, pp. 1–11, 2016.
- [23] S. Sheng, B. Magnien, P. Kumaraguru, A. Acquisti, L. F. Cranor, J. Hong, and E. Nunge, “Anti-phishing phil: the design and evaluation of a game that teaches people not to fall for phish,” in *Proceedings of the 3rd symposium on Usable privacy and security*, pp. 88–99, 2007.
- [24] G. Misra, N. A. G. Arachchilage, and S. Berkovsky, “Phish phinder: a game design approach to enhance user confidence in mitigating phishing attacks,” *arXiv preprint arXiv:1710.06064*, 2017.



- [25] G. Baral and N. A. G. Arachchilage, “Building confidence not to be phished through a gamified approach: Conceptualising user’s self-efficacy in phishing threat avoidance behaviour,” in *2019 cybersecurity and cyberforensics conference (CCC)*, pp. 102–110, IEEE, 2019.
- [26] “How many tlds are there? what are the types? we answer your common tld questions!.”
- [27] V. Le Pochat, T. Van Goethem, S. Tajalizadehkhoob, M. Korczyński, and W. Joosen, “Tranco: A research-oriented top sites ranking hardened against manipulation,” in *Proceedings of the 26th Annual Network and Distributed System Security Symposium*, NDSS 2019, Feb. 2019.

## **Vita**

The author, Saroj Duwal, was born in Bhaktapur, Nepal. He obtained his Bachelor's degree in Computer Science with a minor in Mathematics from the University of New Orleans in 2019. He joined the University of New Orleans Computer Science graduate program to pursue a Masters in Computer Science in Spring 2020. He has been working under Dr. Ben Samuels within the department of Computer Science at the University of New Orleans.