POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Artificial Scams: On the risks of Fully Agentic Spear Phishing

Tesi di Laurea Magistrale in
Computer Science and Engineering - Ingegneria Informatica

Author: **Manuela Maroni**

Student ID: 252121
Advisor: Prof. Stefano Longari
Co-advisors: Prof. Michele Carminati, Francesco Panebianco
Academic Year: 2024-25

Abstract

Phishing is increasingly shifting from generic mass campaigns to highly targeted spear phishing attacks, thanks to Large Language Models (LLMs) enabling the automated generation of coherent, persuasive, and context-aware text at scale. This thesis investigates whether LLM-based systems can be integrated into a fully automated pipeline for generating personalized phishing emails and whether such automation significantly increases attack effectiveness. The system is based on a modular multi-agent architecture which autonomously performs identity inference from the email address, public data extraction, topic selection, and email generation using only publicly available information. Each component of the pipeline is experimentally validated, highlighting both the potential and the limitations of automated identity disambiguation and contextual inference. The overall effectiveness of the system is evaluated through a controlled human-subject experiment comparing personalized LLM-generated phishing emails with a traditional generic phishing message. The results show a substantially higher Click Through Rate (CTR) for personalized emails, even within a technically aware population. A subgroup analysis further indicates that correct recipient name usage is essential to maintain message credibility, while topic relevance plays a decisive role in increasing user engagement. Despite limitations related to participant demographics, data availability, and ethical constraints, the findings demonstrate that LLM-driven automation can significantly enhance the scalability and persuasive power of spear phishing attacks, reinforcing the need for more advanced detection mechanisms and adaptive defensive strategies to counter increasingly personalized and automated phishing threats.

Keywords: phishing, spear phishing, social engineering, Large Language Models, cybersecurity

Abstract in lingua italiana

Il phishing si sta progressivamente evolvendo da generiche campagne di massa a attacchi di spear phishing altamente precisi, grazie ai LLM che consentono la generazione automatica di testi coerenti, persuasivi e contestualizzati su larga scala. Questa tesi valuta se i sistemi basati su LLM possono essere integrati in una pipeline completamente automatizzate per la generazione di messaggi di phishing personalizzati e se tale automazione aumenti significativamente l'efficacia degli attacchi. Il sistema proposto si basa su un'architettura modulare multi-agente che esegue autonomamente l'estrazione dell'identità dall'indirizzo mail, l'estrazione di dati pubblici, la selezione del tema e la generazione dell'email utilizzando esclusivamente i dati disponibili pubblicamente. Tutti i componenti della pipeline sono stati validati sperimentalmente, sottolineando sia il potenziale che i limiti dell'automatizzazione nei processi di disambiguazione dell'identità e di estrazione del contesto. L'efficacia complessiva del sistema è stata valutata tramite un esperimento controllato con soggetti umani, comparando email di phishing personalizzate generate tramite LLM con un messaggio di phishing tradizionale. I risultato mostrano un CTR significativamente più elevato per le mail personalizzate, anche all'interno di una popolazione con competenze tecniche. L'analisi per sottogruppi indica inoltre che il corretto utilizzo del nome del destinatario è essenziale per mantenere la credibilità del messaggio, mentre la rilevanza del tema riveste un ruolo determinante nell'aumentare il coinvolgimento dell'utente. Nonostante le limitazioni legate alle caratteristiche del campione sperimentale, alla disponibilità dei dati e ai vincoli etici, i risultati dimostrano che l'automazione basata su LLM può incrementare significativamente la scalabilità e il potere persuasivo degli attacchi di spear phishing, rafforzando la necessità di meccanismi di rilevamento più avanzati e di strategie difensive adattive per contrastare minacce di phishing sempre più personalizzate e automatizzate.

Parole chiave: phishing, spear phishing, ingegneria sociale, Large Language Models, cybersecurity

Contents

Introduction

Phishing is one of the most effective social engineering attacks against individuals and organizations [14], where attackers impersonate a trusted party and prompt users with a call to action with the goal to trick them into revealing sensible information. Cybercriminals can then use this information for various purposes, e.g., stealing money, identities, or accounts. Phishing is particularly successful because it exploits human psychology, using deceptive methods to push victims to act impulsively before they can reason on the legitimacy of the message. These methods include the use of persuasion principles, e.g., authority and scarcity [4], and the imitation of legitimate messages in visual style, content, and timing.

The message containing the call to action can be delivered in different ways, such as through an email, a text message, or a phone call. In emails and text messages, the call to action usually refers to convincing the user to click on a link contained in the body of the message, which brings the victim to a malicious form. Based on the objective of the attackers, the form can ask for different kinds of information, such as login information, credit card information, or personal information. Login information can be used to gain access to the victim's personal or business account and possibly block them out, causing the extraction of sensible information or company secrets and other disservices. Credit card information is also particularly valuable, as it can be sold on the dark web or it can be used for unauthorized online purchases. Personal information can be used for identity theft and fraud, for example by opening credit and loan accounts using the victim's name or by applying for government benefits.

Phishing attacks can be divided into two categories: classic phishing and spear phishing. Classic phishing is a mass-scale attack, where attackers send generic messages to a wide audience, hoping that a small percentage of recipients responds to the call to action. Classic phishing emails are not tailor-made for the recipient and personalization is minimal, often limited to addressing the recipient by name. Classic phishing emails are relatively easy and inexpensive to produce, however, they can also be detected without much effort. Spear phishing, instead, aims at specific targets by personalizing the message both in content and in style. Writing a spear phishing email assumes prior knowledge

of the victim's interests or professional role, which is typically gathered through an online search of publicly available sources. As a result, spear phishing has traditionally been more resource-intensive and time-consuming than classic phishing, limiting its use to high-value targets rather than large-scale campaigns. However, recent advances in Artificial Intelligence (AI) and LLMs have reshaped the picture [1, 5]. These models can be used to automate both the extraction of significant personal information and the generation of relevant messages, reducing the cost and effort associated with personalization [7]. Attackers can now conduct spear phishing campaigns at a much larger scale while maintaining a high degree of precision and a low cost.

With this thesis, we aim to demonstrate what personal information modern cybercriminals can infer from the Internet, in particular from social media accounts, and how this information can be used to generate spear phishing emails, through custom-made automated pipelines that integrate LLMs and Open-Source Intelligence (OSINT) techniques. We also attempt to raise awareness of the risks derived from such availability of personal information by providing participants with an extensive report that presents the data retrieved by the pipeline in an organized way. Finally, we offer quantitative data based on human-subject campaigns on the effectiveness of personalized LLM-based phishing emails, evaluate the efficacy of various personalization attributes, and provide an effective validation method for the automated pipeline.

Previous work on the elements that contribute to human susceptibility to phishing emails focuses on the common segments in which such emails can be classified [3], on persuasion principles [4, 10], and on demographics and email content [13]. More recent work explores phishing and spear phishing using LLMs, focusing in particular on the use of LLMs in targeted phishing campaigns [9, 12], multi-modal generative attacks [6], and on the use of LLMs for defensive purposes [9, 12]. Finally, there are works that examine large scale phishing campaigns, exploring LLM-based lateral phishing attacks at a large organization [2] and fully-automated human-validated spear phishing campaigns [8]. Section 1.2 summarizes these lines of work in detail.

There are three primary reasons why these approaches fail. First, previous studies on spear phishing assume richer initial inputs or manual steps for personalization, which is not a realistic starting point for an attacker. In this thesis, we only use the email address of the victim as an initial input and no manual steps are needed, making our results closer to those that a cybercriminal might achieve in the real world. Second, these works are missing detailed results on the inferred attributes that influence user responses to the call to action, such as a correctly inferred recipient name or a topic highly relevant to the recipient. Measurement of the impact of these attributes on the CTR is key to

understanding user behavior in front of phishing emails. Finally, other works do not provide a personalized informative report to the participants, which shows what personal information can be inferred from an automated online search based only on the email address.

We propose a new approach on LLM-based spear phishing: a multi-agent pipeline, where each agent has a well-established task and tools at its disposal to complete it. The personalized email generation pipeline runs as follows: the system infers candidate names and organization affiliation from a single email address, discovers and scrapes social media profiles, ranks and validates candidate identities, selects a topic derived from public content, and produces a personalized email. The results of this pipeline are compared to a classic phishing baseline that uses a non-personalized message based on a real-world phishing email. All emails are delivered using the same sending infrastructure, and accesses to the landing page are logged in the same way. On the landing page, participants who received the personalized email have the option to request a detailed report containing the information the pipeline inferred about them. Chapter 2 analyzes in detail the design and operations of this project.

The personalized email generation pipeline is implemented as an automated system that orchestrates multiple LLM-based agents for profile discovery and email generation, combined with web navigation and content extraction techniques to gather publicly available information. Two different landing pages are presented depending on group membership, which debrief the participants who click on the phishing link and collect data on CTRs and, optionally, report generation requests.
Various datasets were used in the experiments: a common email benchmark was adopted to validate the name extraction process with an exhaustive list of real-world email addresses; a custom-made dataset, built using social media references from personal websites and blogs on the Internet, was employed to validate the agents; the dataset of participants who received the personalized phishing email was built by sending a Microsoft Form to students following courses related to cybersecurity and machine learning, which also allowed us to ensure consensual processing of personal data; the dataset of participants that received the classic phishing email was provided by Politecnico's Information Technology (IT) department and consisted of engineering students currently studying at Politecnico di Milano. Chapter 3 provides more implementation details.

The experiments we conducted revealed several important observations on the effectiveness of LLM-based spear phishing and the impact of personalization on phishing strategies. The first experiment focused on the validation of the personalized email generation pipeline. This step was essential in ensuring that the pipeline could generate effective and

tailored messages. The first results of the validation process showed the agents' strong performance in certain tasks, while also providing useful data for further improvements. After making the necessary changes, we repeated the experiment and concluded the validation process.
The second experiment produced two significant results. First, we showed that LLM-based spear phishing techniques are more effective than traditional phishing methods. This suggests that LLMs can generate convincing and personalized attacks, which may be harder for victims to detect. The second finding highlighted the importance of including personalized details in the email body. The inclusion of the target's correct name and ensuring that the email's content was relevant to the target's interests were found to substantially increase the likelihood of the victim engaging with the malicious Uniform Resource Locator (URL) in the email body. Chapter 4 analyzes in detail the results of the experiments.

With this thesis, we make several contributions to the study of phishing attacks and their automation, which we list below:

- A fully automated, multi-agent system that generates personalized phishing emails and requires only the email address as input, reflecting the attack conditions more realistically than previous studies.
- A tool that generates personal reports for study participants and illustrates the extent of personal information that can be inferred from the email address alone, with the goal of educating them on the risks associated with sharing personal data online and on how this information can be exploited in phishing attacks.
- A controlled human-subject evaluation where we demonstrate that LLM-generated spear phishing emails are more effective than classic phishing emails, and a quantitative analysis of specific personalization factors and their impact on CTRs and responses to the call to action.

1 | Motivation

In this chapter, we introduce the problems we want to solve, summarize the state of the art, and present the goals and challenges of this work.

1.1. Problem statement

Cybersecurity threats are becoming increasingly sophisticated, and phishing remains one of the most common attacks. In particular, spear phishing, where attackers send personalized emails to specific targets, has gained popularity and has become a significant threat to both individuals and organizations. With the rapid advancements in AI and LLMs, cybercriminals have the ability to easily launch large-scale spear phishing campaigns with little effort, making them more widespread and more dangerous. These models can autonomously generate convincing, tailored emails by leveraging publicly available personal data, making it significantly more difficult for victims to discriminate between fraudulent messages and legitimate ones.

There is limited empirical evidence on how much more effective fully-automated LLM-driven spear phishing attacks are compared to traditional generic phishing campaigns, in particular when personalization is derived solely from information realistically available to attackers, such as the email address. Therefore, it is necessary to analyze to what extent this type of personalization impacts user susceptibility to spear phishing emails and how much information can be derived through the combination of LLMs and OSINT techniques, so that researchers can assess the real-world risks posed by these attacks.
Furthermore, not all forms of personalization may contribute equally to the success of a phishing attack. Attributes such as a correctly inferred recipient's name, or the relevance of the email's topic may influence user behavior in different ways. However, the importance of these inferred attributes and their effect on user responses to spear phishing calls to action are not well understood. This lack of understanding limits the ability to anticipate attacker strategies and to design effective countermeasures.
Finally, there is an extensive awareness gap among users about how much personal information is publicly available and how it can be used in a spear phishing attack. People often

underestimate how these seemingly innocuous data can be aggregated and exploited to craft persuasive and targeted messages. This disconnection between perceived and actual exposure further aggravates the spear phishing problem.

1.2. State of the art

In this section, we review prior work relevant to the topics addressed in this thesis, including traditional and spear phishing, the use of LLMs in phishing attacks, and the psychological and behavioral factors behind the effectiveness of phishing techniques.

1.2.1. Traditional phishing and human susceptibility

Due to their popularity and effectiveness, phishing attacks have been widely studied and details about the causes of their efficacy have been extensively investigated. In this subsection, we analyze relevant works on the subject.

Burita et al. [3] analyze the content of 200 classic phishing emails collected over a period of two months from two email accounts of one of the authors. The results of the text analysis on selected parameters reveal that phishing emails were delivered significantly more often to the work account than to the personal one and that more than half of the emails were sent from addresses impersonating a male individual. Based on their characteristics, the authors also classified the messages received into the following segments: *Business*, *Charity*, *Fund*, *Transfer*, and *Others*. The *Business* segment includes phishing emails that offer cooperation on a project, investment in the recipient's country, execution of a contract, or realization of a business opportunity. The *Charity* segment typically consists of emails whose sender is an widowed old woman with no children and with only a few weeks to live, whose husband left a large fortune which she wants to donate to charity. If the recipient is willing to open a charity fund, they will be rewarded with part of the said fortune. The *Fund* segment includes phishing emails that promised the recipient money obtained from a fund (compensation, scam, or fraud), a financial gift, or assets from inheritance. The *Transfer* segment included phishing emails in which the sender requested cooperation for money or other asset transfer, in exchange for a commission on the transferred amount. The *Other* segment includes emails of different categories, with low volume with respect to the other segments. Among the identified segments, the *Business* category was the most common.

In his work, Cialdini [4] identifies six principles of persuasion that can be exploited to influence the behavior of individuals and describes them in detail. Reciprocity refers to

the people's tendency to "return the favor", therefore doing something for someone can drive them to do something in return. Commitment and Consistency refer to users' desire to uphold commitments previously made. Social proof refers to how people often look at the actions of others before making their own decisions. Liking refers to people's tendency of saying yes to those they like, that are similar to themselves, or that have similar values and goals. Authority refers to how people tend to avoid questioning directives issued by experts or other authority figures. Scarcity refers to people's reaction when an item's availability is limited. These principles are not limited to the phishing domain and can be used to manipulate people into taking actions that they otherwise would not take.

Khadka et al. [10] systematically summarize research on the use of Cialdini's persuasion principles in phishing attacks. The paper shows that the effectiveness of phishing emails is strongly influenced by psychological manipulation rather than purely technical factors and that attackers combine multiple persuasion principles to obtain better results. The survey also identifies personalization as a key amplifier of these persuasion principles, as tailored details increase the perceived legitimacy of the message. The work helps explain why personalized phishing messages are effective and motivates further research into how modern automated systems implement these persuasion strategies at scale.

Lin et al. [13] study the effect of the age of Internet user and email content, such as weapons of influence and life domains, on spear phishing susceptibility by sending daily simulated phishing emails for 21 days to 158 participants, 100 young and 58 older. The results show that 43% of the participants fell for the simulated phishing emails and that older women showed the highest susceptibility. The paper also reveals that susceptibility in younger participants declined throughout the duration of the study, while it remained stable in older participants. The study demonstrates that susceptibility is in general highest for scarcity and legal emails and lowest for social proof and financial emails. Looking at the different age groups, young users showed greater susceptibility to scarcity and authority emails, while older users showed greater susceptibility to reciprocation and liking emails. Finally, the article highlights that older users reported lower susceptibility awareness compared to younger users.

1.2.2. LLM-based phishing and spear phishing

The generation of phishing and spear phishing emails and their detection using AI and LLMs has attracted significant attention in prior research. In this subsection, we discuss the most relevant related works.

Heiding et al. [9] compare the CTR of four different groups, based on the type of phishing

email received: automatically generated by GPT-4, manually written using the V-Triad, written by combining GPT-4 and the V-Triad, and generic phishing emails. The results indicate that the control group emails received a click-through rate between 19-28%, the GPT-generated emails 30-44%, emails generated by the V-Triad 69-79%, and emails generated by GPT and the V-Triad 43-81%. Participants were also asked to explain why they clicked or did not click on a link in the email, however answers are contradictive and highlight the importance of personal differences. The study also uses popular LLMs to detect the intention of phishing emails, demonstrating a strong ability to distinguish malicious intent that sometimes surpasses human detection, even in non-obvious phishing emails. Finally, the research concludes with an analysis of the economic aspects of AI-enabled phishing attacks, showing how LLMs increase the incentives of phishing and spear phishing by reducing their cost.

Gallagher et al. [6] present two case studies in which LLMs are used to orchestrate phishing attacks and illustrate how LLMs can process and generate content across multiple domains, such as text, code, images, and voice. The first case study focuses on the use of LLMs in facilitating the automation of phishing websites aimed at credential theft, in particular by building a fraudulent e-commerce website. LLMs can enable the systematic orchestration of phishing scams, combining code, text, images, and audio to create numerous websites, product catalogs, and testimonials, although some human intervention was still required. The second case study analyzes the use of LLM in a text-based scam, which uses fake cryptocurrency trading and lures targets through a romantic interest in them. In this case, scammers started using LLM-based responses and were able to successfully steal money from a victim.

Lim et al. [12] use Artificial Intelligence as a Service (AIaaS) products to build a pipeline that generates personalized and persuasive phishing emails based on the target's background and personality. The authors tested the pipeline in three internal phishing campaigns against manually generated phishing emails and found that the AIaaS pipeline matched or exceeded the effectiveness of manually generated emails. The pipeline also allowed Red Team resources to focus on more valuable work. Additionally, the study presented an AIaaS-powered phishing defense framework to detect phishing attacks, allowing security teams to defend against such sophisticated attacks. Finally, the paper presents a discussion of the long-term implications of AI-generated phishing emails and recommends high-level strategies to protect against the abuse of AIaaS products.

1.2.3. Large-scale LLM-driven phishing campaigns

More recently, LLMs have started to be used to launch large-scale and fully-automated spear phishing campaigns, where LLMs take care of all aspects of the attack, from the initial intelligence research to the final phishing email generation. In this subsection, we discuss the research available in this line of work.

Bethany et al. [2] investigate two critical problems concerning the use of LLMs in the phishing context: the lack of specific research on LLM integration for large-scale attacks targeting an entire organization, and the lack of capability to prevent LLM-generated attacks. The study explores the first problem by targeting approximately 9,000 individuals in a university over a period of 11 months, using LLMs to create targeted lateral phishing emails. Then, it evaluates the capability of the email filtering infrastructure to detect such LLM-based phishing attempts, and proposes machine learning-based detection techniques to detect those emails that were missed by the existing infrastructure.

Heiding et al. [8] evaluate the capability of LLMs to carry out personalized phishing attacks and compare their performance with human experts and older AI models. The 101 participants are divided into four groups, based on the type of email received: a control group of arbitrary phishing emails, which received a CTR of 12%, emails generated by human experts (CTR of 54%), fully AI-automated emails (CTR of 54%), and AI emails using a human-in-the-loop (CTR of 56%). The AI-automated emails were written using a custom-built tool that automates the entire spear phishing process, including information gathering and creating personalized vulnerability profiles for each target. The reconnaissance tool used as input the information collected from the initial recruitment survey, such as university affiliation and focus area. Data on the accuracy of the information gathered are also available in the study, with useful information collected in 88% of cases and only 4% of inaccurate profiles produced. The research also focuses on the ability of LLMs to detect the intention of the emails, with the best model scoring above 90% with low false-positive rates. Lastly, the paper analyzes the economics of phishing, highlighting how AI enables attackers to target more individuals at a lower cost.

1.3. Goals and challenges

With this thesis, we address the problems and research gaps illustrated above by providing an empirical analysis of LLM-based spear phishing. In particular, we plan to produce a realistic measurement of the effectiveness of fully-automated LLM-driven spear phishing emails when compared to traditional, non-personalized phishing messages. Effectiveness

is measured through user interaction metrics, such as CTR, allowing for a direct comparison between personalized and generic approaches under controlled conditions. Existing works either focus exclusively on automating the email generation process while assuming rich, pre-existing knowledge about the target, or they depend on extensive background information that would not typically be available to a real-world attacker. In contrast, we adopt a more restrictive and realistic threat model by assuming that the attacker's only initial piece of information is the target's email address. We also explore how publicly accessible information can be inferred using OSINT techniques, with a focus on identifying personal interests and details that may be exploited for personalization.
In addition, we intend to explore the role that individual personalization attributes play in influencing user behavior. Specifically, we intend to understand which types of personal data are most effective in phishing scenarios by analyzing how the different inferred characteristics, such as a correctly guessed recipient's name or interests, affect user responses. This analysis contributes to a more refined understanding of how personalization increases the persuasive power of phishing emails.
Finally, we are also motivated by broader educational considerations. We prioritize transparency toward study participants by allowing them to request a report showing how their publicly available personal information can be collected and exploited, therefore addressing the existing gap between users' perceived and actual online exposure.

The problems addressed in this thesis present several inherent challenges that affect the feasibility of the experiments and the effectiveness of personalized phishing techniques. These challenges are organized below:

- **C1: Identity inference and ambiguity**
  One of the most pressing issues is the inference of a target's identity from an email address. Email addresses often contain partial or ambiguous information about a person's name, and may be based on nicknames, aliases, or unrelated identifiers, providing limited or no reliable clues about the individual's real identity. Additionally, even when a full name can be reasonably deduced from the email address, the risk of misidentification remains due to the presence of common names shared by multiple individuals. This lack of certainty affects the quality of personalization of the phishing emails and, in some cases, reduces their credibility and effectiveness.

- **C2: Limited availability of publicly accessible information**
  A second challenge regards the reliance on OSINT-based online research. Publicly available information may be sparse, incomplete or entirely unavailable for certain individuals, particularly when search results return no profiles or only private accounts. In such cases, the pipeline has little or no data on which to base

personalization of the phishing email, forcing a fallback to generic phishing message.

- **C3: Email delivery constraints in real-world environments**
  Another significant challenge in this domain is the reliable delivery of phishing emails in real-world settings. Even when standard email authentication mechanisms such as Sender Policy Framework (SPF) and DomainKeys Identified Mail (DKIM) are correctly implemented and Domain Name System (DNS) records of the sender's domain are correctly configured, spam filters may still flag messages as malicious and block them based on additional heuristics, likely including factors such as content analysis, sender reputation, and historical interaction patterns. These delivery constraints can interfere with experimental evaluation by preventing emails from correctly reaching the study participants' inboxes, therefore affecting the measurement of phishing effectiveness.

2 | Approach

In this chapter, we present the methodological approach adopted in this work. We provide a high-level overview of the proposed system, introducing the personalized email generation pipeline, the landing page and logging infrastructure, and the on-demand report generation mechanism. We then describe each component in detail, following the sequential flow of the pipeline from email address analysis and online profile discovery to data scraping, identity validation, topic inference, and email generation. The chapter concludes with a description of the security and ethical principles that guided the design of the system and its deployment during the experimental evaluation.

2.1. Approach overview

The proposed approach focuses on a single workflow for generating personalized phishing emails. This workflow, shown in Figure 2.1, integrates a modular email generation pipeline, a delivery module, a landing page and logging infrastructure, and an optional report request mechanism that allows participants to request a detailed analysis of the information used for personalization. For evaluation purposes, the personalized workflow is compared with a traditional phishing baseline, which reuses the same delivery and logging components but differs in content generation and landing page content. We will discuss the baseline configuration in Chapter 4.

The personalized phishing flow (Figure 2.1) relies on a fully automated pipeline that generates spear phishing emails starting from a single email address. As shown in Figure 2.3, the pipeline is structured as a sequence of conceptual modules, each responsible for incrementally enriching the initial input with additional contextual information and finally for the generation of a personalized phishing email. The pipeline begins with an identity inference module that analyzes the email address to infer one or more candidate real-world identities. This is followed by a profile discovery stage, where publicly accessible online profiles associated with the candidate identities are identified and scraped. Then, when multiple candidate identities are available, a validation module evaluates the extracted information for consistency and chooses the most likely candidate identity. Subsequent

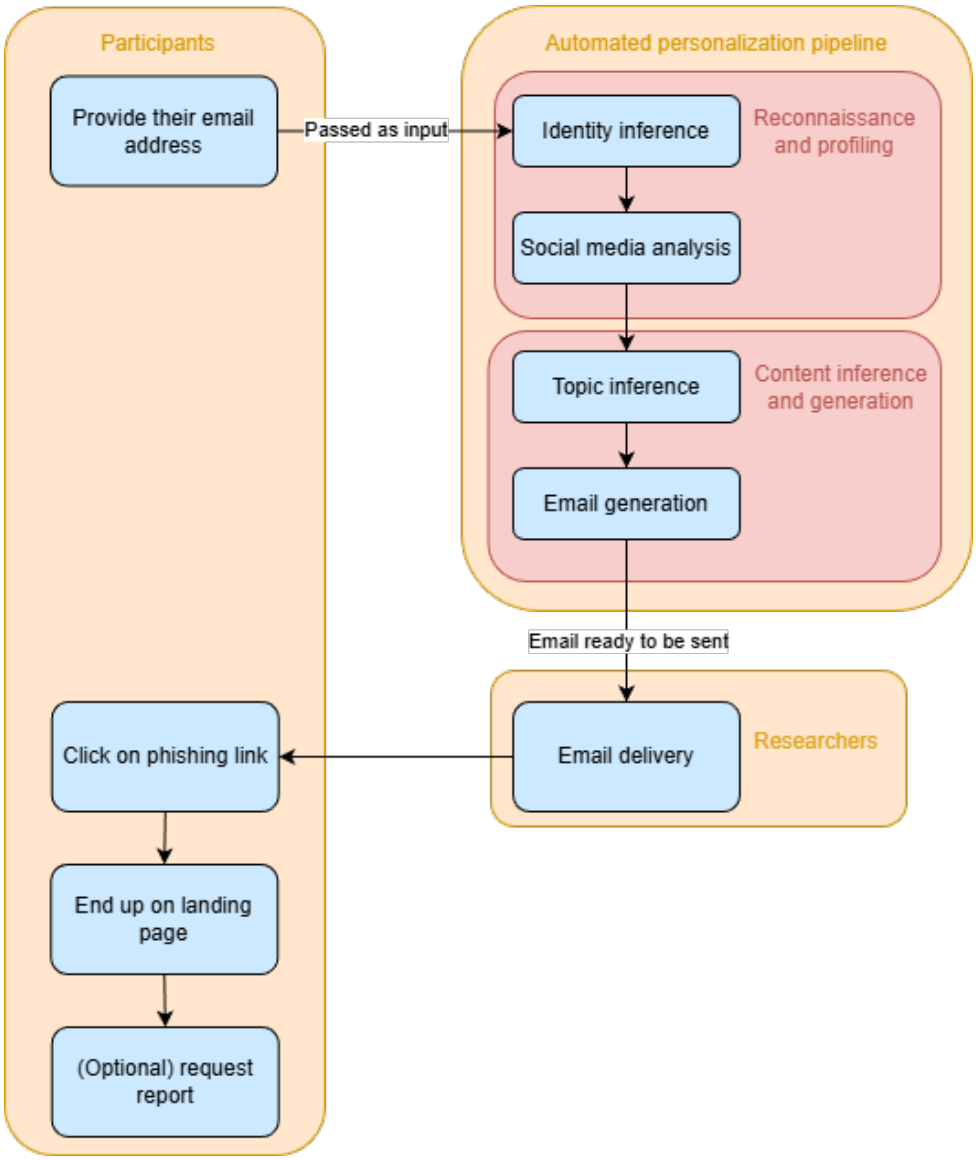

Figure 2.1: Personalized phishing workflow.

modules extract and summarize relevant information from the validated profiles, such as professional context and personal interests, and use this information to infer a topic suitable for targeted communication. The final module of the pipeline uses the aggregated information extracted in the previous stages to generate a personalized phishing email tailored to the inferred interests of the target. The delivery of the generated email is handled in a separate module, independently from the personalized email generation pipeline. After delivery, interaction with the email leads participants to a dedicated landing page, where user actions are recorded for evaluation purposes. An optional stage of the workflow allows participants to request a personalized report detailing the information inferred about them and illustrating how such information can be exploited in targeted

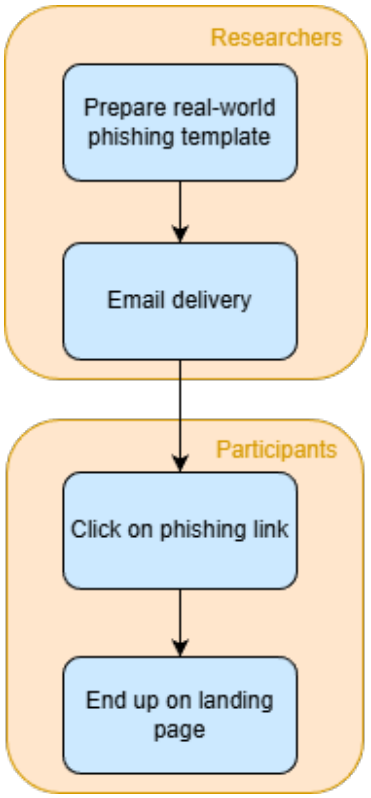

Figure 2.2: Traditional phishing baseline.

social engineering attacks.

We divided participants into two groups, one corresponding to the personalized phishing flow and the other to the traditional phishing flow. Participants assigned to the personalized phishing group were recruited through an online form that outlined the general structure of the experiment without disclosing specific details that could bias behavior. This process also allowed us to collect informed consent for the processing of personal data required by the personalized email generation pipeline. Instead, participants in the control group were provided by Politecnico di Milano's IT department. Since the traditional phishing emails did not rely on the processing of personal data beyond the email address itself, explicit consent was not required for this group.

2.2. Approach details

In this section, we detail the methodology applied to each module and sub-module of the two workflows.

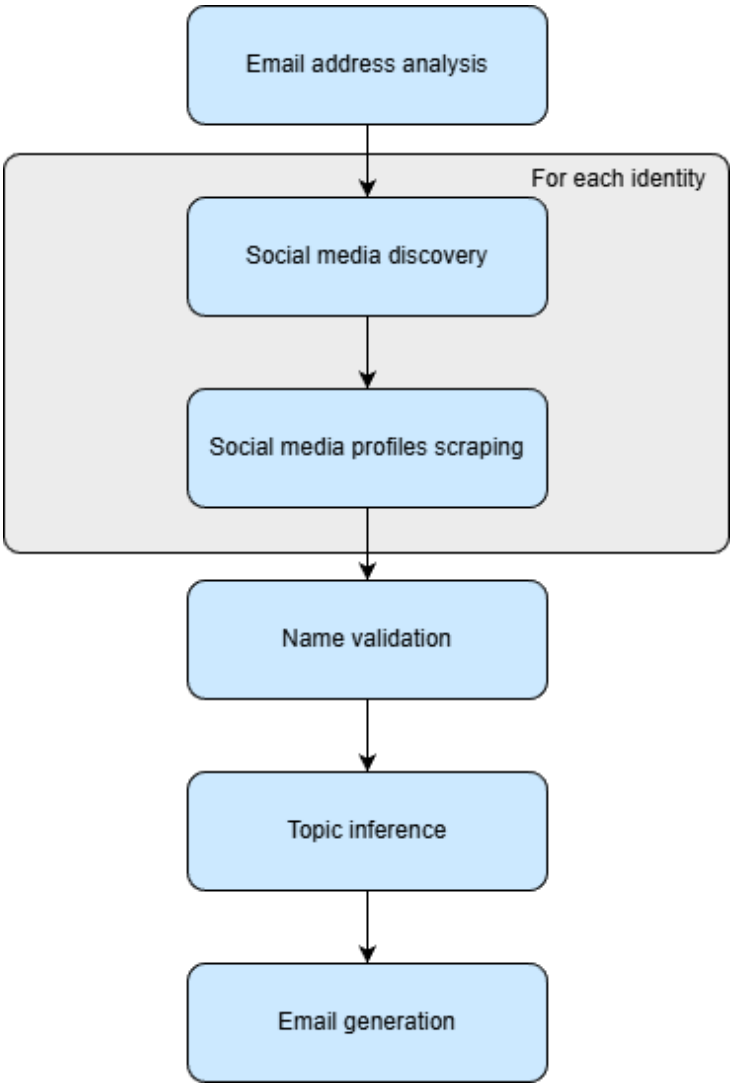

Figure 2.3: Personalized email generation steps.

2.2.1. Fully-automated email generation pipeline

The fully-automated pipeline is responsible for both the reconnaissance and email generation phase. These phases are implemented within a single pipeline composed of multiple sub-modules, each performing a specific task and passing its output to the next stage. These sub-modules are shown in Figure 2.3 and described in detail below.
The reconnaissance phase focuses on extracting as much information as possible starting from the available email address. This process involves inferring the likely identity of the recipient, discovering associated social media profiles through search engine queries, and finally scraping publicly available information from the identified profiles.
The email generation phase then analyzes the extracted data to infer potentially relevant topics or interests and uses this information to generate a personalized phishing email.

Email address analysis

The first step of the pipeline analyzes the input email address to infer one or more candidate names and, when possible, the associated organization or company. A probability score is assigned to each candidate identity, reflecting its likelihood of corresponding to the actual owner of the email address, and therefore directly addressing **C1** (Section 1.3). This analysis relies on common email address patterns, such as combinations of first and last names, initials, or company identifiers. However, the resulting candidates may not correspond to the actual identity of the email address owner if such information cannot be reliably inferred from the email address itself. The most probable candidates are then used as starting points for the online search and profile discovery phase, which is executed for all identities with a probability score higher than a predefined threshold.

Social media discovery

Based on the candidate names inferred from the email address, the pipeline performs an online search to identify potentially matching social media profiles. This sub-module makes multiple queries on a search engine to find profiles that are consistent with the inferred identity and, when available, with the organization or company suggested by the email address. Then, it analyzes the results of the queries and selects the social media profiles that most likely correspond to the considered identity. The output of this step is a set of links to candidate social media profiles that may belong to the target individual.

Social media profiles scraping

Once candidate profiles have been identified, the pipeline extracts publicly accessible information from them. Dedicated scrapers are used to collect profile content from specific social media platforms, such as LinkedIn and Instagram. For platforms for which a scraper is not available, the summary provided by the search engine is used instead. This approach partially addresses **C2** (Section 1.3), although in some cases the data retrieved may still be insufficient to generate a personalized email. The extracted data may include biographical details, professional roles, interests, posts, hobbies, and other publicly shared content. To avoid ethical and privacy related issues, only information that is openly accessible without follow requests is considered.

Name validation

Since the email address analysis may have inferred multiple names, a name validation step is required to select the most plausible identity among the available candidates. This sub-

module evaluates the consistency between the names inferred from the email address and the data extracted from the candidate profiles, considering factors such as name similarity and, when available, professional compatibility. The goal of this step is to choose the profile that most likely corresponds to the intended target, further contributing to the mitigation of **C1** (Section 1.3). However, the outcome of this step may be inaccurate, as the candidate identities may not correspond to the actual owner of the email address, or the publicly information may belong to another person. This limitation is inherently difficult to solve, since different people may share the same name and email addresses may not provide sufficient information to reliably infer the full identity of the target.

Topic inference

After selecting the most plausible identity, the pipeline analyzes the content of the scraped profiles and extracts recurring themes, interests, or professional fields that are relevant to the target and suitable for personalization. Based on this analysis, the pipeline infers a topic to be used to craft a personalized phishing message, a persuasion strategy compatible with the inferred topic and selected according to the principles of influence [4, 10], a credible sender name appropriate to the topic and context, and the language in which the email should be written. If the language cannot be reliably inferred from the available information, English is used as the default. In cases where the availability of publicly available information is limited, as described in **C2** (Section 1.3), the system reverts to generic topics and persuasion strategies, resulting in reduced personalization.

Email generation

Finally, the last step of the pipeline uses the topic, persuasion strategy, and language selected by the previous module to generate a personalized phishing email that is coherent with the inferred interests and professional background of the target and consistent in formality and style with the chosen context. The module is designed to generate a credible email and to embed a natural-sounding call to action to direct the recipient to the phishing landing page, which manages access logging and provides participants with a debriefing message.

2.2.2. Classic phishing email

In addition to personalized phishing emails, the study includes a traditional phishing baseline. These emails are generated using a predefined template derived from a real-world phishing message, with identical content sent to all recipients in the control group. This

email relies on generic persuasive elements, such as a tight deadline and a request from a potential authority, to further persuade participants and encourage them to respond to the call to action. The latter redirects recipients to a landing page for access logging and debriefing, similarly to the personalized emails case. The generic baseline allows us to compare the two approaches in a controlled fashion.

2.2.3. Email delivery

Email delivery is managed through two dedicated scripts operating independently from the generation pipeline. One script automatically sends the personalized phishing emails to all participants in the personalized group, while the other delivers the traditional phishing emails to all participants in the control group. Each script has two tasks: embedding tracking links containing unique alphanumeric identifiers to the email body and delivering the message to the intended recipient. Furthermore, a delivery verification mechanism is implemented to ensure that emails successfully reach the inboxes of the recipients, partially mitigating the delivery constraints described in **C3** (Section 1.3).

2.2.4. Landing page and logging

The landing page is the destination reached by recipients after they click on the phishing URL. Separate landing pages are used for the two experimental groups, as each group is associated with different available functionalities and debriefing messages. Depending on group assignment, the URL embedded in the email redirects recipients to the corresponding landing page.

Both landing pages contain a debriefing message that informs participants about the experiments in which they were involved. The debriefing message for the personalized email group provides additional details about privacy issues and explicitly reassures participants that no personal data was stolen or retained after the completion of the experiment. Furthermore, this landing page offers a mechanism through which participants may request the generation of a report that summarizes the information inferred about them by the pipeline.
The landing page used for the control group contains a more generic debriefing message, limited to reassuring participants that no data were compromised and that the email was part of a simulated phishing scenario.

Information related to user interaction with the landing pages, such as the identifier token, IP address, and user agent of the client, is logged before the redirection for analysis purposes.

2.2.5. On-demand report generation

Participants who received a personalized email and clicked on the phishing link in the email body are given the possibility to request the generation of a report that outlines the results of the pipeline. Report requests are logged similarly to landing page accesses, however only the identifier token and the email address of the requester are logged.

This report presents to the participant that requested it the information that the pipeline inferred about them in an organized fashion. In particular, the report includes the full name, the company affiliation, the profession, the geographical area, the links to social media profiles, a summary of interests and hobbies gathered from the social media profiles available, and an example of a personalized phishing email that could be generated if an attacker wanted to exploits such information.

Since no personal data are retained after the conclusion of the experiment, each report is generated through a new execution of the personalized email generation pipeline. Therefore, the contents of the report may differ from the information used during the experimental phase. Reports are delivered to participants that request them together with a summary of the experimental results and a copy of the complete thesis.

2.3. Security and ethical design

To ensure the security and ethical integrity of our approach, we adopted several precautions throughout the design, implementation, and evaluation of the system. Given the sensitive nature of phishing experiments and the involvement of human participants, we dedicated particular attention to minimizing risks for all parties involved, protecting participants' privacy, and ensuring transparency and informed participation.

Before any interaction with human participants, the personalized email generation pipeline was validated using controlled test inputs. We constructed a dataset of identities from personal websites and blogs available online, which contained email addresses, full names, and links to available social media profiles for each identity. We then executed the pipeline for every identity in the dataset to verify its correct operation. Based on an analysis of the generated outputs, we applied improvements to the pipeline, and executed an additional validation round. This preliminary evaluation allowed us to assess the behavior of the system, verify the correctness of the generated output, and identify and correct potential unintended effects without exposing participants to experimental artifacts.

Participation to the study in the personalized phishing group was voluntary and based on informed consent. Participants explicitly opted in by completing a form distributed via

email by instructors of courses related to the topics of this thesis. Through this form, participants were informed that publicly available personal information could be processed for research purposes and that they might receive a communication generated by the system based such data. Participants were free to provide only the information they were comfortable sharing.
The traditional phishing experiment that was used as baseline was conducted in continuity with the existing anti-phishing training program already deployed by the IT department of Politecnico di Milano. As part of this institutional program, students and staff provide consent to receive simulated phishing emails upon account activation or enrollment. Therefore, no additional consent was required, as participation in the baseline experiment did not involve the processing of new personal data.
The experiments were conducted in accordance with existing organizational policies. Furthermore, the delivery of phishing emails to institutional addresses took place in a controlled setting, ensuring that the study did not interfere with normal operations or cause disruptions within the organization.

The system was deployed in a controlled laboratory environment on a dedicated machine connected to the laboratory network, rather than on our personal computers. This design choice reduced the risk of accidental data leakage and ensured that all experimental data were handled in a secure setting. Intermediate files and collected data during the human-based experiments were retained only for the time strictly necessary to conduct the study and were deleted as soon as they were no longer required. No long-term storage of personal data was performed beyond the duration of the experiment.

The system operates without collecting credentials, passwords, or any other sensitive authentication information from participants. The phishing emails and landing pages were designed exclusively to measure interaction behavior and did not include forms or mechanisms that could result in the harvesting of confidential data. Furthermore, no follow request were sent to private social media profiles in an attempt to access restricted information. Only publicly available data was considered for the purposes of the experiment.

Finally, the personalized reports that outline the data inferred by the pipeline were made available exclusively upon explicit request by participants in the personalized phishing group. The purpose of this report mechanism is to promote transparency and awareness among participants. To avoid long-term retention of personal information, each report was generated on demand with an additional execution of the email generation pipeline and did not rely on previously stored data.

3 | Implementation details

In this chapter, we provide a detailed description of the implementation of the system developed for this thesis. We first present the overall system architecture, describing how different components are orchestrated, how data flows between modules and sub-modules, and how information is stored and managed. Next, we examine in detail the implementation of each component, discussing its purpose, inputs and outputs, and the tools and libraries used. This chapter provides a complete view of the system's implementation, showing how it enables controlled experiments while maintaining the security and ethical standards outlined in Section 2.3.

3.1. System architecture

In this section, we provide an overview of the architecture and implementation choices at the core of our infrastructure. The system is composed of multiple AI-driven agents, an email delivery mechanism, a report generation tool, and a web-based landing page and logging infrastructure.

The core of the system is based on a multi-agent pipeline, orchestrated using LangGraph. As shown in Figure 3.1, LangGraph coordinates the execution of each agent responsible for a specific task in the OSINT-based reconnaissance and email generation processes. Agents are executed synchronously and sequentially, since each stage requires the output produced by the previous steps as input. The pipeline primarily follows a linear structure; however, iterative loops are necessary when multiple candidate identities must be evaluated, allowing the system to process several alternatives before converging to a final output. Moreover, some steps may be unnecessary and therefore skipped, such as when there are no profiles to be scraped or when there is only one candidate identity to explore. Each agent shown in Figure 3.1 is based on the same cloud-based LLM, Gemini 2.5 Flash, accessed through LangChain Application Programming Interfaces (APIs). This model is used consistently across the pipeline to ensure uniform behavior and reduce variability between stages. Agents communicate exclusively through structured prompts and responses and intermediate results are stored in temporary files. These files are deleted as soon as

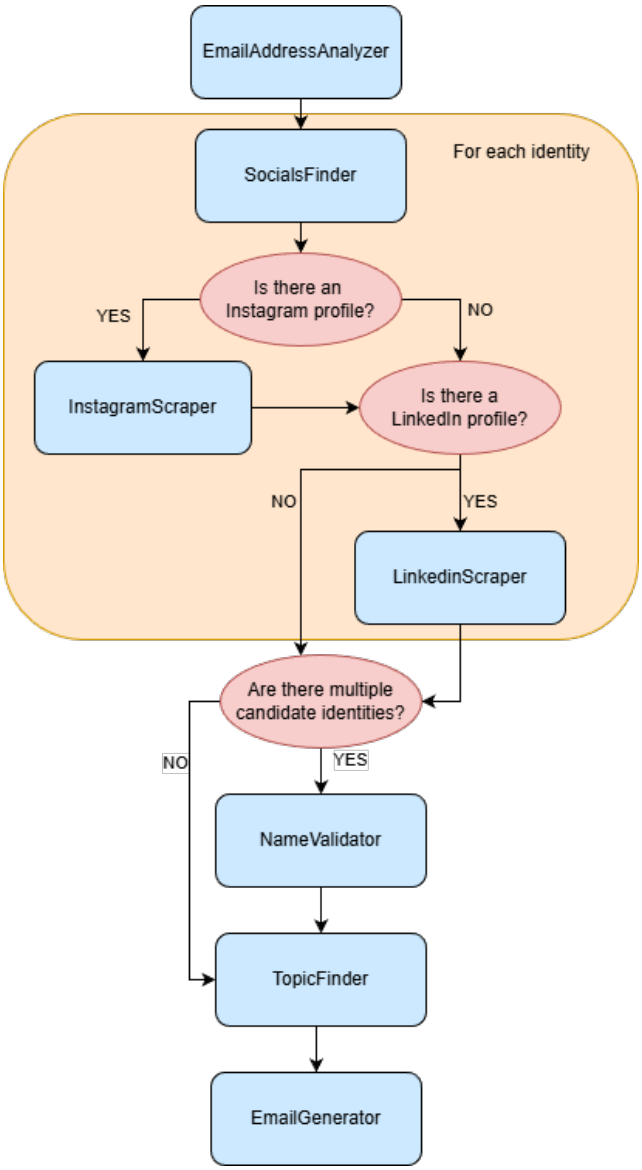

Figure 3.1: Agents' flow for personalized email generation implemented in LangGraph.

they are no longer required, usually at the end of the pipeline execution or after email delivery.

The email delivery mechanism is separate from the email generation pipeline. Once the phishing emails have been generated, a separate delivery script is responsible for sending the message to the intended recipient. This separation allows the same delivery infrastructure to be reused for both personalized and non-personalized phishing emails. We implemented measures to verify email location once delivered and to prevent rate limiting and blocking caused by high email volumes over short time intervals.

The landing page and logging infrastructure are implemented as an external web service,

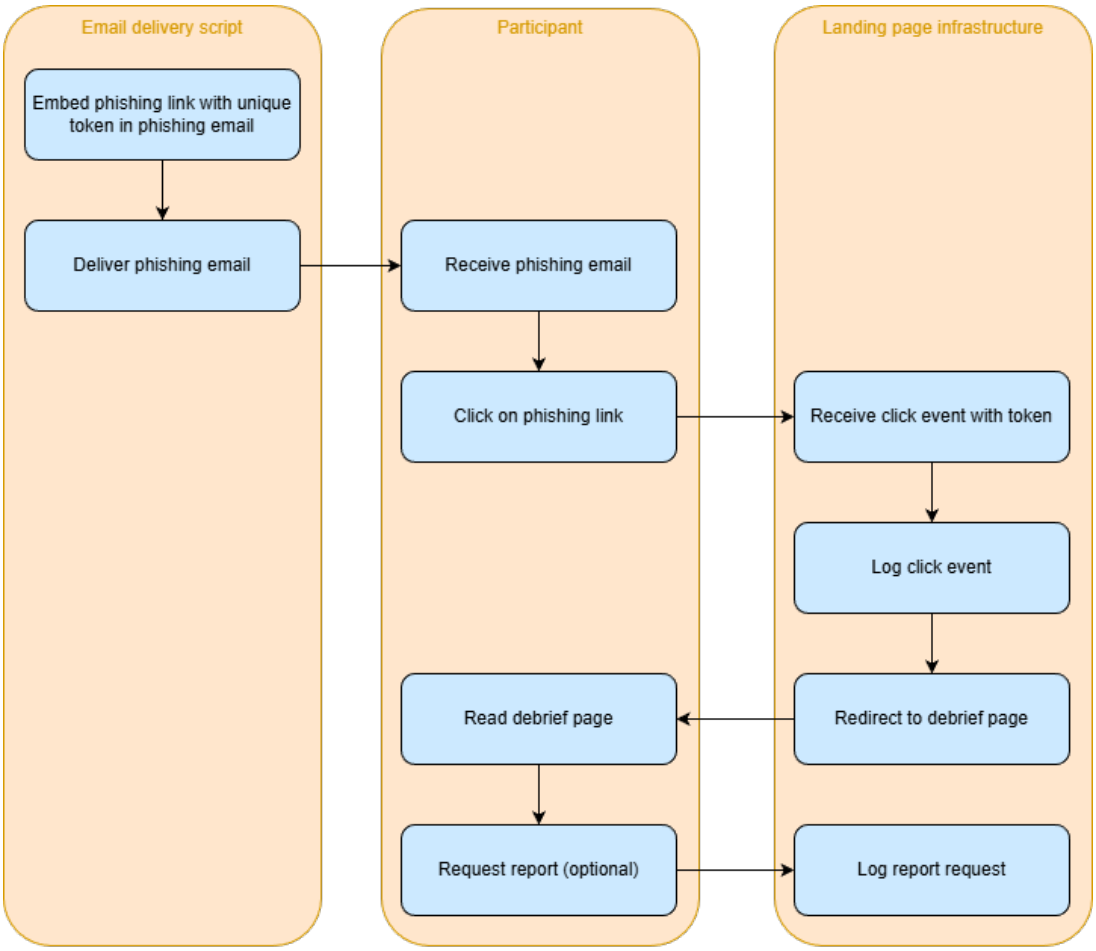

Figure 3.2: Interactions between participants and landing page.

hosted independently from the email generation pipeline. Landing pages are deployed on a third party hosting platform and implemented using a lightweight web server based on PHP. When a recipient clicks on a phishing link, they are redirected to the appropriate landing page, which shows a debriefing message and, for personalized emails, a form to request the report. Access logs, including anonymized identifiers and metadata, are recorded in a Google Sheets document through a PHP-based logging mechanism and are manually processed for analysis. The landing page and its interactions with the rest of the system are illustrated in Figure 3.2.

3.2. System details

In this section, we provide detailed descriptions of each module and sub-module, focusing on implementation specifics.

3.2.1. Personalized email generation pipeline

In this section, we first provide an overview of how the personalized email generation pipeline was implemented, then we describe in detail each AI agent, explaining its purpose, inputs and outputs, any library used, and the tools at its disposal.

The personalized email generation pipeline is implemented as a swarm of AI agents, each responsible for a well-defined task within the reconnaissance or email generation process. Agents are implemented using LangChain and orchestrated through LangGraph, which controls execution order and manages information exchange between components. Each agent is defined in a dedicated source file and instantiated using LangGraph's `create_react_agent()` method, where its role, expected behavior, and output format are specified through a system prompt, together with an optional set of tools. Tools are auxiliary functions exposed to agents to perform specialized operations, and they are explicitly called by the agent as part of its reasoning and execution process. Depending on the task, these tools can rely on deterministic logic or invoke an LLM internally. This modular architecture allows agents to be developed, tested, and refined independently. During execution, agents interact through structured message passing: at each stage, the pipeline uses a user prompt that incorporates the relevant outputs from previous steps, invokes the corresponding agent synchronously, processes its response, and forwards the output to the subsequent agents. This sequential execution model reflects the inherent dependencies between tasks, as each agent relies on the outputs produced in earlier steps to operate correctly.

To ensure reliability and consistency, all agents are instructed to produce a structured output in JavaScript Object Notation (JSON) or predefined text formats. Although the underlying LLM returns plain text responses, explicit constraints at the system prompt level are imposed to enforce a machine-readable structure, and the pipeline validates each output against the expected format before proceeding. If an agent produces a malformed output, the entire pipeline execution is restarted to preserve consistency and avoid the propagation of errors to the following stages. Agents differ in how they leverage LLMs: some rely primarily on their reasoning capabilities, others combine model-based reasoning with external tools, and some use tools almost exclusively while delegating only limited tasks to the model. Model parameters, such as temperature, are fixed at the agent level and tuned to the specific objectives of each agent. Finally, the pipeline incorporates basic failure-handling mechanisms to address errors such as rate limits or quota exhaustion when interacting with external APIs, particularly the LLMs. When such errors occur, an automatic recovery mechanism switches to an alternative API key and restarts the

execution of the pipeline. This approach ensures uninterrupted operation during large-scale experimental runs without requiring manual intervention.

EmailAddressAnalyzer

The `EmailAddressAnalyzer` agent is the first component in the personalized email generation pipeline and is responsible for extracting potential identity information directly from the target email address. Its primary goal is to infer one or more plausible full names associated with the address, which are later used in online searches and personalization steps. When applicable, the agent also attempts to identify the company or organization associated with the email domain, although this is performed conservatively to avoid introducing incorrect assumptions. Through the user prompt, the agent receives as input only the raw email address and operates independently of any additional contextual information.

The agent's model is prompted with a detailed system prompt that encodes linguistic heuristics, name-pattern rules, and constraints aimed at producing high-quality candidate identities. Name inference is based exclusively on the local part of the email address (i.e., the string preceding the `@` symbol) and accounts for a wide range of patterns, including clearly delimited names, reversed name orders, initials, abbreviations, diminutives, and culturally diverse naming conventions. In ambiguous cases, the agent is explicitly instructed to generate multiple plausible full names, assign probabilities that reflect relative likelihood, and ensure gender-balanced distributions when male and female variations of the same root name are possible. With this thorough and detailed system prompt, combined with the possibility of handling multiple candidate identities with assigned probabilities, we address the issue of identity ambiguity described in C1 (Section 1.3). The output is returned as a structured JSON object (shown in Listing 3.1) containing the analyzed email address, a list of inferred full names with associated probabilities, and an optional company field.

Company inference is deliberately restricted to cases where the domain unambiguously corresponds to a well-known organization or institution and only when the agent can determine this with full confidence. Public email providers and personal or ambiguous domains are explicitly excluded, in which case the company field is set to `null`. This conservative strategy reduces noise in other agents that rely on organizational context for topic selection or email personalization, even though it may result in the loss of potentially useful information during the reconnaissance phase. However, since company information can be retrieved from social media profiles, it is only incorporated when it can be inferred with high confidence.

Listing 3.1 JSON structure of EmailAddressAnalyzer's response

```
{
    "agent_name": "EmailAnalyzer",
    "email_address": "{email_address}",
    "full_names": [
        {"name": "Full Name 1", "probability": 0.6},
        {"name": "Full Name 2", "probability": 0.2},
        {"name": "Full Name 3", "probability": 0.2}
    ],
    "company": "Company Name or null"
}
```

SocialProfilesFinder

The `SocialProfilesFinder` agent is responsible for discovering publicly accessible social media profiles associated with a given candidate identity. This agent is executed once for each candidate full name inferred by `EmailAddressAnalyzer`, allowing the pipeline to explore multiple plausible identities in parallel before converging on the most suitable one, further addressing the challenge of identity ambiguity outlined in C1 (Section 1.3). Its primary goal is to identify links to social media profiles belonging to the same individual through Google searches, which are later used to gather contextual information for email personalization. The agent receives as input a single candidate identity at a time, consisting of a full name and an optional company or organization, provided through a structured user prompt.

Rather than performing searches directly, the LLM is restricted to invoking a dedicated tool that automates web searches using a real browser environment based on Selenium. This tool executes multiple Google search queries that combine the target's name and, when available, the associated company, and extracts the most relevant search results. These results are then passed back to the LLM, which analyzes the URLs and the descriptions provided to identify the links corresponding to the social media profiles of the target individual. Additional targeted queries are performed for specific platforms when no profile is initially identified, and the LLM re-evaluates the results to extract relevant links. Rate limits are managed using randomized delays and backoff strategies to prevent blocking. The agent produces a structured JSON output, shown in Listing 3.2, containing, for each supported platform, either a direct link to the identified profile or a `null` value if no reliable match is found. To avoid redundant operations, both the agent's output and the results of the queries are saved to temporary files, which are deleted once the pipeline completes execution.

The disambiguation between individuals sharing the same name remains a fundamental

Listing 3.2 JSON structure of SocialProfilesFinder's response

```
{
    "agent_name": "SocialsFinder",
    "name": "",
    "company": "",
    "instagram_profile": "link_to_the_profile",
    "facebook_profile": "link_to_the_profile",
    "x_profile": "link_to_the_profile",
    "linkedin_profile": "link_to_the_profile",
}
```

challenge and is not explicitly solved by the agent. Instead, `SocialProfilesFinder` attempts to identify profiles that are mutually consistent across platforms and compatible with the company information provided, when available. This limitation reflects the broader challenges of identity inference and ambiguity inherent to OSINT-based approach, as discussed in Section 1.3. When no social media profiles are found, the corresponding fields are set to `null`, and the pipeline falls back to generating a generic phishing email. If multiple plausible profiles are discovered, the agent selects the most likely candidate based on contextual cues available in the search results.

InstagramScraper

The `InstagramScraper` agent is responsible for the extraction of all publicly accessible information from a given Instagram profile and represents one of the most information-rich components of the personalized email generation pipeline. The agent is executed only when `SocialProfilesFinder` has identified a valid Instagram username for the considered candidate identity, otherwise its invocation is skipped entirely. Through a structured user prompt, the agent receives as input a single Instagram username and collects a comprehensive snapshot of the corresponding profile, including both profile metadata and detailed content from recent activity.

The agent is implemented using Selenium and operates through an authenticated Instagram account, allowing access to information that is visible to logged-in users. To protect user privacy, the agent does not send follow requests to either private or public accounts. As a result, both public and private profiles can be processed, although private accounts yield only the information available without following, such as profile metadata and basic statistics. To reduce detection and mitigate automated access restrictions, the scraper incorporates multiple stealth and human-like behaviors, such as randomized delays, scrolling actions, realistic typing patterns, and exponential back-off strategies. Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) detection

is explicitly handled through a retry mechanism, and the entire scraping process is retried multiple times in the event of temporary failures.

Once the profile page is successfully loaded, the scraper extracts the information available in the header: username, displayed name, pronouns, bio, category, external links, link to the Thread profile, highlight names, profile picture, and number of posts, followers and following. Then, it scrapes up to five posts, collecting caption, publication date, location, images, and user interactions such as comments and replies for each post. The images' URLs are passed to another cloud-based LLM to generate a short textual description, which can further enrich the data available.

The output of `InstagramScraper` is returned as a structured textual representation rather than a JSON object. This format is designed to preserve the richness and hierarchical nature of the scraped data while remaining easily consumable by subsequent agents in the pipeline. The resulting text is written to a temporary file and reused if the same profile is requested again, avoiding redundant scraping operations. An example of the structure of the temporary file is shown in Listing 3.3 In case of failures, such as login errors, profile unavailability, or repeated access restriction, the agent retries the operation until a threshold is reached, after which it returns an error. As with the other agents in the system, failures do not interrupt the overall pipeline execution, but, in this case, they may result in reduced personalization, as described by C2 (Section 1.3).

LinkedInScraper

The `LinkedInScraper` agent is designed to extract publicly available information from a LinkedIn profile and is a key component of the personalized email generation pipeline. Similarly to `InstagramScraper`, the agent is executed only when a valid LinkedIn profile is found for the considered identity, otherwise its invocation is skipped. The agent receives as input the LinkedIn profile URL and returns a structured text containing all publicly visible attributes available, such as profile metadata, professional experience, education, activity, certificates, and languages spoken.

The scraper operates without login credentials, leveraging a search-based approach to bypass LinkedIn's authentication walls. Specifically, it performs a Google search for the target profile and accesses the profile through search results, simulating human browsing behavior with scrolling, typing, and randomized delays to reduce the likelihood of detection. Multiple retry attempts with exponential back-off are implemented to handle temporary access issues, including CAPTCHA challenges. Once the profile page is successfully loaded, the HyperText Markup Language (HTML) content is downloaded and

Listing 3.3 Example of the complete structure of `InstagramScraper`'s output file

```
Username:
Posts:
Followers:
Following:
Highlights_names:
Name:
Threads link:
Category:
Bio:
Website Link:
Pronouns:
Profile picture description:
Post 1:
    Location:
    Caption:
    Date:
    Image 1 description:
    Image 2 description:
    Comment 1:
        account_name says:
    Comment 1 replies:
        Reply 1:
            account_name says:
```

stored locally, allowing repeated parsing without additional requests to LinkedIn.

The data available on the profile are scraped using the BeautifulSoup Python library. In particular, the scraper extracts the individual's name, headline, location, the "About" section, the five most recent professional experiences, three items from the education section, publications, languages spoken, certifications, awards, and organizational affiliations. For each section, we implemented error handling procedures to ensure robustness against missing elements.

The information gathered by the scraper is returned as a structured textual representation, preserving the organization of the profile and presenting the data in a way that is easily usable by other agents in the pipeline. The output and the HTML content of the profile are written into temporary files, ready to be reused for other requests for the same profile, thus avoiding redundant executions. An example of the structure of the output file is given in Listing 3.4.

If the profile is not available, restricted, or parsing errors happen, the agent retries the scraping process up to three times. If the failure persists, the agent records it and carries on with the pipeline flow; however, these failures may reduce the richness of the data

Listing 3.4 Example of the complete structure of `LinkedInScraper`'s output file

```
Profile URL:
HTML_File:
Name:
Connections:
Headline:
Location:
About:
Experience: {'title': '', 'company': '', 'duration': '', 'description': ''}, ...
Education: {'school': '', 'degree': '', 'years': ''}, ...
Publications: {'title': '', 'publisher': '', 'content': ''}, ...
Languages: {'language': '', 'proficiency': ''}, ...
Certifications: {'name': '', 'issuer': '', 'date': ''}, ...
Awards: {'title': '', 'issuer': '', 'date', :'', 'description': ''}, ...
Organizations: {'name': '', 'position': '', 'date': '', 'description': ''}, ...
```

available and affect personalization, as outlined in C2 (Section 1.3).

NameValidator

The `NameValidator` agent is responsible for selecting the most plausible real-world identity among multiple candidate names inferred from the target email address. Its role is to resolve ambiguity, and it is executed only when multiple candidate identities are available. If a single candidate identity exists, this validation step is skipped entirely. The agent receives all required information through a structured user prompt. Specifically, the input includes the target email address, the optional company name, the list of candidate full names with associated probabilities, and the complete social media dumps collected for each candidate identity. The output of `NameValidator` is a single validated full name, which is then used for topic extraction and email personalization. The agent is explicitly instructed to return only the validated full name as a plain string, without any additional explanation or metadata.

`NameValidator` relies on a hybrid validation strategy that combines deterministic heuristics with LLM-based semantic reasoning. Rather than making a single evaluation, the agent is constrained by a mandatory multi-step workflow encoded in its system prompt, which enforces consistent and exhaustive evaluation of all candidate identities. This design choice reduces bias, prevents premature convergence, and ensures that all available evidence is systematically considered.

At the core of the validation process is a set of specialized tools invoked by the agent for each candidate identity. First, the `ExtractEmailNameComponents` tool analyzes the local part of the email address to extract potential first and last name components, explicitly

filtering out generic or role-based terms that could otherwise lead to false associations. In parallel, the `ExtractSocialProfileNames` tool processes each social media dump to identify the most likely full name explicitly references in profile metadata, biographies, or descriptive sections.
To quantify the correspondence between the email address and each social profile, the agent invokes the `SemanticSimilarity` tool. This tool performs a structured multi-stage similarity analysis driven by the LLM. The comparison assigns primary importance to the similarity between the email username and the social media username, accounting for common transformations such as separator removal, reordered name components, abbreviations, and numeric suffixes. Secondary factors include alignment between names extracted from the email address and those found in social profiles, as we as supporting evidence from profile content. The result is a normalized score between 0 and 1, reflecting the overall likelihood that the email address and the social profile belong to the same individual.
When a company name is available, the agent additionally invokes the `CompanyPresence` tool for each candidate identity. This tool checks whether the organization name, including minor lexical variations, appears in the social media dump, providing more evidence that the profile is consistent with the inferred organizational context.

After all required tools have been executed for every candidate identity, the agent aggregates the collected signals and selects the identity with the strongest overall coherence. Semantic similarity scores constitute the primary decision criterion, while company presence and initial name probabilities are used as secondary tie-breakers when needed. If none of the candidates exhibits a strong match, the agent defaults to selecting the most probable identity as inferred by `EmailAddressAnalyzer`, ensuring that a valid output is always produced. As imposed by the prompt, the final response consists exclusively of the validated full name, without additional text. This deterministic output format allows for seamless integration with the other agents in the pipeline.

Despite this structured validation process, the selection of an incorrect identity remains possible, directly reflecting the challenges outlined in C1 (Section 1.3). Furthermore, as described in C2, the validation process is constrained by the availability and quality of publicly accessible information: sparse, inconsistent, or private social media profiles may provide insufficient evidence to conclusively disambiguate candidate identities. Consequently, while `NameValidator` significantly reduces ambiguity and improves robustness compared to the inference made by `EmailAddressAnalyzer`, it cannot fully eliminate the risk of incorrect identification inherent to OSINT-based analysis.

`NameValidator` is purposely placed at the end of the reconnaissance and information

gathering process, so that it has all the data at its disposal to accurately perform identity disambiguation. This structural choice improves the reliability of the identity selection process, ensuring that only the most probable identity is forwarded to the personalization phase.

TopicFinder

The `TopicFinder` agent is responsible for identifying a personalized phishing topic and defining the high-level persuasive plan to be used in the generated email. Specifically, the agent selects a single topic, an associated persuasion strategy grounded in Cialdini's principles [4], and the most appropriate language for communication. This agent operates after `NameValidator` has completed identity disambiguation and before `EmailGenerator` generates the email content.

The agent receives its input through a structured user prompt that includes the validated full names of the target individual, the company or organization they are associated with (if available), and the complete social media dumps collected for that identity. These dumps may include detailed Instagram and/or LinkedIn scraping results, as well as supplementary profile descriptions from other platforms such as Facebook or X, extracted from the results of the queries performed by `SocialProfilesFinder`. At this stage, no external tools are used to collect additional data, and the agent relies entirely on LLM-based reasoning over the previously collected information.

The agent selects the topic by analyzing all the information extracted from social media, from which it identifies interests, hobbies, or professional activities about the target. The agent is instructed to prioritize specific details that can be used to start a conversation, such as events attended, travel experiences, or creative projects. The temporal relevance of the topic is also considered in the selection process: a more recent activity is favored against older content, but the latter can still be selected if it is still significant. If the available data is insufficient to select a personalized topic, as described in C2 (Section 1.3), the agent returns a generic topic by default.

In parallel with topic selection, the agent determines an appropriate persuasion strategy by mapping the identified topic and the target's profile to one or two of Cialdini's principles [4]. The chosen strategy must logically align with both the topic and the inferred characteristics of the individual. Explicit constraints prohibit the use of violence, threats, or coercive elements, maintaining alignment with the ethical constraints defined for the system.

Besides the topic and the persuasion strategy, the agents selects the language in which

to write the email by analyzing all the available content in the social media dumps, focusing in particular on the textual cues left in captions, comments, biographies, and the 'Languages' field of the LinkedIn profile. The language selected must be the one in which the target is most active and comfortable. In cases where no other language can be confidently identified or when it is clearly prevalent in the content, English is selected.

`TopicFinder` returns a single JSON object with three fields (shown in Table 4.5), each containing one of the outputs of the reasoning: a detailed description of the selected topic and its relevance to the target, the persuasion strategy to use in the phishing email, and the selected language. The agent's prompt specifies that it must return only the JSON object, without any additional text.

Listing 3.5 JSON structure of TopicFinder's response

```
{
    "topic": "",
    "persuasion_strategy": "",
    "language": ""
}
```

EmailWriter

The `EmailWriter` agent is the final component of the personalized email generation pipeline and is responsible for producing the complete phishing email content. Its task is to transform the high-level strategy defined by `TopicFinder` into a realistic, natural-sounding email that is suitable for delivery while minimizing the likelihood of spam filtering. The agent generates both the subject and the body of the email in a single execution and represents the last processing step before email delivery.

The input is passed to the agent through a structured user prompt, which includes the validated full name of the target, the current date to enable temporally coherent references in the email, and the outputs of `TopicFinder`, which are the selected topic description, the persuasion strategy, and the language for the email. Since the topic is already available, the complete social media data are not passed to the agent to avoid unwanted influences on the email generation process.

`EmailWriter` does not rely on external tools and operates exclusively through LLM-based text generation guided by strict prompt-level instructions. The agent's system prompt encodes a comprehensive set of stylistic, structural, and deliverability rules designed to ensure that the generated emails resemble genuine human correspondence rather than potentially malicious content. The subject line must be concise, contextually appropriate,

and free of characteristics commonly associated with phishing, such as excessive punctuation, capitalization, or trigger words. The email body is written in plain text, kept under a fixed length threshold, and structured to resemble a natural outreach message. The opening establishes a legitimate context before any request or invitation is introduced, and sentence length and rhythm are varied to avoid mechanical patterns. The tone is adapted dynamically based on the topic type (e.g., professional, hobby-related, educational) and remains consistent throughout the message. The agent is instructed to include exactly one call to action, achieved by inserting a hyperlink as a natural continuation of the message. The system prompt also tells the agent to avoid urgent language and explicit instructions to force the recipient to click on the link. The email signature is composed of multiple lines that contain the name of the sender and optional background information, such as role or area of expertise. This helps to give credibility to the sender, without depending on explicit claims of authority. The language selected by `TopicFinder` is used throughout the whole email, including the subject and signature. To avoid weakening the credibility of the message with wrong temporal references, the agent is instructed to use generic terms, such as "recently" or "lately". Moreover, the agent avoids to mention how the information was obtained or their presence on social media, unless it is relevant for the conversation.

The output of `EmailWriter` is a single JSON object containing two fields, shown in Listing 3.6: the email subject and the email body. The agent is told to return only a valid JSON, with no additional comments or metadata.

Listing 3.6 JSON structure of EmailWriter's response

```
{
  "email_subject": "",
  "email_body": ""
}
```

3.2.2. Email delivery module

The email delivery module is responsible for transmitting the generated emails to their intended recipients in a controlled and reproducible manner. This module is intentionally separated from the email generation pipeline, allowing message creation and delivery to be treated as independent stages. This separation enables reuse of the same delivery infrastructure across different experimental conditions and simplifies debugging, validation, and rate control.

We send the phishing emails from a single sender address hosted on a custom domain and associated with Google Workspace. This solution allows us to mitigate the email

delivery constraints described in C3 (Section 1.3) since Google Workspace manages the DNS records and other configurations that might affect reliable message delivery. The emails are sent through Gmail APIs and authentication to the mailbox is handled through Google's OAuth 2.0 mechanism. The first time it is executed, the script initiates a local authentication flow providing a login window to enter the credentials, then stores the resulting token locally. Successive executions reuse the access token and refresh it when expired, avoiding repeated authentication at each launch of the script. Once authenticated, the script iterates over a directory that stores JSON files, each containing the emails to be sent to the participants. The messages are built locally from these files following the Gmail APIs requirements, using standard Multipurpose Internet Mail Extensions (MIME) formatting and later encoding them in base64. The email bodies generated by `EmailWriter` are converted from plain text to HTML, mainly to preserve the hyperlink formatting. We also implemented a mechanism to monitor deliverability and inbox placement: we included an institutional email address in our control in the Blind Carbon Copy (BCC) field of every outgoing message. In this way, we were able to record whether the emails are delivered to the inbox, the spam folder, or blocked entirely. To reduce the likelihood of emails being blocked because of automation detection or rate limits, we introduced a delay of 30 seconds after each email delivery.

3.2.3. Landing page and logging

The landing page and logging module serves two complementary purposes: recording participant interactions with the phishing emails, and providing a transparent debriefing interface after to participants that click on the link.

When a recipient clicks on the link embedded in an email, they are redirected to a dedicated landing page hosted on a website created specifically for this experiment. The URL contains a unique, randomly generated token associated with the recipient, which is used to identify the interaction without relying on explicit personal identifiers. Upon access, the landing page immediately informs the participant that they have clicked on a simulated phishing link and provides a clear summary explaining the research context, institutional affiliation, and purpose of the study. The page emphasizes that no harm has occurred and that the experiment was conducted exclusively for research purposes. Beyond the debriefing content, the page includes a form that allows participants to request a personalized report, which details the data used and the generation process applied in their specific case. This request form is positioned at the bottom of the page to avoid interference with the initial debriefing. Figure 3.3 shows the landing page presented to participants in the personalized phishing email study.

The landing page is implemented entirely in PHP with minimal client-side scripting and no external tracking libraries. It is hosted on a dedicated domain under `altervista.org`, created solely for the purpose of this experiment. The page content is rendered dynamically after the token contained in the URL is validated server-side. This validation step ensures that only tokens generated as part of the experiment are considered valid, while still allowing the page to be displayed even in the presence of invalid or malformed tokens.

We log two types of participant interaction: landing page accesses (i.e., clicks on the link contained in the email) and report requests. In the first case, the system logs a timestamp, the identifier token contained in the URL, the client IP address, and the user agent before the recipient is redirected to the landing page. For report requests submitted through the form, instead, the system records the timestamp, the identifier token, and the submitted email address, then a validation check is executed to ensure that the email address matched the participant associated with the registered token.

All logs are stored using the Google Sheets APIs, with separate sheets for click events, report requests, and valid tokens. This choice provides a lightweight logging backend without requiring a dedicated database server. No client-side logging is performed and all recorded data is generated server-side at the time of interaction.

To comply with the ethical constraints of this study, we delete the logging data as soon as the experiment analysis and evaluation phases are complete. This ensures that the data collected are retained only for the time necessary for our research.

3.2.4. Report generation module

The report generation module provides participants who request it with a detailed, personalized summary of the information that can be inferred about them from a minimal initial input, i.e., their email address. Its primary purpose is educational: to illustrate how publicly available data can be aggregated and leveraged to build a convincing personal profile, and to raise awareness of the privacy and security risks associated with digital footprints.

As for the email generation pipeline, the only external input to the report generation process is the email address voluntarily provided by the participant, first through the study participation form and then through the form on the landing page. Report requests are logged in real time during the experiment, but we manually start the report generation process after the request is received. For each request, the full email generation pipeline is re-executed specifically for the submitted email address. Any intermediate data generated during the original experiment is not reused, as all pipeline outputs are deleted once they

are no longer required. Internally, the report generator invokes all modules of the pipeline, ensuring that the report reflects the same process and capabilities demonstrated during the experiment.

Each generated report contains a structured overview of the information that could be reconstructed from the participant's email address and associated public online presence. The report includes the validated personal name inferred by the pipeline, an inferred company or organization, extracted either directly during email analysis or, when unavailable, via an additional LLM-based analysis of social media content, a list of social media profiles discovered by the pipeline, a natural-language summary of publicly available information extracted from these profiles, a phishing topic inferred from the participant's public content, and an example of a personalized phishing email illustrating how the extracted information might be exploited. Each section of the report is accompanied by a short description explaining how the corresponding information was obtained and why it is relevant from a social engineering perspective. Reports are automatically generated by rendering a template written in HTML and Cascading Style Sheets (CSS) using the Jinja engine, which ensures consistent structure and presentation and allows dynamic insertion of pipeline outputs.

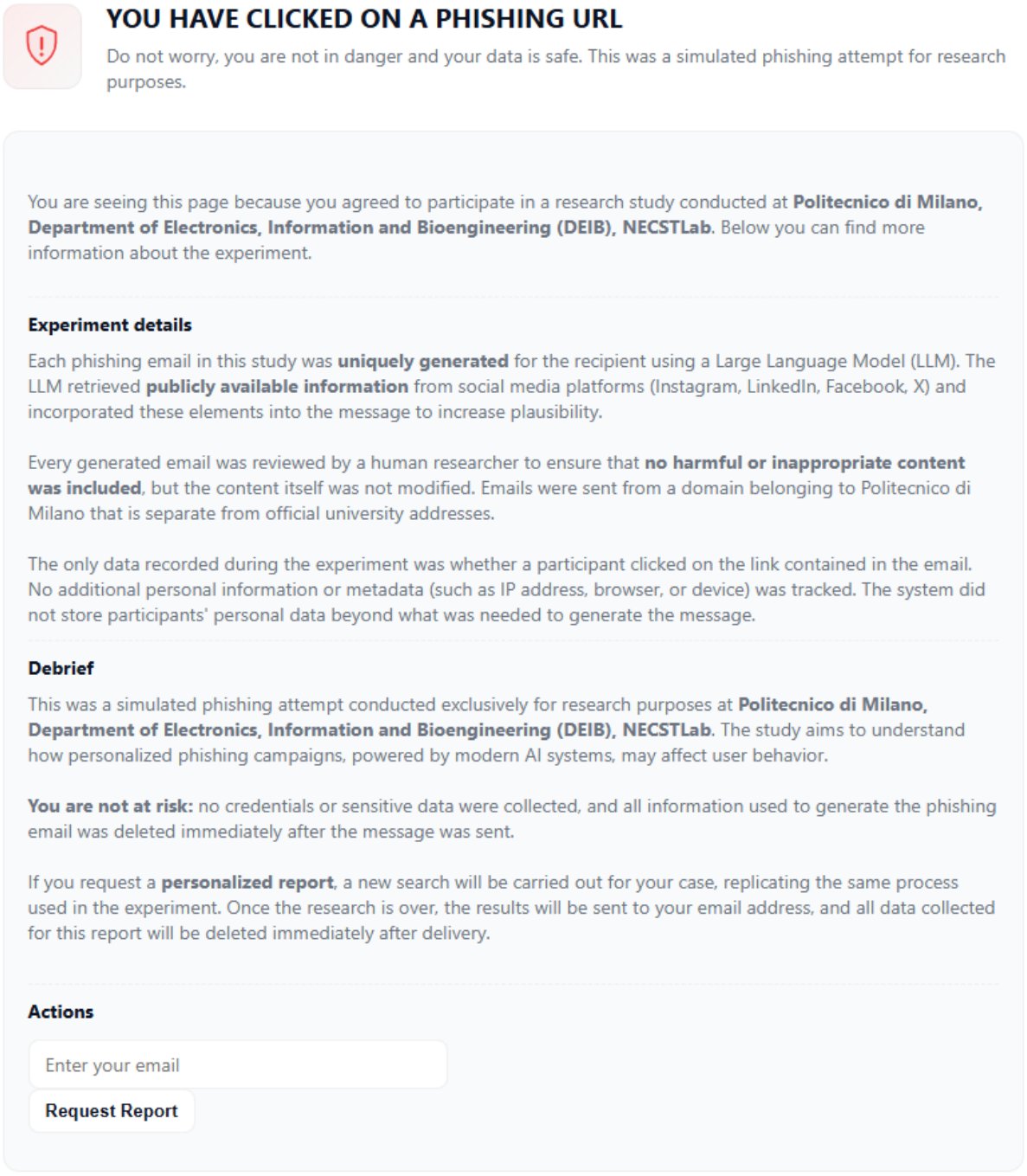

Figure 3.3: Screenshot of the landing page, showing the debriefing message and the report request form.

4 | Experimental validation

In this chapter, we describe the experimental validation of our approach to phishing attacks, focusing on the comparison between LLM-based phishing and traditional phishing methods. We begin by highlighting the goals of the experiments, which include measuring the effectiveness of LLM-based phishing, analyzing how personalization factors impact click rates, and validating the accuracy of each agent of the pipeline. Then, we introduce the datasets used, including the Enron email dataset, a curated collection from personal websites, and data gathered from human participants. Finally, we present the experiments, discuss the criteria for logging and collecting data, report the results, and provide interpretations.

4.1. Goals

In the evaluation phase of our study, we conducted a series of experiments with different goals. These experiments are designed to assess the effectiveness of a fully automated spear phishing pipeline based on LLM agents and to quantify the impact of personalization on recipients' responses to phishing emails.

First, we validate the correctness and reliability of the individual agents that make up the pipeline before employing them in experiments involving human subjects. Each agent is evaluated independently, with a primary focus on accuracy and robustness, using both datasets available from prior research and datasets specifically constructed for this study. This validation step is necessary to ensure that the automated email generation pipeline behaves as intended and to identify potential weaknesses in the pipeline. A key challenge at this stage is the limited availability of ground truth data that includes social media profiles of real individuals, which required the creation of a custom dataset, which we obtained by extracting the information we needed from personal websites and blogs. Furthermore, several aspects of the agents' outputs required manual assessment, particularly for inherently subjective tasks such as evaluating topic relevance or judging the overall quality and plausibility of the generated emails.

Then, we measure the effectiveness of LLM-based spear phishing emails in comparison with traditional, non-personalized phishing emails. This comparison is carried out through a human-subject experiment in which the results of the automated email generation pipeline are compared with a baseline consisting of a real-world generic phishing message. We measured effectiveness primarily using the CTR, defined as the fraction of recipients who interact with the phishing link, while email replies are recorded as an additional indicator of engagement. For this experiment, participants were recruited through a form that also served to collect informed consent for the processing of personal data required by the pipeline. This experiment provides empirical evidence of whether fully automated personalization yields a concrete advantage over conventional phishing techniques.

Finally, we analyze the impact of specific personalization factors on phishing success. Rather than treating personalization as a monolithic property, we investigate the contribution of individual elements to user behavior. In particular, we focus on the correctness of the recipient name used in the email and on the relevance of the selected topic with respect to the recipient's inferred interests. Through subgroup analyses based on these factors, we plan to isolate the contribution of each individual personalization element on CTR and user engagement, providing insight into which aspects of personalization are the most influential.

4.2. Datasets

This section describes the datasets used throughout the experiments. For each dataset, we detail its source, structure, collection process, size, and storage format.

4.2.1. Enron email dataset

The Enron email dataset [11] was used to validate the `EmailAddressAnalyzer` agent, whose goal is to infer a set of candidate identities associated with a given email address. The dataset was obtained from the official Carnegie Mellon University repository and consists of a large collection of real-world corporate emails exchanges by Enron employees.

We developed a Python script to scan the full dataset and extract email addresses and associated full names from the email headers, in particular from the `From`, `To`, `Cc`, and `Bcc` fields. We used regular expressions to identify valid email addresses and extract the corresponding display names when present. Then, we applied a cleaning step to remove malformed entries, normalize name formats, and eliminate duplicates. We extracted a total of 26335 entries, of which 1821 belonged to external domains (i.e., not `@enron.com`).

For testing purposes, we considered 2000 randomly selected Enron email addresses, which we extracted using a fixed seed for reproducibility, and the entire subset of 1821 external email addresses. The external email addresses are particularly relevant for evaluating the agent's ability to infer identities beyond a closed corporate environment. Each record in the final dataset, stored in Comma-Separated Values (CSV) format, consists of two fields: the email address and the associated full name, which serves as ground truth.

4.2.2. Dataset from personal websites and blogs

To validate the remaining agents and the pipeline as a whole, we collected a dataset based on information gathered from personal websites and blog available on the Internet. This dataset was built manually due to the lack of publicly available datasets containing both contact information and linked social media profiles of existing individuals.

We identified personal websites using targeted search queries and Google dorking techniques, focusing on domains commonly associated with personal pages (e.g., websites ending in `.me`) and keywords such as "personal", "blog", or similar terms. Only publicly accessible information was collected. The final dataset contains 200 entries, each corresponding to a distinct individual. Of these, 152 entries include the email address. The dataset is also rich of social media information: 90% of the entries contain a LinkedIn profile, 76.5% an Instagram profile, and 69% an X profile. Other platforms such as GitHub (57%), Facebook (35%), and YouTube (14.5%) are also represented, with smaller fractions linking to Threads, TikTok, Tumblr, Pinterest, and Reddit. Each entry in the dataset is represented as a single row in a CSV file with the following columns: `name`, `company`, `emailAddress`, `instgram`, `linkedin`, `x`, `facebook`, and `others`. The fields populated for each entry depend on the information publicly available on the individual's website.

This dataset serves as ground truth for validating the agents in the email generation pipeline. The presence and correctness of extracted names, companies and social media links can be directly verified against the manually collected data. Other more subjective values must be evaluated manually, inferring the ground truth from the social media profiles available.

4.2.3. Human-subject participant dataset for personalized phishing emails

A dataset of human participants was collected to evaluate the effectiveness of personalized phishing emails generated by the pipeline. Participants were recruited through a form

distributed by professors teaching courses related to cybersecurity and machine learning.

A total of 28 students participated, all of whom received the personalized phishing email. The participants are primarily computer science and engineering students and are assumed to have a technical background, although no information on this was collected. The form was used both for recruitment and to gather consent for the collection and processing of personal data. The collected information includes the participant's email address, full name, and optional links to social media profiles. Among the participants, Instagram was the most common platform (22 out of 28 participants), followed by LinkedIn (18 out of 28), Facebook, (9 out of 28) and X (4 out of 28).

4.2.4. Human-subject participant dataset for traditional phishing emails

To evaluate the baseline scenario involving generic phishing emails, Politecnico di Milano's IT department provided a second dataset of human subject. Such generic phishing campaigns are periodically conducted by the institution as part of security awareness initiatives, and the baseline scenario was part of these.

The dataset consists of 28 participants, all of whom received the generic phishing email. The participants are engineering students, though not necessarily from computer science related programs. No explicit or implicit information about their level of phishing awareness was available.

4.2.5. Data handling and storage

Data used for agent validation experiments were stored on a personal machine, while data related to human-subject experiments were stored on a dedicated physical machine located in the university laboratory. All datasets were anonymized before analysis and presentation of results, ensuring that no personally identifiable information was retained in the reported findings. After the conclusion of the study, all data related to human participants will be permanently deleted, in line with ethical guidelines and data minimization principles.

4.3. Experiments

In this section, we present the experiments conducted to evaluate the proposed approach and the execution environment in which they were performed. We first focus on the

validation of the individual agents composing the pipeline and the assessment of their performance. We then examine the effectiveness of personalized LLM-based phishing emails compared to traditional phishing emails through human-subject experiments. Finally, we analyze the impact of specific personalization factors on user behavior and CTRs by performing subgroup analyses.

4.3.1. Pipeline execution environment

All experiments were conducted using Python 3.11.0. The phishing pipeline, auxiliary tools, and validation scripts were implemented as standalone Python programs executed on generic, off-the-shelf hardware. The physical execution environment differed depending on the type of experiment. Agents validation experiments were executed on a local machine, while pipeline execution and email delivery for the human-subject phishing were performed on a dedicated physical machine located in the university laboratory.

The implementation relies on a set of libraries that support LLM orchestration, web scraping, email delivery, and data handling. In particular, LangChain and LangGraph were used to implement and coordinate the multi-agent pipeline, while LangGraph Swarm and other prebuilt LangGraph components were used to simplify agents interactions. Access to LLMs was provided through the `langchain-google-genai` and `google-genai` libraries.
Web data collection relied on `requests`, `beautifulsoup4`, and `selenium`, email content was rendered with `markdown`, and `jinja2` was used for report templates. The landing pages and logging backend were implemented in PHP. Interactions with Google services and authentication were handled through `google-api-python-client`, `google-auth`, and `google-auth-oauthlib`.
All LLM-based components of the email generation pipeline and the report generation tool were powered by the Gemini 2.5 Flash model, accessed through API keys. The same model version and configuration were used across all experiments to ensure consistency. Agent behavior was controlled through fixed system and user prompts, described in detail in Section 3.2.1.

4.3.2. Agents validation

Before conducting experiments involving human subjects, we validated the accuracy and reliability of the individual agents that compose the automated phishing pipeline. The goal of this validation phase is twofold: to assess whether each agent performs its intended task accurately when compared to ground truth data, and to identify systematic errors

that could propagate through the pipeline and negatively affect other components. Each agent was evaluated independently through offline experiments using task-specific metrics tailored to its functionality. Depending on the agent, evaluation relied either on automatic comparison against ground truth (e.g., name matching or identification of social media profiles) or on manual inspection for inherently subjective tasks, such as topic relevance or email quality.

The validation experiments were conducted using a dedicated evaluation pipeline designed to test each agent in isolation while providing them, when necessary, with the inputs generated during previous validation steps of other agents. This approach ensured realistic operating conditions without evaluating the full pipeline end-to-end. Experiment were performed on the Enron email dataset [11] and on the dataset built from personal websites and blogs. The `EmailAddressAnalyzer` agent was tested on both datasets, whereas all other agents were evaluated exclusively on the latter. The `LinkedInScraper` agent was not included in this validation phase, as it was implemented after the completion of the validation experiments.

Validation of EmailAddressAnalyzer

The `EmailAddressAnalyzer` agent is responsible for inferring candidate identities and, when possible, an associated company from a given email address. The agent outputs a ranked list of candidate full names and a candidate company name. A prediction is considered correct if the ground truth name appears anywhere in the returned list (Top-N accuracy); additionally, we report whether the correct name appears as the first candidate (Top-1 accuracy). Company inference was evaluated only on the dataset containing information extracted from personal websites and blogs, where ground truth information is available.

The agent was evaluated on a subset of the Enron dataset [11] and on a dataset composed of information extracted from personal websites and blogs. The Enron dataset [11] provides reliable ground truth for both internal and external email addresses, allowing evaluation in a controlled corporate setting as well as in a more heterogeneous context. The second dataset complements this analysis by focusing on personal domains and non-corporate email addresses, which are more representative of real-world spear phishing targets.

The results, shown in Table 4.1, reveal a strong dependency between the accuracy of inference and the structure of the email address. For internal Enron addresses, which follow relatively consistent naming conventions, the agent achieves high results, exceeding 90%

Metric	Results
Internal addresses (Enron, 2000)	
Correct name (Top-N)	**90.70%**
Correct name (Top-1)	79.60%
Incorrect	9.30%
External addresses (Enron, 1821)	
Correct name (Top-N)	**46.68%**
Correct name (Top-1)	35.26%
Incorrect	53.32%
Dataset from personal websites and blogs (152)	
Name match (Top-N)	**55.92%**
Company match	0.00%

Table 4.1: EmailAddressAnalyzer validation results

Top-N accuracy. In contrast, performance on external Enron addresses is significantly lower: these addresses exhibit much greater heterogeneity and ambiguity, often containing partial names, initials, or arbitrary identifiers that do not map cleanly to real identities. Additionally, results on the dataset built from personal websites and blogs are slightly better than the ones obtained by external Enron addresses.
Qualitative analysis showed that incorrect attribution of the company name, often caused by confusing the personal domains of the email address with the organization name, leads to errors in successive agents, especially in the topic selection process, and thus in less credible emails. Therefore, company prediction was intentionally limited to high confidence cases, since the analysis indicated that, in phishing scenarios, credibility and coherence with the target's profile are more important than completeness.

Results on the dataset built from personal websites and blogs are slightly better than the ones obtained with external addresses from the Enron dataset [11]. Qualitative analysis revealed that incorrect company attribution, often caused by confusing personal domains with organizational ones, frequently misled downstream agents and resulted in less credible phishing emails. As a result, company prediction was intentionally restricted to cases of high confidence. This design choice reflects a broader principle emerging from the validation: in phishing scenarios, plausibility and consistency are more critical than completeness.

Most errors in these settings are attributable to systematic ambiguities such as inverted name orderings, the presence of nicknames, or non-person identifiers embedded in the address. The system prompt of the agent explicitly addresses these ambiguities and suggests strategies to deal with them, such as generating multiple name orderings, explicitly

modeling nicknames, increasing candidate diversity when only initials are available and excluding generic prefixes, therefore substantially reducing errors.

The validation demonstrates that `EmailAddressAnalyzer` performs reliably in structured corporate environments, while performance remains inherently limited on external and personal email addresses, where the information available is often insufficient or ambiguous, as described in C1 (Section 1.3).

Validation of SocialsFinder

The `SocialsFinder` agent identifies social media profiles associated with a target individual across Instagram, LinkedIn, Facebook, and X. For each platform, outcomes were classified into one of the following categories: correctly identified profile, profile not identified when present in the ground truth, profile identified when not present in the ground truth, wrongly identified profile, and correctly not identified profile when not present in the ground truth. This granular categorization allows the analysis of both false negatives and false positives, which are particularly important in phishing scenarios where incorrect attribution of interests can reduce credibility.

Table 4.2 reports the validation results for `SocialsFinder` across all supported social media platforms. LinkedIn achieves the highest proportion of correctly identified profiles, combined with relatively low rates of incorrect identifications, indicating a strong performance. Instagram and X show moderate correct identification rates, but both exhibit non-negligible levels of wrongly identified profiles and missed detections. Faceboook presents the most challenging scenario: while correctly not identifying absent profiles in a third of the cases, it also shows a high rate of identifying profiles when none exist in the ground truth and a comparatively low correct identification rate. The data suggest that `SocialsFinder` performs more reliably on professional-oriented platforms such as LinkedIn, whereas platforms with more ambiguous or common user identifiers lead to higher misclassification rates.

Validation of InstagramScraper

The goal of the `InstagramScraper` agent is to extract all publicly available information from an Instagram profile. Since this agent does not have any reasoning tasks, its validation focuses on the availability of profile data, on the robustness of the scraper, and on its operational efficiency. In particular, we focused the evaluation on the presence of profile elements in the output of the agent.

Table 4.3 reports the validation results for `InstagramScraper`, which demonstrate high ro-

Metric	Results
Instagram	
Correctly identified	**35.38%**
Not identified when present	13.36%
Identified when not present	11.91%
Wrongly identified	26.35%
Not identified when not present (Correct)	**13.00%**
LinkedIn	
Correctly identified	**56.68%**
Not identified when present	20.94%
Identified when not present	3.61%
Wrongly identified	11.19%
Not identified when not present (Correct)	**7.58%**
X	
Correctly identified	**33.57%**
Not identified when present	22.74%
Identified when not present	10.83%
Wrongly identified	10.83%
Not identified when not present (Correct)	**22.02%**
Facebook	
Correctly identified	**11.55%**
Not identified when present	7.94%
Identified when not present	32.85%
Wrongly identified	14.80%
Not identified when not present (Correct)	**32.85%**

Table 4.2: SocialsFinder validation results

bustness in extracting informative profile and content attributes. Profiles are successfully scraped in 96.28% of cases, indicating stable and reliable execution. Core identity elements show very high availability, with the display name present in 97.02% of the scraped profiles, bio information available in 76.60%, and profile picture descriptions generated in 98.51% of cases. At the content level, an average of 4.58 posts is collected per profile, with captions available for 78.90% of posts. Posts contain on average 1.40 images, and image descriptions are successfully generated in 96.82% of cases, ensuring broad coverage od visual information. These data confirm that `InstagramScraper` reliably retrieves the key textual and visual elements necessary for personalization tasks.

Validation of NameValidator

When multiple candidate identities are available, the `NameValidator` agent analyzes the social media dumps gathered by previous agents and selects the most plausible identity to

Metric	Results
Profiles successfully scraped	**96.28%**
Name available	97.02%
Bio available	**75.60%**
Profile picture description availability	**98.51%**
Private profiles	30.06%
Threads link available	44.05%
Website URL available	57.14%
Average posts per profile	**4.58**
Caption of post available	78.90%
Average number of images per post	1.40
Image description availability	**96.82%**

Table 4.3: InstagramScraper validation results

be used for topic selection and email generation. This disambiguation step takes advantage of the data available to the pipeline to make an informed decision about the identity of the target.

Metric	Results
Correctly identified names	**50.66%**
Wrongly identified names	49.34%

Table 4.4: NameValidator validation results

Table 4.4 summarizes the validation results for `NameValidator`. A prediction is considered correct if the selected name matches the ground truth available in the dataset. The agent correctly identifies the target name in 50.66% of cases, while 49.34% of predictions result in an incorrect selection. These results indicate a balanced but still challenging task, as the agent must choose among multiple candidate names inferred from the email address. Given the direct impact of name selection on the credibility of the generated email, the near-equal distribution between correct and incorrect predictions highlights both the progress achieved and the remaining margin for improvement in candidate disambiguation and selection strategies.

Validation of TopicFinder

The `TopicFinder` agent infers a topic to be used as the main theme of the phishing email and the language to be used, based on the information extracted from the target's social media presence. Topic relevance was evaluated manually using a binary criterion:

a topic is considered relevant if it can be directly associated with the target individual and is present in at least one of the available social media profiles. In addition to topic relevance, we evaluated whether the agent correctly identified the language in which the email should be written based on information and texts available on the social media profiles. All samples were manually reviewed by a single evaluator to ensure consistency, as topic relevance is inherently subjective and difficult to automate reliably.

Metric	Results
Relevant topic (predicted profile)	85.50%
Relevant topic (ground truth profile)	**50.50%**
Language correctly predicted	96.50%

Table 4.5: TopicFinder validation results

Table 4.5 shows the validation results for `TopicFinder`. When evaluated against the predicted profiles, the agent generates a relevant topic in 85.50% of the cases,indicating that the inferred themes are largely consistent with the information available from the profiles identified by the pipeline. The remaining 14.50% of cases in which the inferred topic is not relevant to the predicted profile correspond to those instances where insufficient information is available from social media profiles to support reliable inference. Language detection achieves 96.50% accuracy, showing that the agent reliably infers the appropriate language for the email based on textual signals extracted form social media content.
When evaluation is performed against the ground truth profiles, topic relevance decreases to 50.50%. The gap between the two relevance scores highlights the impact of profile identification errors on topic inference. Since topic generation depends directly on the information collected from previously selected profiles, inaccuracies in earlier stages propagate to this component. Nevertheless, the results indicate that, given the available input data, the agent is generally able to infer content and contextually relevant themes.

Validation of EmailWriter

The last agent of the email generation pipeline, `EmailGenerator`, is responsible for the generation of the final phishing email and its subject, using the full names, topic, persuasion strategy, and language identified by previous agents. The output was evaluated qualitatively, by focusing on attributes that affect email credibility and effectiveness, such as the absence of grammatical and spelling errors, the correct use of the inferred recipient's name in the email body, and coherence with the selected topic. These criteria are subjective but crucial to assess the quality of the generated emails for phishing attempts.

Metric	Results
Email well written	**93.50%**
Correct target name used	78.50%

Table 4.6: EmailGenerator validation results

Table 4.6 summarizes the evaluation of `EmailGenerator`. A total of 93.50% of the generated emails are judged as well written, indicating a high level of grammatical correctness, coherence and overall readability. This suggests that the agent consistently produces messages that meet a minimum quality threshold for plausibility in phishing scenarios. The correct target name is used in 78.50% of the emails. Since name selection depends on the output of previous components, this result reflects both the reliability of the pipeline and `EmailGenerator`'s ability to correctly integrate the selected name into the final message. The results indicate that the agent is capable of producing coherent and personalized emails in the majority of cases, while errors are primarily associated with incorrect name inputs.

4.3.3. Personalized vs. classic phishing email effectiveness

In this section, we detail the experimental settings for the human-subject phishing campaigns. In particular we go over the design of the phishing campaign, the email delivery infrastructure, the landing pages, and how results were collected. Then, we show the results obtained by the campaigns and provide an interpretation.

Experimental setup

This experiment compares the effectiveness of personalized phishing emails generated by the proposed LLM-based pipeline with that of a non-personalized phishing email. The goal is to evaluate whether automated personalization increases user engagement, measured mainly through the CTR.

The experiment involved two distinct groups of human participants. The personalized phishing group consisted of 28 participants recruited through a form, each of whom received a phishing email generated individually by the automated pipeline. The generic phishing group consisted of 28 participants selected from an email list provided by Politecnico di Milano's IT department, and each participant received a non-personalized phishing email based on a real-world example. Therefore, each participant received one phishing email and no participants appeared in both groups. The two campaigns were conducted

at different times: personalized phishing emails were sent first, while the generic phishing campaign was carried out approximately three weeks later. This temporal separation is reported for completeness, although no attempt was made to control possible time-related effects. It should be noted that not all generic phishing emails were successfully delivered. One email in the generic phishing group bounced due to the destination address being unavailable, resulting in 27 emails effectively received in that group. This occurrence is reported for transparency and was not further analyzed.

The emails were sent using a Google Workspace account created specifically for the experiments. Email delivery was performed using the Gmail API and OAuth authentication. A single sender address was used for all campaigns and was shortly created before the experiments, using a custom domain designed to mimic the institutional one. Emails were sent sequentially with a fixed delay of 30 seconds between consecutive messages to approximate manual sending behavior and reduce the risk of automated rate limiting or blocking. Due to technical issues, emails belonging to the personalized phishing group were sent in two batches separated by three days, whereas generic emails were sent in a a single batch. Generic phishing emails were based on a real-world phishing email previously received by the author of this thesis. The content was only minimally modified to preserve realism and plausibility. A single generic template was used for all recipients to avoid introducing additional variability.

The phishing links embedded in the emails redirected users to custom-built landing pages implemented in PHP and hosted on a third-party hosting service. Each link contained a unique identifier token associated with a specific recipient. Upon access, the landing page logged the request and redirected the user to a debrief page. Two separate endpoints were used: `/track.php?id=TOKEN` for personalized emails and `/trace.php?id=TOKEN` for generic emails. This separation was required to provide different debriefing content, as participants in the personalized phishing group needed to be informed about the use and processing of their personal data and have the possibility of requesting the report. To request a report, the user had to submit their email address in a form at the bottom of the page, which triggered a POST request to the `/report.php?email=EMAIL_ADDRESS&id=TOKEN` endpoint.

Access logs were stored in a Google Sheets file and included the identifier token, timestamp, and basic request metadata, such as IP addresses and user agents. Report request logs included a timestamp, the identifier token, and the email address to which the report must be sent. Email replies were collected directly through the Gmail inbox associated with the sender account and manually associated with the corresponding participants

Several exclusion criteria were applied during the analysis of click logs. Duplicate clicks from the same participants were ignored, and only the first click was considered for the CTR. Automated accesses, identified through combinations of non-residential IP addresses and user agents strings indicative of security tools or crawlers, were not counted as valid clicks. However, their presence was noted as an interesting outcome, as it suggests active inspection of phishing links by some participants. Although no invalid tokens were registered during the experiments, these would not have been considered toward the count of the CTR.

The primary metric used to evaluate the effectiveness of the phishing emails is the CTR, which is defined as the percentage of participants who clicked at least once on the phishing link. We did not consider duplicate clicks from the same participant. We also excluded from the CTR any automated access originating from security tools or URL checkers, identified through combinations of non-residential IP addresses and typical user agent strings. Nevertheless, these requests were noted and interpreted as additional defensive behaviors triggered by the phishing emails. Similarly, email replies and report requests were recorded as secondary indicators of engagement with the phishing emails.

Experiment results

Table 4.7 shows the results of the human-subject experiment, delineating for both groups the number of delivered emails, the CTR, the number of email replies, the number of automated link inspections, and the number of report requests. These numbers clearly show that the personalized phishing emails achieved a considerably higher response rate than the generic phishing emails, with 28.57% of participants in the personalized group clicking on the phishing links compared to 3.57% in the generic group. Therefore, it is possible to conclude that, in the phishing landscape, automated personalization significantly increases the likelihood of user interaction.

Metric	Personalized phishing	Generic phishing
Participants	28	28
Emails received in inbox	28	27
Unique clicks	8	1
Click-through rate (CTR)	**28.57%**	**3.57%**
Email replies	**3**	**0**
Automated link inspections	**3**	**1**
Report requests	**2**	**N.A.**

Table 4.7: Comparison of the effectiveness between personalized LLM-based phishing emails and generic phishing emails.

Beyond link clicking behavior, personalized phishing emails triggered a broader range of interactions. Three participants replied to the personalized emails. Two of these replies pointed out inaccuracies in the personalization, specifically an incorrect recipient name and an incorrect inferred interest, suggesting that personalization errors can prompt corrective or skeptical responses. The third reply explicitly recognized the email as a phishing attempt. No replies were observed in the generic phishing group. Similarly, user-triggered automated URL inspections were more frequent in the personalized group. This suggests that while personalized emails are more effective at inducing engagement, they also prompt closer scrutiny, particularly among technically-aware users.

Overall, the results support the claim that personalized LLM-based phishing emails are more effective than traditional generic phishing emails, even in a population with a technical background. At the same time, the diversity of responses highlights the dual effect of personalization: while it increases the likelihood of successful interaction, it also raises the stakes of accuracy, as personalization errors can expose the attack and trigger defensive or investigative behavior. These findings motivate a deeper analysis of which personalization factors contribute most significantly to this increased effectiveness, which is explored in the following section.

4.3.4. Personalization factors analysis

To better understand the aspects of personalization that contribute most to the effectiveness of phishing emails, we performed a subgroup analysis on the participants who received personalized phishing messages. The goal of this analysis is to isolate the impact of individual personalization factors on user behavior, measured through differences in

CTR across subgroups. In particular, we focus on two personalization factors that are central to spear phishing attacks and are explicitly handled by the proposed pipeline: the correctness of the recipient name used in the email body, and the relevance of the topic selected for the phishing message with respect to the recipient's inferred interests.

Each participant in the personalized phishing group was assigned to different subgroups based on the accuracy of each personalization factor. In particular, we determined the correctness of the recipient name by matching the one resulted in the email body with the participant's ground truth name, i.e., the one entered in the form; common variants of the name were also considered correct. We evaluated the relevance of the topic by linking it to information found in at least one social media profile of the participant. For each subgroup, we computed the CTR and analyzed the results.

Table 4.8 reports the CTRs and the number of email replies observed for each subgroup of the personalized phishing experiment. When we consider name correctness, a clear difference emerges. Emails that correctly addressed the recipient achieved a CTR of 34.78%, whereas no clicks were observed for emails containing an incorrect recipient name. Therefore, we can conclude that correct usage of the recipient's name appears to be a necessary condition for successful phishing attempts: incorrect naming acts as a strong credibility breaker, effectively preventing link clicks and prematurely influencing the decision of the user before they read the full email. In this subgroup, we observed a single email reply, which informed the sender of what the recipient presumed was a contact error.
Topic relevance shows an even stronger effect on click behavior. Emails whose topic was assessed as relevant to the recipient achieved a CTR of 50.00%, compared to only 7.14% for emails with a non-relevant or generic topic. This major difference highlights the importance of content-level personalization in lowering suspicions and in increasing engagement with phishing links once basic credibility is established. Email replies were observed in both subgroups, indicating that topic relevance may influence different forms of engagement in distinct ways. This finding supports the intuition underlying spear phishing attacks, where attackers exploit contextual knowledge to craft highly targeted messages.

Personalization factor	Participants	CTR	Email replies
Correct recipient name used	**23**	**34.78%**	**2**
Incorrect name	5	0%	1
Relevant topic selected	**14**	**50.00%**	**1**
Non-relevant or generic topic	14	7.14%	2

Table 4.8: Click-through rates observed for different subgroups of the personalized phishing experiment, based on individual personalization factors.

These results suggest that each personalization factor has a different effect. The use of the correct name at the beginning of the email is essential to avoid immediate detection and suspicion, whereas the relevance of the topic is crucial to persuade recipients to interact with the phishing link. From a defensive perspective, these findings emphasize the need to train users to identify phishing emails that contain highly personalized content and accurate personal information and the necessity for the implementation of automated tools that can detect and flag these kind of emails.

5 | Limitations

This work presents an automated pipeline for generating personalized phishing emails and evaluates both its technical components and its effectiveness through a controlled human-subject experiment. While the results provide evidence of the feasibility and impact of large-scale automated personalization, several limitations must be acknowledged. In this chapter, we describe these limitations, in particular concerning the characteristics of the participant population, the operational scope of the pipeline, and the constraints imposed by ethical and legal considerations.

The first limitation concerns the composition of the participant population involved in human-subject phishing campaigns. Participants were predominantly students affiliated with a technical university, and it can reasonably be assumed that a large portion of them possess above-average familiarity with cybersecurity concepts or phishing risks. This characteristic may have influenced both initial awareness and response behavior, potentially leading to more cautious interactions than would be observed in a general population. In addition, participants in the personalized phishing group were recruited through a form and were therefore aware that they might receive a communication generated by the system. Although the exact nature and timing of the phishing email were not disclosed, this prior awareness may have caused some participants to expect an email or to scrutinize incoming messages more carefully. As a result, the observed CTRs and interaction patterns may underestimate the effectiveness of personalized phishing emails in real-world scenarios where targets are unaware of being observed or studied.

Another challenge lies in the operational scope of the data collection pipeline. The personalization process relies exclusively on publicly available information, primarily extracted from social media platforms and web search results. Consequently, the pipeline is inherently limited when targets maintain private profiles, have minimal online presence, or use pseudonyms that cannot be reliably linked to their real identity. As observed during agent validation, some social media profiles were either inaccessible or incorrectly attributed, and missing data propagated to the rest of the pipeline, affecting topic inference and name selection. This means that the pipeline's effectiveness is uneven across individuals

and strongly dependent on the visibility and quality of their digital footprint. In environments where privacy settings are stricter or where users intentionally minimize their online presence, the benefits of automated personalization may be significantly reduced.

Finally, ethical and legal constraints represent another significant limitation of this study. The experiments were designed to comply with ethical guidelines and data protection regulations, including informed consent and data minimization principles. These safeguards necessarily restrict the realism of the phishing campaigns. For instance, participants in the personalized phishing group had to be informed about the use of their personal data. Such transparency is incompatible with real-world phishing attacks, where deception is unconstrained and attackers can freely exploit sensitive or semi-private information. As a consequence, the study likely underestimates both the aggressiveness and the potential effectiveness of malicious actors who operate without ethical boundaries.

6 | Future work

The research presented in this thesis has demonstrated the feasibility and effectiveness of a fully automated, large-scale pipeline for generating personalized phishing emails. Although the current work provides empirical evidence and validates technical components, we identified several directions for extending and refining this research. In this chapter, we provide an overview of these paths, which can be broadly categorized into three areas: expanding the participant population for the studies, improving the technical robustness of the pipeline, and exploring defensive applications.

A key limitation of the present evaluation is the relatively small and homogeneous participant population. All human-subject experiments were conducted with students affiliated with a technical university, a group likely to possess above-average familiarity with cybersecurity practices and phishing risks. Therefore, observed CTRs and engagements behaviors may not generalize to broader populations with different demographic characteristics, technical backgrounds, or cultural contexts.
Future studies should involve larger and more diverse participant pools, including individuals of different ages, occupations, and education. Expanding the participant base would allow for a more comprehensive assessment of the effectiveness of automated personalization in real-world scenarios. In addition, other studies could explore how repeated exposure to personalized phishing attempts affects user behavior, learning, and susceptibility. Such studies could incorporate multiple campaign waves and controlled variations in the style of the emails.

Although the current pipeline achieves high plausibility in generated emails, several technical challenges remain. A noteworthy area for improvement is the accuracy of identity disambiguation and the reliability of public data extraction. As observed in the validation of the `EmailAddressAnalyzer` and `SocialsFinder` agents, incorrect name inference or misattribution of social media profiles can reduce the effectiveness of personalization. Future work could investigate possible ways to increase precision in identity matching. Similarly, topic inference and content personalization depend heavily on the availability and quality of social media data. Research into more sophisticated content summariza-

tion, interest inference, and integration of additional data sources could enhance the relevance of phishing content. For example, web search queries can provide a wide range of publicly available information beyond social media, such as personal websites, news coverage, and other online references, which would enable richer and more context-aware personalization.

This thesis explores the offensive capabilities of LLMs and how they can exploited to carry out automated spear phishing campaigns, however the insights found can be directly applied to defensive research. Future work could focus on the training of models to detect LLM-generated phishing emails using a combination of linguistic analysis and metadata heuristics. Furthermore, other mitigation strategies could be explored, such as a system that alerts users about the presence of known personalization patterns in the incoming email.

7 | Conclusions

This thesis investigated the feasibility, effectiveness, and security implications of a fully automated pipeline for generating personalized phishing emails based on LLMs. The primary goal of this work was to assess whether modern LLM-based systems can autonomously collect publicly available information, infer contextual signals about a target, and generate highly personalized phishing messages capable of increasing user engagement compared to traditional, non-personalized attacks.

The proposed solution consists of a modular multi-agent architecture designed to replicate the typical stages of a spear phishing campaign: identity inference, social media discovery, information extraction, topic selection, and email generation. Each component was independently validated using curated datasets, including the Enron email dataset [11] and a manually built dataset derived from personal websites and social media profiles. This validation phase highlighted both the strengths and limitations of automated identity disambiguation and data extraction processes, showing that performance is highly dependent on the structure of available public data and the clarity of digital footprints.
Beyond technical validation, the thesis presented a controlled human-subject experiment comparing personalized LLM-generated phishing emails with a traditional generic phishing email. The results demonstrate a substantial increase in CTR when personalization is applied. Even within a technically aware population, personalized emails achieved significantly higher engagement levels than their generic counterparts. These findings provide empirical evidence that automated personalization, when supported by LLMs and publicly accessible data, can meaningfully increase the effectiveness of phishing attempts.
The subgroup analysis further revealed that personalization is not a monolithic factor but operates in layers. Surface-level personalization, such as correct use of the recipient's name, appears to be a necessary condition for credibility. Deeper content-level personalization, particularly the selection of a topic relevant to the recipient's interests, has an even stronger influence on engagements. These results reinforce the conclusion that contextual knowledge plays a central role in successful social engineering attacks.

At the same time, we acknowledge several limitations of this work. The participant

population was relatively small and homogeneous, which may limit the generalization of the findings. The pipeline's effectiveness is also constrained by the availability and quality of publicly accessible data, as well as by unavoidable inaccuracies in identity resolution and profile attribution. Furthermore, ethical and legal safeguards required for conducting human-subject research reduce the realism of the simulated attacks compared to unconstrained malicious campaigns.

Despite these limitations, these results have important implications for cybersecurity and spear phishing. We demonstrated that LLMs are able to automate the generation of personalized and coherent phishing emails that are relevant to the target's interests. This lowers the technical and economical barriers that prevented attackers to launch large-scale spear phishing campaigns, as manual reconnaissance and careful email crafting can be reliably automated.

From a defensive perspective, the findings of this work suggest the urgent need to improve the detection mechanisms available at the moment and to spread awareness to users on the capabilities of these models. Traditional indicators of phishing, such as poor grammar or generic content, are not reliable anymore. Defensive systems must integrate new personalization and behavioral patterns and user training programs should include examples of personalized, well-written phishing emails to better reflect the evolving phishing landscape.

Bibliography

[1] AI in Cybersecurity: Key Stats & Insights — zerothreat.ai. https://zerothreat.ai/blog/ai-in-cybersecurity-statistics. [Accessed 09-01-2026].

[2] M. Bethany, A. Galiopoulos, E. Bethany, M. B. Karkevandi, N. Vishwamitra, and P. Najafirad. Large language model lateral spear phishing: A comparative study in large-scale organizational settings. *arXiv preprint arXiv:2401.09727*, 2024.

[3] L. Burita, P. Matoulek, K. Halouzka, and P. Kozak. Analysis of phishing emails. *AIMS electronics and electrical engineering*, 5(1):93–116, 2021.

[4] R. Cialdini. *Influence: The Psychology of Persuasion*. Collins Business Essentials. HarperCollins e-books, 2009. ISBN 9780061899874. URL https://books.google.it/books?id=5dfv0HJ1TEoC.

[5] A. Devasia. AI Cyber Threat Statistics: The 2025 Landscape of AI-Powered Cyber-attacks — thenetworkinstallers.com. https://thenetworkinstallers.com/blog/ai-cyber-threat-statistics/. [Accessed 09-01-2026].

[6] S. Gallagher, B. Gelman, S. Taoufiq, T. Vörös, Y. Lee, A. Kyadige, and S. Bergeron. Phishing and social engineering in the age of llms. In *Large Language Models in Cybersecurity: Threats, Exposure and Mitigation*, pages 81–86. Springer Nature Switzerland Cham, 2024.

[7] F. Heiding, B. Schneier, and A. Vishwanath. AI Will Increase the Quantity—and Quality—of Phishing Scams. https://www.schneier.com/academic/archives/2024/06/ai-will-increase-the-quantity-and-quality-of-phishing-scams.html. [Accessed 09-01-2026].

[8] F. Heiding, S. Lermen, A. Kao, B. Schneier, and A. Vishwanath. Evaluating large language models' capability to launch fully automated spear phishing campaigns: Validated on human subjects. *arXiv preprint arXiv:2412.00586*, 2024.

[9] F. Heiding, B. Schneier, A. Vishwanath, J. Bernstein, and P. S. Park. Devising and

detecting phishing emails using large language models. *IEEE Access*, 12:42131–42146, 2024.

[10] K. Khadka, A. B. Ullah, W. Ma, E. M. Marroquin, and Y. Alem. A survey on the principles of persuasion as a social engineering strategy in phishing. In *2023 IEEE 22nd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 1631–1638. IEEE, 2023.

[11] B. Klimt and Y. Yang. The enron corpus: A new dataset for email classification research. In *European conference on machine learning*, pages 217–226. Springer, 2004.

[12] E. Lim, G. Tan, T. Hock, and T. Lee. Turing in a box: Applying artificial intelligence as a service to targeted phishing and defending against ai-generated attacks. *Black Hat USA, Las Vegas*, 2021.

[13] T. Lin, D. E. Capecci, D. M. Ellis, H. A. Rocha, S. Dommaraju, D. S. Oliveira, and N. C. Ebner. Susceptibility to spear-phishing emails: Effects of internet user demographics and email content. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 26(5):1–28, 2019.

[14] J. Lindner. Social Engineering Statistics Statistics: Market Data Report 2025 — gitnux.org. `https://gitnux.org/social-engineering-statistics/`. [Accessed 09-01-2026].

List of Acronyms

AI Artificial Intelligence. 2, 5, 7–9, 23, 26

AIaaS Artificial Intelligence as a Service. 8

API Application Programming Interface. 23, 26, 37, 38, 45

BCC Blind Carbon Copy. 37

CAPTCHA Completely Automated Public Turing test to tell Computers and Humans Apart. 29, 30

CSS Cascading Style Sheets. 39

CSV Comma-Separated Values. 43

CTR Click Through Rate. i, iii, 2–4, 7, 9, 10, 42, 45, 52, 54–57, 59, 61, 63

DKIM DomainKeys Identified Mail. 11

DNS Domain Name System. 11, 37

HTML HyperText Markup Language. 30, 31, 37, 39

IT Information Technology. 3, 15, 21, 44, 52

JSON JavaScript Object Notation. 26–30, 35–37, 71

LLM Large Language Model. i, iii, 2–9, 23, 26, 28, 30, 32–35, 39, 41, 42, 45, 52, 55, 62–64

MIME Multipurpose Internet Mail Extensions. 37

OSINT Open-Source Intelligence. 2, 5, 10, 23, 29, 33

SPF Sender Policy Framework. 11

URL Uniform Resource Locator. 4, 19, 28, 30, 37, 38, 54, 55

List of Figures

List of Listings

List of Tables