Automatically Generating Contextualised Responses to Phishing Reports

Sean Strain

MInf Project (Part 1) Report
Master of Informatics
School of Informatics
University of Edinburgh

2022

Abstract

In this project, I propose a novel system that will respond to email phishing reports with contextualised information and advice. The intention of this first stage of the project is to develop a key component of this system - known as the Auto-Responder - which analyses a given email for a range of phishing indicators and dynamically generates this response. This response is intended to display the evaluation of these indicators to the user to provide them with support in distinguishing phishing emails from non-phishing. The Auto-Responder was gradually developed to extract and evaluate a number of phishing indicators, and was verified to be able to do so consistently, reliably, and quickly. The predictive power of the system was evaluated by comparing its outputs over two large corpora, one of phishing emails and the other legitimate ones. The design process of the response was also initiated. The response design was confirmed to conform to a number of aspect ratios and can be delivered through the email framework. The goals of the second stage of this project are also outlined. As the incipient project in the development of this system, the project lays the groundwork for future work to complete the user interaction aspects of the tool.

Acknowledgements

I would like to thank my computer for not dying and erasing all my work, my chair for being so comfortable, and my coffee for keeping me awake.

Table of Contents

Chapter 1

Introduction

1.1 Motivations

“Phishing” is a catch-all term for a variety of social-engineering attacks where an attacker sends a fraudulent message with the intent to deceive someone into performing an action they would not otherwise do. Examples of such actions include: revealing some information to the attacker [83], misdirecting them to a fraudulent site [62], tricking them into downloading malware on their device [105], etc. . These attacks cause a growing amount of harm to society; they often result in significant financial loss and breaches of data [64, 87]. Further, phishing is the most common form of cybercrime in the USA and the UK [92, 55]. 36% of corporate data breaches worldwide were caused by phishing attacks in 2021 - up from 25% in 2020 [113]. Thus, it is an issue at the forefront of cybercrime prevention.

Phishing attacks can occur through a range of communication media, such as through text messages, instant messaging, and the telephone, but by far the most prevalent medium is email. Emails are very common in today’s world, over half of the world’s population possesses at least one email account, and many individuals have multiple accounts [58, 100]. They have many important and legitimate uses and exist as an almost ubiquitous form of communication used frequently in institutions, workplaces and universities worldwide. However, this prevalence provides a potential vector for malicious actors to attack, with 96% of phishing attacks being delivered through email in 2021 [113, 103]. As it constitutes such a high proportion of attacks I will refer to email phishing as simply phishing (or phish) henceforth.

A large amount of research has studied technical solutions that detect and destroy phish before it reaches a user’s email inbox. Determining whether a given email is phish or not is a complex issue. Attackers constantly update and change their behaviours and techniques in order to stay ahead of phishing prevention measures [38]. With the advent of machine learning, many email services have started to employ phishing filters using these technologies. Today, they eliminate the vast majority of phishing attacks. In 2021, Google used their TensorFlow machine-learning framework to build a filter that was able to block upwards of 99.9% of phishing attacks, spam, and malware on the GMail platform - over 100 million emails per day [78]. However, despite the successes of automatic phishing filters, Khonji et al. states no system or group of systems is enough to mitigate all the vulnerabilities we face in preventing phishing - we cannot block them all [69]. Phishing emails will continue to arrive in email inboxes for the foreseeable future.

When emails such as these make it through the automated defences, it comes down to the user who opens and reads the email to prevent the attack from progressing by correctly identifying the email's true nature and refusing to interact with it. These users have contextual information that automated tools can't possess, such as account details, which enables them to be a valuable component in the anti-phishing framework [116]. Thus, supporting those that receive a phishing email in correctly identifying the email as malicious is important for data security and mitigating the damage caused by these attacks [97, 90, 116].

One way to provide this support is email reporting. Organisations commonly encourage their members to report emails they consider suspicious for review by an expert. This way, an information security expert at the organisation can give their members bespoke advice and support. Such reporting has become trivially easy, with phishing report buttons becoming *"easy buttons"* [95], resulting in a high false-positive report rate - 61% in 2021 [48]. This ease of use places high demands on Security Operation Centres (SOCs), and they are typically unable to respond to all reports in a timely manner. SOCs take on average 17 to 25 minutes to inspect and respond to a potentially malicious email [57], but the time to data being compromised following a phishing attack can be measured in seconds to minutes [112]. So, with quickly responding to reports being essential, SOCs typically reply automatically to a user's report with an automatic, standard response (an auto-response) with generic advice while they analyse the report.

However, these auto-responses do not take into account the context of the email that was reported, responding with the exact same language regardless of an email's content. For example, the University of Edinburgh Information Security team will respond with the same stock response to every report, containing advice such as "don't click on any attachments" regardless of whether the email has an attachment or not. This is despite the fact that phishing emails typically contain a number of cues that a user could utilise in determining whether that email is phish [51], but users often don't have the knowledge [115, 51] or the motivation [117] to extract and analyse these cues.

Thus, in this project I propose the creation of a system that can improve upon these responses. Such a system would facilitate the reporting of an email by a user, automatically analyse phishing cues in that reported email, dynamically create a response to the user containing the information obtained in a readable form, and send this response to the user for their perusal. This would be with the intent to assist the user in identifying a phishing email by making the process of analysing an email easier. In my research I did not discover a similar system that provides such a function; I propose that this system is novel.

I intend to complete this system in two stages. In this project I outline the first stage of development, in which I create one of the system's components known as the Auto-Responder. This component will be responsible for the analysis of an email and the generation of the response. Further, I discuss the future objectives the second stage of development is intended to complete in finishing this system.

This project was conducted alongside the research of the TULiPS research group at the University of Edinburgh. This research group supplied this project with materials, such as genuine phishing emails, which helped inform the development of the Auto-Responder. The system is intended for use by students and staff at the University of Edinburgh and will be designed accordingly.

This project directly follows the work done by Zhang in 2021, who designed template responses and investigated potential features an Auto-Responder could use [122].

1.2 Goals

In this project, I completed the following goals:

- **Investigated** what information could be feasibly extracted from a given email by machine analysis.
- **Researched** the related literature to establish what phishing indicators a user would typically look for in a suspicious email, and to gather the literature's recommendations for systems similar to this project.
- **Developed** a component of a larger system, responsible for analysing an email for a range of phishing indicators and dynamically generating a response that can be shown to a user.
- **Initiated** the design process for that response.
- **Analysed** the differences in the information we can find in phish and legitimate emails to inform what is most indicative of a phishing email.
- **Tested** the component across a number of metrics.
- **Outlined** the future direction of work in the second stage of the project.

Chapter 2

Email Phishing

Phishing is a *socio-technical* problem [32]. That is to say that it is a problem that exploits how people interact with computers; to succeed, an attack must overcome both human and machine defences. In this chapter I will discuss what an email is and the technological vulnerabilities emails possess, before explaining and how email can be exploited to manipulate and deceive end-users.

2.1 What the RFC5322 is an Email?

There have been many iterations of the definition of "electronic mail" (email). The current definition of is defined in Request For Comments (RFC)[1] 5322, issued in 2008 [101]. This mandates that an email consist of two parts: The header and the body. Users send and receive email using computer programs known as email clients, e.g., Microsoft Outlook [15]. The Simple Mail Transfer Protocol (SMTP) defines the basic standard used in communicating an email between two email clients [72].

The header contains a series of data fields that contain information useful to the receiver, such as where the email came from, when it was sent, its subject, etc. . It can also contain technical details for use by email clients, e.g., a *Message-ID* to uniquely identify the email. Emails today contain optional headers[2], with the Internet Assigned Numbers Authority (IANA) maintaining a list of 184 header fields that could be used in an email [73], although some are now obsolete.

The body contains the contents of the email, i.e., the message the sender wants the recipient to see. The RFC5322 standard only supports plain-text[3] in the body. However, this was extended by the Multipurpose Internet Mail Extensions (MIME) standard [56]. This standard is defined by a series of RFCs. It introduced new character encodings, the attachment of files, and allows for *multipart* emails - emails that have distinct parts

[1] RFCs are a series of peer-reviewed technical reports from the leading standard-setting bodies on the Internet [18].

[2] It may be surprising to some that an email doesn't strictly need a subject or even a *To* address. The only mandatory headers are the *From* address and *Date* the email was sent.

[3] More concretely, it only supports American Standard Code for Information Interchange (ASCII).

which can have different encodings. Another addition that MIME brought to email was the HyperText Markup Language (HTML). HTML allows for images and embedded programs, etc., to be directly placed in the body.

2.1.1 The SMTP Authentication Problem

SMTP does not define any way for the sender to authenticate. In this context, authentication is the guarantee that the email originated from the sender. Thus, with SMTP it is possible for any user to send an email as if they were from any email address - a phenomenon known as "email spoofing" [79]. This is done by manipulating the header fields to make the email appear to be from a different sender, allowing for the impersonation of the sender.

The email may contain header fields that could be used to show a discrepancy in the email caused by spoofing, such as the *Mail-From* header, but this requires the recipient to investigate the header fields. Most of these headers are hidden by email clients and investigating the entire source message - while possible - is rarely done by most users [116].

In an effort to solve the email spoofing problem, the Domain-based Message Authentication, Reporting, and Conformance (DMARC) protocol was created. It is an email authentication and reporting protocol. It extends and combines two other widely used protocols, namely the Sender Policy Framework (SPF) and Domain Keys Identified Message (DKIM):

- **SPF** allows the owner of a domain to specify a list of authorised addresses that may send mail on behalf of that domain. Any email originating from an address outside this list would be considered fraudulent and deleted.
- **DKIM** uses public key cryptography to verify an email was sent from an authorised sender. The sender uses their private key to attach a digital signature to the email, which the receiver then verifies using the sender's public key.

These two protocols were introduced to allow for authentication in the SMTP. However, there are several areas where these protocols alone are insufficient [49]: Many email environments have multiple systems sending mail often and this can include 3rd party service providers. This means authenticating every email is a complex task. Therefore, many senders send authenticated and unauthenticated email. However, the receiver cannot delete all unauthenticated mail because it cannot know if the sender authenticates every email it sends or not. This limited the usefulness of the protocols.

DMARC was introduced to solve these issues [49]. DMARC allows the receiver to ask the sender if an unauthenticated email should be accepted or not. Upon receiving an email purporting to be from the sender in the *From* field, the receiver gets the DMARC Domain Name System (DNS)[4] record for that sender and applies the sender's policy to the received SPF and DKIM authentication results. If the sender's policy does not permit unauthenticated email, and the email fails authentication, the receiver now knows it can delete the email. Furthermore, it adds "alignment". A received email "aligns"

[4] This is a system that translates URLs into Internet Protocol (IP) addresses [24].

if the *From* domain is equal to (or a subdomain of, depending on the DMARC policy) the DKIM domain and SPF domain headers. The results of all three protocols are then added to the header of the email.

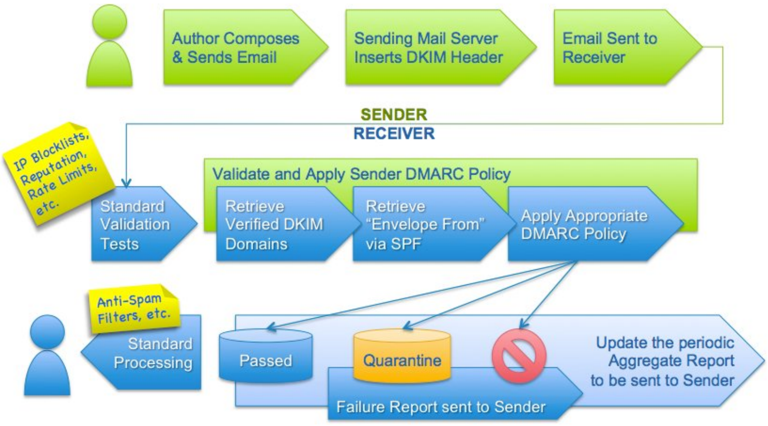

Author Composes & Sends Email
Sending Mail Server Inserts DKIM Header
Email Sent to Receiver
SENDER
RECEIVER
IP Blocklists, Reputation, Rate Limits, etc.
Standard Validation Tests
Validate and Apply Sender DMARC Policy
Retrieve Verified DKIM Domains
Retrieve "Envelope From" via SPF
Apply Appropriate DMARC Policy
Anti-Spam Filters, etc.
Standard Processing
Passed
Quarantine
Update the periodic Aggregate Report to be sent to Sender
Failure Report sent to Sender

Figure 2.1: DMARC and the Email Authentication Process [49].

DMARC, when used properly, reduces email spoofing to near-zero, but adoption is slow with only 32% of Fortune 500 domains having an acceptable level of protection [48] over ten years since DMARC was announced [50]. A plurality of domains with DMARC have a policy of "none" [48], meaning emails that fail to authenticate from that sender do not get deleted. Thus, it can be useful to check the authentication result headers to see if an email failed DKIM and SPF, as the email can be treated with added suspicion.

2.1.2 Email Addresses

Email spoofing allows the sender to make their from address appear to be exactly that of another sender. Attackers can also create email addresses that *closely resemble* that of the address of another sender [31]. These manipulations cannot be solved by the DMARC protocol [49].

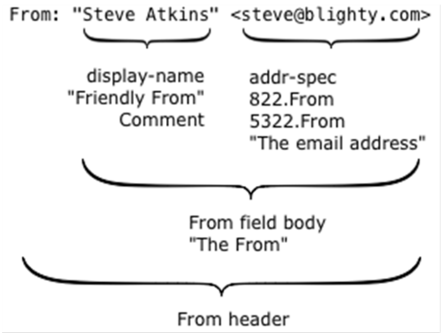


Figure 2.2: Anatomy of a *From* header [35].

These attempts to mislead the recipient revolve around the recipients misunderstanding of how a domain works. As these manipulations revolve around the email address' domain, it is interchangeable with the methods attackers use to manipulate URLs, discussed in section 2.2.

Irwin states that a key identifier of a phishing email is that it is from a publicly available domain [66]. He writes “No legitimate organisation will send emails from an address that ends ‘@gmail.com’. Not even Google.”. This can be used to identify phishing attacks. If an email claims to be from a legitimate organisation, but the *From* header reveals it is from a publicly available email address, it maybe an illegitimate email. There exists a large amount of free email domains [10], however, so knowing them all may be difficult for a user.

2.1.3 Attachments

MIME allows emails to have attached files. While this has legitimate uses in sending files through the email framework, it presents a vector for attack. A phishing email could attach a file that contains malware. If the recipient is deceived by the phishing email into downloading the attachment, a malicious program could be run on the recipient’s machine, e.g., ransomware. While some email services use a list of unacceptable file types for attachments [94], attackers consistently create new ways of circumventing filters like these. 36% of phish contains an attachment [44] and 39% of users reported receiving an email with a suspicious attachment in 2021 [99]. To prevent this, recipients are advised to only open attachments in emails they are sure they can trust [66].

2.1.4 HTML

HTML allows for a sender to add colour, style, images, etc., to an email. It is worthwhile to note that HTML also allows for the execution of JavaScript code in emails. This presents a serious security vulnerability. Allowing code to execute on a machine through email means that an attacker could embed malware. Thus, many email clients block the execution of JavaScript in an email [118].

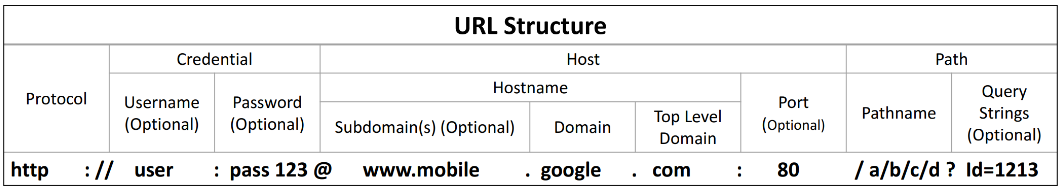
More relevant is the addition of HTML’s *a* (anchor) tag. This tag allows the sender to attach a URL to custom text. There is nothing preventing the sender from making that text another URL, misdirecting the recipient into thinking the URL points towards a webpage it does not. Furthermore, a recent trend is that of HTML *smuggling*, where an attacker obfuscates a malicious HTML file behind an anchor tag made to look like a safe file [106]. Once the tag is clicked, this downloads the malicious file to the victim’s computer. This can be detected by extracting the real URL from the tag, but many users lack the knowledge necessary to do this [51].

2.2 Uniform Resource Locators (URLs)

A URL is a reference to a web resource. A web resource is any entity that exists in a networked information system, such as a file, document or web page. These can be placed within emails for legitimate reasons.

URLs have a flexible, adaptable structure which makes them useful in many purposes. However, this flexibility means fully understanding them is difficult; despite their prevalence in everyday use, a body of research exists demonstrating how little users understand URLs [34, 31] even after training [47, 31].

As users don't understand URLs very well, URLs are the most commonly used and most successful vector of attack within phishing emails [99]. Such attacks revolve around making the URL seem as though it points towards a web page it doesn't point to. This involves the creation of a counterfeit web page. These pages are discussed in section 2.2.2. This prevalence means significant focus in this project should be placed on how best to assist users in understanding URLs to reduce their susceptibility to URL manipulation.



URL Structure

Protocol	Credential		Host				Path	
	Username (Optional)	Password (Optional)	Hostname			Port (Optional)	Pathname	Query Strings (Optional)
			Subdomain(s) (Optional)	Domain	Top Level Domain			
http ://	user :	pass 123 @	www.mobile .	google .	com :	80	/ a/b/c/d ?	Id=1213

Figure 2.3: Structure of a URL [34].

2.2.1 How a URL can be manipulated

Volkamer et al. identified four URL manipulation techniques [114]:

- **Camouflage** - Use the brand name as as subdomain or together with an extension, e.g., amazon-canda.com vs. amazon.com.
- **Mangle** - Use letter substitutions or misspellings, e.g., amaz0n.com vs. amazon.com.
- **Mislead** - Embed the authentic name within the URL, e.g., amazon.slz.com vs. amazon.com
- **Obfuscate** - Use an IP address or arbitrary name, e.g., slsdz.com or 123.32.22.123 vs. amazon.com.

Identifying these URL manipulation tricks is not easy for many users [31, 114]. There also may be legitimate reasons to change a URL. For instance, shortening a URL is not always malicious in its intent. There exists URL shortening tools, like ShortURL [19], that create a smaller URL to reduce its length. These change the domain name to that of the URL shortening tool's domain name. There exists tools that obtain the original URL given a shortened one, but this involves users be aware of such tools and be motivated to use them, which is discussed in section 3.1.2.

One example of a phishing attack that manipulated URLs used Google Translate [40]. The attack put their URL into the Google Translate tool [13], which prepended the legitimate appearing "translate.google.com" to the URL, which could make it appear safe to some users [31].

Much research exists on how to assist users in identifying these manipulations, as discussed in section 3.2.1.

2.2.2 Identifying Phishing Websites by Analysing the URL

The ultimate purpose of these URL manipulation tricks is to deceive the user into going to a webpage they believe is legitimate or controlled by someone trustworthy. These webpages are designed to look closely like that of the organisation being emulated. It can be useful to analyse a URL and its associated domain for certain indicators that may reveal its true nature.

Phishing webpages are typically hosted on Newly Registered Domains (NRDs) [59]. When an NRD is used in a phishing attack, they are identified fairly quickly, with 84% being taken down within a day of their registration [104]. Therefore, it can be useful to determine the age of a website pointed at by a URL, as an NRD can be treated with added suspicion. Tools like WhoIs [23] reveal domain registration information, which includes the domain age. However, there is evidence that phishing attacks are using artificially aged domains by leaving them dormant for some time [109] and some NRDs are legitimate, meaning this indicator is not perfect. Regardless, it remains that NRDs are more likely to be used in phishing attacks [59].

Another indicator is relative popularity. For example, Google.com is the most popular domain on the Internet [1, 81]. If a URL appears to be pointing towards the Google domain, but actually points towards a relatively unpopular domain, it follows that the URL has been manipulated [81].

2.3 Quick Response (QR) Codes

A QR code is a machine-readable, 2-dimensional optical label that stores data that can be read by a computer [108]. The uses of QR codes are many [86], including holding the data for strings of text, telephone numbers, email addresses, etc., but the most pertinent use of a QR code is that it can be used as an alternative representation of a URL. As noted by Yao and Shin, QR codes are ”more machine-readable than human-readable” [120]. They go on to state that this presents a potential vector of attack, as an attacker can hide their malicious website behind a QR code representing a URL.

QR codes can be embedded within an email as an image which may be rendered by a user’s email client. The user may then scan the QR code using a QR code scanner (a tool that reads the data stored in a QR code). Yao and Shin identified 31 available QR code scanners, of which only 2 had security warnings pertaining to malicious websites. Identifying a malicious QR code in an email would involve scanning the QR code for the URL it represents, and analysing it using the same identifiers discussed in section 2.2.

Figure 2.4: An example of a QR Code.

2.4 Spotting Phish in an Email's Body

Phishing attacks commonly have predictable characteristics in their use of language that can be used to identify them [66, 32]:

- Absence of the recipient's name - Phishing attacks often use a generic salutation like "Dear Customer" or "Dear User" instead of the name of the recipient [36]. This is common in bulk phishing attacks, discussed in section 2.4, as the email is sent to a large amount of recipients.
- Urgent language - the use of language to create a sense of urgency is a typical hallmark of phish [32]. This is to encourage the recipient to engage with the email and the attack vector (URL/attachment, etc.) the phishing attack uses.
- Poor grammar or spelling - Often phishing emails will have spelling mistakes or grammatical errors, as noted by Alkalilh et al. [32]. Alkalilh et al. theorise this is to circumvent spam filters that are looking for certain keywords commonly associated with phish. However, Harrison et al. noticed in their study that users often miss typographical errors; they found this indicator had no impact on the user's ability to detect phish [60].

2.5 Types of Phish

With the various technical methods that can be exploited in an email identified, it is useful to discuss the types of phishing attacks. The two types of phishing attacks relevant to email phishing are [99]:

- Bulk Phishing - The most common form of email phishing [99], bulk phishing is deployed en-masse. These attacks are not targeted to a specific person. Instead, they send mail to many recipients to maximise their success rate. The attacker will try to closely emulate legitimate companies' communications.
- Spear Phishing - A focused phishing attack personalised for a specific individual. The term "Whaling" is also used when the spear phishing attack targets a high-value individual such as a company's CEO. Spear phishing is considered a much more sophisticated attack than bulk, with attackers often performing research on their target(s) before they execute the attack [36].

In the 2022 State of the Phish Report [99], organisations self-reported what attacks they had experienced. 86% were targeted by bulk phishing and 79% by spear phishing.

Chapter 3

Related Work

In the previous chapter I discussed what can be technically exploited in an email to deceive users, but it is also important to investigate what features users look for when deciding if an email is phish to inform what the tool can do to help them. Moreover, investigating the literature regarding what users are being taught to look for, what information other similar tools to the Auto-Responder try to show to the user, and the recommendations that the literature provides for such tools, can also inform the tool.

3.1 User Training

User training involves increasing the awareness and level of knowledge users have about phishing through education. Research has been conducted on a variety of methods of delivering this education, including: giving classes [102], providing materials that explain key phishing features [41], simulating phishing attacks [76], creating interactive games [75], etc. . Through training, researchers aim to lower the amount of users that are deceived by phishing that bypasses the technical filters.

Early work in the field was undecided on the effectiveness of user training. Herley used an economically based argument to rebuke user training, claiming that the ever-changing landscape of phishing invalidates previous learning; users must be constantly retrained and this outweighs the costs of breaches in security [61]. He concluded that user training has to be simple and clear, otherwise it is ineffective and wastes time. Martin Overton, a security expert, remarked that “user education is a complete waste of time” [54].

However, more recent work [97, 115, 116], stresses that humans can be useful in the security framework and that a user-centred approach to security is valuable. Pleeger et al. stated that training users can enable them to become key components in the security framework [97]. Wash et al. [115] furthered this by postulating that the user has “unique knowledge and valuable capabilities” in identifying phish. Ultimately, 99% of organisations had security awareness training for employees in 2021, with email-based phishing being the most commonly trained topic [99]. As such, a large body of research exists regarding how to best educate users about phish.

3.1.1 What Users Look For in an Email

To better inform what anti-phishing training should focus on, a number of studies have investigated how users interact with an email they consider suspicious.

To understand why users fall for phish, Downs et al. investigated security-naive[1] users' sensitivity to features that are commonly used to identify phish [51]. They surveyed 20 individuals about phishing emails they had interacted with in the past and what they found within them. The participants identified spoofed *From* headers, suspicious URLs, etc. . The researchers discovered that what cues were found were not used correctly by the participants, with some interpreting the cues as something wrong with their computer as opposed to a security risk. The researchers concluded that this was not good enough; users do not possess enough knowledge about phishing and recommended phishing education be used in the future to improve this.

Wash et al. conducted a similar study on what non-experts[2] notice in phishing emails [116]. They surveyed 297 users, and discovered that users look for more *"contextual"* information, i.e., how the email relates to their expectations and what they've seen previously. These are features that a technical anti-phishing tool can't possess. However, they discovered that non-experts do not often identify common *"conclusive distinguishers"* - features that can be used to conclusively determine an email's legitimacy, e.g., the sender's name, attachments, or suspicious URLs. Further, Harrison et al. noticed in their study of 113 university students that typographical errors in a simulated phishing email they created went unnoticed by all participants, and the participants were more likely to think the email originated from the participants' university despite it coming from a freely available address [60].

Wash also conducted a study on what experts[2] notice in a phishing email [115]. Wash surveyed 21 IT experts and asked them about a phishing email they had dealt with previously. Wash then asked them what feature in the email revealed to them that it was phish. The most common cue was an *"action link"* which Wash describes as a URL that asks the user to perform an action, e.g., login to a website. The paper also describes a reasonable upper bound on how well a user can be expected to do in identifying phish, with the experts not always performing a complete investigation of an email they consider suspicious. Wash claims that the thorough investigation of every URL in an email is unrealistic for users, even for experts.

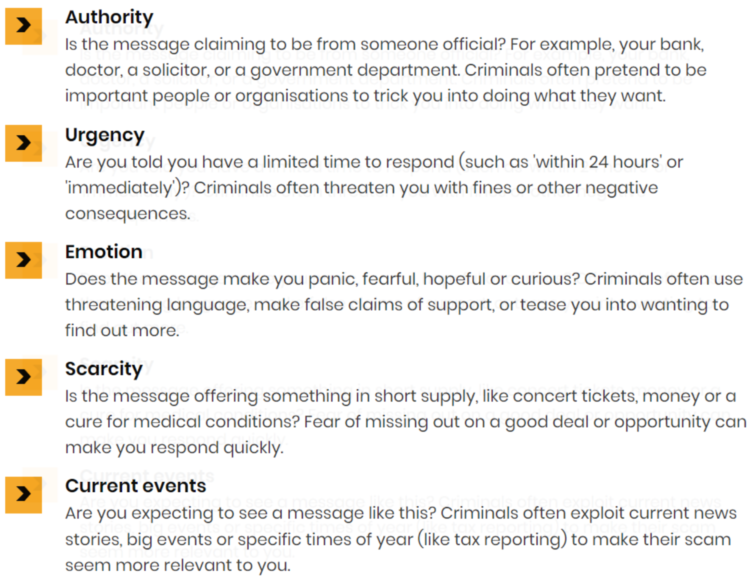
3.1.2 Preventative Training

Preventative training (also known as upfront training) aims to teach users "best-practices" when it comes to phishing before a phishing attack occurs. This way, it is thought a user will learn how to identify suspicious looking websites and can judge the trustworthiness of communications they receive. Governments [93, 41] and companies [52, 65] provide online materials, such as the one provided by The UK Government's National

[1] Their criteria for "naive" included having never adjusted security preferences on their computer or having never helped another user with their computer, etc. .

[2] They differentiated between experts and non-experts using Klein and Hoffman's criteria for expert [71].

Cyber Security Centre - shown in Figure 3.1 - that explain what to look for in an email. Kumaraguru et al. discovered that online materials such as these can be effective when they are read by users [77].



Authority
Is the message claiming to be from someone official? For example, your bank, doctor, a solicitor, or a government department. Criminals often pretend to be important people or organisations to trick you into doing what they want.

Urgency
Are you told you have a limited time to respond (such as 'within 24 hours' or 'immediately')? Criminals often threaten you with fines or other negative consequences.

Emotion
Does the message make you panic, fearful, hopeful or curious? Criminals often use threatening language, make false claims of support, or tease you into wanting to find out more.

Scarcity
Is the message offering something in short supply, like concert tickets, money or a cure for medical conditions? Fear of missing out on a good deal or opportunity can make you respond quickly.

Current events
Are you expecting to see a message like this? Criminals often exploit current news stories, big events or specific times of year (like tax reporting) to make their scam seem more relevant to you.

Figure 3.1: How to spot scam messages or calls [41].

Lastdrager et al. explored a classroom approach to preventative training [80]. They studied 353 children aged 9-12, splitting them into control and intervention groups. The intervention group received a 40-minute anti-phishing presentation. Both groups were asked to complete a test of 10 questions at 0, 2, and 4 weeks after the presentation. The intervention group was initially better than the control at identifying phishing, but this effect decayed over time; the 4 week test showed an insignificant difference between the scores of the groups. However, they did note that the invention group correctly identified legitimate emails more often than the control. Robila and Raguci did a similar study with university students. They involved 48 students in an anti-phishing course, finishing with a phishing IQ test. This test contained 6 phish and 6 non-phish emails. The participants were asked to classify each email. The average IQ was 57.29%, meaning slightly under one in two emails were misclassified by the participants [102].

This method of training is criticised for its lack of effectiveness. It necessitates the user be motivated to access and read the online tools or engage in the course. Whitten and Tygar noted that security is a secondary goal for the user [117]: The user does not want to sit at their computer and learn about security; they want to engage in other activities and have the security in place for them while they do these things. Kumaraguru et al. furthered this notion by claiming that users have difficulty with motivation for two key reasons: They believe phishing is unlikely to occur, and may be overconfident in what they know [75]. Moreover, recent research [67] claims that users reject this type of training due to of boredom.

To combat this, a change of direction was required. Later research began to shift towards the use of embedded training with the thinking that placing training materials in the users' daily routines would be more effective.

3.1.3 Embedded Training

Embedded training is a real-time training method. The training is *embedded* in the user's day-to-day life. Kumaraguru et al. investigated the effectiveness of this in their seminal study on applying the concept of embedded training to phishing [76]. In the study, the researchers exposed participants to a list of emails, some of which emulated phishing emails containing a trapped URL that points to a webpage the researchers controlled. This webpage then explained the nature of the email and provided the user with training materials to prevent them from falling for real phish in the future. They repeated this experiment a week later to measure the participants improvement vs. a control group who had been exposed to preventative training instead. They concluded that the embedded materials were more effective than the preventative.

Embedded training gives the user a strong incentive to learn as it necessitates that they just made a mistake - giving a sense of shock and creating a solid memory anchor point; this was noted in the paper by Kumaraguru et al. [74], one user remarked that they related with the training because it made them think "You could've just gotten scammed! You should be more careful". Furthermore, it is believed to save time over other methods of training as it is run continuously and automatically [67].

Despite this, Caputo et al. obtained contrary results when studying embedded training [39]. Their results split users into three separate groups, "all-clickers", "no-clickers" and "inexplicables". The all-clickers always clicked despite training, the no-clickers never clicked, and the inexplicables appeared to behave randomly. They concluded that anti-phishing training did not help the treated group detect phish. They did not elaborate on how these users could be better trained. However, they recommended that a phishing report tool would be a meaningful addition to the phishing defence system.

3.2 Support Tools

Users have difficulty responding effectively to training, and automatic systems cannot catch all phishing emails. Therefore, support tools, also known as combination tools [116], look to combine automated methods with the user to enhance protection. This involves augmenting the user decision making process by presenting the user with machine obtained information. The key aspect of these tools is in helping the user find and evaluate the features that can differentiate legitimate email from phish. The proposed Auto-Responder falls under the definition of a user support tool.

The first anti-phishing defences were combination tools. SpoofGuard, developed in 2004 by Chou et al., is a browser plug-in that uses a traffic light colour system to classify websites the user navigates to [43]. Today, a user support tools come in many forms.

3.2.1 Anti-Phishing URL Tools

As discussed in section 2.2, a key attack vector used in phishing attacks is URLs. Phishing attacks manipulate URLs to mislead the recipient. Consequently, a large amount of anti-phishing tools and research pertain to helping users understand URLs.

3.2.1.1 How Well Can Users Understand URLs Without Assistance?

To inform how anti-phishing tools can help users understand URLs, Albakry et al. investigated users' skills in interpreting URLs unassisted [31]. 1,929 participants were asked a series of questions about 23 URLs with varying structures. The questions were aimed at determining whether the participant knew what website the presented URL pointed towards, whether the URL was safe, among others. These participants were separated into three groups depending on their experience with computers. They identified that the most competent group, while better at analysing URLs than the other two groups, were only able to correctly answer 55.8% of the questions asked. They suggested that users be given better support and training in reading URLs.

3.2.1.2 Assisting the User in Understanding a URL

To improve upon users' understanding of URLs, researchers have investigated various ways of making the URL easier to read and interpret for a user. Methods include highlighting the domain, providing warnings, providing feedback, etc. .

Petelka et al. evaluated different active methods of warning users about suspicious URLs [96]. They compared various methods of warning placement and functionality, such as only allowing a URL to be interacted with through a warning interface. They concluded that a warning placed near a suspicious link is more effective than a banner at the top of the email that contained the link, and that forced interaction with warnings was the most effective method of reducing click through rate. This corroborates the finding by Kumaraguru et al. in their embedded training research - both suggest that in-the-moment, obvious phishing defence is more effective than measures users have to choose to interact with.

Lin et al. [82] did a similar study where they highlighted the domain of a URL to make the URL easier to understand. This improved phishing URL detection rates, but the researchers stated that the effect was "not as well as we would have liked". To improve this, they suggested future tools should show the domain name in a URL in a "dedicated area separate from the rest of the URL". They also suggested "pruning" the URL (i.e removing the nonessential information) could be useful. This was corroborated by Volkamer et al. [114] who analysed 411 participants evaluating 16 URLs, half phish, half legitimate. They noted an increase in the participants detecting malicious URLs when the URL had been pruned, stating that "pruning undeniably improves detection rates".

A more sophisticated approach to making URLs more understandable was that done by Althobaiti et al. [34]. They created a chatbot that interacts with a user on the messaging service Slack. Given a URL from the user, it detects the four methods of URL manipulation discussed in section 2.2, before providing the user with feedback on the given URL. Their findings showed that using the chatbot increased the users' ability to detect phishing URLs, even unassisted after using the bot. In a similar study by Althobaiti et al. [33], the researchers identified several key features about a URL and it's associated domain that could be useful to show to a user. This included showing the top-level-domain, the domain's popularity, and the domain's age. They discovered that showing the user these features helped them to decide if the URL was safe or not.

There are a wide range of freely available URL analysis tools on the Internet. WhereGoes [29] is a tool that detects if a URL redirects to a different page. A redirect is an intermediate webpage that automatically directs the user to a different webpage. This could be used in a phishing attack to circumvent Domain Name System BlackLists (DNSBLs), discussed in section 3.2.2. WhoIs [23] is a tool that provides a comprehensive overview of the domain in a URL, including contact info for the domain registrar. Tranco [1] is a list of popular domains created as a result of the research done by Pochat et al. [81]. It is made for use in security, as popular domains are less likely to be manipulated into stealing information without authorisation (although this is not impossible [62]).

3.2.2 Browser Based Defences

Some Internet browsers have built in phishing defences, while others allow for the browser to be extended by plug-ins[3] that can be used in phishing defence. One such defence is Google Safe Browsing (GSB) [12], which is integrated into Google products.

GSB uses a DNSBL. A DNSBL is a list of IP addresses, with each list having different criteria for determining if a given IP should be added to the list, such as if it was identified in a phishing attack [9]. These lists can be used by anti-phishing tools to warn users when they are about to navigate to a blacklisted phishing website. However, there are some criticisms of these lists, with claims that DNSBLs such as that used by GSB could be used as a powerful tool for Internet censorship [53]. Regardless, Jung and Sit identified that DNSBLs contained 80% of the spam sources they identified, making them valuable tools in phishing defence [68].

Other plug-ins, such as Netcraft [17], display information about the website the user is visiting, such as the page's popularity and location, the usefulness of which was discussed in section 3.2.1.

3.2.3 Similar Tools to the Auto-Responder

Of all the anti-phishing tools, the most similar to the proposed Auto-Responder are *triage* systems. Triage systems are made for use by SOCs, allowing them to prioritise an email's investigation depending on how dangerous that email is calculated to be by the system. Examples include TruSTAR's Phishing Triage Intel Workflow [28], CoFense Triage [16] and ThreatQuotient [21]. There does not exist any scientific literature on these tools, and their functionality is limited. Of the tools mentioned, the

[3] A plug-in is a piece of software that adds new functions to other software, but doesn't change the original software itself.

most comprehensive is that of ThreatQuotient. It extracts and displays a variety of features of a given email, such as its DKIM-Signature and other headers, and information regarding attachments. However, such a tool would not be very useful from a user perspective given the literature discussed previously. For example, it does not present the technical information in an understandable way for a non-expert user, which was recommended by Downs et al. [51]. Ultimately, it is useful to be aware of these tools, but the amount of knowledge we can draw from them is limited.

3.3 Summary

In this section, I discussed the methods currently in use to prevent phishing. I discovered that there does not currently exist a solution that automatically generates a contextual report of an email for a user like the proposed Auto-Responder. That does not mean that there is not a large amount of anti-phishing research that can inform the creation of this novel system.

A prevailing theme is observed when analysing the previous literature on phishing training - user-support tools can be valuable in assisting users detect phishing. Downs et al [51] suggested improvements for future anti-phishing security tools, writing that most users won't understand information about domains, website certification and other technical concepts. They suggest that there should be tools that provide information on such concepts in a meaningful way that doesn't use technical jargon. Wash et al. [115] suggested that a "help me troubleshoot this email" button with relevant information about the email would be beneficial. This would help non-expert users exploit features they don't usually consider in their decision making process. Wash [116] showed that even experts do not analyse each URL thoroughly; this suggests that analysing each URL automatically would be of benefit even to information security experts. This suggests the use of automated support tools would be effective in assisting users when analysing their emails.

However, while user-support tools are useful, Yang et al. argues that combination tools and user training must be used in combination in order to increase the effectiveness of both [119]. Furthermore, user-support tools do not often relate to the email itself, rather the vectors of attack within the email - such as the URLs or attachments. There does not currently exist a system that displays all the relevant information in an email to the user. A system could be constructed to incorporate the key functionalities across the spectrum of tools already in place to centralise the defenses.

By fulfilling these goals, the system would fill a gap in the existing literature and anti-phishing defence framework.

Chapter 4

Implementation and Design

The Auto-Responder is a component of a larger system being developed in two stages. In this chapter I outline the first stage of development - the creation of the Auto-Responder. In the first section I will describe the intended functionality of the system as a whole, and in doing so contextualise the Auto-Responder and the work done in this project. Then, I will describe how the Auto-Responder was developed, the decisions made and their justifications. Finally, I will outline the design of the response of the Auto-Responder, and the process used to create this design.

4.1 Implementation

4.1.1 System Overview

Before I began developing of the system, I investigated what the functionality and goals of the finished system should have. To inform this investigation, I outlined the primary use case of the system:

> "A user reports an email and receives in response contextualised information and advice about that email"

Within this use case there are 6 key questions that must be answered:

1. How does the user report an email?
2. How is the response delivered to the recipient?
3. How is the email analysed?
4. What in the email should be analysed?
5. What should the response look like?
6. What advice should be given to the user?

To begin answering these questions, I first drew a plan for the system as a whole, shown in Figure 4.1. This system diagram shows the planned flow of control throughout the system. It has been simplified for areas that will be completed in the second stage of the project. With the diagram complete, I began work on answering each of these questions using the diagram to inform this process, which I describe in the rest of this chapter.

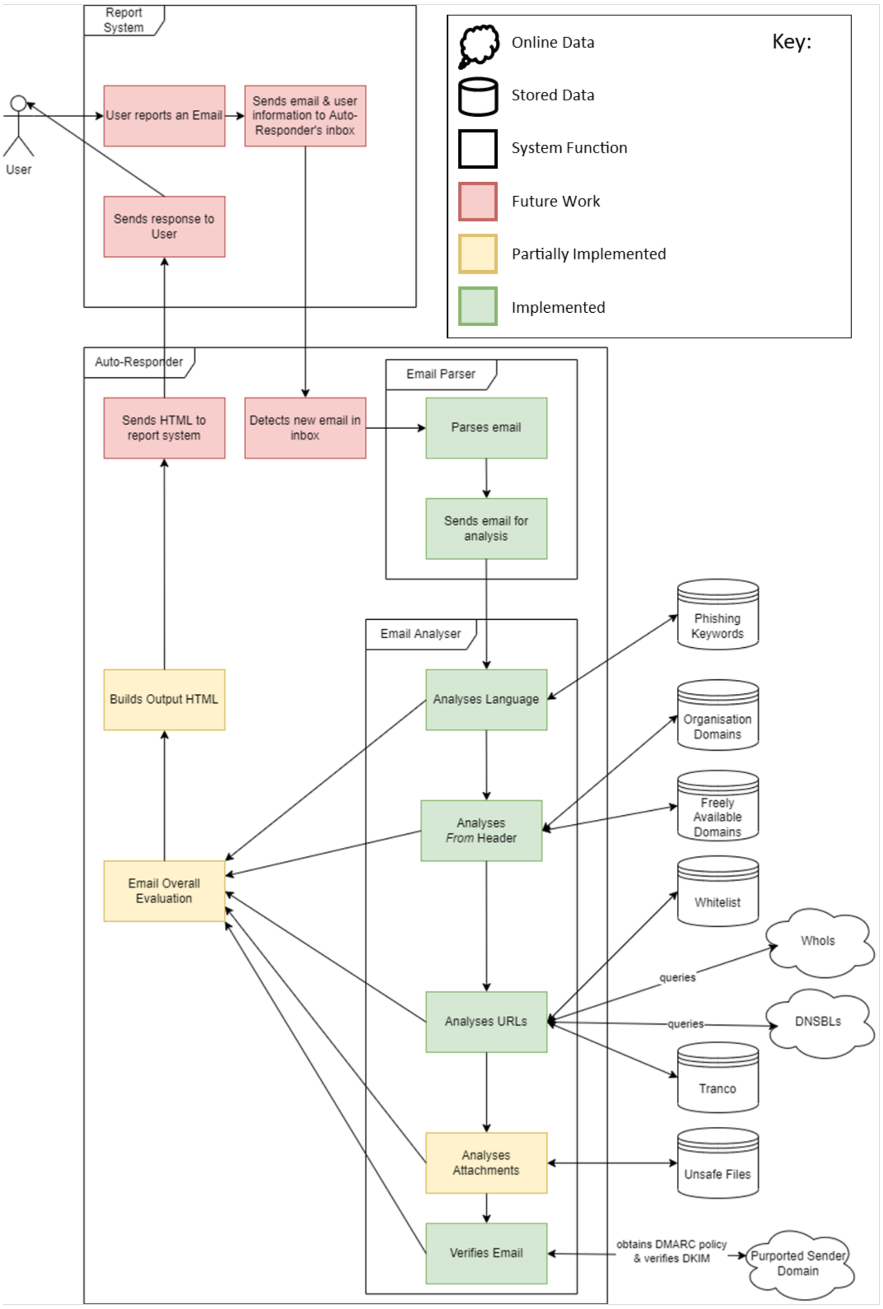


Figure 4.1: Overall System Control Flow.

4.1.2 Initial Investigations

Initially this stage of the project included the email reporting system as described above. I spent a significant amount of time looking into this aspect of the system.

Whitten and Tygar, as noted in section 3.1.2, emphasised that users will not go out of their way to ensure their security [117]. Instead, it should be as easy as possible for a user to practice security. Furthermore, Kumaraguru et al. showed that placing security in user's usual routines is effective [76]. Therefore, it is important to investigate whether the tool can be integrated into or extend the technologies already in use by the target user. This will allow the tool to be as undemanding as possible to use for the user.

To do this, I identified the target userbase of the overall system and the technologies these users already use for their email. As discussed in Chapter 1, the system will be made for use by students and staff at the University of Edinburgh. These users use the email client Microsoft Outlook [15]. It was identified that Microsoft Outlook provides a way to extend the email client through the development of "add-ins" [46]. An add-in is a software module that integrates into the email client, adding additional functionality and features. Creating an add-in could allow the tool to be accessed by a user with relative ease. As shown in figure 4.2[1] it could allow the user to click one button to report an email.

A significant portion of time was devoted to investigating the feasibility of creating an add-in. The creation of the add-in proved to be difficult; it involved an in-depth understanding how this professionally made, commercially available email client worked. I had to spend some time learning the system before learning how to develop an add-in. Eventually I successfully created an add-in which added a button to the Outlook's drop-down menu. I created this with the intent of using this button to allow the user to report the email. The button is seen in figure 4.2[1].

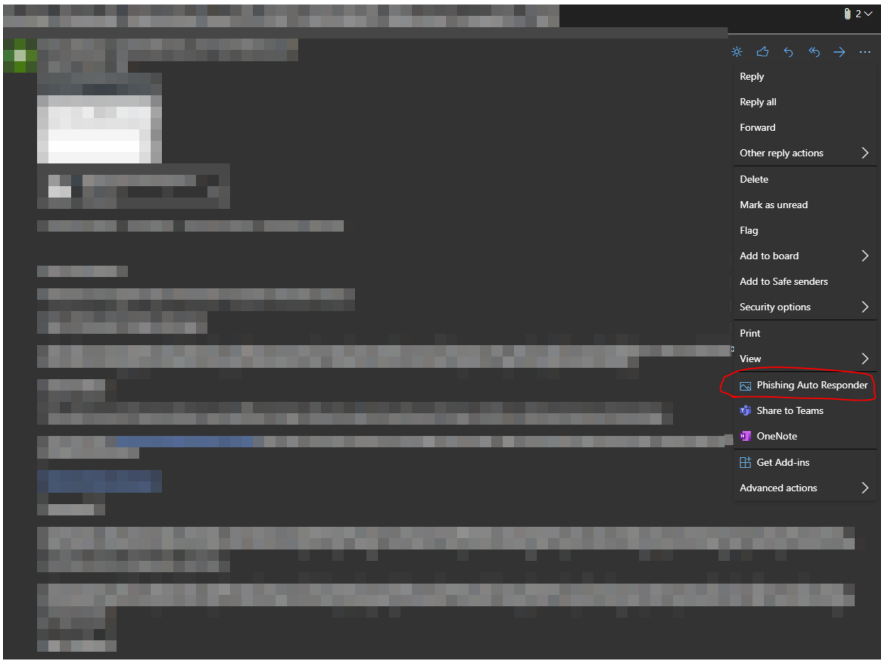


Figure 4.2: The add-in button (circled in red) displayed in the Microsoft Outlook Email Client.

However, it was discovered that Outlook denies important information in the email

[1] The email used to display the button has been blurred to protect the privacy of the email's sender.

from being accessed by add-ins, such as DKIM-signatures. This could not have been known without investigating the add-in first to see what was possible. I decided that the restrictions an add-in places on the functionality of the tool were to great. Furthermore, the API used by Outlook for add-ins is set to be decommissioned in November 2022 [27], meaning that the tool would become obsolete. As a result, I cancelled work on this avenue of investigation.

The question of "how does the user report an email" remains unanswered. Due to time limitations I decided to move focus to the analysis of the emails as done by the Auto-Responder; I considered the analysis more critical, as the reporting system relies on the output of the analysis in order to respond to a report. The user-reporting aspect of the system was delegated to a PhD student working within the TULiPS research group and is to be integrated with the Auto-Responder in the second stage of this project.

4.1.3 System Features

With the focus now on creating a system to analyse an email, I began answering the question "what in the email should be analysed?". This revolved around identifying what in an email can be used for phishing attacks, which I describe in Chapter 2, and exploring the literature regarding email phishing, which I describe in Chapter 3. Below is a list of potential features I identified and the current state of implementation. If I made the decision to not implement a feature, I justify the reasons why this decision was made:

- **Authentication - Implemented** - As discussed in section 2.1.1, DMARC and its associated technologies (SPF and DKIM) allow a sender to guarantee to the receiver that the email originated from them. Some email clients do this authentication and append the results to the email header. However, an attacker can manipulate these headers. For instance, DKIM signatures can be manipulated such that a genuine signature can be preserved while the contents of the email changed [110]. Therefore, I decided that authentication would be done independently within the Auto-Responder itself.
- **Language - Implemented** - As discussed in section 2.4, there are several indicators in the contents of the email that can be used to determine whether an email is a phishing email. The Auto-Responder could analyse the language of an email using to detect poor spelling and urgent language. It could also detect the recipients name to determine if has been used in an email.
- ***From* Header - Implemented** - As discussed in section 2.1.2, the *From* address of a sender can be manipulated. The Auto-Responder could detect manipulations in the *From* header. It could also check the domain to see if it is within the University and, if not, whether it is a freely available domain such as "@gmail.com".
- **URL and Websites - Implemented** - As discussed in section 2.2 and later section 3.2.1, URLs are the most common attack vector used in phishing emails. The Auto-Responder could extract the URLs in an email and evaluate them by looking at both the URL and the website it points towards.

- **Attachments - Partially Implemented** - As discussed in section 2.1.3, emails can have attached files and this is a potential vector of attack for phishing emails. The Auto-Responder could analyse emails for attached files and evaluate whether they are dangerous. This could also reveal the use of HTML smuggling as discussed in section 2.1.4. Analysing an attachment that could potentially be dangerous is a complex task, so given the restriction of time, this stage of the project only outlines a cursory examination of attachments.
- **User Advice and Training Materials - Partially Implemented** - The ultimate goal of the Auto-Responder is to assist a user in identifying a phishing email. The Auto-Responder could give the user advice on how to act and how to make their decision, and provide materials pertaining to phishing to help them understand the concepts within the response. Fulfilling this feature is scheduled for the second stage of this project as delays in ethics approval prevented studies into how best to proceed with this aspect of the system. Currently, the Auto-Responder gives limited advice, outlined in section 4.1.7.
- **QR Codes - Not Yet Implemented** - As discussed in section 2.3, QR codes can be embedded in an email as an image. The Auto-Responder could extract the data in a QR code and analyse the data as it would an attachment and/or URL. The reasons this feature wasn't implemented is twofold: First, I decided to prioritise creating the logic to analyse URLs first, as this feature would depend upon that. Second, implementing this feature would be complex, and therefore time-consuming, as QR codes are embedded in images; it would require either rendering images and detecting QR codes within them, or detecting QR codes within an image's data.

4.1.4 System Design

To begin answering the question "how is the email analysed?", I first had to decide what technologies I would use.

I decided that the Auto-Responder would use the Electronic Mail Format (eml) file type. I made this decision because, as Prom reports, it has "achieved a certain status as de facto standard[]" [98] among email clients. This prevalence means the tool will be usable by the most users on the most email clients, allowing it to maximise the benefit it can provide. Furthermore, eml is stored in plain-text, meaning it can be changed to ubiquitous file types like "txt", and can be opened in Internet browsers [25]. This means that the files are easily read by a person, which is useful when comparing the output of the system vs. the eml file input. This decision was accepted by those working on other parts of the larger system.

The Auto-Responder needs to display its information in a readable format. As the response will likely be delivered to the user in the form of an email, I decided to use HTML as this can be directly embedded within the email. This format is also flexible in that it can be displayed in a variety of ways, such as on a webpage, giving the response a degree of flexibility if the method of delivering the response changes. Further, it can be made to dynamically adapt to different screen resolutions, which may be beneficial for usability.

A programming language had to be selected for the codebase. As the output is HTML, JavaScript was the first option as it is designed for use with HTML [30]. However, as Python is better suited for use in other components of the larger system and will be used for them, I decided Python will be used by the Auto-Responder as well. This will support compatibility across the whole system.

A key consideration is how the Python script interacts with the HTML template file. I investigated three ways of doing this: 1) The Django web-framework [7], 2) The Flask web framework [22], and 3) The `BeautifulSoup` Python module [3]. Django and Flask were created to allow Python to be used in web-development. As such, their functionalities go beyond that of editing a single HTML file. This means they have many unnecessary features and are challenging to learn. Thus, I decided that these technologies over-complicated the interaction between the python script and the HTML file, and I elected to use the `BeautifulSoup` Python module instead.

In Chapter 1 I discussed that The Auto-Responder project is part of a larger body of research being conducted by the TULiPS research group. The code-base behind the tool may be used by other researchers, or incorporated into a larger system. With this in mind, I adhered to the PEP 8 Style Guide for Python Code [111] to ensure readability, and commented my code heavily.

There are a number of lists that inform the Auto-Responder's analysis with certain phishing indicators. These are text files and can be edited. This means the functionality of the Auto-Responder can be customised easily, allowing it to be portable to other institutions/companies with different needs than the University of Edinburgh.

4.1.5 Email Parsing

Before the phishing indicators in an email can be analysed, the email is processed into a usable data structure, i.e., parsed. To parse the eml file, the Auto-Responder uses Python's integrated `email` module, which can be used as it compatible with the eml file type.

Emails are flexible in their structure and the data they include. As discussed in section 2.1, MIME allows an email to have several "payloads". Each payload may have a *Content-Type* header denoting how the data in a payload should be interpreted (JSON, HTML, etc.), and it may also have a *Content-Transfer-Encoding* header denoting how that data is represented (base64, quoted-printable, etc.). A payload may also be a list of other payloads, forming a MIME tree (see figure 4.3). This complex structure is made navigable by the `email` module, which separates the payloads automatically.

The key information required for the Auto-Responder in the email headers was extracted using Python's `re` module. This module provides regular expression matching operations. For instance, the purported domain of the sender was extracted from the *From* header using the regular expression "`(?<=@)[\w-]+\.[\w.-]+`".

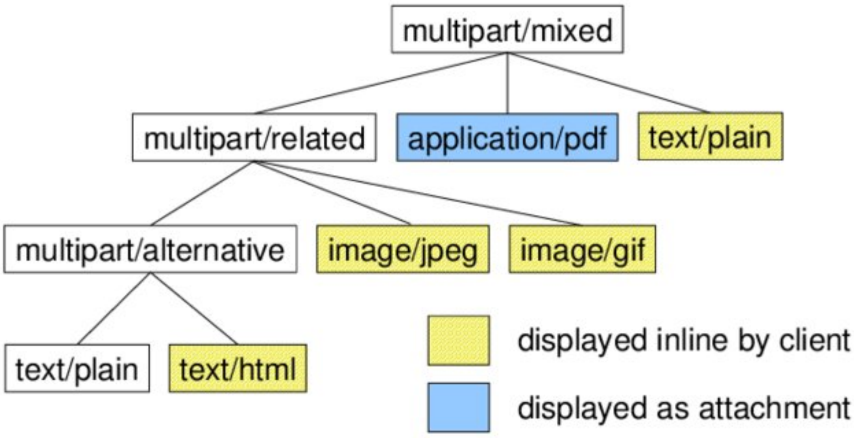


Figure 4.3: An example of a MIME tree [85].

4.1.6 Phishing Indicators

In this subsection, the methods used by the Auto-Responder for each phishing indicator it analyses are outlined and discussed. Each phishing indicator was identified by examining the background and literature behind phishing emails (Chapters 2 & 3).

Each implemented phishing indicator is analysed using the methods described below, and each is evaluated according to the results of my analysis of two large email corpora, outlined in section 4.2.

4.1.6.1 Authentication Indicators

To authenticate an email, the Auto-Responder first extracts the relevant headers (if present) from the email. It will authenticate the DKIM signature(s) against the purported sender domain. Then, it will query the purported sender domain for the permitted sender IP addresses, and checks this against the email's sending IP address to authenticate SPF. Finally, it queries the domain in the email's *From* address to receive the domain's DMARC policy. It then uses the DMARC policy, if there is one, to determine whether the email's headers align.

4.1.6.2 Language Indicators

There are 4 aspects to the Auto-Responder's language analysis. Namely, the use of the recipients name, the emotion of the language used in the email, the number of spelling mistakes, and the number of phishing keywords.

Before analysing the language indicators, each word in the body is extracted, then filtered using regex to exclude non-word text such as email addresses, numbers, etc. .

1. **Recipient Name** - The Auto-Responder will analyse an email's body and *Subject* header to examine whether or not the recipient's name is used. If the recipient's name is not present, it will mark the "recipient name used" phishing indicator with a warning. The filtering of the email using regex excludes instances where the recipient has their name in their email address. I believe this is desirable as

using a recipient's email address does not constitute use of the recipients name.

2. **Emotive Language** - The Auto-Responder uses the `text2emotion` python module to evaluate the emotion of the email's text. This evaluates five categories of emotion: 1) Happy, 2) Angry, 3) Surprise, 4) Sad, and 5) Fear. For each emotion, the Auto-Responder will determine if it is normal for a legitimate or not, as informed by the analysis described in section 4.2.

3. **Misspellings** - The Auto-Responder uses the `pyspellchecker` python module to check every word used and evaluate whether it was misspelled or not. It counts the occurrences of misspelled words in the email, and determines whether the amount of misspelling in the email is excessive or not. However, This spellchecker is unsophisticated, and different languages are always considered "misspelled" as it is configured for only one language at a time. In the Auto-Responder's case this is English. This leads to every university email being labelled with "excessive misspellings" because of the Scottish Gaelic attached at the the end. A more useful spellchecker should be found for this feature.

4. **Keywords** - The Auto-Responder uses the list of phishing keywords described in the paper by Bergholz et al. [37]. It counts the occurrences of phishing keywords in the email, and determines whether the amount of keywords in the email is unusual or not.

4.1.6.3 *From* Header Indicators

The University of Edinburgh attaches a special warning to every email that is sent to a University of Edinburgh address and from an address external to the University. This is because external emails are considered to be more dangerous than internal ones. I wanted the Auto-Responder to include this functionality and warn against external emails as well. As such, I enquired with a staff member at the University if it was possible to obtain a list of University of Edinburgh email domains for the purposes of this project, but I was told this was not possible. Instead, the Auto-Responder uses a limited list of University domains extracted from emails I have from the University. The *From* header is compared against this list to see if the from domain is contained within the list, if not a warning is added to the response. This list would need to contain a comprehensive list of organisational domains if this feature is to be used for security purposes.

Furthermore, as noted in section 2.1.2, phishing emails can originate from freely available domains. Knowing this, I decided that the Auto-Responder should extract the domain in the *From* header and determine if it was sent from a freely available domain. To do this, the Auto-Responder uses a list of freely available email domains created for use by the Customer Relationship Management (CRM) system called HubSpot [10]. This list was not created for this use case, but it was the most comprehensive list I could find in my research. If the Auto-Responder or a system like it was to be deployed for use by an organisation or company, this list should be improved upon first by replacing it with a security-orientated one.

As a result of my analysis described in section 4.2, freely available domains were more prevalent in the legitimate email corpora after adjusting for corpus size. This suggests that, while this feature may still useful if a user is not expecting the email to be from a freely available domain, its predictive power in determining if an email is legitimate or phish is limited.

4.1.6.4 Attachment Indicators

Originally, attachments in an email were found by analysing the email's MIME tree for instances where the payload header *Content-Disposition* was "attachment". However, I later discovered that payloads can have a *Content-Disposition* of "inline". The email client will attempt to open and render payloads that are inline, which is useful for rendering HTML, but these can also be files such as images or pdfs. Such files are not strictly considered to be attachments, but as they can be downloaded by the user, for the purpose of the Auto-Responder I decided to include inline payloads with a filename in the attachment analysis process.

While it is possible to thoroughly examine an attachment to determine whether it is safe or not, due to the restriction of time this was not implemented. Instead, a cursory examination of attachments has been implemented, the process of which is described below.

The Auto-Responder accepts as an input a curated list of "unsafe file extensions". This is intended to be a list of file types (e.g., exe, js) that are considered too dangerous to be downloaded because they can be used to harbour malware [121].

In my research, I found 3 different Unsafe File Lists. The "Internet Explorer Unsafe File List" [121], The extensions.org "dangerous and malicious file extension list" [6] and the Chromium "download file types" list [11]. I decided against the Chromium list as it includes 3,609 lines of file extensions, only some of which are marked as dangerous, so it would have required a degree of pre-processing to extract the dangerous file extensions. In the end, I decided to use the Internet Explorer Unsafe File List [121] as I had not heard of the website "extensions.org", and thus felt the latter more trustworthy.

The usefulness of this list is limited, as emails using these attachments will often either fail to send or be deleted before their arrival in an inbox as some email clients operate with similar lists. Indeed, if the Auto-Responder is being used with Microsoft Outlook, as described earlier, all of these attachments will be blocked by Outlook first as it uses the same list, and this renders the list redundant. However, using a different email client I attempted to send myself emails with empty files that use some of the file extensions listed to test this function and was able to succeed, demonstrating that the list has some purpose outside of Outlook.

The Auto-Responder will mark the attachment phishing indicator as dangerous if an unsafe file is detected. If an attachment is present, but isn't an unsafe file type, the Auto-Responder will mark the attachment phishing indicator with a warning. If no attachments are detected, it will be marked safe.

4.1.6.5 URL and Website Indicators

As discussed in section 2.2, URLs are the most common attack vector in phishing emails. With this in mind, particular consideration was placed on extracting and analysing the URLs in an email. It accounts for the majority of the processing done by the Auto-Responder. Each link is extracted from the email, processed into a dictionary of domains, and each domain is then analysed for their safety.

For this phishing indicator, I drew upon the work of Althobaiti et al. [33] who designed a "URL nutrition label" to inform what information I could extract from a URL. However, there is more that can be found from a URL in an email than in a URL by itself: In section 2.1.4, it was discussed that a HTML *anchor* tag can be manipulated to show a different URL than the one given in the *href* attribute. This is called "Domain Mismatching". Furthermore, the domain used in the URL may be different to that of the *From* header domain. In my analysis I found this to be common in legitimate emails, but I decided to keep this feature as if the user doesn't expect this mismatch it may be useful to them.

Due to the limitation of time, some heuristics that Althobaiti et al. identified have not been implemented yet, but are set to be implemented in the second stage of this project as discussed in the Future Work chapter. The heuristics I have implemented are:

1. **Domain** - The most important feature in a URL is the domain itself [33]. I used the Python module `tldextract` to find the domain name within a URL.

2. **Domain Age** - In section 2.2.2, it was shown that a 84% of phishing domains are NRDs (domains registered within a day). I took the same approach as Althobaiti et al. and use WhoIs [23] to find the domain age.

3. **Domain Popularity** - the relative traffic of a domain, as discussed in section 2.2.2, is a reliable indicator of a domain's safety. I originally set out to use the Alexa Page Rankings [2], which Althobaiti et al. used, but I discovered this tool is set to be removed in May 2022. Instead I used Tranco [1], which was discussed in section 3.1.2.1. This decision is supported by Pochat et al. [81], who discuss that the Alexa Page Rankings are easily manipulated by malicious actors looking to make their domain appear more popular than it otherwise would. Tranco combines the rankings of Alexa, Cisco Umbrella [63] and Majestic [14] and is designed to resist manipulations. This asserts that Tranco, which was built with security in mind, is a preferable choice over that made by Althobaiti et al. . Further, I use a "whitelist" - a list of always trusted domains - which currently only contains "ed.ac.uk".

4. **DNSBL Results** - As discussed in section 3.2.2, DNSBLs can be used to check whether or not the IP address associated with a domain is considered malicious. The Auto-Responder queries a list of 49 DNSBLs curated by DNSBL.info [8]. If the domain is found in one or more of these blacklists, it will show the URL as blacklisted and therefore dangerous.

5. **Domain Mismatching** - The Auto-Responder will find all anchor tags with a URL in the *href* attribute. Then, it will check each tag's string to check if it is also

a URL, and if so it will extract the domain of both. If they don't match, this mean the domains are mismatched, and thus the indicator is labelled as dangerous.

6. ***From* Header Mismatching** - The Auto-Responder will find all anchor tags with a URL in the *href* attribute. It extracts the domain and compares it with the *From* header domain, if they don't match then this indicator is labelled with a warning.

The Auto-Responder also counts the occurrences of a given domain in the email. However, this is not as simple a task as it would seem. When implementing this feature I noted that domains are often counted more times than expected. Upon investigating, I discovered that in many emails with a URL in a plain-text payload, the URL is again featured in a HTML payload, meaning it is analysed again. This may be confusing to a user as the Auto-Responder will count more occurrences of a domain than the user sees rendered by the email client. The Auto-Responder still analyses and counts links in plain-text - it is possible for emails to be in only plain-text with URLs - so I decided it was more important to analyse a link twice than not analysing it at all. In the future, either the count should be removed from the design or a way to count only the rendered links if the Auto-Responder is deployed for use by an organisation or company.

4.1.7 Summary

Table 4.1 shows the values each indicator can take and what classification each indicator will be displayed as at a given value. These were decided based on my analysis of two large email corpora, shown in section 4.2.

With each phishing indicator analysed, the The Auto-Responder will use each of the individual indicators as a heuristic in calculating its classification of the email. The Auto-Responder classifies emails into three classes: “Safe”, “Suspicious” and “Dangerous” depending on the results of the indicators. The weighting of each indicator into this classification was determined as a result of my analysis in section 4.2.

4.2 Corpora Analysis

I used the Auto-Responder to evaluate two large email corpora that were recommended to me by a phishing expert at the University of Edinburgh [91]. The motivation behind this analysis was to determine the differences in the values of the extracted phishing indicators between phish and non-phish emails. This was to inform how each phishing indicator should be classified, and the weighting they should have on the final classification. All results I discuss in this section are available in Appendix B.

The first corpus contains 2,239 genuine phishing emails, each were reported and verified to be phish [91]. The second contains 4,279 real, non-phish emails obtained from the Enron email dataset [91, 45]. The Auto-Responder was used on each of these emails to obtain the results of its analysis. I then graphed the results, and graphed again after normalising for corpus size.

To inform the weightings of each attribute in the final classification, I fit a logistic regressor to the data I obtained and extracted its weights and bias. Setting aside 1000

Email Indicator	Safe	Warning	Danger
DKIM	Succeeded	Failed	N/A
SPF	Succeeded	Failed	N/A
ARC	Succeeded	Failed	N/A
DMARC	Succeeded	No Policy/Spam	Should be Deleted
Recipient Name	Used	Not Used	N/A
In/out University	Inside	Outside	N/A
Email Domain	Legitimate domain	Free Domain	N/A
Blacklisted	No Blacklisted URL	N/A	Blacklisted URL(s)
Attachment	No Attachments	File(s) Attached	Has an Unsafe File
Emotion	Normal	Unusual	N/A
Misspellings	Acceptable	Unusual	Suspicious
Keywords	Acceptable	Unusual	Suspicious
URL Indicator			
Domain Age	> 1 year	> 3 months	≤ 3 months
Popularity	Whitelisted/On Tranco	N/A	Not Listed on Tranco
Blacklisted	Not Blacklisted	N/A	Blacklisted
href Mismatch	Not Mismatched	N/A	Mismatched
From Mismatch	Not Mismatched	Mismatched	N/A

Table 4.1: Each indicator's classification given the results of the indicator.

emails for testing, the logistic regressor was able to correctly classify over 91% of them before optimising. Further, I inspected the data I obtained in an attempt to notice patterns and differences between the corpora.

However, this inspection revealed that the two corpora have significant drawbacks. The enron corpus has no emails with an attachment, meaning all emails with an attachment are classified as phish which is not the case in reality. Further, the emails in both corpora are old, with some emails being over 20 years old. This limits the usefulness of the data. The logistic regressor only correctly classified 8 of 10 legitimate emails I collected from my personal inbox, and was only able to correctly classify 3 of a small set of 10 phishing emails detected this year that were sent to me by a phishing expert at the University, suggesting either the regressor over-fit to the data, or the data is not in keeping with modern emails. All this being said, the weights are useful for some of the indicators, such as the emotions.

Thus, the weights I chose were set by manual guess-work, informed by my analysis and the regressor. The weights are displayed in table C.1. I make use of the logistic function and separate its classification into three, with suspicious emails being classified above a result of 0.5 and dangerous above 0.7. These were also set manually. This approach is unsophisticated, and if the Auto-Responder were to be used for security, these weights would have to be adjusted and better informed.

Both corpora did not possess the headers necessary for email authentication. Thus, the authentication indicators are currently only used to override the outcome of the weighted indicators if they recommend a safer classification, and flag an email as suspicious if the DMARC policy recommends it be quarantined, or dangerous if the DMARC policy recommends the email be deleted.

Using these criteria, I was able to correctly classify all the small set of 10 phishing emails as either suspicious or dangerous, and all legitimate emails from my inbox. However, this is not a comprehensive measure of performance given the size of the sets.

Ultimately, as classification was not a primary goal for this stage of the project, the weightings of the indicators are not crucial. However, it should be a key goal of the next stage of the project to obtain weights that are better justified by better data. It was useful to conduct this analysis, as it lays a solid framework for the next stage of the project and informed how each individual indicator should be classified.

4.3 Design

Ultimately, the Auto-Responder will display the technical information it extracts in a readable format for a user. The problem of displaying technical concepts to a non-expert user in an understandable way has been the subject of a large amount of research. A successful design would incorporate the findings of this research.

I chose to undertake a “iterative design” process, which, as Nielsen notes, is a process where you repeatedly improve upon the design by incorporating feedback from users [89]. In doing so, I aim to answer the questions “what should the response look like?” and “what advice should be given to the user?”.

It should be noted that the design was a secondary goal of this project. The initial stages of the design process were completed, but delays in receiving ethics approval for a usability study resulted in the design being superseded in priority by the functionality of the system described above. The process of designing how the response from the Auto-Responder is, therefore, incomplete. This section outlines the goals completed so far.

4.3.1 Requirements Gathering

Before beginning the design process, it was important to understand what goals the design had to achieve. I decided to create a list of requirements my design should fulfill. To inform this list, I analysed the technical limitations of the design and investigated the relevant literature on presenting phishing concepts to users.

4.3.1.1 Technical Limitations

As decided in section 4.1.4, the output of the Auto-Responder will be a HTML file. This means that the design must use HTML and its associated technology Cascading Style Sheets (CSS). As these technologies are quite flexible in the tools they give a designer, I found this limitation to not be a large problem, although at times it took some trial and error to display what I wanted.

The response is intended to be sent through email which places some additional limitations upon it. Emails can be received and read on a variety of devices and therefore a variety of screen aspect ratios. As HTML can be used to be reactive to aspect ratio it is easier to address this limitation, but care must be used to ensure that the design remains readable across devices.

4.3.1.2 How to Present Phishing Concepts to a User

Prior to designing the first draft design, I explored the methods that similar anti-phishing tools used in their design process, and gathered advice the anti-phishing literature has regarding design.

Althobaiti et al. used a traffic light sysem in their "URL nutrition label" to highlight the varying degrees of danger of a given indicator [33]. They used red to designate reliable indicators that strongly correlated with a dangerous URL, yellow for when there is a weaker correlation, and green to show that there was no issue with that indicator. Their focus group noted some issues with this, including how different cultures associate colours with different meanings. Ultimately, they kept this colour scheme and included a key describing the meaning of the colours. Further, Kirlappos and Sasse [70] evaluated the effectiveness of a anti-phishing tool called "Solid" which uses this same colour scheme to display the authenticity of websites. They studied 36 participants, asking them to choose a website to buy a ticket from, and found that users who had the colour warnings chose safer options. Therefore, I decided to emulate these findings in my own design by using traffic light colours and a key describing them.

As noted in section 3.2.1.2, pruning and highlighting a URL improved phishing detection rates [114]. Lin et al. [82] also suggested URLs should be displayed in a dedicated area. I decided to incorporate these suggestions in my design by providing an area for each unique URL detected by the Auto-Responder's analysis.

Kirlappos and Sasse remark on what makes a design trustworthy for a user [70]. They note that a previous experience with a design induces a willingness to trust that design in the user. As does using "trust symbols", i.e., a clickable image or logo that directs a user to certification for the website that contains it. As such, I decided that the design should look like a University of Edinburgh email and contain a clickable University logo that directs the user to the University.

4.3.2 Initial Design

Initially, I created a prototype of my design using Desktop Publishing (DTP) software. Doing this allowed me to visualise, test, and get feedback on my design before implementing it. I elected to do this instead of using methods such as paper prototyping as Covid-19 restrictions limited my ability to interact with people in-person. Thus, feedback had to be received through online methods.

Tidwell discusses how use of the "Gestalt Principles" makes a design more effective [107]. These principles improve not only aesthetics, but also the functionality and usability of a design [42]. The principles are: *Proximity* - the placement of related design elements close together, *Similarity* - designing related design elements in the same size, shape or colours, *Continuity* - related design elements placed in a "visual path", and *Closure* - surrounding related design elements with white space or a border. I attempted to conform to these principles where possible.

To fulfill the suggestions by Kirlappos and Sasse [70], the design should look like an email coming from the University of Edinburgh. To that end, I extracted the design scheme (fonts, colours, layout, etc.) of the University of Edinburgh's "Myed Student and Staff Portal" and downloaded the University of Edinburgh logo for use in the design. These are seen regularly by students and staff at the university. Further, I incorporated the suggestion from Marcus [84] who suggested that graphical designs should use between 3-7 colours. This informed the colour palette seen in figure 4.4.

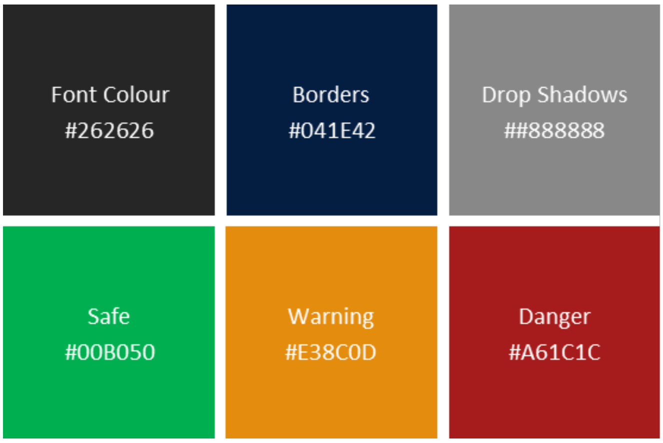
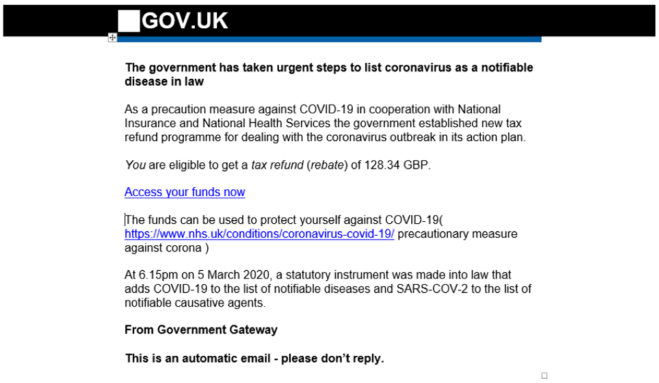


Figure 4.4: The colour palette used in the design process.

To inform the first iteration of my design, I found an example phishing email (shown in figure 4.5) to help me base my design off of a real world example. I manually extracted features in the email that the literature found relevant, such as whether the recipient's name was used. For features that I couldn't extract manually, such as the domain age of the "action link" [115], I filled in what I suspected the features would be.



GOV.UK

The government has taken urgent steps to list coronavirus as a notifiable disease in law

As a precaution measure against COVID-19 in cooperation with National Insurance and National Health Services the government established new tax refund programme for dealing with the coronavirus outbreak in its action plan.

You are eligible to get a *tax refund* (*rebate*) of 128.34 GBP.

Access your funds now

The funds can be used to protect yourself against COVID-19(https://www.nhs.uk/conditions/coronavirus-covid-19/ precautionary measure against corona)

At 6.15pm on 5 March 2020, a statutory instrument was made into law that adds COVID-19 to the list of notifiable diseases and SARS-COV-2 to the list of notifiable causative agents.

From Government Gateway

This is an automatic email - please don't reply.

Figure 4.5: The example phishing email used to inform the design process [26].

I drew inspiration from Zhang [122] who had previously investigated how an Auto-Responder's response would be designed. However, my first prototype design differed greatly from Zhang's; I identified different goals the design needed to fulfill and thus our designs differ. The initial design is shown in figure A.3.

4.3.3 Iterating on the Design

With the initial design complete, I began the iterative design process. I informed this process by following the advice on iterative design given by Nielsen [89]. To obtain the feedback required for this process, I performed a series of informal interviews. I gathered 8 participants from various backgrounds and levels of computer knowledge to help inform the design process.

I created a storyboard, as seen in Appendix A, to guide the participants through a use case of the Auto-Responder. Storyboards are commonly used in usability research as a way to gain early-stage design feedback [20]. They guide the user through the steps it takes for them to see the design, helping to contextualise the design and inform the user of its purpose. As these interviews were informal, the participants names and their responses have not been recorded. Using their feedback, I iterated upon the design several times. Nielsen notes that iterations yield diminishing returns, and that iterations should stop when the designer deems it practical [88]. As such, I stopped iterating when the feedback I received from the participants stopped yielding any actionable advice.

One notable participant had deuteranopia, a type of colour-blindness. They were able to differentiate between the traffic light colours used. I also used the Coblis tool [4] and verified that the colours were distinguishable from each other across the spectrum of colour blindness. However, a more thorough investigation into the colour scheme should be completed through user studies.

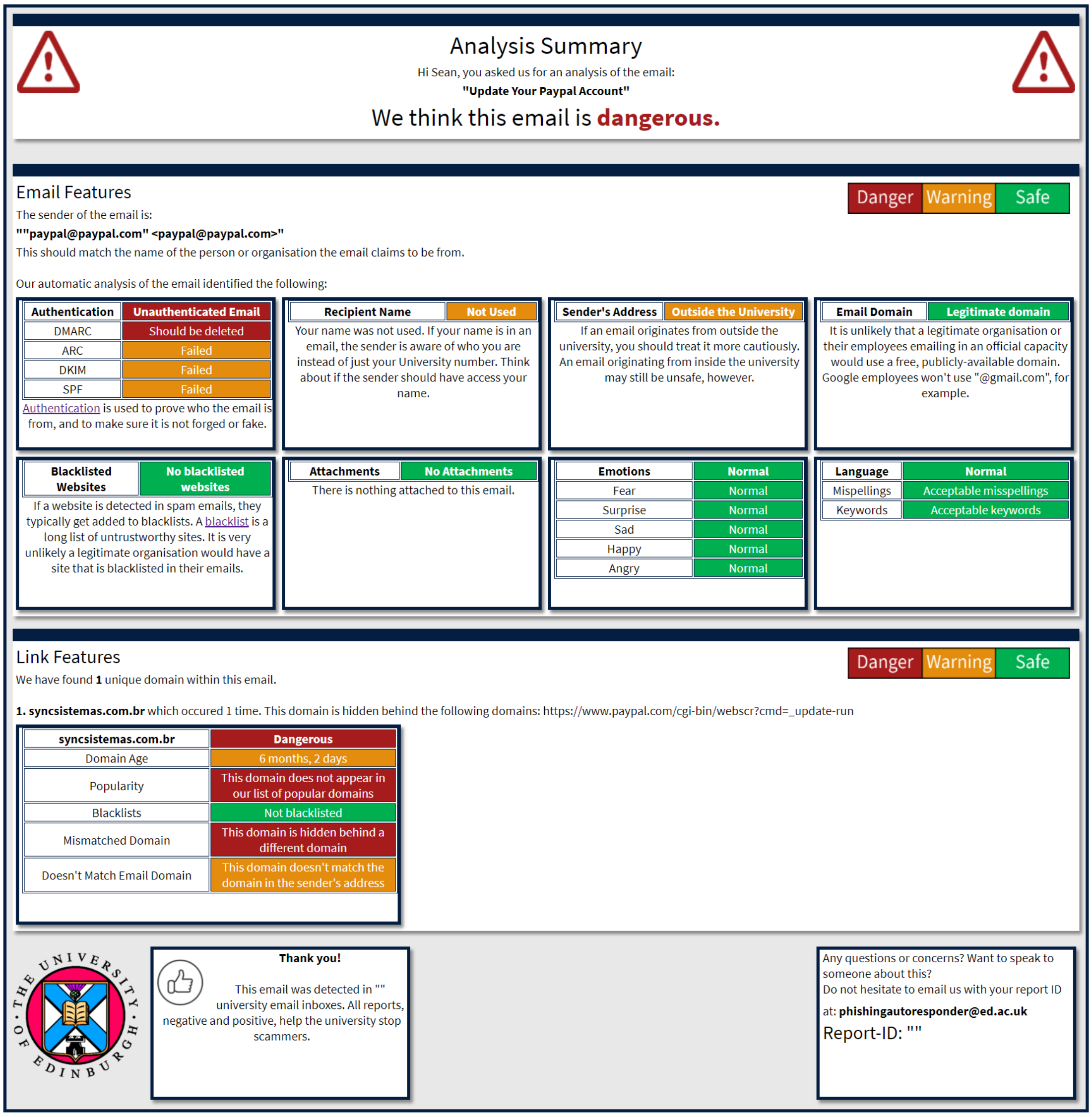
With the design iterations done, I had to move the design from the DTP to the intended format, HTML. This necessitated the design change somewhat to incorporate the change in software.

4.3.4 Current Design

As noted in this section's introduction, the design process has not reached a conclusion. The future steps of the design are described in the Future Work Chapter. The current design is shown in figure 4.6. This design incorporates all the features shown in the Implementation section.

4.3.4.1 Conforming to the Size of the Screen

HTML allows for the design to dynamically reshape depending on the size of the screen. However, this necessitates a level of understanding of how HTML does this. Originally, I naively had the sizes of elements in the HTML styled using CSS to have sizes defined in "em" units. This unit scales depending on the size of the screen. However, upon testing this on mobile and other taller aspect ratios, the response became crowded and information was lost over the sides of the screen. Thus, the CSS required



Analysis Summary

Hi Sean, you asked us for an analysis of the email:

"Update Your Paypal Account"

We think this email is **dangerous.**

Email Features

The sender of the email is:

""paypal@paypal.com" <paypal@paypal.com>"

This should match the name of the person or organisation the email claims to be from.

Our automatic analysis of the email identified the following:

Authentication	Unauthenticated Email
DMARC	Should be deleted
ARC	Failed
DKIM	Failed
SPF	Failed

Authentication is used to prove who the email is from, and to make sure it is not forged or fake.

Recipient Name	Not Used

Your name was not used. If your name is in an email, the sender is aware of who you are instead of just your University number. Think about if the sender should have access your name.

Sender's Address	Outside the University

If an email originates from outside the university, you should treat it more cautiously. An email originating from inside the university may still be unsafe, however.

Email Domain	Legitimate domain

It is unlikely that a legitimate organisation or their employees emailing in an official capacity would use a free, publicly-available domain. Google employees won't use "@gmail.com", for example.

Blacklisted Websites	No blacklisted websites

If a website is detected in spam emails, they typically get added to blacklists. A blacklist is a long list of untrustworthy sites. It is very unlikely a legitimate organisation would have a site that is blacklisted in their emails.

Attachments	No Attachments

There is nothing attached to this email.

Emotions	Normal
Fear	Normal
Surprise	Normal
Sad	Normal
Happy	Normal
Angry	Normal

Language	Normal
Mispellings	Acceptable misspellings
Keywords	Acceptable keywords

Link Features

We have found **1** unique domain within this email.

1. syncsistemas.com.br which occured 1 time. This domain is hidden behind the following domains: https://www.paypal.com/cgi-bin/webscr?cmd=_update-run

syncsistemas.com.br	Dangerous
Domain Age	6 months, 2 days
Popularity	This domain does not appear in our list of popular domains
Blacklists	Not blacklisted
Mismatched Domain	This domain is hidden behind a different domain
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

Thank you!

This email was detected in "" university email inboxes. All reports, negative and positive, help the university stop scammers.

Any questions or concerns? Want to speak to someone about this?
Do not hesitate to email us with your report ID at: **phishingautoresponder@ed.ac.uk**

Report-ID: ""

Figure 4.6: An example of the current design.

refactoring to dynamically alter the design for taller screens. The difference in the design for taller screens and wider screens can be seen in figures D.1 and D.2

Chapter 5

Evaluation

5.1 Software Testing

In order to perform the corpora analysis described in section 4.2, the Auto-Responder was used on over 6,500 emails. It successfully ran to completion on every email without exception. Further, a testing suite was built to achieve maximal branch coverage[1]. It reaches over 97% of statements within the Auto-Responder. A coverage report is supplied in Appendix C. This demonstrates that the system is robust.

During the analysis, I also calculated the average time it takes for the Auto-Responder to complete its process. In section 1.1 it was noted that SOCs usually respond to phishing reports within 17-25 minutes. In comparison, the Auto-Responder took on average 2.46 seconds to analyse the contents of a given email and generate the output HTML file. This demonstrates that the Auto-Responder can do its analysis quickly.

I compared the expected results of the Auto-Responder to its outputs to confirm the system worked as expected. I randomly selected 10 emails and manually extracted the indicators the Auto-Responder does automatically, then compared these values. This analysis and a brief discussing of findings can be found in Appendix C. The Auto-Responder consistently analysed URLs and authenticated emails as expected, and adequately counted occurrences of misspellings and keywords.

I also tested the HTML output of the Auto-Responder to ensure that it worked in an email. I embedded the HTML output into an email, sent it to my university inbox and viewed it on a mobile phone and on a computer. Images of this can be seen in Appendix D.1 and D.5 respectively. This shows the output can be successfully delivered via email.

I then tested the output among a variety of prevalent screen aspect ratios, and confirmed that the design conformed to fit each. These tests can be seen in Appendix D. This demonstrates that the design can be used on a wide array of devices, increasing its potential usability.

[1] It is important to note that - as a necessity to completing a suite to maximise branch coverage - some of these testing emails are dangerous and contain malicious URLs and/or attachments.

5.2 Limitations of the Component

There are a number of limitations that I have noted during my analysis.

Should the Auto-Responder be used to analyse an email it considers dangerous due to the URLs the email contain, the output email would also be considered dangerous as it too contains those (pruned) URLs. This creates a potential “recursive analysis” problem, i.e., a user deems the Auto-Responder’s analysis as untrustworthy as it is a dangerous email. As users are not currently using the system this problem is not a present concern, but it is important to note for the future.

The Auto-Responder is not in a state in which it can be used for security purposes. This is due to a number of reasons, all of which have previously been discussed. To summarise, a number of the Auto-Responders functions have limitations within them that are not conducive to an effective, security-focused software. There should be work done in the second stage to attempt to improve upon these features.

Chapter 6

Future Work

In this paper I describe one half of a two-part project. As such, there remains an amount of work to be done before the project is completed.

6.1 Analysis

The Auto-Responder touches on a range of functions, but only a few are totally complete. It should be an objective of the second stage of the project to bring more of these functions into a security ready state.

- The URL analysis, while already thorough, can be improved upon by incorporating more of the functionality of tools such as Faheem [34] or the “URL Nutrion Label“ [33]. This includes: analysing URLs for similarities to frequently impersonated domains, comparing URLs to the top search result of running each URL through a search engine, etc. .
- Attachments can be processed and evaluated more thoroughly by inspecting the contents of the attachment and evaluating whether it is safe.
- The language processing functions could incorporate a more robust spellchecker that can be used on multiple languages at once, limiting the number of falsely identified misspellings.
- The Auto-Responder could also analyse an email to extract QR codes and treat them to the same analysis as an attachment and/or URL.

In addition, I noted a number of limitations with the inputs and modules the Auto-Responder uses for its analysis, such as the spellchecker. Work should be done to find improvements in the future to find more comprehensive and security focused elements.

6.2 Design

As discussed previously, time limitations meant the design could not be evaluated by a user study. This is an important stage in the design process and should be completed in the next stage of the project. Such a study could focus on a number of criteria, such as: how well users understand the concepts within the design, the effectiveness of any training materials it contains, and whether the design helps users detect phish, etc. .

Moreover, a study to compare the differences in susceptibility to falling for phish between standard auto-responses and the Auto-Responder's response could be done. Such a study would involve testing two groups against simulated phishing attacks, with one group having access to the Auto-Responder. It would then measure the Click Through Rate (CTR) of the participants to measure the effectiveness of the Auto-Responder. Alternatively, a range of both phish and non-phish emails could be shown to the participants, with one group being given the Auto-Responder's analysis of those emails. This would then evaluate each group and how often they correctly classify each email.

Industry standard auto-responses revolve around giving advice to the user on how to act. Currently the Auto-Responder's response contains a limited amount of advice for the user. The design should fully explain each indicator comprehensively yet concisely, and provide high-level advice beyond that of the headline "we think this email is..." the design currently contains. This should be informed by user feedback and research.

Previously, I briefly touched on the accessibility of the design when I discussed colour blindness. The design should undergo a process to evaluate and, if needed, increase its accessibility, and should determine what can be improved or included in this area.

6.3 Reporting

The second stage of development needs to be completed. This stage will involve creating a system that allows a user to report an email and receive the Auto-Responder's response. This system would have to allow for the user to easily send the reported email to an inbox where it can be downloaded into a format the Auto-Responder can use. Then, the system should take the output of the Auto-Responder, attach it to an email or another format before delivering that to the user. As this constitutes the "front-end" of the reporting system proposed, it should be subject to a usability study. The system should prioritise ease of use to encourage users to report emails. Further, the system should undergo testing, including stress testing, to assure its reliability.

Chapter 7

Conclusion

In this paper, I proposed that it is possible to improve upon the industry standard auto-responses to phishing reports through a system that makes these responses dynamic, useful and contextual. I began work on this novel system by creating a component of the system - known as the Auto-Responder - that analyses an email for a number of phishing indicators and generates a readable report of the information found by this analysis.

In my research across the wide range of literature regarding phishing prevention, I identified a number of indicators within an email that can potentially identify phishing and that can be beneficial for a user to know. I developed the Auto-Responder, which could extract these indicators. Using the data collected from the Auto-Responder's analysis of two large email corpora, I justified that these indicators can be used to effectively differentiate between phish and non-phish emails. However, I noted that these corpora had limitations that reduced their efficacy in this justification, and discussed that better data should be used in the future.

The Auto-Responder incorporates the functions of a number of other tools in analysing these indicators, including the analysis of URLs, attachments, language, and authentication, etc. . However, I identified several areas in which this analysis can be improved upon, and established that the current state of the system is not sufficient for use in security. This will be improved upon in the second stage of this project.

I also outlined the initial stages of the design process for the output of this component. Due to delays in ethics approval, a full usability study could not be conducted. Despite this, I drew upon previous research to inform this design, followed basic design principles and processes, and created a design that is functional.

The Auto-Responder was tested on over 6,500 emails; it was proven to be fast, robust, and consistently correct. The design was proven to be able to adapt to a variety of screen aspect ratios and is capable of being delivered through the email framework.

Bibliography

[1] A Research-oriented Top Sites Ranking Hardened Against Manipulation - Tranco. URL: https://tranco-list.eu/. Online; Accessed 12-April-2022.

[2] Alexa - Top sites. URL: https://www.alexa.com/topsites. Online; Accessed: 12-April-2022.

[3] Beautiful Soup. URL: https://www.crummy.com/software/BeautifulSoup. Online; Accessed: 12-April-2022.

[4] Coblis — Color Blindness Simulator. URL: https://www.color-blindness.com/coblis-color-blindness-simulator/. Online; Accessed: 12-April-2022.

[5] Coverage.py. URL: https://coverage.readthedocs.io/en/6.3.2/. Online; Accessed: 12-April-2022.

[6] Dangerous and Malicious File Extensions List. URL: https://www.file-extensions.org/filetype/extension/name/dangerous-malicious-files. Online; Accessed: 12-April-2022.

[7] Django. URL: https://www.djangoproject.com/. Online; Accessed: 12-April-2022.

[8] DNSBL Information - Complete DNSBL List. URL: https://www.dnsbl.info/dnsbl-list.php. Online; Accessed: 12-April-2022.

[9] DNSBL Information - Spam Database and Blacklist Check. URL: https://www.dnsbl.info/. Online; Accessed: 12-April-2022.

[10] Domains blocked from form submissions. URL: https://knowledge.hubspot.com/forms/what-domains-are-blocked-when-using-the-forms-email-domains-to-block-feature. Online; Accessed: 12-April-2022.

[11] Download File Types - Chromium Code Search. URL: https://source.chromium.org/chromium/chromium/src/+/main:components/safe_browsing/core/resources/download_file_types.asciipb;drc=af17ad3f07c1d8a24381eb7669bec0c2ffb86521. Online; Accessed: 12-April-2022.

[12] Google Safe Browsing. URL: https://safebrowsing.google.com. Online; Accessed: 12-April-2022.

[13] Google Translate. URL: https://translate.google.co.uk/. Online; Accessed: 12-April-2022.

[14] Majestic Million. URL: https://majestic.com/reports/majestic-million. Online; Accessed: 12-April-2022.

[15] Microsoft Outlook. URL: https://www.microsoft.com/en-gb/microsoft-365/outlook/email-and-calendar-software-microsoft-outlook. Online; Accessed: 12-April-2022.

[16] Phishing Email Analysis, Identification & Mitigation Automation. URL: https://cofense.com/product-services/cofense-triage/. Online; Accessed: 12-April-2022.

[17] Protection for Browsers. URL: https://www.netcraft.com/apps/browser/. Online; Accessed 12-April-2022.

[18] RFCs. URL: https://www.ietf.org/standards/rfcs/. Online; Accessed: 12-April-2022.

[19] Short URL. URL: https://www.shorturl.at/shortener.php. Online; Accessed 12-April-2022.

[20] Storyboard | Usability Body of Knowledge. URL: https://www.usabilitybok.org/storyboard. Online; Accessed 12-April-2022.

[21] ThreatQuotient | Use Case | Spear Phishing. URL: https://www.threatq.com/spear-phishing/. Online; Accessed 12-April-2022.

[22] Welcome to Flask. URL: https://flask.palletsprojects.com/en/2.1.x/. Online; Accessed: 12-April-2022.

[23] WHOIS Search, Domain Name, Website, and IP Tools - Who.is. URL: https://who.is/. Online; Accessed 12-April-2022.

[24] Domain Names - Concepts and Facilities. Request for Comments RFC 1034, Internet Engineering Task Force, November 1987.

[25] Email (Electronic Mail Format). URL: https://www.loc.gov/preservation/digital/formats/fdd/fdd000388.shtml, April 2014. Online; Accessed: 12-April-2022.

[26] Coronavirus: How hackers are preying on fears of Covid-19. *BBC News*, March 2020.

[27] Outlook REST API v2.0 Production and Beta Endpoint Deprecation. URL: https://devblogs.microsoft.com/microsoft365dev/outlook-rest-api-v2-0-deprecation-notice/, November 2020. Online; Accessed: 12-April-2022.

[28] Overview: Phishing Triage. URL: https://support.trustar.co/article/ndwbmmzqyw-phishing-panel-basics, May 2020. Online; Accessed 12-April-2022.

[29] URL Redirect Checker | WhereGoes. URL: https://wheregoes.com/, December 2020. Online; Accessed 12-April-2022.

[30] What is JavaScript? - Learn Web Development | MDN. URL: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript, March 2022. Online; Accessed: 12-April-2022.

[31] Sara Albakry, Kami Vaniea, and Maria K. Wolters. *What is This URL's Destination? Empirical Evaluation of Users' URL Reading*, page 1–12. Association for Computing Machinery, New York, NY, USA, April 2020.

[32] Zainab Alkhalil, Chaminda Hewage, Liqaa Nawaf, and Imtiaz Khan. Phishing Attacks: A Recent Comprehensive Study and a New Anatomy. *Frontiers in Computer Science*, 3, 2021.

[33] Kholoud Althobaiti, Nicole Meng, and Kami Vaniea. I Don't Need an Expert! Making URL Phishing Features Human Comprehensible. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, May 2021. Association for Computing Machinery.

[34] Kholoud Althobaiti, Kami Vaniea, and Serena Zheng. Faheem: Explaining URLs to people using a Slack bot. In *Symposium on Digital Behaviour Intervention for Cyber Security*, pages 1–8, April 2018. AISB 2018 Symposium on Swarm Intelligence & Evolutionary Computation, AISB 2018.

[35] Steve Atkins. The Anatomy of From. URL: https://wordtothewise.com/2014/04/the-anatomy-of-from/, April 2014. Online; Accessed 12-April-2022.

[36] Joe A. J. Beaumont. Spear Phishing - What It Is And How To Prevent It. URL: https://www.bulletproof.co.uk/blog/what-is-spear-phishing, May 2021. Online; Accessed 12-April-2022.

[37] André Bergholz, Gerhard Paaß, Frank Reichartz, Siehyun Strobel, and Schloß Birlinghoven. Improved Phishing Detection Using Model-based Features. In *Fifth Conference on Email and Anti-Spam, CEAS*, 2008.

[38] Akashdeep Bhardwaj, Varun Sapra, Aman Kumar, Naman Kumar, and S Arthi. Why is Phishing Still Successful? *Computer Fraud & Security*, 2020(9):15–19, September 2020.

[39] Deanna Caputo, Shari Pfleeger, Jesse Freeman, and M.Eric Johnson. Going Spear Phishing: Exploring Embedded Training and Awareness. *Security & Privacy, IEEE*, 12:28–38, January 2014.

[40] Larry Cashdollar. Phishing Attacks Against Facebook / Google via Google Translate. URL: https://www.akamai.com/blog/security/phishing-

attacks-against-facebook-google-via-google-translate, February 2019. Online; Accessed 12-April-2022.

[41] National Cyber Security Centre. How to Spot a Scam Email, Text Message or Call. URL: https://www.ncsc.gov.uk/collection/phishing-scams/spot-scams, November 2021. Online; Accessed 12-April-2022.

[42] Cameron Chapman. Exploring the Gestalt Principles of Design. URL: https://www.toptal.com/designers/ui/gestalt-principles-of-design. Online; Accessed: 12-April-2022.

[43] Neil Chou, Robert Ledesma, Yuka Teraguchi, and John Mitchell. Client-Side Defense Against Web-Based Identity Theft. January 2004.

[44] Cofense. Cofense Annual Report 2021. Technical report, Cofense.

[45] William W. Cohen. Enron Email Dataset. URL: https://www.cs.cmu.edu/~enron/, May 2015. Online; Accessed 12-April-2022.

[46] Office 365 Developers. Outlook Add-ins Overview - Office Add-ins. URL: https://docs.microsoft.com/en-us/office/dev/add-ins/outlook/outlook-add-ins-overview. Online; Accessed 12-April-2022.

[47] Rachna Dhamija, J. D. Tygar, and Marti Hearst. Why Phishing Works. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, page 581–590, New York, NY, USA, April 2006. Association for Computing Machinery.

[48] Agari Cyber Intelligence Division. 2021 Email Fraud and Identity Deception Trends. Technical report, 2021.

[49] DMARC.org. Overview – DMARC.org. URL: https://dmarc.org/2022/01/dmarc-announced-ten-years-ago/. Online; Accessed 12-April-2022.

[50] DMARC.org. DMARC Announced Ten Years Ago – DMARC.org. URL: https://dmarc.org/2022/01/dmarc-announced-ten-years-ago/, January 2012. Online; Accessed 12-April-2022.

[51] Julie S. Downs, Mandy B. Holbrook, and Lorrie Faith Cranor. Decision Strategies and Susceptibility to Phishing. In *Proceedings of the Second Symposium on Usable Privacy and Security*, SOUPS '06, page 79–90, New York, NY, USA, 2006. Association for Computing Machinery.

[52] eBay Inc. Customer Services. Recognising Phishing Phone Calls and Emails. URL: https://www.ebay.co.uk/help/account/protecting-account/recognising-spoof-emails?id=4195. Online; Accessed 12-April-2022.

[53] Robert Epstein. The New Censorship. *US News & World Report*, June 2016.

[54] Joris Evers. Security expert: User education is pointless. *CNET*, October 2006.

[55] UK Government Department for Digital, Culture, Media and Sport. Cyber Security Breaches Survey 2021. Technical report, March 2021.

[56] Ned Freed and Nathaniel S. Borenstein. Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies. Request for Comments RFC 2045, Internet Engineering Task Force, November 1996.

[57] Jeremy Fuchs. Email Security Can Be Time Consuming. We Quantified The Exact Amount. URL: `https://www.avanan.com/blog/email-security-can-be-time-consuming.-we-quantified-just-how-much`, 2021. Online; Accessed 12-April-2022.

[58] The Radicati Group. Email Statistics, 2019-2023. Technical report, The Radicati Group, 2019.

[59] Shuang Hao, Nick Feamster, and Ramakant Pandrangi. Monitoring the Initial DNS Behavior of Malicious Domains. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC '11, page 269–278, New York, NY, USA, November 2011. Association for Computing Machinery.

[60] Brynne Harrison, Elena Svetieva, and Arun Vishwanath. Individual processing of phishing emails. *Online Information Review*, 40:265–281, April 2016.

[61] Cormac Herley. So Long, And No Thanks for the Externalities: The Rational Rejection of Security Advice by Users. In *Proceedings of the 2009 Workshop on New Security Paradigms Workshop*, NSPW '09, page 133–144, New York, NY, USA, January 2009. Association for Computing Machinery.

[62] Joel Hruska. Equifax Sent Customers to a Phishing Site, Hacked Months Earlier. *Extreme Tech*, September 2017.

[63] Dan Hubbard. Cisco Umbrella 1 Million. URL: `https://umbrella.cisco.com/blog/cisco-umbrella-1-million`, December 2016. Online; Accessed 12-April-2022.

[64] IBM. IBM Cost of a Data Breach 2021 - Key Findings and Resources. Technical report, 2021.

[65] Amazon.com Inc. Identifying whether an email, phone call, text message or web page is from Amazon - Amazon Customer Service. URL: `https://www.amazon.co.uk/gp/help/customer/display.html?nodeId=G4YFYCCNUSENA23B`. Online; Accessed 12-April-2022.

[66] Luke Irwin. 5 Ways to Detect a Phishing Email: With Examples. *IT Governance UK Blog*, March 2022.

[67] K. Jansson and Rossouw Solms. Phishing for Phishing Awareness. *Behaviour & Information Technology - Behaviour & IT*, 32:1–10, January 2011.

[68] Jaeyeon Jung and Emil Sit. An Empirical Study of Spam Traffic and The Use of DNS Black Lists. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, IMC '04, pages 370–375, New York, NY, USA, October 2004. Association for Computing Machinery.

[69] Mahmoud Khonji, Youssef Iraqi, and Andrew Jones. Phishing Detection: A Literature Survey. *IEEE Communications Surveys and Tutorials*, 15:2091–2121, 2013.

[70] Iacovos Kirlappos and Angela Sasse. Security Education against Phishing: A Modest Proposal for a Major Rethink. *IEEE Security & Privacy*, 10:24–32, March 2012.

[71] Gary Klein and Robert Hoffman. *Seeing the Invisible: Perceptual-Cognitive Aspects of Expertise*, pages 203–226. February 2020.

[72] Dr. John C. Klensin. Simple Mail Transfer Protocol. RFC 5321, October 2008.

[73] Graham Klyne. Message Headers. URL: `https://www.iana.org/assignments/message-headers/message-headers.xhtml`, February 2022. Online; Accessed 12-April-2022.

[74] Ponnurangam Kumaraguru, Justin Cranshaw, Alessandro Acquisti, Lorrie Cranor, Jason Hong, Mary Blair, and Theodore Pham. School of Phish: A Real-world Evaluation of Anti-phishing Training. *SOUPS 2009 - Proceedings of the 5th Symposium On Usable Privacy and Security*, January 2009.

[75] Ponnurangam Kumaraguru, Yong Rhee, Alessandro Acquisti, Lorrie Cranor, Jason Hong, and Elizabeth Nunge. Protecting People from Phishing: The Design and Evaluation of an Embedded Training Email System. *Conference on Human Factors in Computing Systems - Proceedings*, pages 905–914, April 2007.

[76] Ponnurangam Kumaraguru, Yong Rhee, Steve Sheng, Sharique Hasan, Alessandro Acquisti, Lorrie Cranor, and Jason Hong. Getting Users to Pay Attention to Anti-phishing Education: Evaluation of Retention and Transfer. *Proc. the antiphishing working group's 2nd annual eCrime researchers summit*, 269:70–81, January 2007.

[77] Ponnurangam Kumaraguru, Steve Sheng, Alessandro Acquisti, Lorrie Cranor, and Jason Hong. Teaching Johnny Not to Fall for Phish. *ACM Trans. Internet Techn.*, 10, May 2010.

[78] Neil Kumaran. Spam Does Not Bring us Joy. URL: `https://cloud.google.com/blog/products/g-suite/ridding-gmail-of-100-million-more-spam-messages-with-tensorflow`, February 2021. Online; Accessed 12-April-2022.

[79] Pandove Kunal, Jindal Amandeep, and Kumar Rajinder. Email Spoofing. *International Journal of Computer Applications*, 5, August 2010.

[80] Elmer Lastdrager, Inés Carvajal Gallardo, Pieter Hartel, and Marianne Junger. How Effective is Anti-Phishing Training for Children? In *Proceedings of the Thirteenth USENIX Conference on Usable Privacy and Security*, SOUPS '17, page 229–239, USA, July 2017. USENIX Association.

[81] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczynski, and Wouter Joosen. Tranco: A Research-Oriented Top Sites Ranking

Hardened Against Manipulation. In *Proceedings 2019 Network and Distributed System Security Symposium*, San Diego, CA, 2019. Internet Society.

[82] Eric Lin, Saul Greenberg, Eileah Trotter, David Ma, and John Aycock. Does Domain Highlighting Help People Identify Phishing Sites? In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2075–2084, Vancouver BC Canada, May 2011. ACM.

[83] Eric Lipton, David E. Sanger, and Scott Shane. The Perfect Weapon: How Russian Cyberpower Invaded the U.S. *The New York Times*, December 2016.

[84] Aaron Marcus. Principles of Effective Visual Communication for Graphical User Interface Design. In *Readings in human–computer interaction*, pages 425–441. Elsevier, 1995.

[85] Jens Müller, Marcus Brinkmann, Damian Poddebniak, Sebastian Schinzel, and Jörg Schwenk. Re: What's Up Johnny? Covert Content Attacks on Email End-to-End Encryption. In *Applied Cryptography and Network Security: 17th International Conference, ACNS 2019, Bogota, Colombia, June 5–7, 2019, Proceedings*, page 24–42, Berlin, Heidelberg, April 2019. Springer-Verlag.

[86] Codrut Neagu. What is a QR code? What are QR codes used for? *Digital Citizen*, March 2021.

[87] Jess Nelson. Email Phishing Attacks Estimated To Cost $1.6M Per Incident. URL: `https://www.mediapost.com/publications/article/267680/email-phishing-attacks-estimated-to-cost-16m-per.html`. Online; Accessed 12-April-2022.

[88] Jakob Nielsen. Iterative Design of User Interfaces. URL: `https://www.nngroup.com/articles/iterative-design/`, November 1993. Online; Accessed 12-April-2022.

[89] Jakob Nielsen. Parallel & Iterative Design + Competitive Testing = High Usability. URL: `https://www.nngroup.com/articles/iterative-design/`, January 2011. Online; Accessed 12-April-2022.

[90] Norbert Nthala and Rick Wash. How Non-Experts Try to Detect Phishing Scam Emails. *Workshop on Consumer Protection*, May 2021.

[91] Diego Ocampo. MachineLearningPhishing. URL: `https://github.com/diegoocampoh/MachineLearningPhishing`, March 2022. Online; Accessed 12-April-2022.

[92] Federal Bureau of Investigation. 2020 Internet Crime Report. Technical report, Federal Bureau of Investigation.

[93] Federal Bureau of Investigation. Spoofing and Phishing. URL: `https://www.fbi.gov/scams-and-safety/common-scams-and-crimes/spoofing-and-phishing`. Online; Accessed 12-April-2022.

[94] Outlook. Blocked Attachments in Outlook. URL: `https://support.microsoft.com/en-us/office/blocked-attachments-in-`

outlook-434752e1-02d3-4e90-9124-8b81e49a8519. Online; Accessed 12-April-2022.

[95] Geoffrey Parker. Automating Response to Phish Reporting. *SANS Technology Institute*, May 2019.

[96] Justin Petelka, Yixin Zou, and Florian Schaub. Put Your Warning Where Your Link Is: Improving and Evaluating Email Phishing Warnings. *CHI '19: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–15, April 2019.

[97] Shari Pfleeger, Angela Sasse, and Adrian Furnham. From Weakest Link to Security Hero: Transforming Staff Security Behavior. *Journal of Homeland Security and Emergency Management*, 11, December 2014.

[98] Christopher J. Prom. *Preserving Email.* Number 11-01 in DPC Technology Watch Report. Digital Preservation Coalition, December 2011.

[99] ProofPoint. State Of The Phish – An In-depth Look at User Awareness, Vulnerability and Resilience. Technical report, 2022.

[100] Sara Radicati. Email Statistics Report, 2014-2018. Technical report, 2014.

[101] Pete Resnick. Internet Message Format. Request for Comments RFC 5322, Internet Engineering Task Force, October 2008. Num Pages: 57.

[102] Stefan A. Robila and James W. Ragucci. Don't Be a Phish: Steps in User Education. In *Proceedings of the 11th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, ITICSE '06, page 237–241, New York, NY, USA, January 2006. Association for Computing Machinery.

[103] Maddie Rosenthal. Phishing Statistics (Updated 2022) - 50+ Important Phishing Stats. URL: https://www.tessian.com/blog/phishing-statistics-2020/, January 2022. Online; Accessed 12-April-2022.

[104] Tara Seals. 84% of Phishing Sites Last for Less Than 24 Hours. URL: https://www.infosecurity-magazine.com/news/84-of-phishing-sites-last-for-less/, December 2016. Online; Accessed 12-April-2022.

[105] Rebecca Smith. How a U.S. Utility Got Hacked. *Wall Street Journal*, December 2016.

[106] Microsoft 365 Defender Threat Intelligence Team. HTML smuggling surges: Highly evasive loader technique increasingly used in banking malware, targeted attacks. *Microsoft Security Blog*, November 2021.

[107] Jenifer Tidwell. *Designing Interfaces: Patterns for Effective Interaction Design.* O'Reilly Media Inc., November 2005.

[108] Sumit Tiwari. An Introduction to QR Code Technology. *2016 International Conference on Information Technology (ICIT)*, pages 39–44, December 2016.

[109] Bill Toulas. Silent danger: One in Five Aged Domains is Malicious, Risky, or Unsafe. *BleepingComputer*, December 2021.

[110] Steffen Ullrich. Breaking DKIM - on Purpose and by Chance. Technical report, October 2017.

[111] Guido van Rossum, Barry Warsaw, and Nick Coghlan. PEP 8 – Style Guide for Python Code. Technical report.

[112] Verizon. 2019 Data Breach Investigations Report. Technical report, 2019.

[113] Verizon. 2021 Data Breach Investigations Report. Technical report, Verizon, 2021.

[114] Melanie Volkamer, Karen Renaud, and Paul Gerber. Spot the Phish by Checking the Pruned URL. *Information & Computer Security*, 24(4):372–385, January 2016. Publisher: Emerald Group Publishing Limited.

[115] Rick Wash. How Experts Detect Phishing Scam Emails. *Proceedings of the ACM on Human-Computer Interaction*, 4, October 2020.

[116] Rick Wash, Norbert Nthala, and Emilee Rader. Knowledge and Capabilities that Non-Expert Users Bring to Phishing Detection. In *Seventeenth Symposium on Usable Privacy and Security (SOUPS 2021)*, pages 377–396. USENIX Association, August 2021.

[117] Alma Whitten and J. D. Tygar. Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0. In *Proceedings of the 8th Conference on USENIX Security Symposium - Volume 8*, SSYM'99, page 14, USA, August 1999. USENIX Association.

[118] Wikipedia contributors. Comparison of Email Clients. URL: `https://en.wikipedia.org/w/index.php?title=Comparison_of_email_clients&oldid=1077003658`, March 2022. Page Version ID: 1077003658.

[119] Weining Yang, Aiping Xiong, Jing Chen, Robert W. Proctor, and Ninghui Li. Use of Phishing Training to Improve Security Warning Compliance: Evidence from a Field Experiment, 2017.

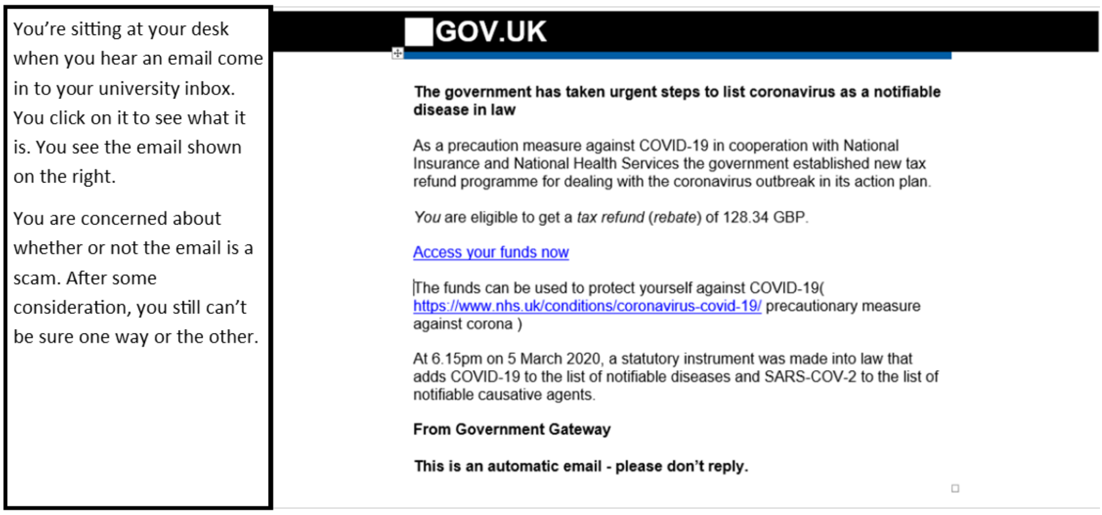
[120] Huiping Yao and Dongwan Shin. Towards Preventing QR Code Based Attacks on Android Phone Using Security Warnings. In *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security*, ASIA CCS '13, page 341–346, New York, NY, USA, 2013. Association for Computing Machinery.

[121] Haiying Yu. Information About the Unsafe File List - Browsers. URL: `https://docs.microsoft.com/en-us/troubleshoot/developer/browsers/security-privacy/information-about-the-unsafe-file-list`. Online; Accessed 12-April-2022.

[122] Zeyu Zhang. Designing an Autoresponder for Phishing Email Reports. 2021.

Appendix A

Storyboard



You're sitting at your desk when you hear an email come in to your university inbox. You click on it to see what it is. You see the email shown on the right.

You are concerned about whether or not the email is a scam. After some consideration, you still can't be sure one way or the other.

GOV.UK

The government has taken urgent steps to list coronavirus as a notifiable disease in law

As a precaution measure against COVID-19 in cooperation with National Insurance and National Health Services the government established new tax refund programme for dealing with the coronavirus outbreak in its action plan.

You are eligible to get a *tax refund* (*rebate*) of 128.34 GBP.

Access your funds now

The funds can be used to protect yourself against COVID-19(https://www.nhs.uk/conditions/coronavirus-covid-19/ precautionary measure against corona)

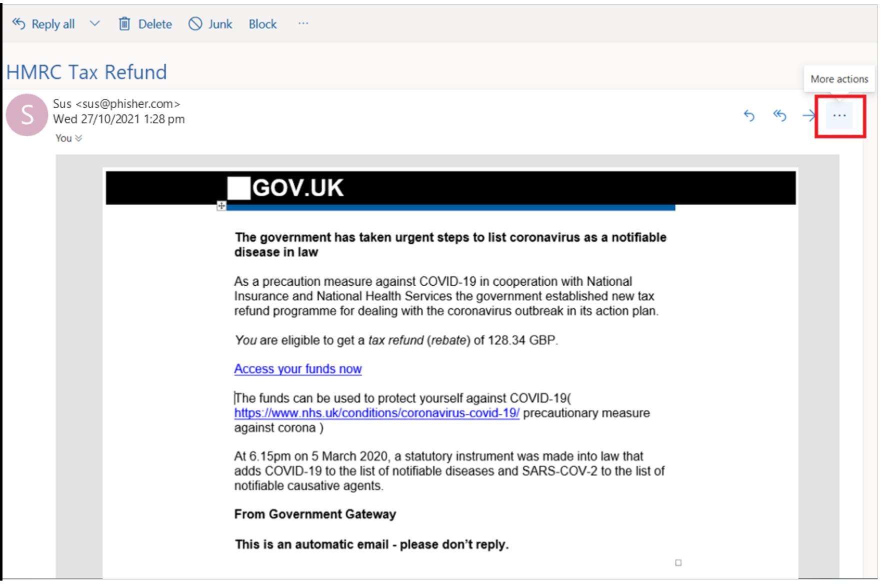
At 6.15pm on 5 March 2020, a statutory instrument was made into law that adds COVID-19 to the list of notifiable diseases and SARS-COV-2 to the list of notifiable causative agents.

From Government Gateway

This is an automatic email - please don't reply.

Figure A.1: Storyboard Part 1

You decide to report the email to the university using the email report add-in. You know the response from the add-in will give you more information about the email.



HMRC Tax Refund

Sus <sus@phisher.com>
Wed 27/10/2021 1:28 pm

GOV.UK

The government has taken urgent steps to list coronavirus as a notifiable disease in law

As a precaution measure against COVID-19 in cooperation with National Insurance and National Health Services the government established new tax refund programme for dealing with the coronavirus outbreak in its action plan.

You are eligible to get a *tax refund* (*rebate*) of 128.34 GBP.

Access your funds now

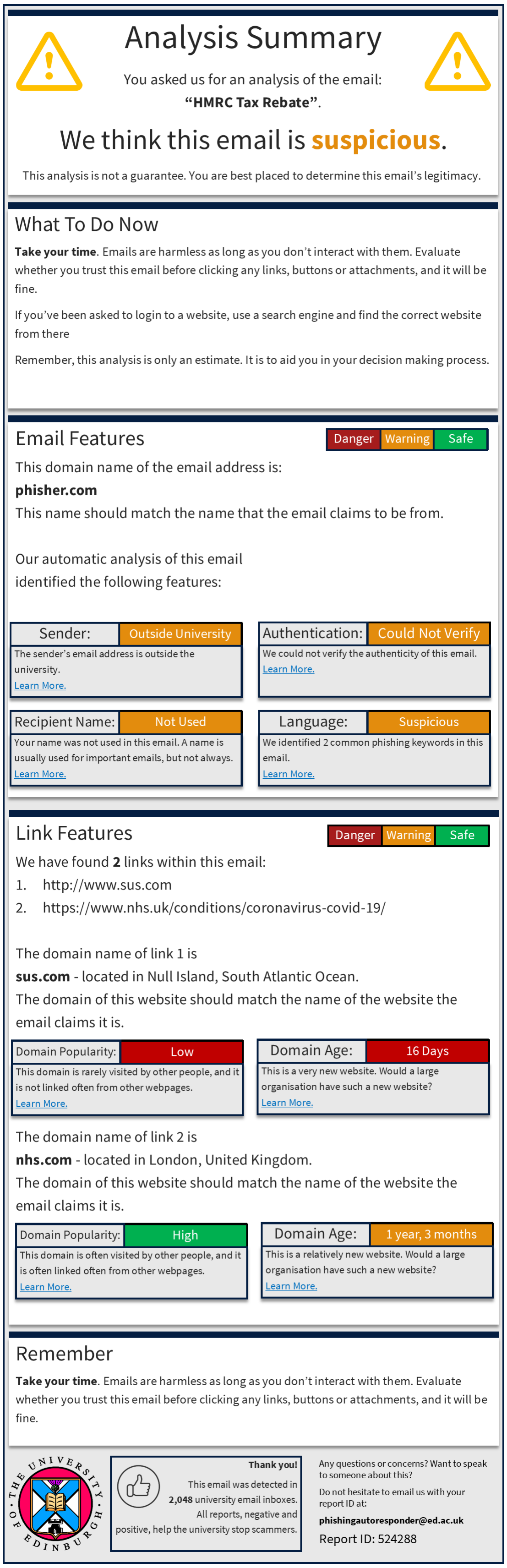
The funds can be used to protect yourself against COVID-19(https://www.nhs.uk/conditions/coronavirus-covid-19/ precautionary measure against corona)

At 6.15pm on 5 March 2020, a statutory instrument was made into law that adds COVID-19 to the list of notifiable diseases and SARS-COV-2 to the list of notifiable causative agents.

From Government Gateway

This is an automatic email - please don't reply.

Figure A.2: Storyboard Part 2



Analysis Summary

You asked us for an analysis of the email:
“HMRC Tax Rebate”.

We think this email is **suspicious**.

This analysis is not a guarantee. You are best placed to determine this email’s legitimacy.

What To Do Now

Take your time. Emails are harmless as long as you don’t interact with them. Evaluate whether you trust this email before clicking any links, buttons or attachments, and it will be fine.

If you’ve been asked to login to a website, use a search engine and find the correct website from there

Remember, this analysis is only an estimate. It is to aid you in your decision making process.

Email Features

Danger | Warning | Safe

This domain name of the email address is:
phisher.com
This name should match the name that the email claims to be from.

Our automatic analysis of this email
identified the following features:

Sender: Outside University
The sender’s email address is outside the university.
Learn More.

Authentication: Could Not Verify
We could not verify the authenticity of this email.
Learn More.

Recipient Name: Not Used
Your name was not used in this email. A name is usually used for important emails, but not always.
Learn More.

Language: Suspicious
We identified 2 common phishing keywords in this email.
Learn More.

Link Features

Danger | Warning | Safe

We have found **2** links within this email:

1. http://www.sus.com
2. https://www.nhs.uk/conditions/coronavirus-covid-19/

The domain name of link 1 is
sus.com - located in Null Island, South Atlantic Ocean.
The domain of this website should match the name of the website the email claims it is.

Domain Popularity: Low
This domain is rarely visited by other people, and it is not linked often from other webpages.
Learn More.

Domain Age: 16 Days
This is a very new website. Would a large organisation have such a new website?
Learn More.

The domain name of link 2 is
nhs.com - located in London, United Kingdom.
The domain of this website should match the name of the website the email claims it is.

Domain Popularity: High
This domain is often visited by other people, and it is often linked often from other webpages.
Learn More.

Domain Age: 1 year, 3 months
This is a relatively new website. Would a large organisation have such a new website?
Learn More.

Remember

Take your time. Emails are harmless as long as you don’t interact with them. Evaluate whether you trust this email before clicking any links, buttons or attachments, and it will be fine.

THE UNIVERSITY OF EDINBURGH

Thank you!
This email was detected in **2,048** university email inboxes. All reports, negative and positive, help the university stop scammers.

Any questions or concerns? Want to speak to someone about this?

Do not hesitate to email us with your report ID at:
phishingautoresponder@ed.ac.uk

Report ID: 524288

Figure A.3: Storyboard Part 3 and the first draft design of the Auto-Responder’s response

Appendix B

Corpora Analysis

B.1 Hypotheses

- Phishing emails will have a lower value of “Happy”, but greater values of “Angry”, “Surprise”, ”Sad” and ”Fear”.
- Phishing keywords will be more common in the phishing corpus.
- Misspellings will be more prevalent in the phishing corpus.
- The phishing corpus will contain more emails from a freely available email domain.
- The phishing corpus will contain significantly more blacklisted domains.
- There will be more mismatched URLs in the phishing corpus.
- There will be more unpopular domains in the the phishing corpus.
- Domains in the phishing corpus will be generally younger and will contain more URLs below 1 month.
- The enron corpus will have more attachments.

B.2 Key Findings

- Phishing emails tend to be slightly less emotional. The exception is “Fear”, which was significantly greater in phishing emails.
- Phishing keywords are much more prevalent in the phishing corpus than the enron. It is a powerful indicator, with the median value of keywords in phish being 6.5x greater than that of the enron.
- The median value of misspellings in a phishing email is less than that of the enron corpus. However, the phishing emails had more extreme values far from the median. This suggests that a high value of misspellings is more indicative of

Phishing Indicator	Weight	Potential Values
Bias	-0.2	1
Blacklisted Domain	3	0 or 1
Unpopular Domain	0.7	0 or 1
Young Domain	0.5	0 or 1
URL Age	-0.0001	0 or more
Mismatched URL	3	0 or 1
Within Organisation	0.2	0 or 1
Freely Available *From* Domain	0.2	0 or 1
Happy	-0.3	0 to 1
Angry	-0.1	0 to 1
Surprise	-0.15	0 to 1
Sad	-0.2	0 to 1
Fear	0.4	0 to 1
Misspellings	0.1	0 or more
Phishing Keywords	0.3	0 or more
Attachment	3	0, 0.1 or 1

Table B.1: Manually set weights used by the Auto-Responder's logistic function.

phish, but it becomes more difficult to differentiate between the two classes for lower values of this indicator.

- Significantly more of the enron corpus came from a freely available domain in comparison to the phishing corpus. This is surprising, and I propose that it should be validated by different corpora.
- Some of the enron corpus contains blacklisted domains. I was surprised by this, but I independently verified some of these emails and confirmed that they did in fact contain blacklisted domains. Despite this, this indicator was roughly double as prevalent in the phishing corpus.
- The URLs in the corpora are generally quite old, which is expected given the ages of the corpora. The Phishing corpus did contain younger URLs.
- The enron corpus had no attachments. This suggests they were removed, possibly for privacy reasons. This limits the usefulness of the results of this indicator.

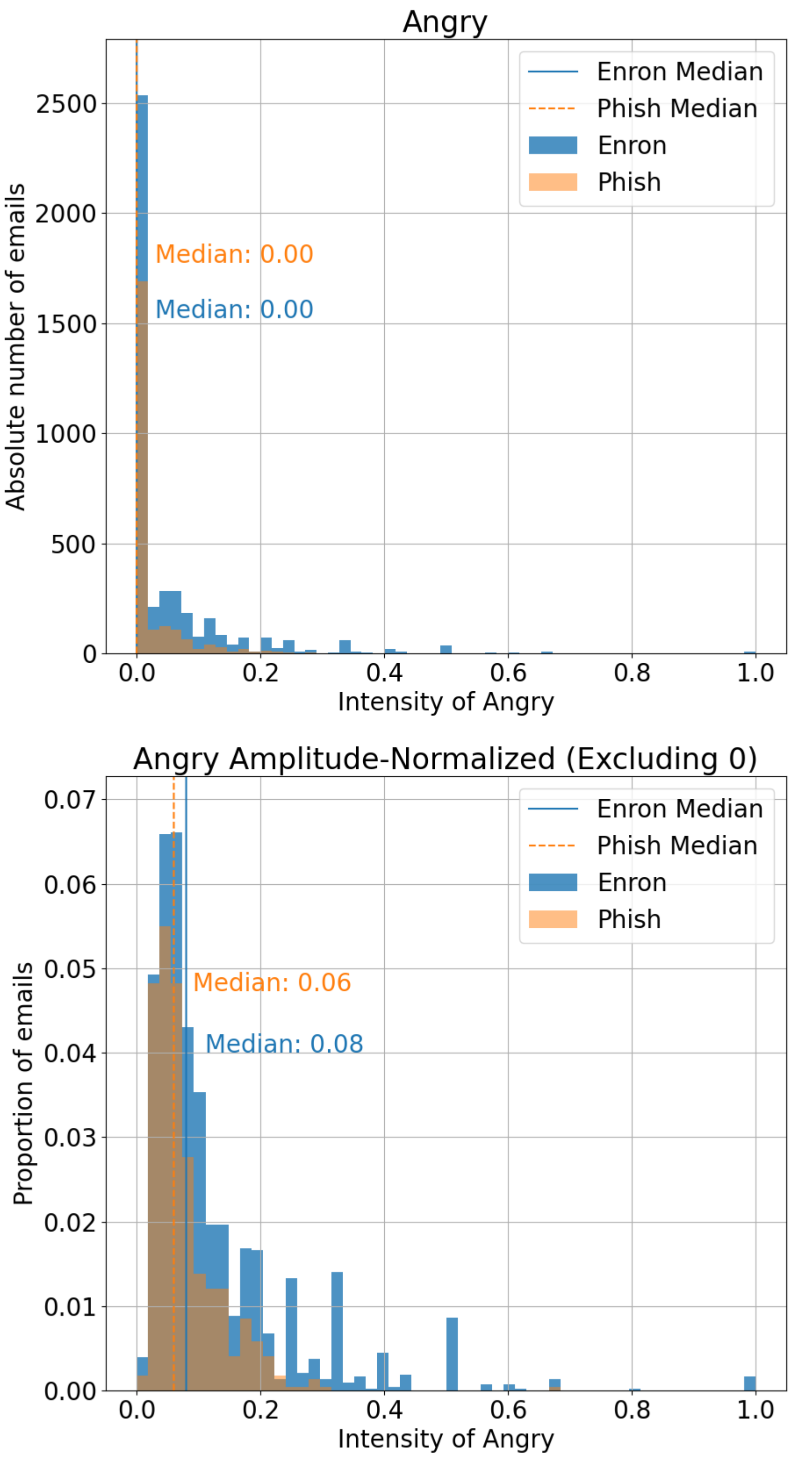


Figure B.1: Enron and Phishing corpora - Angry

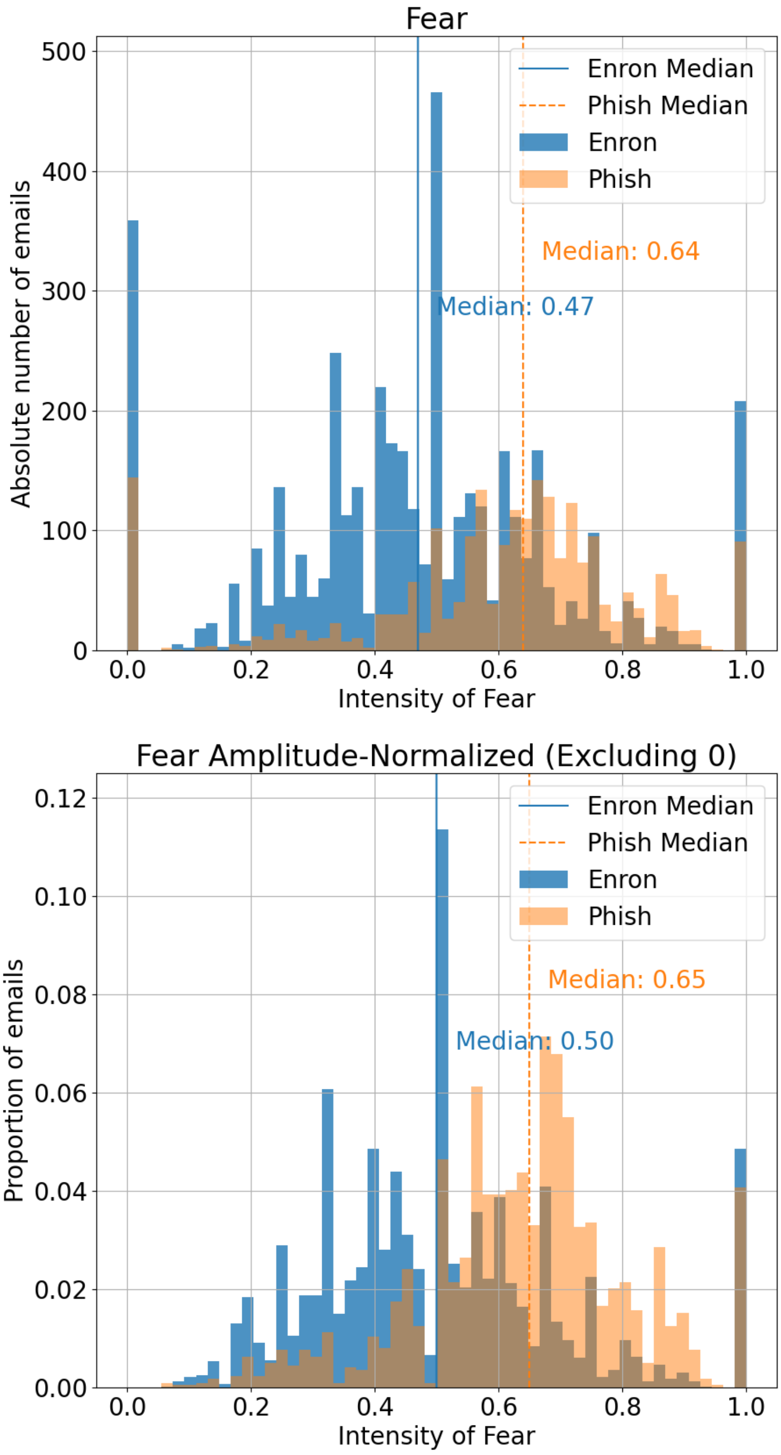


Figure B.2: Enron and Phishing corpora - Fear

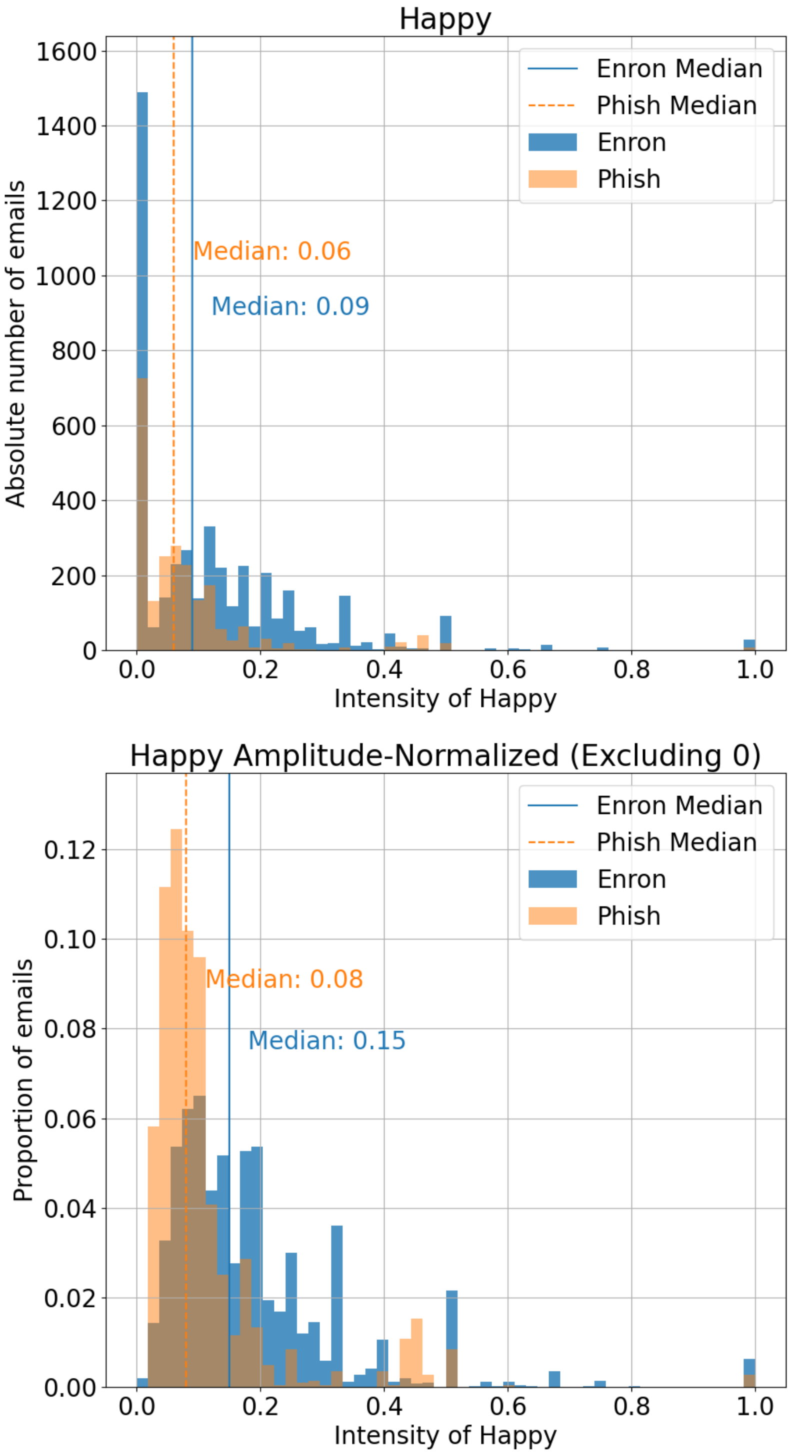


Figure B.3: Enron and Phishing corpora - Happy

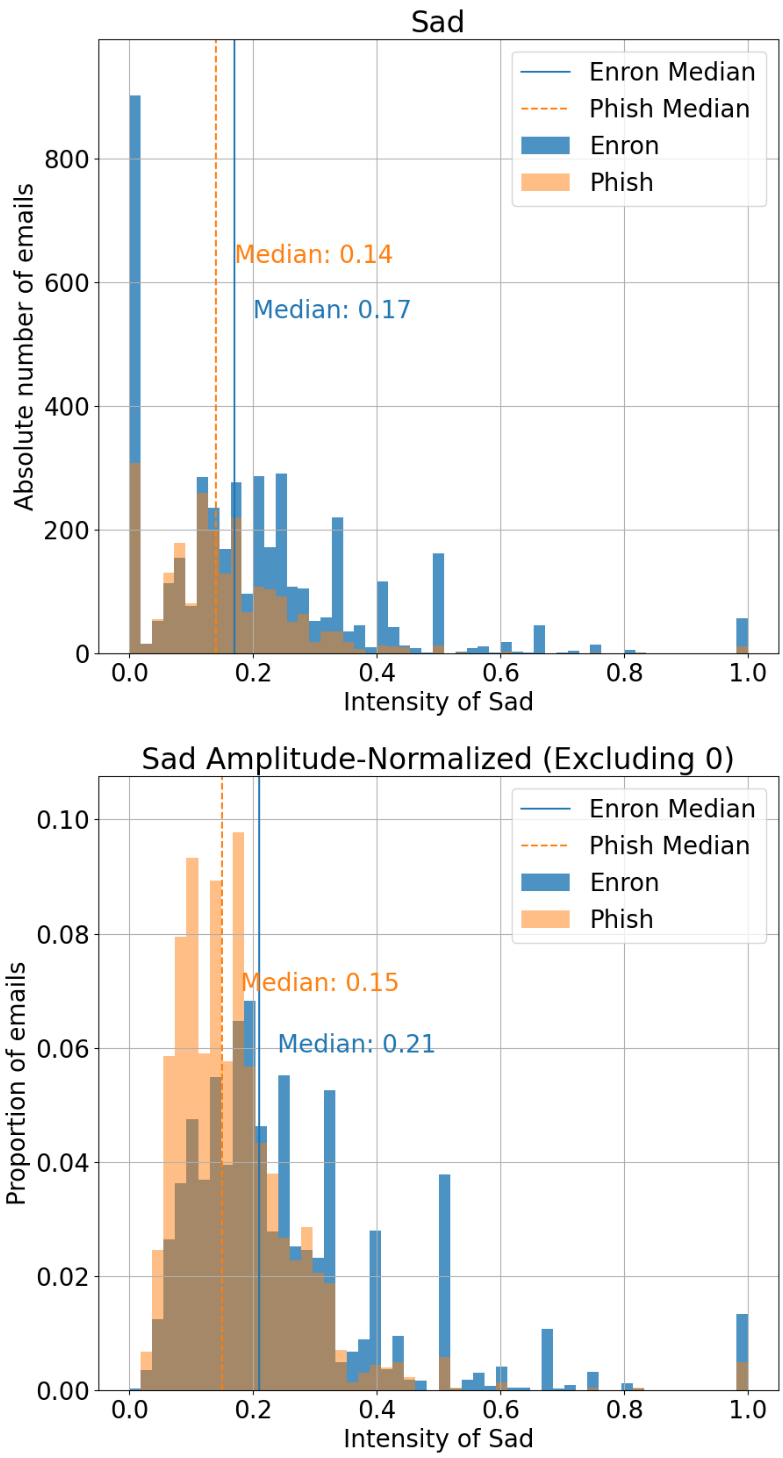


Figure B.4: Enron and Phishing corpora - Sad

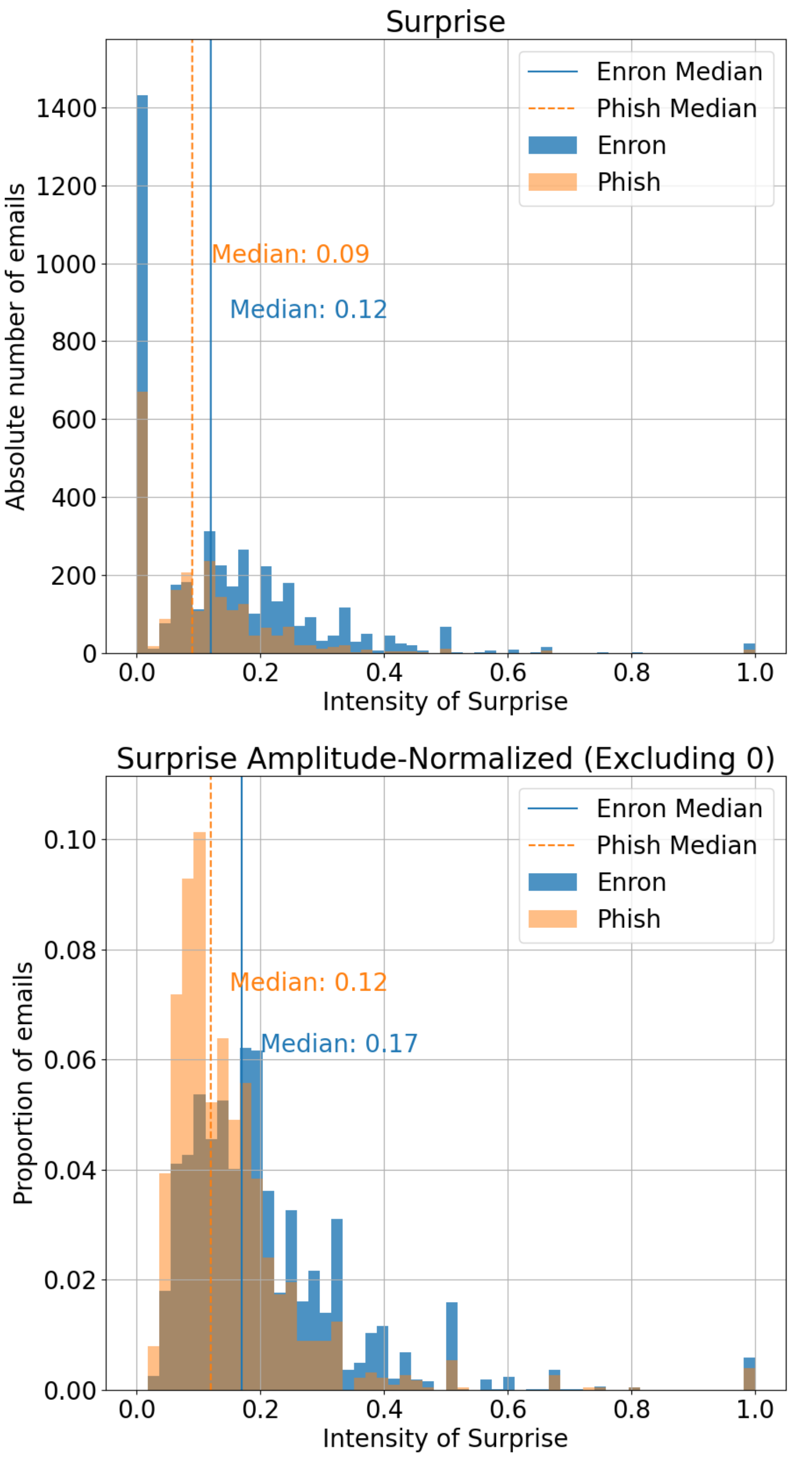


Figure B.5: Enron and Phishing corpora - Surprise

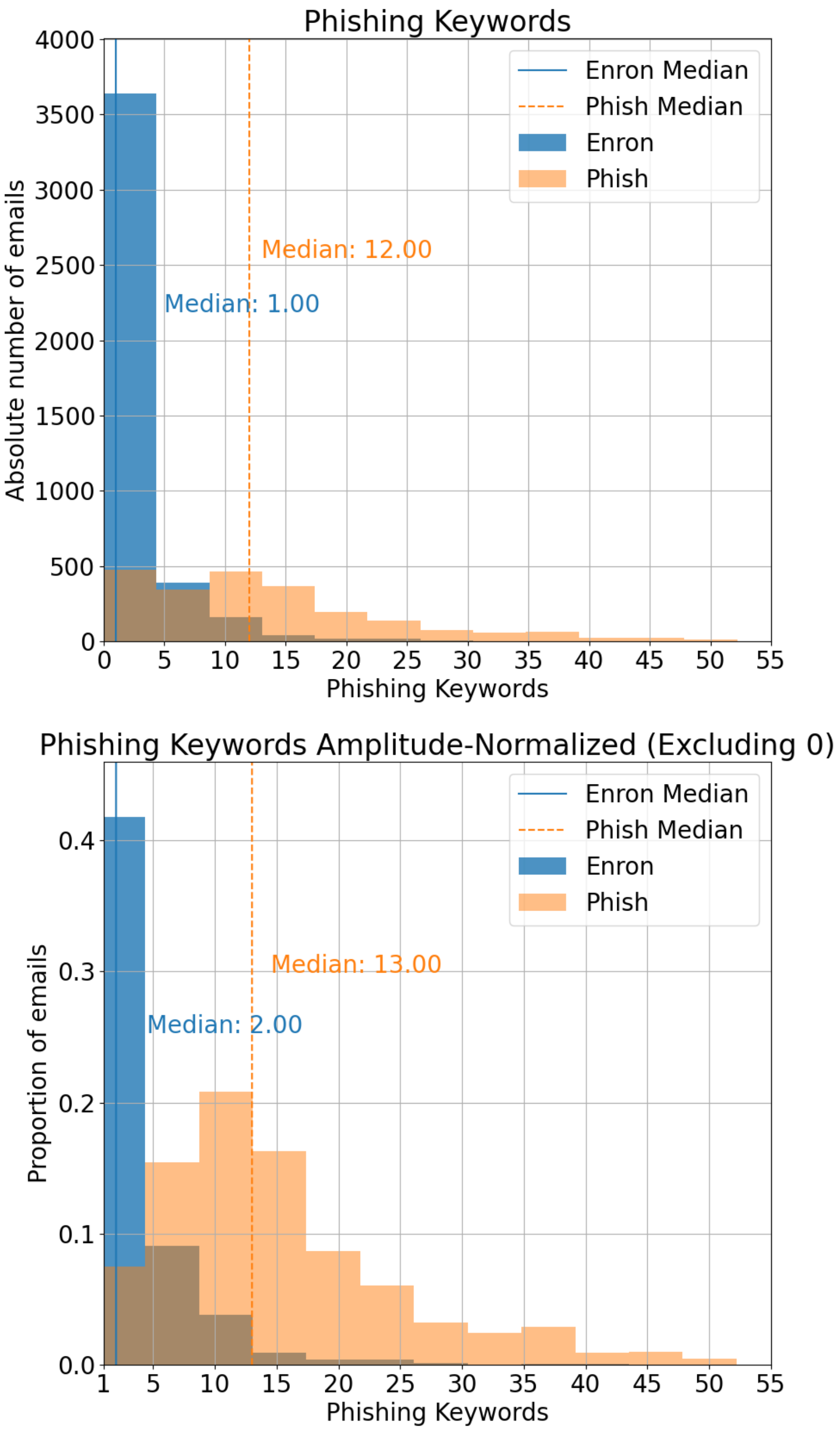


Figure B.6: Enron and Phishing corpora - Sad

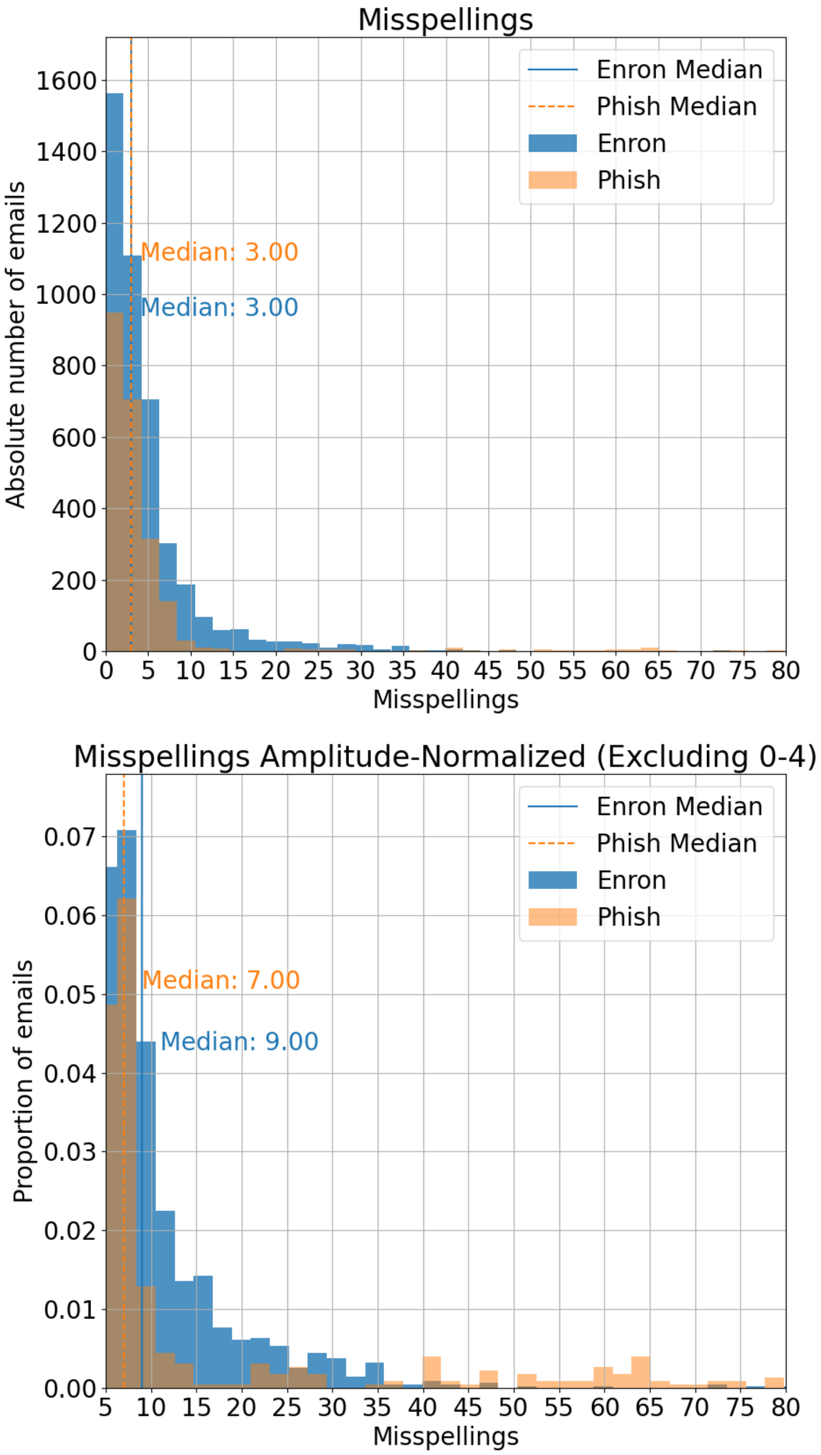
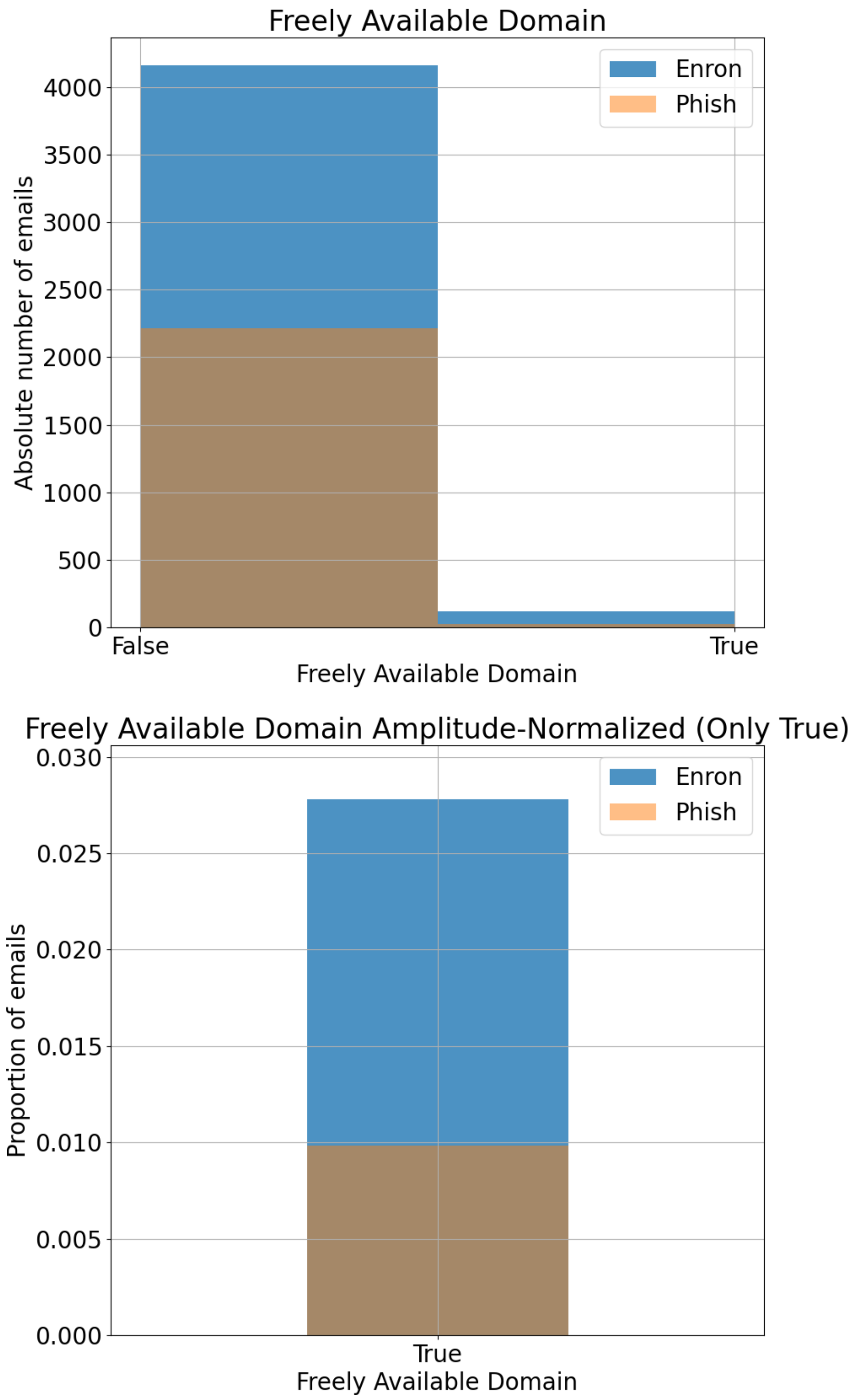


Figure B.7: Number of misspelled words in email.



Freely Available Domain

Freely Available Domain Amplitude-Normalized (Only True)

Figure B.8: Whether email is from freely registered domain.

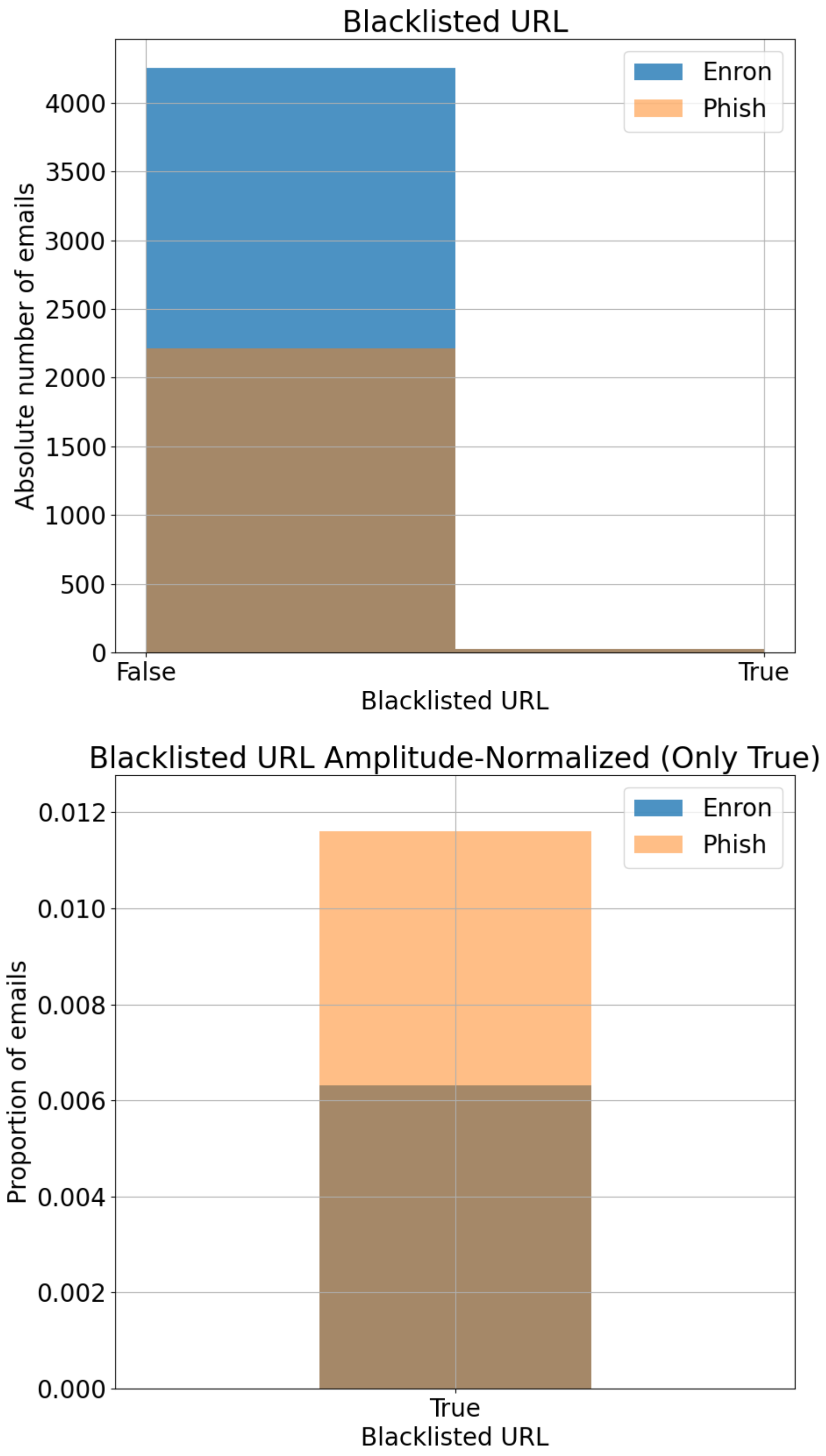


Figure B.9: Whether email contains a blacklisted domain.

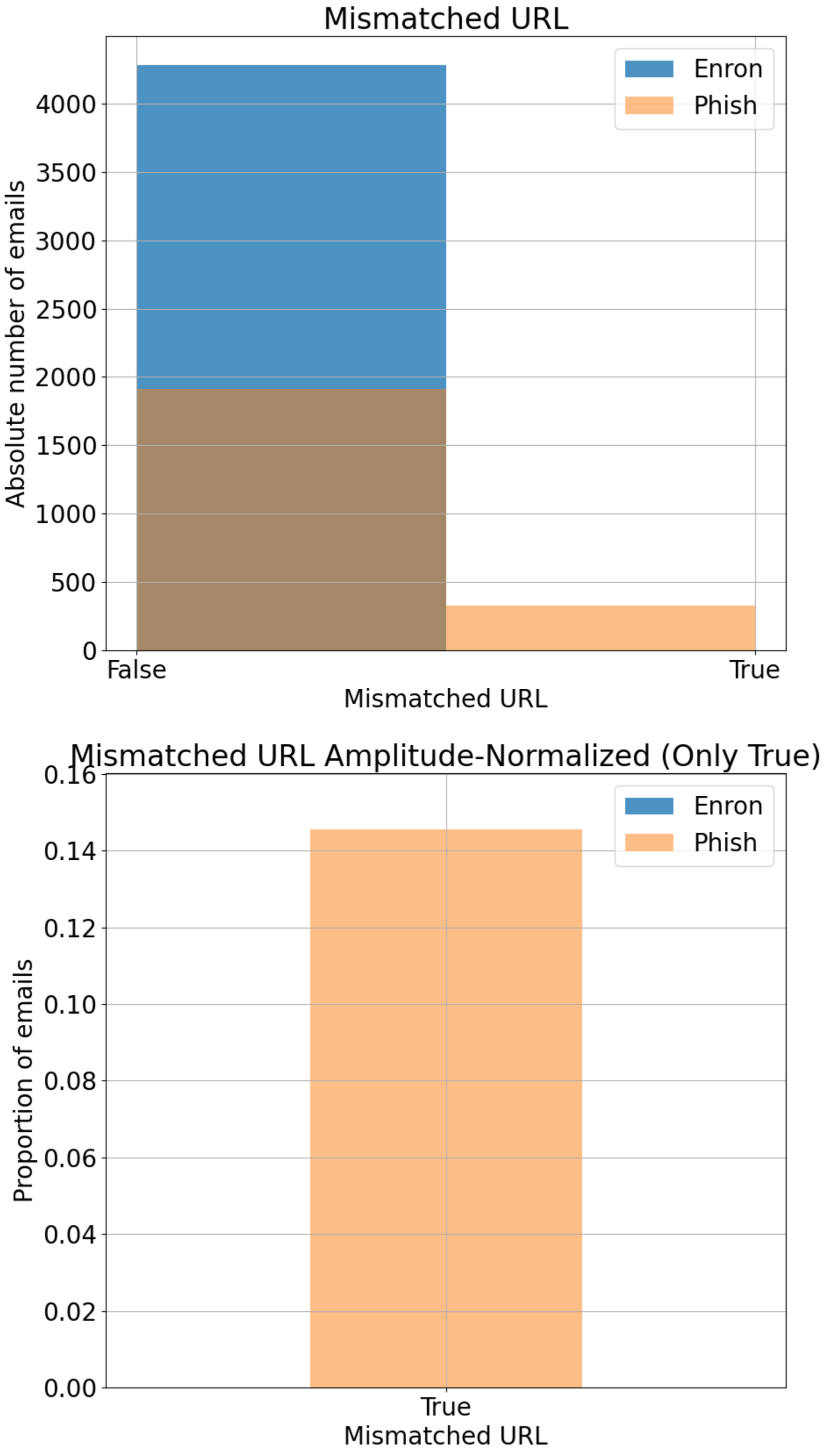


Figure B.10: Whether email contains a URL with a domain hidden behind a different domain

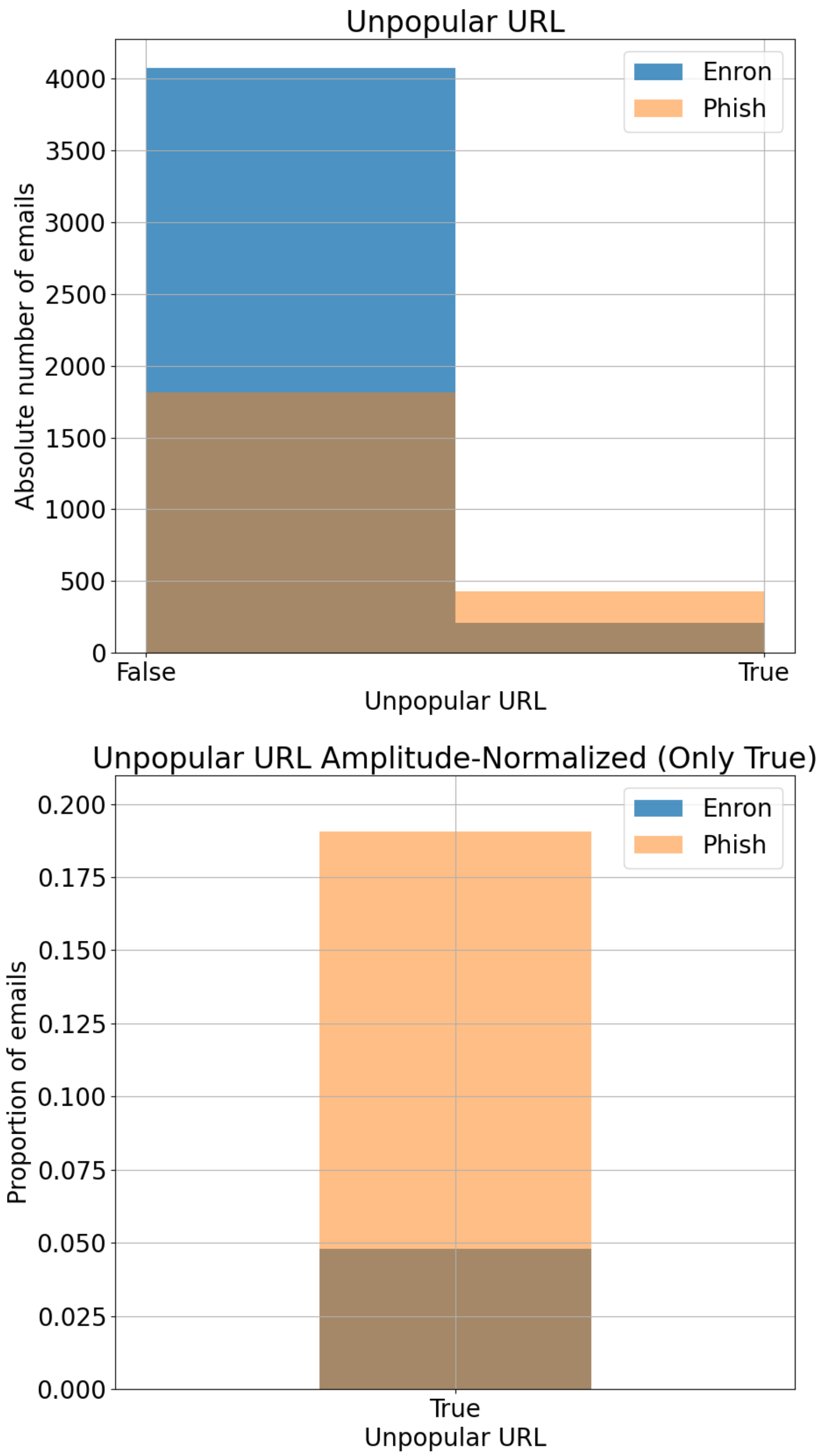


Figure B.11: Whether URL is on the Tranco list [1].

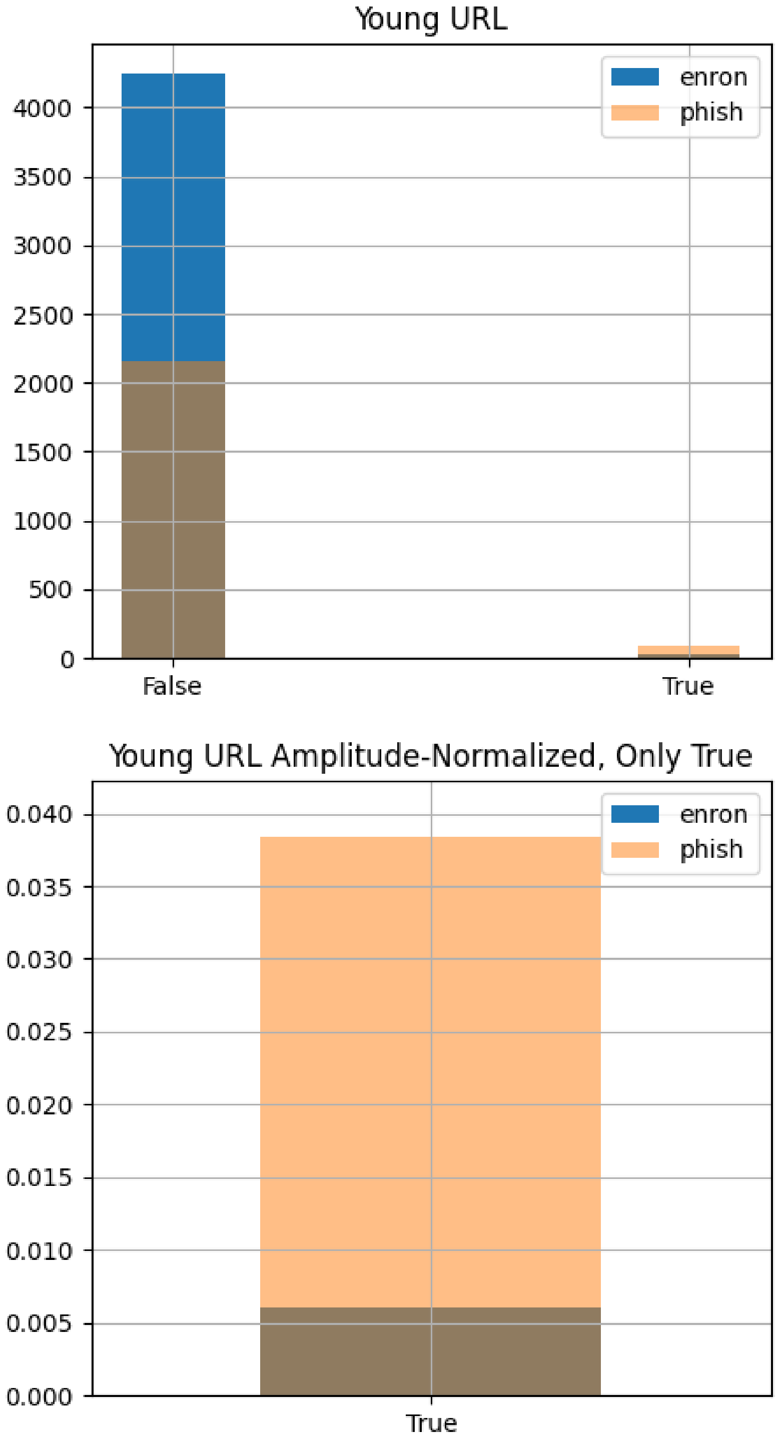


Figure B.12: URLs below 1 month old.

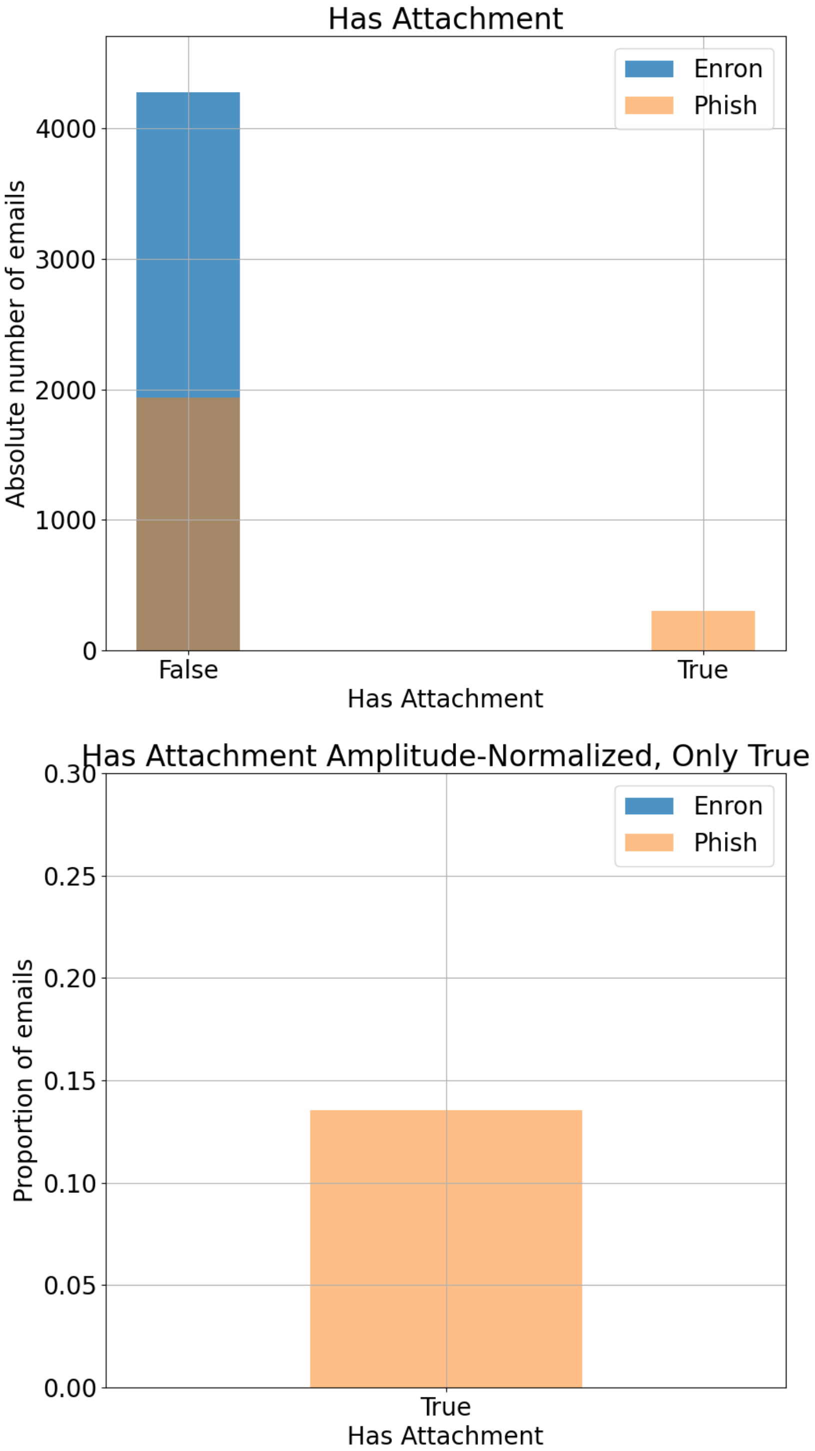


Figure B.13: Whether email has an attached file.

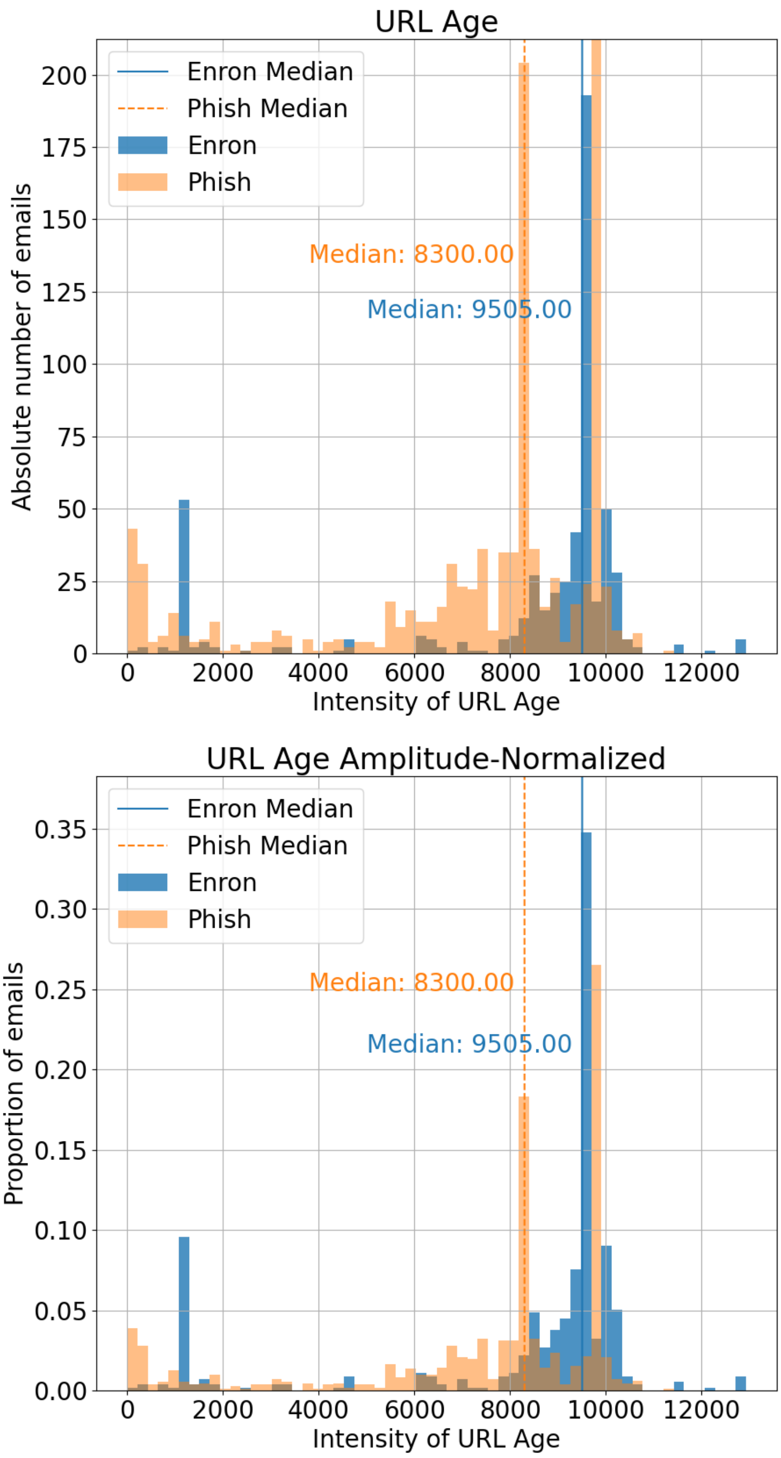


Figure B.14: URL days since registration.

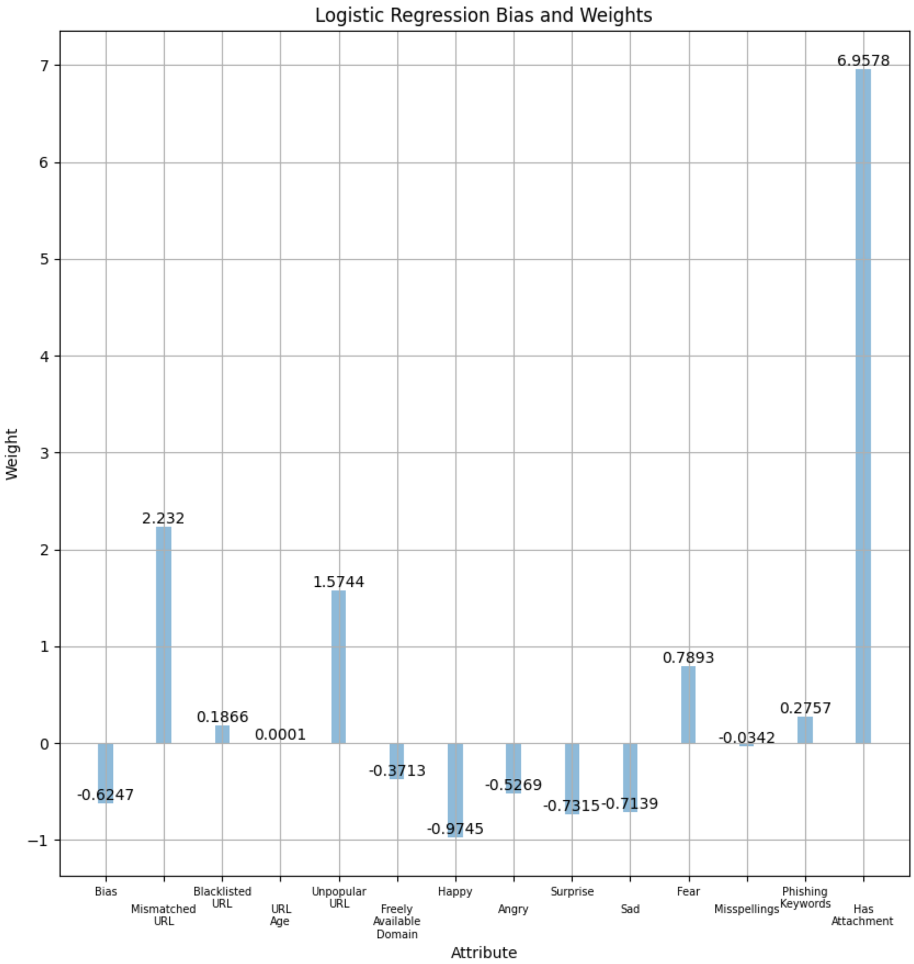


Figure B.15: Logistic Regression Bias and Weights.

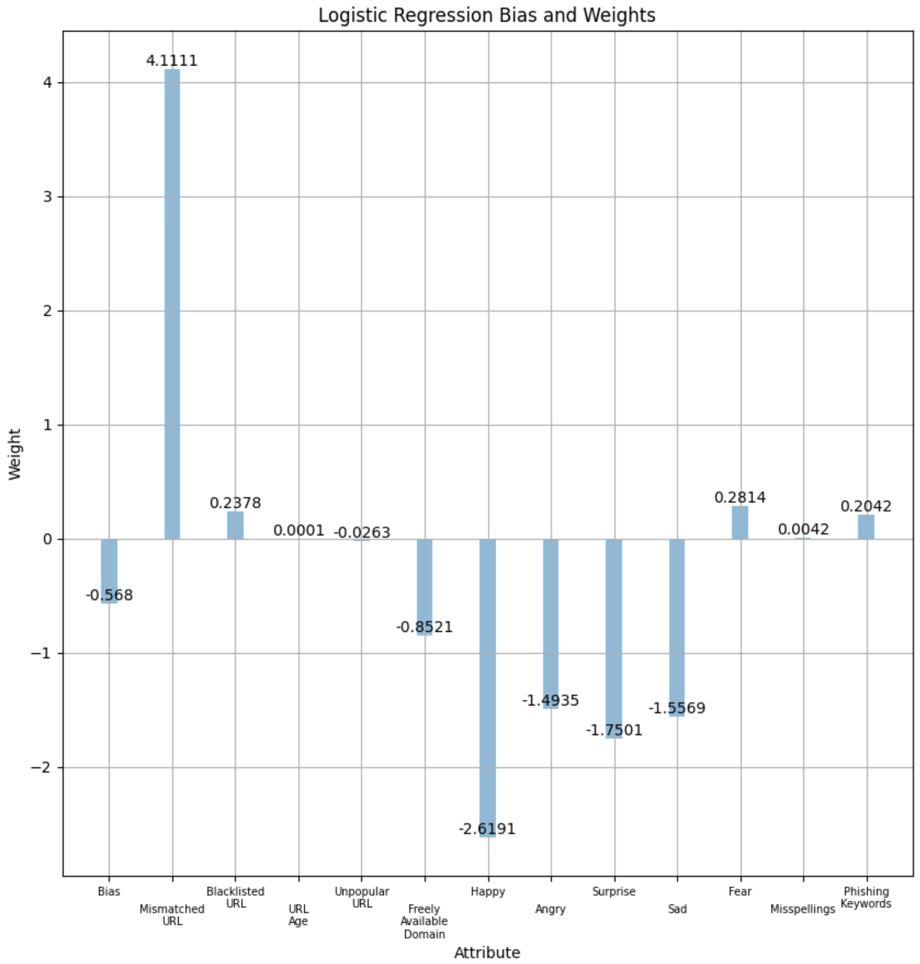


Figure B.16: Logistic Regression Bias and Weights without Attachments.

Appendix C

System Testing

C.1 Expected Results vs Output Results

The following methods were used to manually evaluate the expected results (Table C.1) of an email:

- URLs were evaluated by using WhoIs [23].
- Free Email Domain was evaluated using the list of free domains found here [10].
- Misspellings and Keywords were found by importing the text of an email into a Word Processing Software (WPS) tool and using the tool's in-built word finder and spell checker.

An element of human error should be expected in the manually calculated results.

Key findings from comparing expected results with the Auto-Responder's results (Table C.2):

- URLs are consistently correct.
- Misspellings are usually wrong by some amount. I suspect this is due to differences between the dictionaries used by the two spellcheckers. For example, Auto-Responder's spellchecker considers the word 'c' (such as in a copyright notice) to be misspelled, but the WPS did not. Conversely, the Auto-Responder's spellchecker counts names as correct, but the WPS does not.

Emails in the table are named by their numbering in the file system submitted alongside this project.

C.2 Branch Coverage

The tests were run and coverage report (figure C.1) created using the `coverage` [5] tool. The missed statements include the systems "main" function which is impossible to reach as I used an external file to run the Auto-Responder. The testing suite was

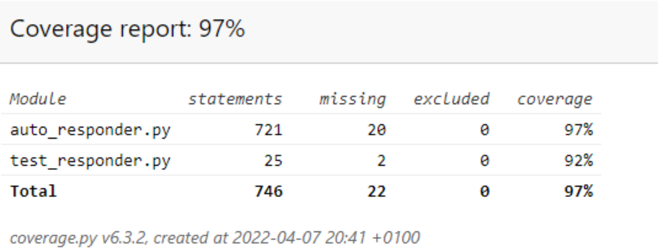
Email	Mismatch	Blacklist	URL Age	Unpopular URL	Free Email Domain	#Misspellings	#Keywords	Attachment
Phish 2	No	No	11394	No	No	4	8	Yes
Phish 618	No	No	Age Unknown	Yes	No	0	0	Yes
Enron 2515	N/A	N/A	N/A	N/A	No	10	0	No
Enron 578	N/A	N/A	N/A	N/A	No	5	0	No
Enron 2586	N/A	N/A	N/A	N/A	No	0	0	No
Phish 1728	No	No	9741	Yes	No	8	16	Yes
Phish 1862	Yes	No	5992	Yes	No	2	21	No
Phish 13	Yes	Yes	6850	Yes	No	7	22	No
Enron 2907	No	No	10019	No	No	0	1	No
Enron 4278	N/A	N/A	N/A	N/A	NO	0	0	No

Table C.1: Expected results of the Auto-Responder.

Email	Mismatch	Blacklist	URL Age	Unpopular URL	Free Email Domain	#Misspellings	#Keywords	Attachment
Phish 2	No	No	11394	No	No	4	8	Yes
Phish 618	No	No	Age Unknown	Yes	No	0	0	Yes
Enron 2515	N/A	N/A	N/A	N/A	No	5	0	No
Enron 578	N/A	N/A	N/A	N/A	No	9	0	No
Enron 2586	N/A	N/A	N/A	N/A	No	10	0	No
Phish 1728	No	No	9741	Yes	No	8	16	Yes
Phish 1862	Yes	No	5992	Yes	No	5	21	No
Phish 13	Yes	Yes	6850	Yes	No	7	22	No
Enron 2907	No	No	10019	No	No	0	1	No
Enron 4278	N/A	N/A	N/A	N/A	NO	0	0	No

Table C.2: Actual results of the Auto-Responder.

minimised to include only emails that increase branches covered. It currently contains 23 emails and 25 tests.



Coverage report: 97%

Module	statements	missing	excluded	coverage
auto_responder.py	721	20	0	97%
test_responder.py	25	2	0	92%
Total	**746**	**22**	**0**	**97%**

coverage.py v6.3.2, created at 2022-04-07 20:41 +0100

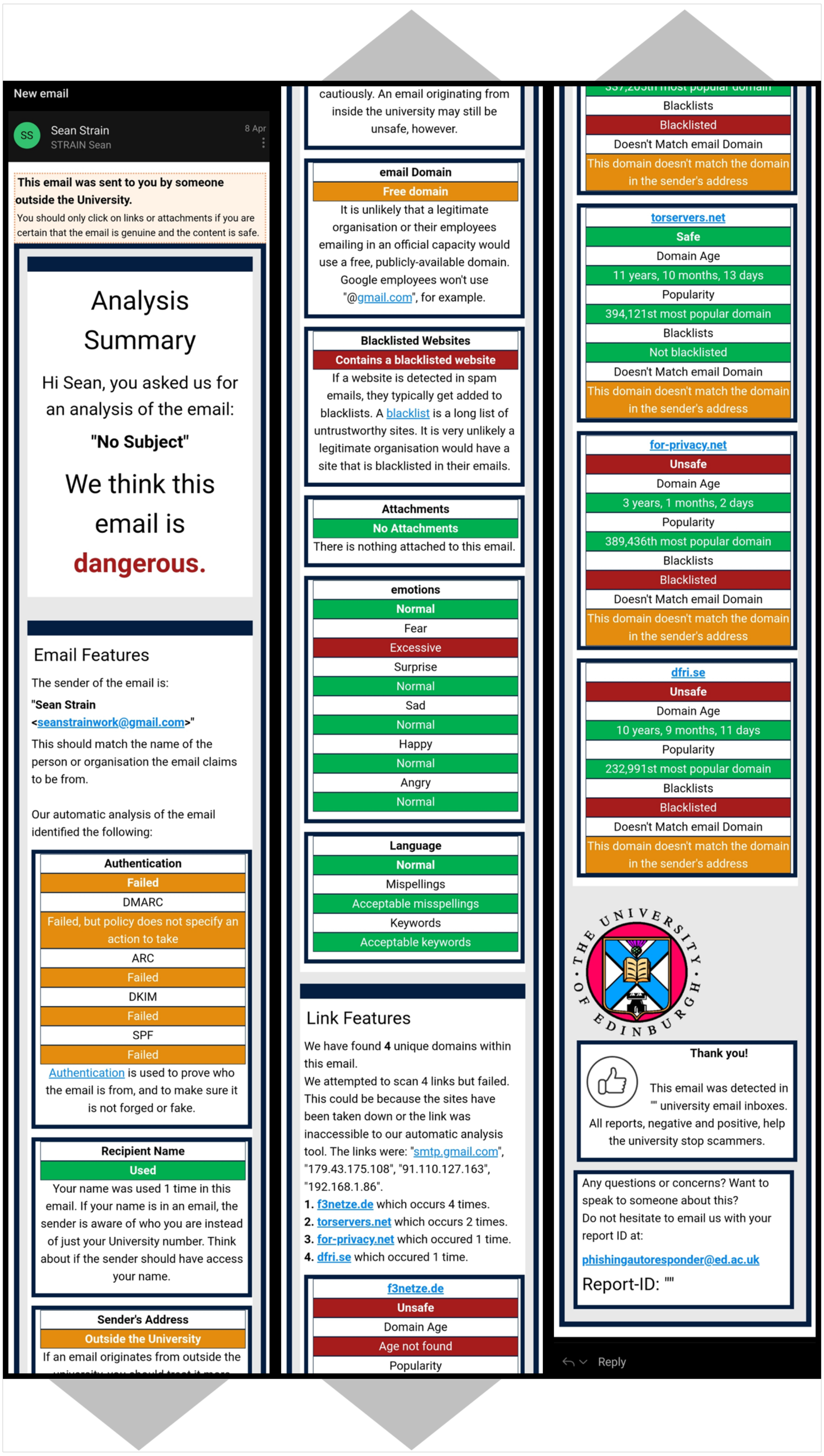
Figure C.1: Coverage Report

Appendix D

Output Testing

The Auto-Responder's output was tested among various popular aspect ratios. This demonstrates that the design can adapt to users' devices.

Further, it was tested on the email client Outlook [15] to demonstrate that the design can be embedded into email successfully. This can be seen in figure D.1 and D.5.



New email

SS Sean Strain
STRAIN Sean
8 Apr

This email was sent to you by someone outside the University.
You should only click on links or attachments if you are certain that the email is genuine and the content is safe.

Analysis Summary

Hi Sean, you asked us for an analysis of the email:

"No Subject"

We think this email is **dangerous.**

Email Features

The sender of the email is:

"Sean Strain <seanstrainwork@gmail.com>"

This should match the name of the person or organisation the email claims to be from.

Our automatic analysis of the email identified the following:

Authentication
Failed
DMARC
Failed, but policy does not specify an action to take
ARC
Failed
DKIM
Failed
SPF
Failed
Authentication is used to prove who the email is from, and to make sure it is not forged or fake.

Recipient Name
Used
Your name was used 1 time in this email. If your name is in an email, the sender is aware of who you are instead of just your University number. Think about if the sender should have access your name.

Sender's Address
Outside the University
If an email originates from outside the

cautiously. An email originating from inside the university may still be unsafe, however.

email Domain
Free domain
It is unlikely that a legitimate organisation or their employees emailing in an official capacity would use a free, publicly-available domain. Google employees won't use "@gmail.com", for example.

Blacklisted Websites
Contains a blacklisted website
If a website is detected in spam emails, they typically get added to blacklists. A blacklist is a long list of untrustworthy sites. It is very unlikely a legitimate organisation would have a site that is blacklisted in their emails.

Attachments
No Attachments
There is nothing attached to this email.

emotions
Normal
Fear
Excessive
Surprise
Normal
Sad
Normal
Happy
Normal
Angry
Normal

Language
Normal
Mispellings
Acceptable misspellings
Keywords
Acceptable keywords

Link Features

We have found **4** unique domains within this email.
We attempted to scan 4 links but failed. This could be because the sites have been taken down or the link was inaccessible to our automatic analysis tool. The links were: "smtp.gmail.com", "179.43.175.108", "91.110.127.163", "192.168.1.86".

1. f3netze.de which occurs 4 times.
2. torservers.net which occurs 2 times.
3. for-privacy.net which occured 1 time.
4. dfri.se which occured 1 time.

f3netze.de
Unsafe
Domain Age
Age not found
Popularity

337,205th most popular domain
Blacklists
Blacklisted
Doesn't Match email Domain
This domain doesn't match the domain in the sender's address

torservers.net
Safe
Domain Age
11 years, 10 months, 13 days
Popularity
394,121st most popular domain
Blacklists
Not blacklisted
Doesn't Match email Domain
This domain doesn't match the domain in the sender's address

for-privacy.net
Unsafe
Domain Age
3 years, 1 months, 2 days
Popularity
389,436th most popular domain
Blacklists
Blacklisted
Doesn't Match email Domain
This domain doesn't match the domain in the sender's address

dfri.se
Unsafe
Domain Age
10 years, 9 months, 11 days
Popularity
232,991st most popular domain
Blacklists
Blacklisted
Doesn't Match email Domain
This domain doesn't match the domain in the sender's address

THE UNIVERSITY OF EDINBURGH

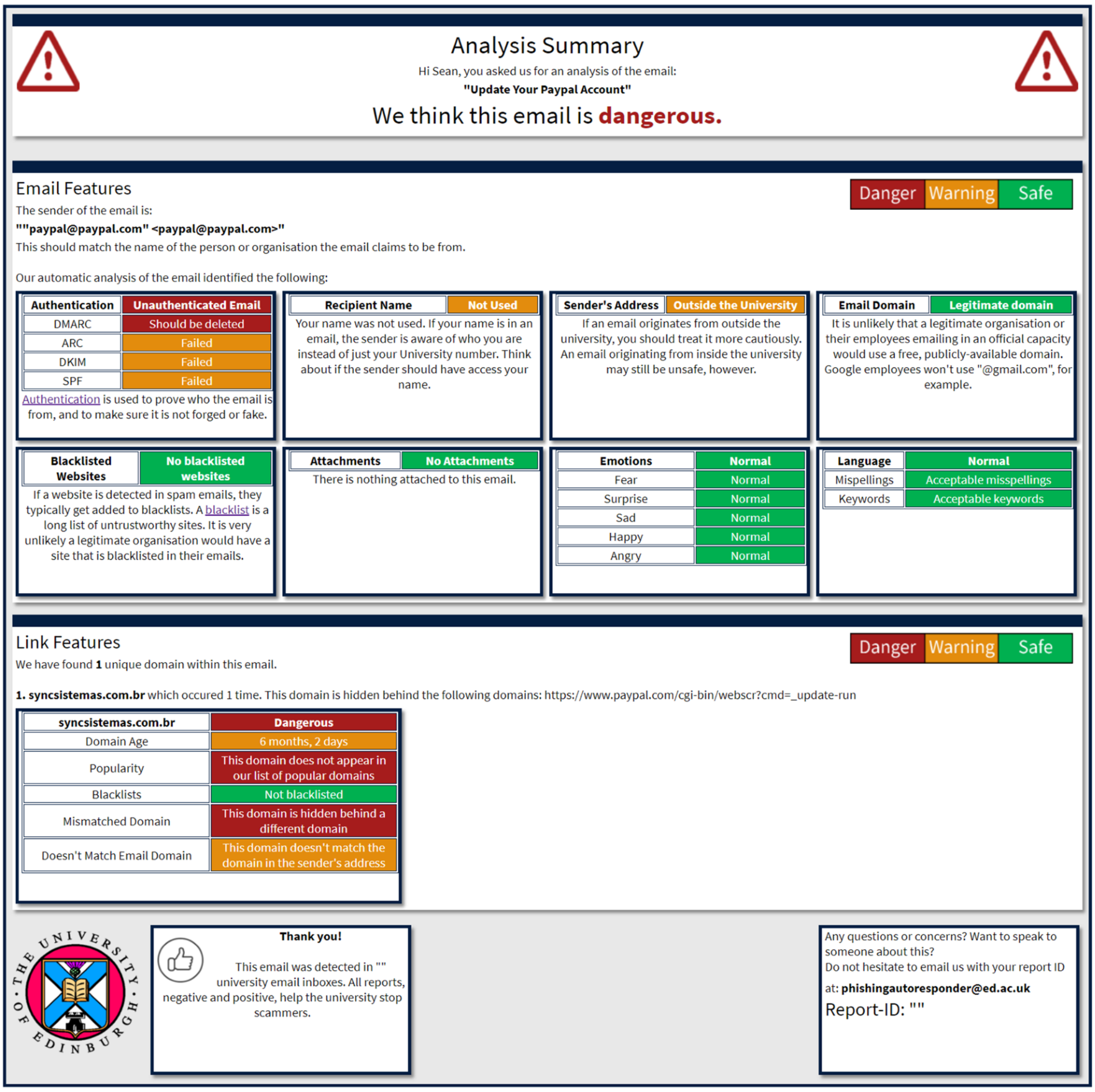
Thank you!
This email was detected in "" university email inboxes.
All reports, negative and positive, help the university stop scammers.

Any questions or concerns? Want to speak to someone about this?
Do not hesitate to email us with your report ID at:
phishingautoresponder@ed.ac.uk

Report-ID: ""

Reply

Figure D.1: The Auto-Responder's output in 9:20 (mobile) in Outlook.



Analysis Summary

Hi Sean, you asked us for an analysis of the email:

"Update Your Paypal Account"

We think this email is **dangerous.**

Email Features

Danger Warning Safe

The sender of the email is:

""paypal@paypal.com" <paypal@paypal.com>"

This should match the name of the person or organisation the email claims to be from.

Our automatic analysis of the email identified the following:

Authentication	Unauthenticated Email
DMARC	Should be deleted
ARC	Failed
DKIM	Failed
SPF	Failed

Authentication is used to prove who the email is from, and to make sure it is not forged or fake.

Recipient Name	Not Used

Your name was not used. If your name is in an email, the sender is aware of who you are instead of just your University number. Think about if the sender should have access your name.

Sender's Address	Outside the University

If an email originates from outside the university, you should treat it more cautiously. An email originating from inside the university may still be unsafe, however.

Email Domain	Legitimate domain

It is unlikely that a legitimate organisation or their employees emailing in an official capacity would use a free, publicly-available domain. Google employees won't use "@gmail.com", for example.

Blacklisted Websites	No blacklisted websites

If a website is detected in spam emails, they typically get added to blacklists. A blacklist is a long list of untrustworthy sites. It is very unlikely a legitimate organisation would have a site that is blacklisted in their emails.

Attachments	No Attachments

There is nothing attached to this email.

Emotions	Normal
Fear	Normal
Surprise	Normal
Sad	Normal
Happy	Normal
Angry	Normal

Language	Normal
Mispellings	Acceptable misspellings
Keywords	Acceptable keywords

Link Features

Danger Warning Safe

We have found **1** unique domain within this email.

1. syncsistemas.com.br which occured 1 time. This domain is hidden behind the following domains: https://www.paypal.com/cgi-bin/webscr?cmd=_update-run

syncsistemas.com.br	Dangerous
Domain Age	6 months, 2 days
Popularity	This domain does not appear in our list of popular domains
Blacklists	Not blacklisted
Mismatched Domain	This domain is hidden behind a different domain
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

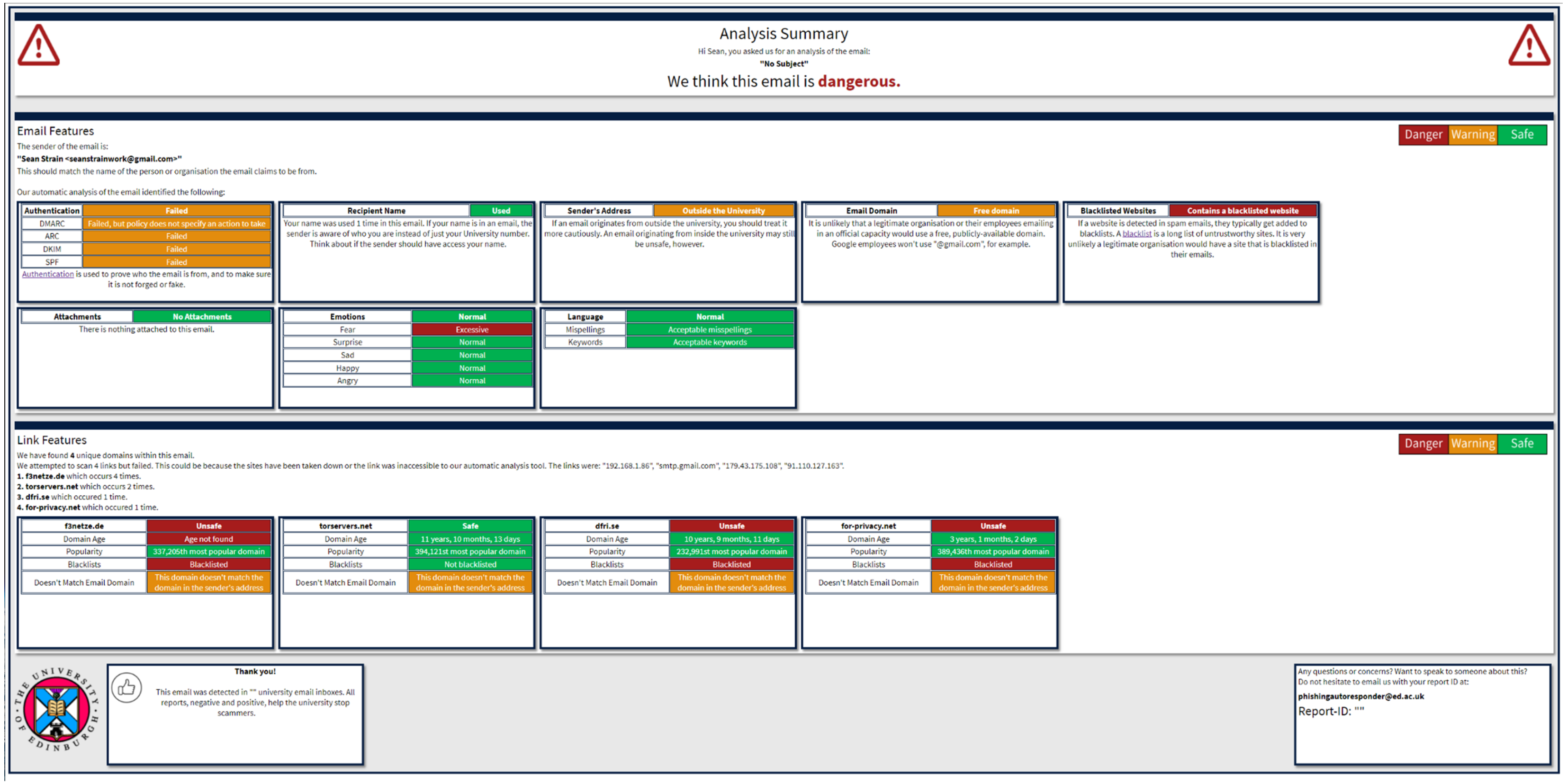
Thank you!

This email was detected in "" university email inboxes. All reports, negative and positive, help the university stop scammers.

Any questions or concerns? Want to speak to someone about this?
Do not hesitate to email us with your report ID at: **phishingautoresponder@ed.ac.uk**

Report-ID: ""

Figure D.2: The Auto-Responder's output in 1:1 (square).



Analysis Summary

Hi Sean, you asked us for an analysis of the email:

"No Subject"

We think this email is **dangerous.**

Email Features

Danger Warning Safe

The sender of the email is:

"Sean Strain <seanstrainwork@gmail.com>"

This should match the name of the person or organisation the email claims to be from.

Our automatic analysis of the email identified the following:

Authentication	Failed
DMARC	Failed, but policy does not specify an action to take
ARC	Failed
DKIM	Failed
SPF	Failed

Authentication is used to prove who the email is from, and to make sure it is not forged or fake.

Recipient Name	Used

Your name was used 1 time in this email. If your name is in an email, the sender is aware of who you are instead of just your University number. Think about if the sender should have access your name.

Sender's Address	Outside the University

If an email originates from outside the university, you should treat it more cautiously. An email originating from inside the university may still be unsafe, however.

Email Domain	Free domain

It is unlikely that a legitimate organisation or their employees emailing in an official capacity would use a free, publicly-available domain. Google employees won't use "@gmail.com", for example.

Blacklisted Websites	Contains a blacklisted website

If a website is detected in spam emails, they typically get added to blacklists. A blacklist is a long list of untrustworthy sites. It is very unlikely a legitimate organisation would have a site that is blacklisted in their emails.

Attachments	No Attachments

There is nothing attached to this email.

Emotions	Normal
Fear	Excessive
Surprise	Normal
Sad	Normal
Happy	Normal
Angry	Normal

Language	Normal
Mispellings	Acceptable misspellings
Keywords	Acceptable keywords

Link Features

Danger Warning Safe

We have found **4** unique domains within this email.

We attempted to scan 4 links but failed. This could be because the sites have been taken down or the link was inaccessible to our automatic analysis tool. The links were: "192.168.1.86", "smtp.gmail.com", "179.43.175.108", "91.110.127.163".

1. f3netze.de which occurs 4 times.
2. torservers.net which occurs 2 times.
3. dfri.se which occured 1 time.
4. for-privacy.net which occured 1 time.

f3netze.de	Unsafe
Domain Age	Age not found
Popularity	337,205th most popular domain
Blacklists	Blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

torservers.net	Safe
Domain Age	11 years, 10 months, 13 days
Popularity	394,121st most popular domain
Blacklists	Not blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

dfri.se	Unsafe
Domain Age	10 years, 9 months, 11 days
Popularity	232,991st most popular domain
Blacklists	Blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

for-privacy.net	Unsafe
Domain Age	3 years, 1 months, 2 days
Popularity	389,436th most popular domain
Blacklists	Blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

Thank you!

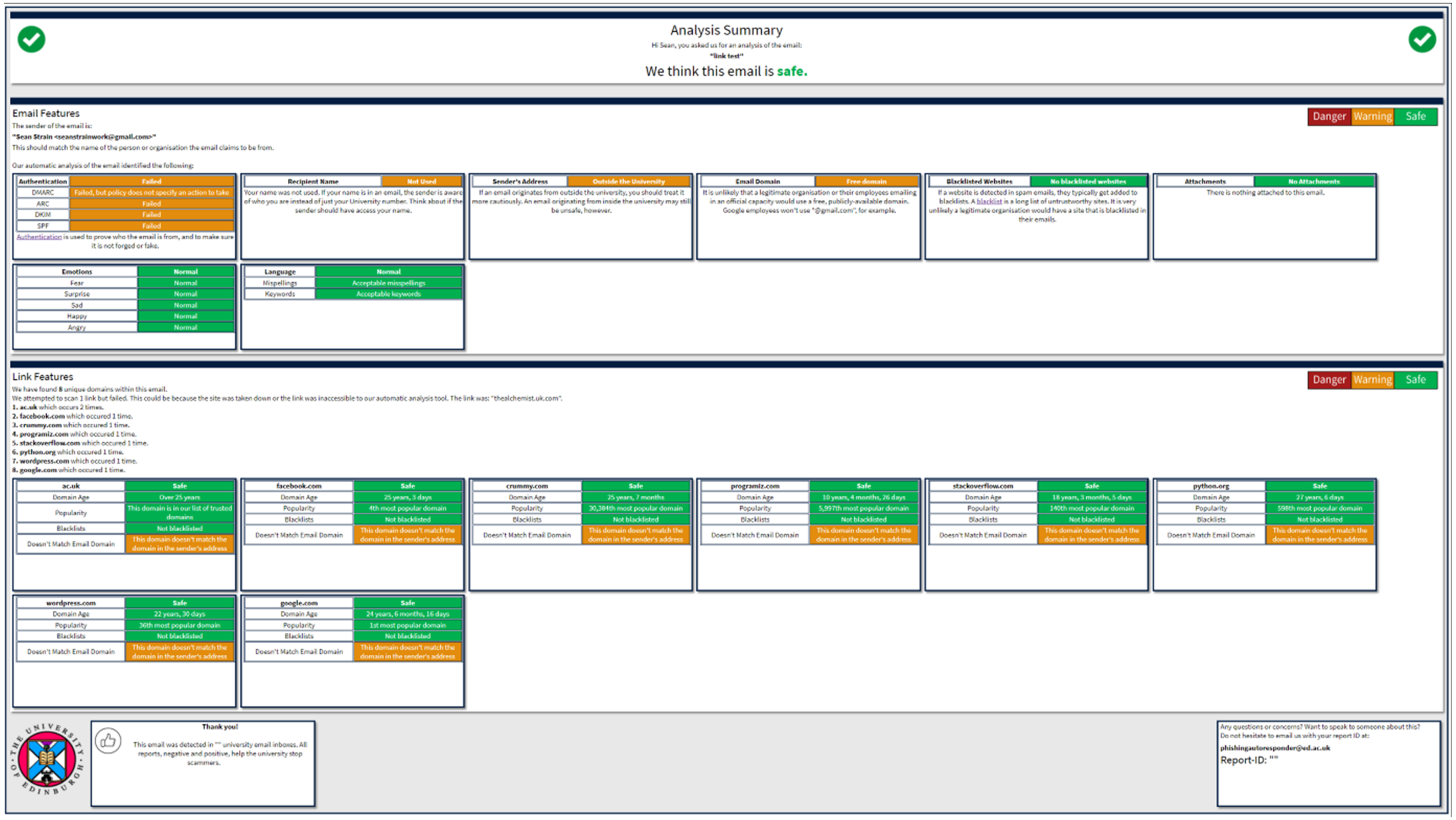
This email was detected in "" university email inboxes. All reports, negative and positive, help the university stop scammers.

Any questions or concerns? Want to speak to someone about this?
Do not hesitate to email us with your report ID at:

phishingautoresponder@ed.ac.uk

Report-ID: ""

Figure D.3: The Auto-Responder's output in 2:1.



Analysis Summary

Hi Sean, you asked us for an analysis of the email:

"link test"

We think this email is safe.

Email Features

Danger Warning Safe

The sender of the email is:

"Sean Strain <seanstrainwork@gmail.com>"

This should match the name of the person or organisation the email claims to be from.

Our automatic analysis of the email identified the following:

Authentication	Failed
DMARC	Failed, but policy does not specify an action to take
ARC	Failed
DKIM	Failed
SPF	Failed

Authentication is used to prove who the email is from, and to make sure it is not forged or fake.

Recipient Name	Not Used

Your name was not used. If your name is in an email, the sender is aware of who you are instead of just your University number. Think about if the sender should have access your name.

Sender's Address	Outside the University

If an email originates from outside the university, you should treat it more cautiously. An email originating from inside the university may still be unsafe, however.

Email Domain	Free domain

It is unlikely that a legitimate organisation or their employees emailing in an official capacity would use a free, publicly-available domain. Google employees won't use "@gmail.com", for example.

Blacklisted Websites	No blacklisted websites

If a website is detected in spam emails, they typically get added to blacklists. A blacklist is a long list of untrustworthy sites. It is very unlikely a legitimate organisation would have a site that is blacklisted in their emails.

Attachments	No Attachments

There is nothing attached to this email.

Emotions	Normal
Fear	Normal
Surprise	Normal
Sad	Normal
Happy	Normal
Angry	Normal

Language	Normal
Mispellings	Acceptable misspellings
Keywords	Acceptable keywords

Link Features

Danger Warning Safe

We have found 8 unique domains within this email.

We attempted to scan 1 link but failed. This could be because the site was taken down or the link was inaccessible to our automatic analysis tool. The link was: "thealchemist.uk.com".

1. ac.uk which occurs 2 times.
2. facebook.com which occured 1 time.
3. crummy.com which occured 1 time.
4. programiz.com which occured 1 time.
5. stackoverflow.com which occured 1 time.
6. python.org which occured 1 time.
7. wordpress.com which occured 1 time.
8. google.com which occured 1 time.

ac.uk	Safe
Domain Age	Over 25 years
Popularity	This domain is in our list of trusted domains
Blacklists	Not blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

facebook.com	Safe
Domain Age	25 years, 3 days
Popularity	4th most popular domain
Blacklists	Not blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

crummy.com	Safe
Domain Age	25 years, 7 months
Popularity	30,384th most popular domain
Blacklists	Not blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

programiz.com	Safe
Domain Age	10 years, 4 months, 26 days
Popularity	5,997th most popular domain
Blacklists	Not blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

stackoverflow.com	Safe
Domain Age	18 years, 3 months, 5 days
Popularity	140th most popular domain
Blacklists	Not blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

python.org	Safe
Domain Age	27 years, 6 days
Popularity	598th most popular domain
Blacklists	Not blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

wordpress.com	Safe
Domain Age	22 years, 30 days
Popularity	36th most popular domain
Blacklists	Not blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

google.com	Safe
Domain Age	24 years, 6 months, 16 days
Popularity	1st most popular domain
Blacklists	Not blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

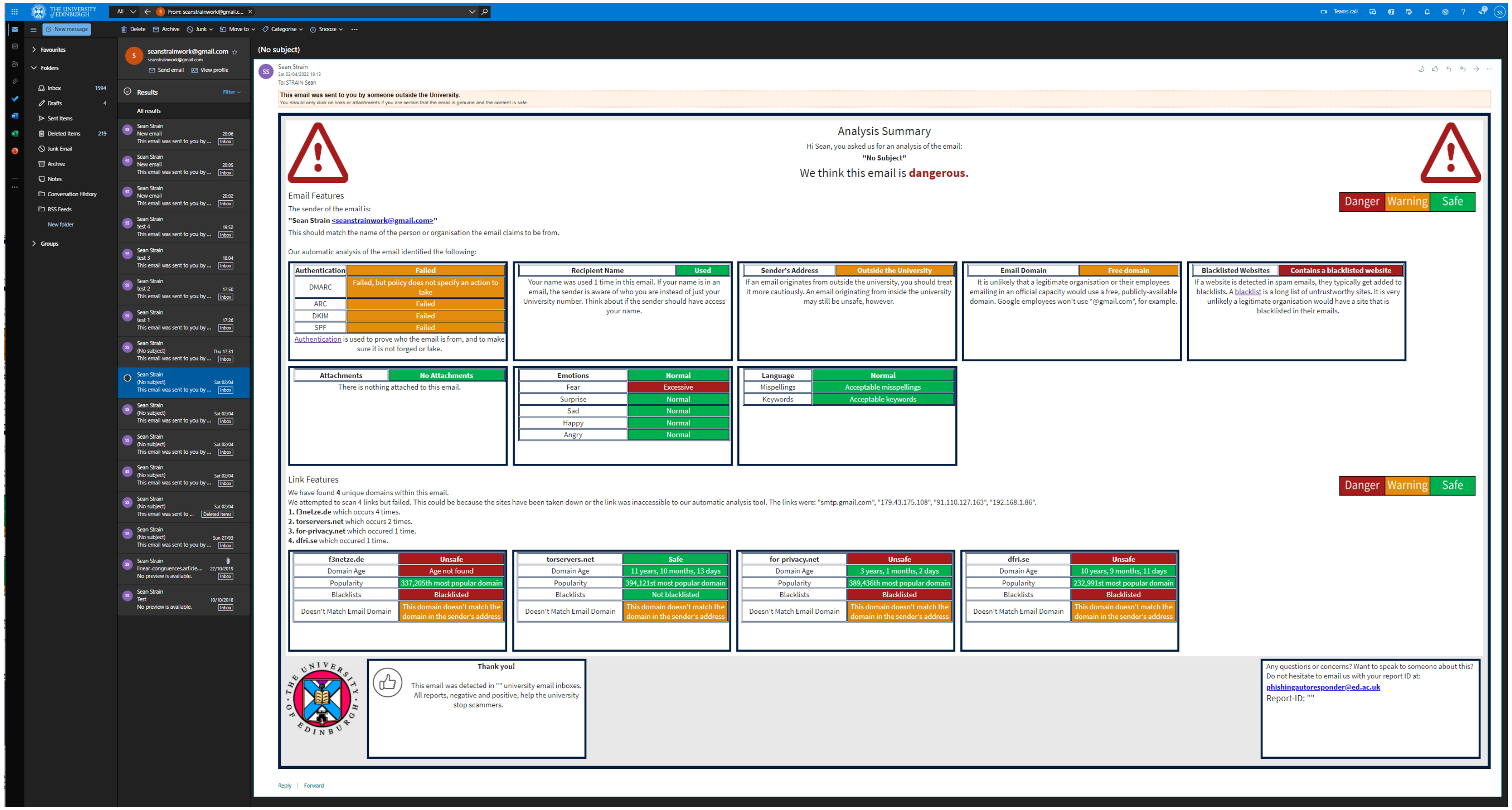
THE UNIVERSITY OF EDINBURGH

Thank you!

This email was detected in "" university email inboxes. All reports, negative and positive, help the university stop scammers.

Any questions or concerns? Want to speak to someone about this?
Do not hesitate to email us with your report ID at:
phishingautoresponder@ed.ac.uk
Report-ID: ""

Figure D.4: The Auto-Responder's output in 16:9 (widescreen).



Sean Strain
New email 20:06
This email was sent to you by ... Inbox

Sean Strain
New email 20:05
This email was sent to you by ... Inbox

Sean Strain
New email 20:02
This email was sent to you by ... Inbox

Sean Strain
test 4 19:52
This email was sent to you by ... Inbox

Sean Strain
test 3 18:04
This email was sent to you by ... Inbox

Sean Strain
test 2 17:50
This email was sent to you by ... Inbox

Sean Strain
test 1 17:26
This email was sent to you by ... Inbox

Sean Strain
(No subject) Thu 17:31
This email was sent to you by ... Inbox

Sean Strain
(No subject) Sat 02/04
This email was sent to you by ... Inbox

Sean Strain
(No subject) Sat 02/04
This email was sent to you by ... Inbox

Sean Strain
(No subject) Sat 02/04
This email was sent to you by ... Inbox

Sean Strain
(No subject) Sat 02/04
This email was sent to you by ... Inbox

Sean Strain
(No subject) Sat 02/04
This email was sent to ... Deleted Items

Sean Strain
(No subject) Sun 27/03
This email was sent to you by ... Inbox

Sean Strain
linear-congruences.article.... 22/10/2019
No preview is available. Inbox

Sean Strain
Test 10/10/2018
No preview is available. Inbox

(No subject)

Sean Strain
Sat 02/04/2022 19:13
To: STRAIN Sean

This email was sent to you by someone outside the University.
You should only click on links or attachments if you are certain that the email is genuine and the content is safe.

Analysis Summary

Hi Sean, you asked us for an analysis of the email:

"No Subject"

We think this email is **dangerous.**

Danger | Warning | Safe

Email Features

The sender of the email is:

"Sean Strain <seanstrainwork@gmail.com>"

This should match the name of the person or organisation the email claims to be from.

Our automatic analysis of the email identified the following:

Authentication	Failed
DMARC	Failed, but policy does not specify an action to take
ARC	Failed
DKIM	Failed
SPF	Failed

Authentication is used to prove who the email is from, and to make sure it is not forged or fake.

Recipient Name	Used

Your name was used 1 time in this email. If your name is in an email, the sender is aware of who you are instead of just your University number. Think about if the sender should have access your name.

Sender's Address	Outside the University

If an email originates from outside the university, you should treat it more cautiously. An email originating from inside the university may still be unsafe, however.

Email Domain	Free domain

It is unlikely that a legitimate organisation or their employees emailing in an official capacity would use a free, publicly-available domain. Google employees won't use "@gmail.com", for example.

Blacklisted Websites	Contains a blacklisted website

If a website is detected in spam emails, they typically get added to blacklists. A blacklist is a long list of untrustworthy sites. It is very unlikely a legitimate organisation would have a site that is blacklisted in their emails.

Attachments	No Attachments

There is nothing attached to this email.

Emotions	Normal
Fear	Excessive
Surprise	Normal
Sad	Normal
Happy	Normal
Angry	Normal

Language	Normal
Mispellings	Acceptable misspellings
Keywords	Acceptable keywords

Link Features

Danger | Warning | Safe

We have found **4** unique domains within this email.
We attempted to scan 4 links but failed. This could be because the sites have been taken down or the link was inaccessible to our automatic analysis tool. The links were: "smtp.gmail.com", "179.43.175.108", "91.110.127.163", "192.168.1.86".

1. f3netze.de which occurs 4 times.
2. torservers.net which occurs 2 times.
3. for-privacy.net which occured 1 time.
4. dfri.se which occured 1 time.

f3netze.de	Unsafe
Domain Age	Age not found
Popularity	337,205th most popular domain
Blacklists	Blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

torservers.net	Safe
Domain Age	11 years, 10 months, 13 days
Popularity	394,121st most popular domain
Blacklists	Not blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

for-privacy.net	Unsafe
Domain Age	3 years, 1 months, 2 days
Popularity	389,436th most popular domain
Blacklists	Blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

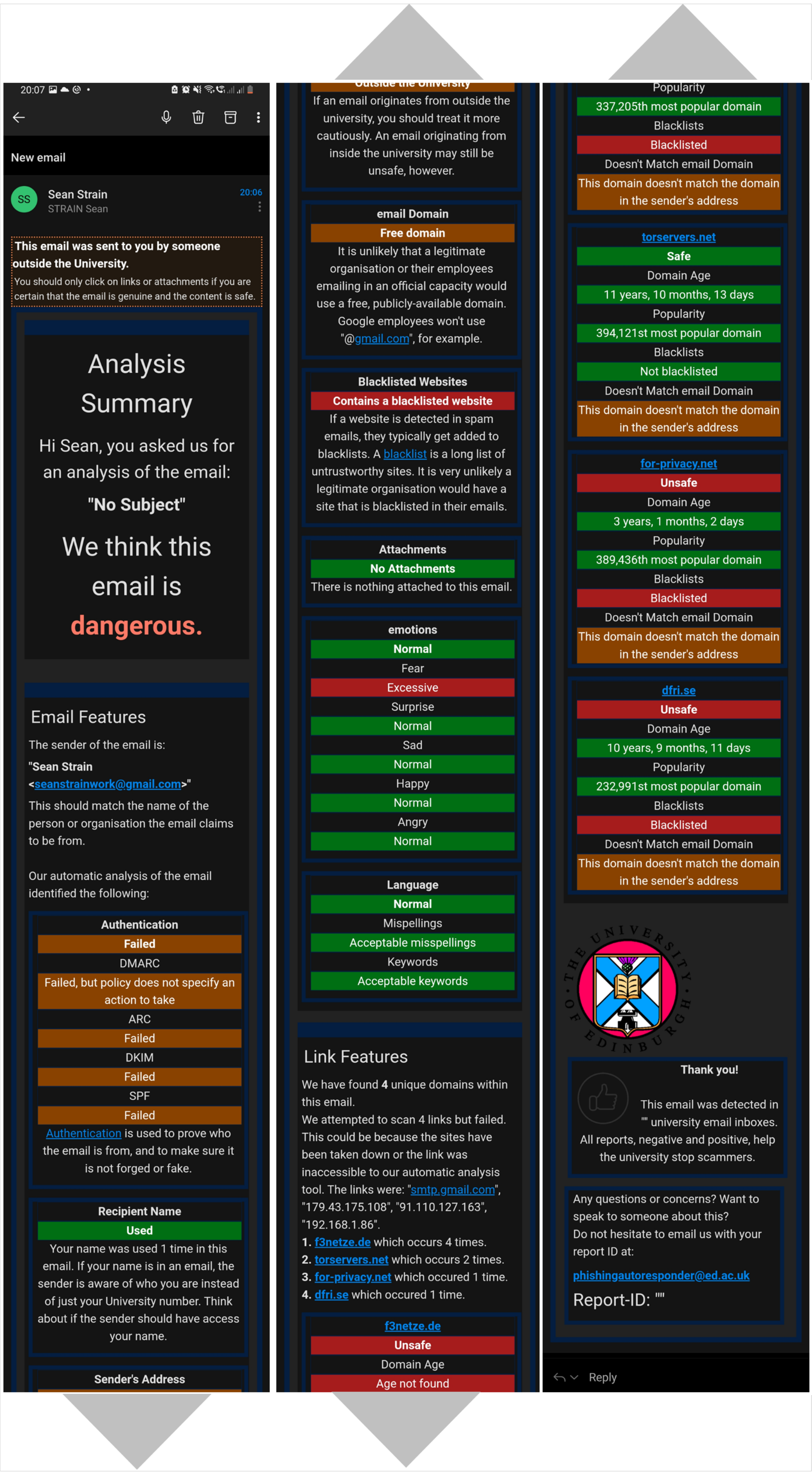
dfri.se	Unsafe
Domain Age	10 years, 9 months, 11 days
Popularity	232,991st most popular domain
Blacklists	Blacklisted
Doesn't Match Email Domain	This domain doesn't match the domain in the sender's address

Thank you!

This email was detected in "" university email inboxes. All reports, negative and positive, help the university stop scammers.

Any questions or concerns? Want to speak to someone about this? Do not hesitate to email us with your report ID at:
phishingautoresponder@ed.ac.uk
Report-ID: ""

Figure D.5: The Auto-Responder's output in Outlook.



20:07

New email

SS Sean Strain 20:06
STRAIN Sean

This email was sent to you by someone outside the University.
You should only click on links or attachments if you are certain that the email is genuine and the content is safe.

Analysis Summary

Hi Sean, you asked us for an analysis of the email:

"No Subject"

We think this email is **dangerous.**

Email Features

The sender of the email is:

"Sean Strain <seanstrainwork@gmail.com>"

This should match the name of the person or organisation the email claims to be from.

Our automatic analysis of the email identified the following:

Authentication
Failed
DMARC
Failed, but policy does not specify an action to take
ARC
Failed
DKIM
Failed
SPF
Failed
Authentication is used to prove who the email is from, and to make sure it is not forged or fake.

Recipient Name
Used
Your name was used 1 time in this email. If your name is in an email, the sender is aware of who you are instead of just your University number. Think about if the sender should have access your name.

Sender's Address

Outside the University
If an email originates from outside the university, you should treat it more cautiously. An email originating from inside the university may still be unsafe, however.

email Domain
Free domain
It is unlikely that a legitimate organisation or their employees emailing in an official capacity would use a free, publicly-available domain. Google employees won't use "@gmail.com", for example.

Blacklisted Websites
Contains a blacklisted website
If a website is detected in spam emails, they typically get added to blacklists. A blacklist is a long list of untrustworthy sites. It is very unlikely a legitimate organisation would have a site that is blacklisted in their emails.

Attachments
No Attachments
There is nothing attached to this email.

emotions
Normal
Fear
Excessive
Surprise
Normal
Sad
Normal
Happy
Normal
Angry
Normal

Language
Normal
Mispellings
Acceptable misspellings
Keywords
Acceptable keywords

Link Features

We have found **4** unique domains within this email.
We attempted to scan 4 links but failed. This could be because the sites have been taken down or the link was inaccessible to our automatic analysis tool. The links were: "smtp.gmail.com", "179.43.175.108", "91.110.127.163", "192.168.1.86".
1. f3netze.de which occurs 4 times.
2. torservers.net which occurs 2 times.
3. for-privacy.net which occured 1 time.
4. dfri.se which occured 1 time.

f3netze.de
Unsafe
Domain Age
Age not found

Popularity
337,205th most popular domain
Blacklists
Blacklisted
Doesn't Match email Domain
This domain doesn't match the domain in the sender's address

torservers.net
Safe
Domain Age
11 years, 10 months, 13 days
Popularity
394,121st most popular domain
Blacklists
Not blacklisted
Doesn't Match email Domain
This domain doesn't match the domain in the sender's address

for-privacy.net
Unsafe
Domain Age
3 years, 1 months, 2 days
Popularity
389,436th most popular domain
Blacklists
Blacklisted
Doesn't Match email Domain
This domain doesn't match the domain in the sender's address

dfri.se
Unsafe
Domain Age
10 years, 9 months, 11 days
Popularity
232,991st most popular domain
Blacklists
Blacklisted
Doesn't Match email Domain
This domain doesn't match the domain in the sender's address

THE UNIVERSITY OF EDINBURGH

Thank you!
This email was detected in "" university email inboxes.
All reports, negative and positive, help the university stop scammers.

Any questions or concerns? Want to speak to someone about this?
Do not hesitate to email us with your report ID at:
phishingautoresponder@ed.ac.uk
Report-ID: ""

Figure D.6: The Auto-Responder's output in 9:20 (mobile) in Outlook, shown in dark-mode.