

Another Phishing Training - Role playing as an attacker

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

Bachelor in Science

University of New Orleans, 2019

May, 2022

Table of Contents

	Page
List of Figures	iv
List of Tables	v
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	6
1.3.3 Phishing Link (URL) training.....	7
1.3.4 Role playing game	8
1.4 Objective.....	9
2. System Description	11
2.1 Game Description	11
2.1.1 Game Story	11
2.2 Mechanics	12
2.2.1 Components.....	12
2.2.1.1 Attacker	12
2.2.1.2 Marketplace.....	13
2.2.1.3 Emails.....	17
2.2.2 Email efficiency	21
2.2.3 Previous Iteration	23
2.2.4 Weekly Goals	24
3. Evaluation.....	30
3.1 Insights	30
4. Discussion	31
4.1 Limitations	31
4.2 Future Work	31
4.3 Conclusion.....	31

References	32
------------------	----

List of Figures

FIGURE	Page
1.1 Phishing email sent to John Podesta	2
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	8
2.1 Screenshot of initial state of the game	11
2.2 Screenshot of the attacker module on week 4	14
2.3 The marketplace accepts any valid domain name	15
2.4 Top 10 top level domains present Tranco list	15
2.5 Different stages of the marketplace	18
2.6 Emails generated after training vs before training on passive skills	25
2.7 Example of a targeted email generated by the system.....	26
2.8 Examples of hiding the actual link behind text or button	26
2.9 Example of a URL shortner	27
2.10 Example of a email generated with link confusion	27
2.11 Fake email sender	28
2.12 Spoofing option in game	28
2.13 Initial version of the game	29

List of Tables

TABLE	Page
1.1 Different type of training games and their main objectives	9
2.1 Different skills and their effect in the game.....	13
2.2 Different second level domain and their similarity with "paypal"	16
2.3 Example of different URL shortner	20
2.4 Efficiency of each option	22
2.5 Similarity of spoofed email domain and points assigned	23
2.6 Weekly goals	24

Abstract

This is place holder for the abstact sectionLaboris sit sunt nulla elit nisi adipisicing incididunt esse. Occaecat anim ex fugiat aute fugiat enim cillum. Magna do pariatur voluptate cupidatat consequat magna enim occaecat tempor laborum ea aliqua amet. Ex tempor nulla est amet irure dolor veniam occaecat do deserunt cillum ullamco occaecat.

Keywords: keywords, keywords

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, baiting, quid pro quo, and tailgating 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]. 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].

```
> *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]e@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

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, APWG saw 260,642 phishing attacks [3]. Additionally, Proofpoint 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.

1.2 Current Mitigations

The prevention of phishing attacks can be divided into three steps [10]—the first step in 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.

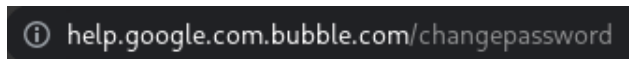


Figure 1.2: Browsers uses different shades to indicate the primary link.

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 and found some positive signs. Prevailing works can be classified into three main categories (discussed below).

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. Players have to recognize six out of eight URLs to move to the next level.

There are other similar games to Anti-Phishing Phil. Phish Phinder [24] builds upon Anti-Phishing Phil with more user interaction, such as asking for hints and levels. Another example of a link training game is developed by Baral et al. [25]. Baral et al.'s game story is based on a balloon shooter where the main character has to shoot balloons with legitimate URLs.

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 will serve as a suitable training module. Anti-Phishing Phil results show that their game has a good impact compared to existing training material while detecting legitimate sites.

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.

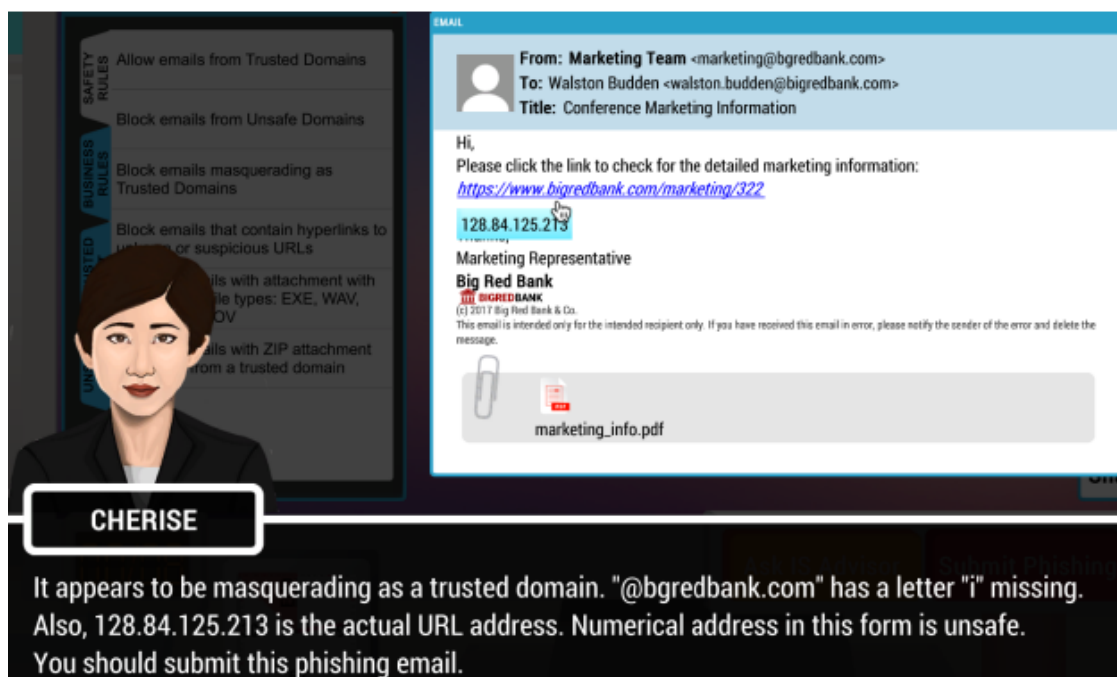


Figure 1.5: What.Hack gameplay

"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 real emails (both phishing and legitimate), 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 were able to incorporate the 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
Link Based	Link Based Game	✓		
Board Game	Teach players to identify phishing URLs in a cartoon-like game without phishing attempts			
What.Hack	Teach players to identify phishing URLs in a cartoon-like game without phishing attempts			
Our Game	Teach players to identify phishing URLs in a cartoon-like game without phishing attempts			

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

Our goals for the study can be summarized as:

- Teach email phishing defense by creating realistic emails as an attacker
- Engage the player by setting clear goals and tasks that requires player to explore more skills

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 React¹ with Chakra UI². We use Supabase³ to keep logs of user interaction with the game. Figure 2.1 shows the initial state of the game.

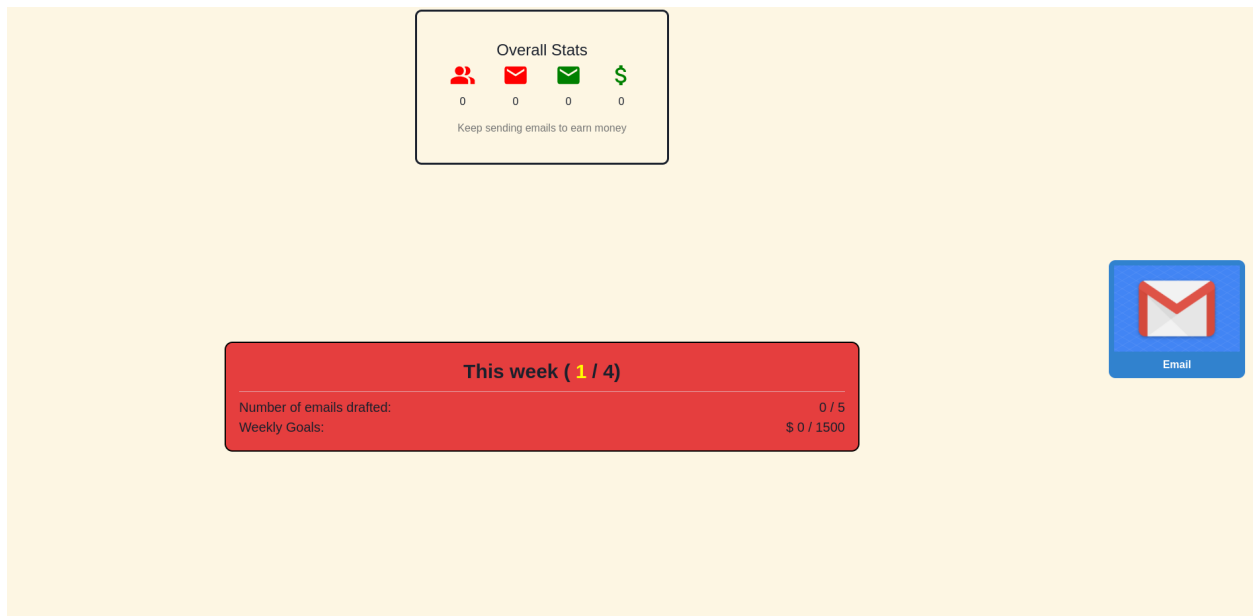


Figure 2.1: Screenshot of initial state of the game

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

¹<https://reactjs.org/>

²<https://chakra-ui.com>

³<https://supabase.com/>

through hard work and uses phishing tricks to scam people. The main character hires a helper to create phishing emails. The emails generated by the helper impersonate PayPal. The player has four weeks to pay off the loan with weekly payments.

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 email. We will talk more about certain week progression in later sections. Players can use the unlocked skills to generate emails. Each email efficiency will be based on the option chosen by the user.

2.2.1 Components

Before we deep dive into the flow of the game, let us talk about each individual component.

2.2.1.1 Attacker

The attacker module takes care of training the helper. 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 system without any additional input from the user. In contrast, active skills will give the user more options while generating the email. For example, after you train you helper on spellings, the attacker will stop making spelling errors without any additional input from the user. However, training the helper on links will allow the user to choose how 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 while talking about email generation.

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. Skills were not made free to represent training requires some resources. However, 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.

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, etc. Therefore, the decision to limit the number of skills was primarily made to limit the number of properties users were actively concentrating on. 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 wide range of emails. Emails generated by the system still include these properties.

2.2.1.2 *Marketplace*

The marketplace allows the player to purchase domains that can be used 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

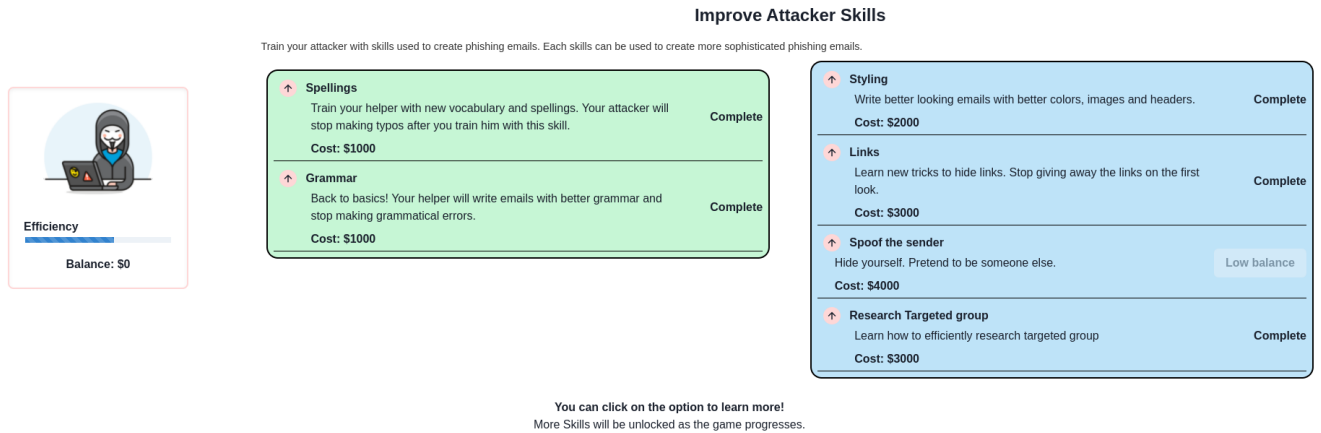


Figure 2.2: Screenshot of the attacker module on week 4

gameplay, the attacker "Link" skill teaches how phishing emails hide links to trick victims into clicking the link.

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. For example, "test.com" is a valid domain where "test" is a second-level domain and "com" is the top-level domain. The validity of the second-level domain is based on the characters in the domain. We do not allow special characters (Fada Accent) in our domain for simplicity. The following regex code validates the domain:

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

There are 1,500 top-level domain [26]. 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. We filtered 262 top-level domains from the Tranco list that occur at least one hundred times. This limited number of top-level domains allowed us to incorporate commonly used domains while maintaining the game's simplicity.

Players can choose any combination of valid characters for the second-level domain (validated

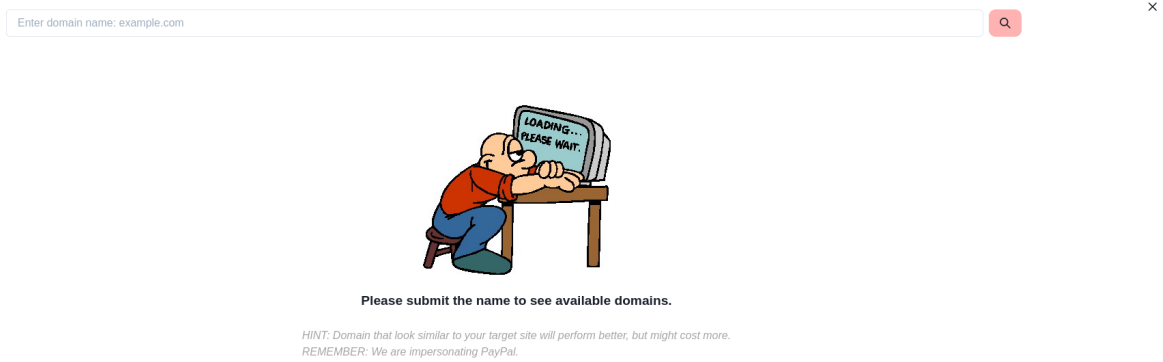


Figure 2.3: The marketplace accepts any valid domain name

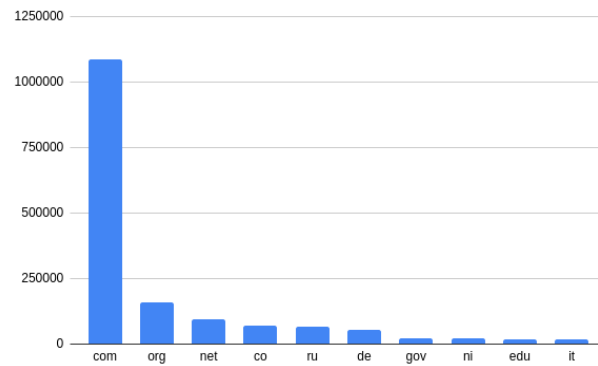


Figure 2.4: Top 10 top level domains present Tranco list

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 1000 as processing required more time, impacting the gameplay. This list covers the most popular domains, enough to get the idea of domains to the users.

Our game uses Paypal to trick the attackers, so domains closer to PayPal will perform better.

Purchasing a new domain requires some money. Domains similar to popular services (determined in the top 1000 domains) will have a higher cost. Due to this, 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 coefficient⁴ 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 domains along with their similarity. We can see that a domain similar to "paypal" has a higher similarity. Therefore, we treat higher similarity as more efficient for our game. We did not use the top-level domain for cost calculation as it would offset the string similarity (as majority of the player chose ".com"). However, the top-level domain is later used when sending an email.

Custom Domain	Similarity with "paypal"
paypl	0.66
paypale	0.90
appl	0
palpay	0.8
test	0

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

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

⁴en-academic.com/dic.nsf/enwiki/5165495

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 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:

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

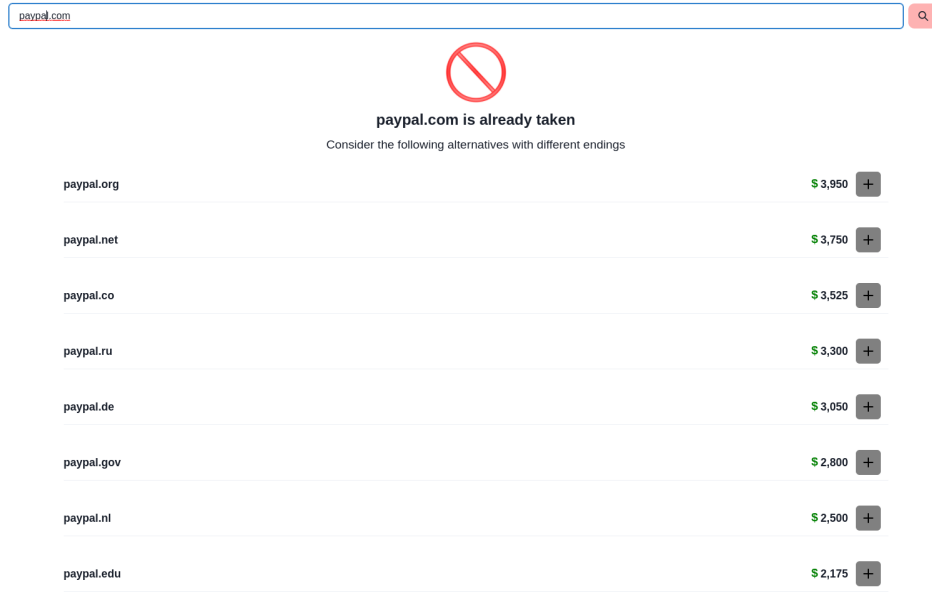
where index is the ranking of the top level domain based on frequency in Tranco list.

The initial formula to calculate the cost was based on 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. We later scaled the cost to adjust the game's difficulty.

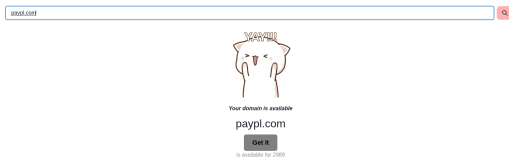
2.2.1.3 Emails

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 emails and tricks used by attackers such as log-in, welcome emails, limited account emails, etc.

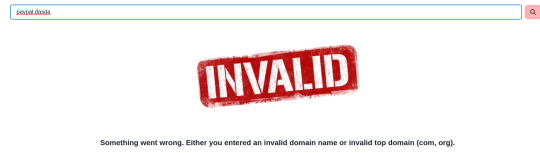
Before discussing sending email and efficiency, let us discuss how each skill impacts the email generation process. As mentioned in the attacker component section, passive skill does not require additional input from the player and improves the generated email after training them. Spelling, grammar,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



(a) Marketplace unavailable



(b) Marketplace available



(c) Marketplace invalid

Figure 2.5: Different stages of the marketplace

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 with legitimate errors. Since we do not want the user to spend all their time finding language errors, we generate emails with proper grammar and spelling after training.

Similarly, styling increase the visual appealingness of the email with the help of images, header, footer, and better-styled emails. Emails sent by an organization generally contain images and styling. Attackers use this fact and try to trick victims by including company logos and images.

Active skills give the user more options while generating emails. Our game's active skills are links, research, and spoofing. Each of these options allows the user to modify the generated email. We discuss each of these skills in detail below.

Research skill allows the player to generate targeted emails or generic emails. Our game approaches the concept of spear-phishing with targeted email. Before training on research, all the generated emails will be generic, and players do not get an option to send targeted emails. Generic emails do not contain any specific user information, whereas targeted email contains some form of user-specific information such as an address and name.

Both targeted and generic emails are pre-defined in the system. The system filters out targeted/generic emails and randomly chooses an email from the remaining emails based on the user input.

Our links skill attempts to cover URL/link training many current phishing training games cover. Training helper on link unlocks different ways to hide the link when generating an email. These options were chosen based on real-world phishing emails. The game includes four different ways to hide 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 either some text or button but redirects to another destination. To familiarize users with different text and buttons, we randomly hide the text either behind the button (Example figure 2.8a) or a randomly chosen text (Example figure 2.8b). Then, players can hover over the text or button to visualize the link (similar to real email clients). Since we did not want the text or button to look the same every time the option was chosen, we randomly selected pre-determined texts and links such as "Go to PayPal," "Click here," "www.paypal.com/help," and similar alternatives.

The primary goal of this option is to train the user to not trust the displayed link or text and be careful about clicking on the link.

2. Link shortner

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

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

Table 2.3: Example of different URL shortner

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.

Figure 2.9 shows an example of email generated with shortner 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, we want the user to know about multiple 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. Table 2.3 shows different links included in the game.

3. *Confusion*

The confusion option tries to train users to recognize legit-looking domains. We focus on subdomains for this option as subdomains are free and can be anything (including existing organization names). Phishing links attempt to confuse the users by including the organization name as a subdomain. For example, "paypal.xyz.com" may look like a PayPal domain but is a page in xyz.com.

4. *Display link as is*

The display link as is option allows the user to see the actual link without any modification. This option is useful when the domain purchased by the player is very similar to PayPal. For example, "paypai.com" (with an i) looks similar enough to paypal.com. This can easily trick the users into clicking the link if they are not 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. One of the primary goals of link hiding techniques was to teach the player how each option changes the emails. Therefore, our gameplay immediately visualizes the player's changes to the generated email. This allows the players to see how the links can be used in context to the email.

The final active skill in the game is spoofing. Existing games do not focus on training users 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 email address of the sender to any valid email address. The primary goal is to show the user that the sender can be any one and user have to pay attention to other details of the emails too.

2.2.2 Email efficiency

The efficiency of the email generated by the system depends on the options chosen by the user. Each skill improves the efficiency of the email. The efficiency of the email is calculated as

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

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. Generating targeted (spear phishing) is more efficient as it contains

personal information in the emails. To replicate this, we add 20 points to the efficiency if the generated email is targeted.

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

Table 2.4: Efficiency of each option

Different link hiding skill have different efficiency although really 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 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.

We want to encourage users to notice that they can pretend to send the email as anyone. We compare the domain in the email chosen by the user with "paypal.com". We assign variable points to the efficiency based on the similarity.

We considered different keywords used in real emails. If the name included keywords "contact", "help", "info", "no-reply" or "noreply", we add 5 points to the efficiency. Similarly, if the email contains "paypal" in the name, we add 10 points.

,

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

Table 2.5: Similarity of spoofed email domain and points assigned

2.2.3 Previous Iteration

Initially, 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 watch the time.

We noticed a few drawbacks in the initial testing of the game. First, we realized that players were worried about running out of time and were not reading all the emails and options. This 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 finish 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 ensuring all the players explored the training module in the same order. It also allowed the player to train on multiple modules simultaneously because of which players might not learn all the objectives of each option.

We developed a new design to mitigate these challenges that ensured players focused on each training module. Our current design locks the training skills and forces users to use specific training skills each week.

2.2.4 Weekly Goals

The game's current design divides the game into four parts (week). Each week unlocks new skills that the user must train on to reach the goal. Table 2.6 shows what skills are unlocked each week and the skills we want the user to focus on. 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.

Week	Trainable Skills	Skill focus	Weekly Goal
1	None	Spelling, Grammar	5000
2	Spelling, Grammar, Links	Links	5000
3	Marketplace, Styling, Research	Spear phishing and domains	5000
4	Spoofing	Spoofed emails	5000

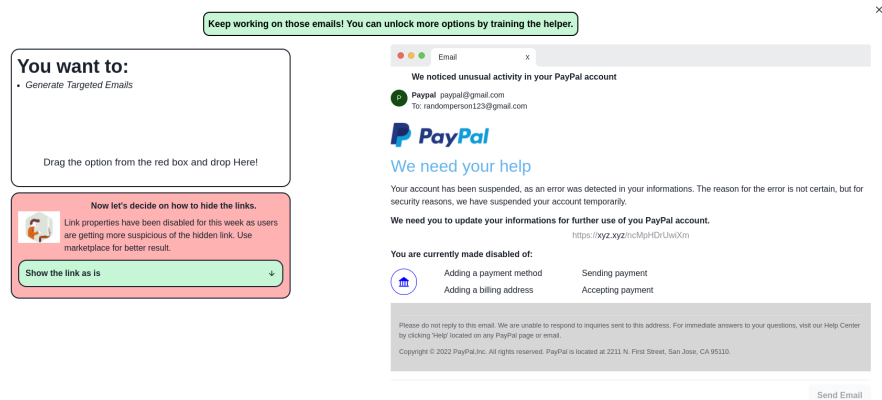
Table 2.6: Weekly goals

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 or grammar problems, whereas official emails are usually proofread and do not contain these issues.

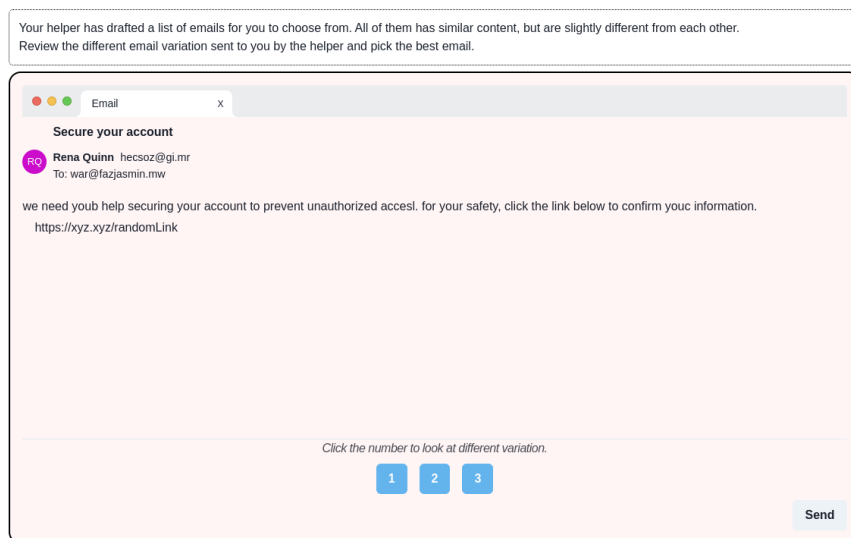
Week 2 lets the user train on spelling, grammar, and links. Players can remove spelling and grammar errors by training language skills and entirely focus on different link hiding techniques.

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. We want users to pay close attention to the link they are clicking.

Week 4 disables the marketplace and unlocks spoofing. We let the user hide the link and let them be anyone they want.



(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

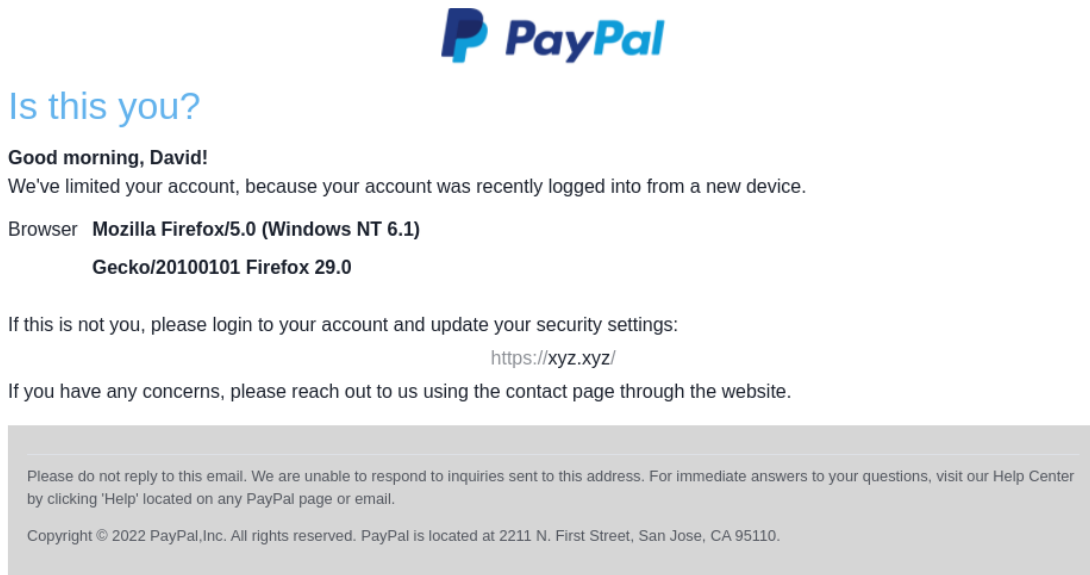
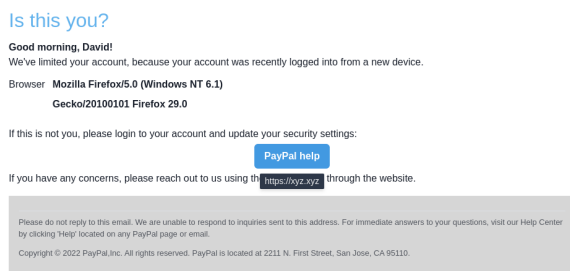
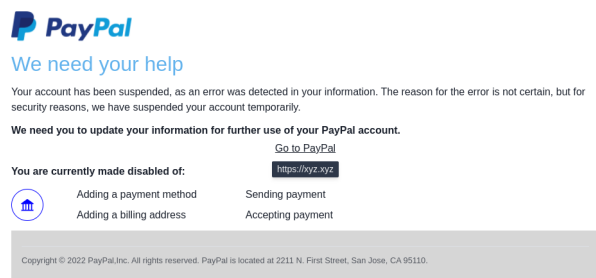


Figure 2.7: Example of a targeted email generated by the system



(a) The actual link is hidden behind the button



(b) The actual link is hidden behind the text

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

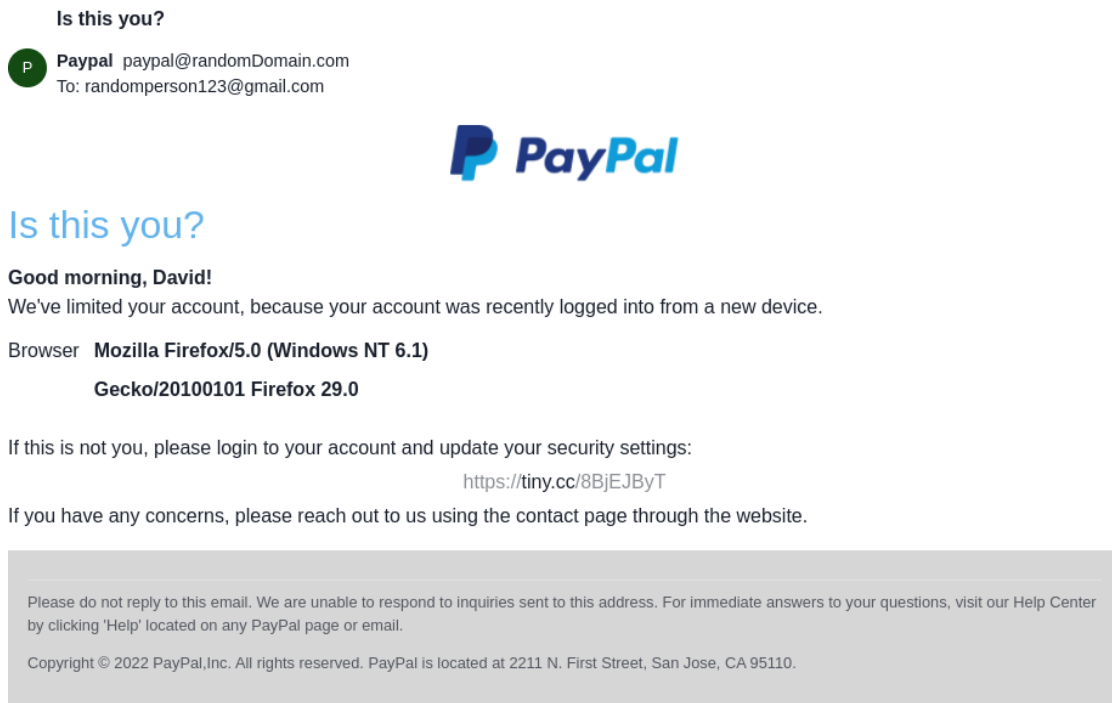


Figure 2.9: Example of a URL shortner

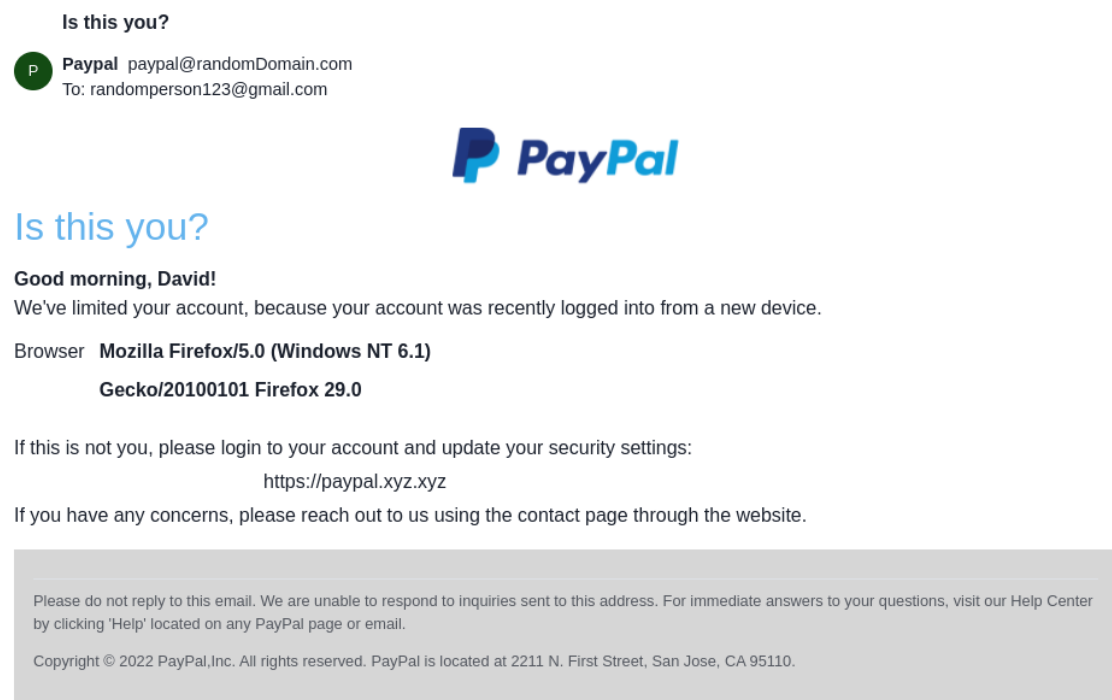


Figure 2.10: Example of a email generated with link confusion

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 by [sending it from your PC](#), if this doesn't do what you want.


To:

From:

Subject:

Message:

Font: sans Size: 12

Security:  Please enter the code in the box to the left (only 2)


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

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

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!**

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

Spoofo email address ↓

No Spoofo email address ↓

Figure 2.12: Spoofing option in game

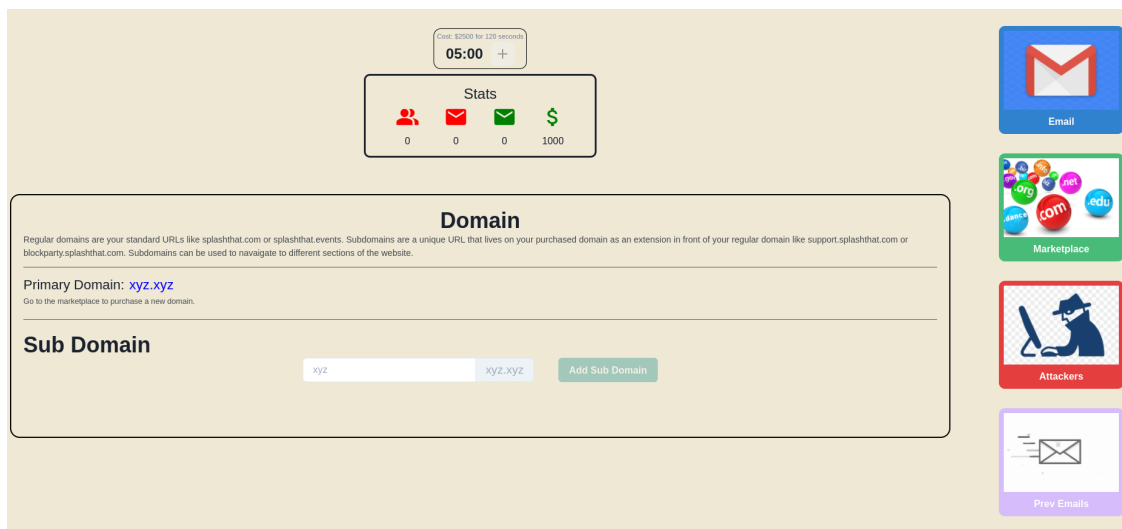


Figure 2.13: Initial version of the game included timer to incentivize the user to complete the game.

3. Evaluation

3.1 Insights

- Subdomains and confusion about Subdomains - spoofing and confusion

4. Discussion

- sendInBlue

4.1 Limitations

4.2 Future Work

4.3 Conclusion

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. Dirir, “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.