Generating Account Access Graphs Using Open-Source Intelligence Data

Nickon Koorosh Tajali

4th Year Project Report
Artificial Intelligence
School of Informatics
University of Edinburgh

2024

Abstract

Account access graphs enable the comprehensive modelling and analysis of an individual's account ecosystem to identify security vulnerabilities exploitable by malicious actors [8]. These models incorporate the connections between accounts, credentials and devices to facilitate their analysis. However, there is often a lack of consideration regarding the extent to which an individual's account setup can be uncovered through data available in the public domain, potentially leaving it vulnerable to exploitation. In an attempt to address this, we conducted a participant study. This study aimed to determine how much of an individual's data can be uncovered in the public domain using Open-Source Intelligence (OSINT) tools. We then model this data using account access graphs to analyse and identify security vulnerabilities. Subsequently, we evaluate the accuracy of these tools by comparing the account access graphs generated to participant-provided data. Using the participant-provided data, we further analyse their security setups and highlight common security behaviour across participants. We also developed and implemented a tool for generating account access graphs.

By leveraging OSINT tools, we uncovered over half of each participant's data. We successfully discovered at least one email account for each participant and, in some cases, uncovered all their email accounts. Additionally, half of the participants had their phone numbers revealed, along with a substantial number of accounts. In some cases, more accounts were discovered for a participant than they had provided. The abundance of data uncovered using OSINT tools and the resulting account access graph that can be generated using this data stresses the extent to which an individual's security setup can be revealed online. This emphasises the need for robust authentication methods and the reduction of interconnections between accounts.

Research Ethics Approval

This project obtained approval from the Informatics Research Ethics committee.
Ethics application number: 724975
Date when approval was obtained: 2023-11-10

The participants' information sheet is included in Appendix C and the consent form is included in Appendix D.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Nickon Koorosh Tajali*)

Acknowledgements

Firstly, I would like to thank my supervisor, Dr. David Aspinall. Your support and your expertise greatly benefited me throughout my project. I am grateful for your guidance, and I thoroughly enjoyed our meetings. Your enthusiasm for cybersecurity is both inspiring and motivating, and I hope to apply the skills I have learned from this project to my future endeavours.

I would also like to express my thanks to Sandor Bartha for his support and his willingness to take the time to read my draft chapters. Your feedback was very valuable and greatly appreciated.

Lastly, I would like to thank my family and friends for their everlasting love and support throughout my time at university. The things you have done for me will never be forgotten.

Table of Contents

Chapter 1

Introduction

1.1 Motivation

The ever-increasing digitisation of society has required digital users to maintain a more significant number of accounts across a variety of different services. Consequently, individuals are consistently urged to adopt unique passwords and enable two-factor authentication across all their accounts [1]. This advice stems from the belief that an individual's overall online security is solely correlated with the protection of their individual accounts through robust authentication methods. However, a notion that is often overlooked is that an individual's accounts are not isolated and are instead interconnected, either directly or indirectly. Therefore, while robust authentication methods may enhance their security, they may not fully protect against security vulnerabilities stemming from account connections. The interconnection of accounts within a user's online setup can pose substantial risks to an individual's overall security, especially if a critical account is compromised. Breaching an individual's online setup could lead to the compromise of vital accounts, potentially enabling access to bank accounts and facilitating theft. For instance, using an email address as a recovery mechanism for a social media account links a personal email address to a specific username on a social network. This same email address may also be linked to their Revolut account, illustrating how a compromise in one account could lead to a compromised chain of accounts.

Previous research proposed modelling a user's online account setup using account access graphs that illustrate the connections between accounts, credentials, and devices [8]. These models identify weaknesses in an individual's account ecosystems to prevent malicious actors from compromising them. By enabling tailored advice rather than generic recommendations designed to fit all individuals in society, these models offer a more personalised approach to enhancing online security.

While modelling an individual's online account ecosystem using provided data is highly beneficial for analysing and identifying vulnerabilities in one's setup, it fails to consider the data a malicious adversary can obtain online. This is concerning because individuals might not realise the extent to which their online account setup can be extrapolated and modelled to pinpoint weaknesses. Furthermore, this process enables a malicious actor to uncover accounts and credentials that an individual has overlooked or not considered, allowing them to identify the most efficient entry points into an individual's security setup. Therefore, it is essential to consider the data that could be obtained about an individual through the public domain and emphasis should be especially placed on ensuring adequate protection of these accounts that may be uncovered online. The ability of a malicious adversary to uncover an individual's credentials and accounts from the internet to model their online security setup and compromise it serves as a strong motivation for this project.

1.2 Project's Goals and Contributions

The goals of this project were as follows:

- Evaluate the proficiency of OSINT tools in uncovering a participant's account connection setup using data found on the internet
- Carry out analysis on participants' actual account connection setup using provided data to identify vulnerabilities.
- Design and implement a robust and intuitive Java tool that generates account access graphs using provided data.
- Identify common trends in security behaviour among participants.

The project's contributions were as follows:

- Assessment of the effectiveness and accuracy of various OSINT tools in gathering data regarding participants' account connection setups. Identification of strengths and limitations of OSINT tools in this context.
- Identification of vulnerabilities, weaknesses and patterns in participants' account ecosystems. Insights into common security weaknesses and areas for improvement among participants.
- Implementation of a user-friendly tool for generating account access graphs.

1.3 Outline

This report is structured as follows: Chapter 2 delves into account access graphs and introduces Open-Source Intelligence. Chapter 3 outlines the design of the participant study, the methodology for gathering data using OSINT tools, and the design of the Java tool. Chapter 4 discusses the implementation and testing of the Java tool. Chapter 5 presents the experimental results and their evaluation. Finally, Chapter 6 summarises the report and discusses potential future work.

Chapter 2

Background Research

2.1 Account Access Graphs

Comprehensive modelling of an individual's unique security setup enables the analysis of the security of individual accounts and the connections that form among the user's accounts, devices, credentials, keys, and documents. This is achievable through the use of a formalism called an Account access graph [8], which facilitates the discovery of vulnerabilities within a user's setup that could be exploited by an attacker. The formal definition of an account access graph can be stated as:

Definition 1. *An account access graph is a directed graph $G = (V_G, E_G, C_G)$, where V_G are vertices, C_G are colors, and $E_G \subseteq V_G \times V_G \times C_G$ are directed colored edges.*

An example of an account access graph is illustrated in Figure2.1. The online shop account acc_{shop} can be accessed using the password pwd_{shop} or recovered using the email account acc_{mail}. The email account requires two-factor authentication with password pwd_{shop} and a code that is generated using an authenticator app on the device. This device can be unlocked with either the PIN or fingerprint.

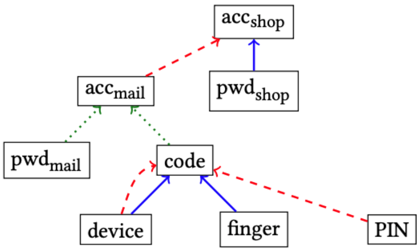


Figure 2.1: Example Account Access Graph

The edge colours of a graph signify the access permissions associated with a vertex. If multiple edges of the same colour point to vertex v, then all source vertices are used in conjunction to authenticate the user. Equally, different coloured edges pointing to vertex v represent the various authentication methods a user can use to authenticate themselves. The edges of an account access graph serve a multitude of purposes, such as connecting a credential to an account to indicate its use for user authentication or linking one account to another for single-sign-on purposes.

The *access set* of an account can be defined as the minimal sets of credentials that are sufficient to provide access to the account. Using this definition, we can formally define the access base of a vertex as:

Definition 2. *The access base* $\mathrm{AccessBase}(v, V_{init})$ *of a vertex v with respect to a set of initial vertices* V_{init} *consists of the minimal access sets V that only contain vertices from* $V_{init.}$. $\mathrm{AccessBase}(v, V_{init}) := \{V \in \mathrm{MinAccessTo}(v) \mid V \subseteq V_{init}\}$.

The access base of an account models the elements needed to compromise it, as indicated by the credential or account leaf nodes pointing to it in the graph. Each access set within the access base is distinguishable by the shared colour of the originating edges.

Assessing the effectiveness of a user's security setup involves considering additional, sometimes hard-to-measure, factors beyond what is visible in their account access graph. For instance, not all accounts a user possesses carry the same importance. An email account used as a recovery mechanism for all of the user's social media and online banking accounts will hold greater significance than an account with a high-street retailer. Furthermore, individuals may also face varying levels of risk of having their possessions stolen. To account for these hidden factors, a security scoring scheme introduced in [8] evaluates the security of an account using its access base, where a higher score signifies a more secure account. The formal definition of the security scoring scheme for an account access graph can be defined as:

Definition 3. *A security scoring scheme for an account access graph G is a 6-tuple (* $D, \leq, V_{init}$*, Init, Eval, Combine), where:*

- *D is the domain over which scores are defined.*
- $\leq$ *is a partial order relating elements in D.*
- $V_{init} \subseteq V_G$ *is called the set of initial vertices.*
- *Init :* $V_{init} \to D$ *maps initial vertices to initial scores.*
- *Eval:* $\mathcal{P}_{\mathcal{M}}(D) \to D$ *maps a multiset of initial scores (of vertices in an access set) to an intermediate score (for that access set).*
- *Combine:* $\mathcal{P}(D) \to D$ *maps a set of intermediate scores (of access sets in a vertex's access base) to a score (for that vertex).*

Given Eval, we define an auxiliary function EvalSet: $\mathcal{P}(V_{init}) \to D$ *that directly maps an access set to its intermediate score by first computing the initial scores of its vertices and then applying Eval.*

$$\mathrm{EvalSet}(S) := \mathrm{Eval}(\{\mathrm{Init}(v') \mid v' \in S\}).$$

We then define the following score function Score: $V_G \to D$*, which directly maps a vertex to its (final) score:*

$$\mathrm{Score}(v) := \mathrm{Combine}(\{\mathrm{EvalSet}(S) \mid S \in \mathrm{AccessBase}(v, V_{init})\}).$$

Intermediate scores can be computed using either a maximum or sum function to capture the conjunction of credentials required to access an account. The vertex's overall score is determined using a minimum function, representing the disjunction of required access sets. A variety of scoring schemes can be defined depending on the purpose and the complexity of the graph. For example, a simple scoring scheme assigns natural numbers to vertices to calculate both intermediate and overall scores that determine how secure an account is. Meanwhile, a complex scoring scheme factors in attacker-related variables such as location and skill set, assigning higher capabilities to local attackers than remote ones. It also categorises skill levels as none, some, or expert. Vertices in the graph are scored using attribute tuples denoting attacker location and skill set. This scoring scheme evaluates each account based on the weakest attacker capable of compromising it, preventing any misrepresentation of the attacker's capabilities. For instance, if an account has a score of (remote, some), then a remote attacker with some special hacking skills could compromise that account. An extension of the account access graph in Figure 2.1 made to include attacker attribute tuples is illustrated in Figure2.2.

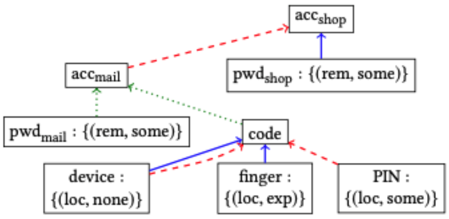


Figure 2.2: Account Access Graph Extended to Include Attacker Attribute Tuples

Moreover, scoring schemes can help identify critical vertices in a user's graph that pose a significant risk if compromised due to their access to numerous other vertices. In a centrality scoring scheme, vertices are assigned scores based on how frequently they appear in the access set of other vertices. Each vertex has a minimum centrality score of one, as it is an element of its own access set. Therefore, vertices that appear in the access sets of a large number of other vertices will receive high centrality scores, indicating their importance in the user's security setup. For example, a participant study detailed in [7] found that the most central vertex in their participants' account access graphs was often an internet-connected device. It was found that participants frequently used their devices to access accounts through open sessions or passwords saved on the device.

Identifying the most central vertices in an account access graph aids in revealing the presence of cycles within the graph. Cycles occur when multiple vertices are interconnected through both incoming and outgoing edges. A cycle that includes a critical vertex can result in high centrality scores for the other vertices within the cycle because of the abundance of connections the critical vertex has with other vertices. Therefore, it is necessary to ensure the adequate protection of these central vertices.

The scoring scheme previously introduced can be adapted to assess the likelihood of a user getting locked out of their account by considering the account's recovery mechanisms. Recovery mechanisms allow users to regain access if they forget or lose their authentication credentials. While convenient, these mechanisms may also pose security risks, providing an additional access point that adversaries could exploit. A high recoverability score suggests a user is less likely to experience account lockout. However, to consider the recoverability of an account, the scoring scheme defined in Definition 3 needs to be adapted to evaluate the lockout base of an account instead of its access base. Before defining the lockout base of an account, we need to define the lockout set for a vertex v as the set of vertices V that if the user does not have access to any credential or account in V, the user cannot access v. We can then proceed to define the lockout base of a vertex v as:

Definition 4. *The lockout base LockoutBase (v, V_{init}) of a vertex v with respect to a set of initial vertices V_{init} consists of the minimal lockout sets that only contain vertices from $V_{init.}$* . $\text{LockoutBase}(v, V_{init}) := \{V \in \text{MinLockout}(v) \mid V \subseteq V_{init}\}$.

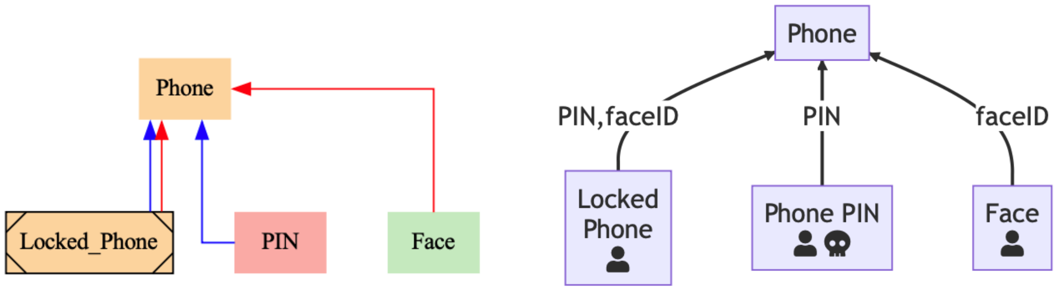
The application of scoring schemes for recoverability enables the analysis of whether an account possesses backdoor access that is more easily exploitable through recovery mechanisms than primary authentication methods. This analysis involves comparing the scores derived from the original account access graph with those obtained from a modified version where the recovery edges have been removed. If the score of the original graph is lower than the modified one, it indicates a potential backdoor into the account. Furthermore, scoring can also be applied for risk analysis to determine which accounts' security should be prioritised. A higher score suggests it is a more critical account that warrants more robust security measures.

2.2 Account Access Graphs With State

Account access graphs effectively visualise a user's security setup and facilitate the analysis of its static security properties. However, a limitation of these models is their inability to consider modifications to a user's account access graph, which results from an attacker disconnecting an account or device from the user's account ecosystem. To address this limitation, [2] proposes considering account access graphs as states in a state transition system, where tactics are used to transition from one account access graph to another. These extended graphs include state information that indicates which vertices are accessed by different parties, such as the user or an adversary. States are assigned to a vertex using a mapping technique that assigns a set of parties to each vertex. To formally define an account access graph with state, we will first introduce several key elements. Let $\mathcal{V}$ be a countably infinite set of vertices (representing, e.g., accounts, devices, credentials), ranged over by (possibly indexed) variables u and v. Let $\mathcal{L}$ be a countably infinite set of labels (for access methods) ranged over by l and $\mathcal{A}$ be a set of participants, typically a user and an attacker, ranged over by a. We can now formally define an account access graph with state:

Definition 5. *(Account Access Graphs with State). An account access graph with state is a triple $G = (V, E, A)$ where $V \subset \mathcal{V}$ is a finite set of vertices, $E : (V \times V) \to 2^{\mathcal{L}}$ is a map labelling pairs of vertices with finite sets of access methods, pairs of vertices labelled with a non-empty set of access methods are edges, and $A : V \to 2^{\mathcal{A}}$ is a map labelling vertices with a finite set of participants.*

In Definition 5, the original concept of multiple coloured edges pointing to a vertex is replaced by a single edge assigned a set of labels denoting access methods. Edges with the same label pointing to a vertex v are used together for authentication, while different labels indicate alternative methods. A user can access a vertex v if they have access to all vertices with an edge pointing to v that share the same label. The differences in the original account access graph's definition and its extended definition are illustrated in Figure 2.3.



(a) Account Access Graph (b) Account Access Graph with State

Figure 2.3: Illustration of Differences in Account Access Graphs

The creation of a state transition system enables the graph to capture the evolution of account or credential access over time. This system accounts for changes due to access route utilisation and the addition or removal of accounts and credentials by both users and adversaries. Moreover, it introduces the ability to represent account takeover attacks, which couldn't be represented in the original account access graphs. Modelling these attacks is essential to understand how changes in device or account access, as a result of a user's or adversary's actions, can impact the overall security of the user's ecosystem. Furthermore, properties can serve as predicates to capture graph structure and the impact of user or adversary actions on the vertices in the graph. These properties can denote the effect transition relations have on an account access graph without altering its structure. Transition relations offer a methodology for altering access to vertices, either individually or collectively, through account access graph operations such as changing the graph state, discovering new accounts, or losing access to an account. These operations involve adding or removing vertices and edges, which can only be executed by a user or an attacker.

Account access graph operations can be executed using tactics that are expressed using clearly defined semantics. Tactics provide a mechanism to model and record attacker techniques or resilience strategies by creating small programs in a domain-specific language. The effectiveness of a tactic is assessed based on the predefined criteria established for the graph. From a user's perspective, tactics represent the user trying to satisfy a security requirement and ensure the security of an account/credential. Meanwhile, from the perspective of an attacker, they represent an adversary trying to break the security requirements a user has in place. Tactics allow us to gain an understanding of attacks that target a particular set of accounts. This was demonstrated in the case study outlined in [2], where tactics were employed to examine potential methods by which an attacker could compromise a user's account ecosystem. The scenario was based on a news article published by The Wall Street Journal ([18]).

The article detailed an incident where thieves targeted an individual in a bar, ultimately obtaining the victim's phone PIN and stealing their iPhone. With knowledge of the PIN and possession of the device, they locked the victim out of their accounts and gained unauthorised access to their sensitive data and bank accounts by removing the victim's access to their data from all their Apple devices. By modelling this scenario using tactics, the authors of [2] could disprove two claims made by the Wall Street Journal. The first claim made was Apple's screen time feature could have acted as a defence measure against the attack. Secondly, the article suggested that a similar attack could be carried out on an Android phone. Through the use of tactics, it was determined that the Google account logged into an Android phone could not be compromised in the same manner as the AppleID account presented in the article. As a result, the adversaries would not have experienced the same outcome presented in the article had they stolen an Android Phone. However, it was found that specific Android devices that required their own manufacturer accounts could be compromised in a manner similar to Apple devices.

2.3 Limitation of Account Access Graphs

While account access graphs provide a practical and precise method for analysing security attributes within a user's security ecosystem, they do not differentiate between an account's primary and backup authentication methods. This distinction is crucial as the process and effectiveness of these mechanisms may vary in practice. Consequently, assessing the risks associated with authentication within an ecosystem using account access graphs may not be as precise as initially believed.

To address this limitation, the authors of [16] proposed the Authentication Analysis Framework (AAF) to evaluate authentication risks logically. The AAF approach suggests separating the analysis of authentication methods from that of the account access graph. It considers the quality and reliability of both primary and fallback authentication methods and the account type. This assessment involves ranking authentication methods and account types based on associated security and accessibility risks using maturity tables as reference. Using these tables, account access graphs can now be analysed to derive a more accurate security score for each user account.

The global security score for an account is calculated based on the connections between accounts and the individual account score. A protection score is then generated using a scale that reflects the type of account being analysed. The protection score is then compared against the actual security score to determine how protected an account is. If the actual score exceeds the protection score, the account is secure. If they match, the account is adequately protected. If the actual score is lower, security improvements are needed.

Additionally, account accessibility is assessed to determine the risk of user access loss. Accordingly, an accessibility score is generated to indicate the diversity of the independent recovery authentication options available. A high accessibility score signifies a low risk of a user losing access, while a score below 1 indicates a high risk.

2.4 Account Access Graphs in Other Contexts

The underlying concept of account access graphs can extend beyond user security setups. For instance, they can be adapted to represent file access control within a *SecureSim* for a certificate-based authentication scheme ([23]). Within this SIM, access control acts as an organisational framework categorising all SIM files into six distinct classes, enabling permission customisation for various files. Access graphs model the relations in the SIM's access control using three node types: file node, profile node, and schema node. A file node represents an individual file, a profile node denotes a collection of files, and a schema node outlines the policy for an entity. Profile nodes collate files into predefined categories, each linked to a profile node. With the access graph generated, as illustrated in Figure 2.4, the management of access control for the nodes can be modeled. This allows the determination of the file access permissions granted by the *SecureSim* to an entity based on the privilege of its leaf node.

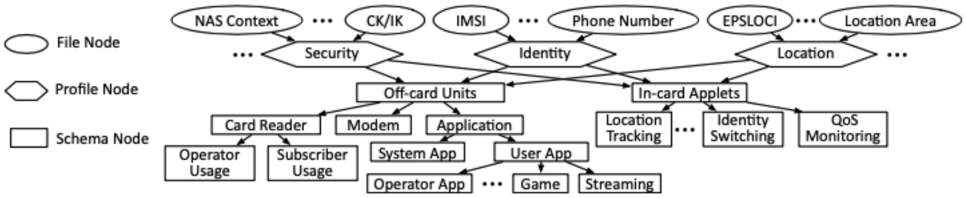


Figure 2.4: SIM File Access Modelled Using an Access Graph

2.5 Open-Source Intelligence

Open-Source Intelligence (OSINT) involves collecting and analysing data from open sources such as public records, social media, websites, and the dark web [4]. It is utilised by various groups or individuals, including governments, investigators, and hackers. OSINT comes in two forms: passive and active. Passive collection involves gathering data without interacting with the target, while active OSINT consists in engaging with the target directly, such as adding them on social media or messaging them. Passive OSINT carries a low risk of attribution, whereas active OSINT can lead to a higher risk of attribution [6]. Open-Source Intelligence typically works in a four-stage process:

1. Collection: Gathering publicly available information on a target using various sources.
2. Processing: Processing the data obtained to remove duplicate, irrelevant, or inaccurate data and filtering and categorising based on relevance and importance.
3. Analysis: Identifying trends, patterns, and relationships in the data obtained using data visualisation tools, data mining, and natural language processing.
4. Dissemination: Presenting the intelligence to the necessary subject

While national security teams and law enforcement commonly employ Open-Source Intelligence to protect organisations and society from threats, its use has sparked controversy over the utilisation of the data obtained using these tools. Concerns arise regarding legality, where although accessing and analysing the data obtained through these tools is legal, it can be used to support malicious actors in illegal activities. Additionally, ethical concerns arise regarding the appropriate use of the obtained information, emphasising the need for its use in legitimate and legal contexts and not to perform harm to others. Lastly, the data obtained raises privacy concerns due to the extensive personal information available in the public domain. Individuals often share such data without considering its full implications, allowing for the creation of detailed profiles that can compromise privacy [4].

2.6 Case Study - Exploiting Publicly Available Data

The following case study illustrates how attackers exploit publicly available information to compromise an individual's account ecosystem, posing significant risks to individuals, as demonstrated by the targeted attack on journalist Mat Honan[10]. Below, the steps of the attack are outlined:

1. Hackers identified the victim's Gmail address by generating random email prefixes using the journalist's name. They then located the victim's recovery email address through the recovery method of the Gmail account, successfully deducing the complete address through character analysis and brute force despite partial obscuration.

2. The domain of the newfound email address indicated to the hackers that the victim had an Apple ID account. Further reconnaissance in the public domain revealed the victim's Amazon account associated with the newly uncovered email address. The hackers then deceived Amazon customer support by impersonating the victim and adding a new credit card to the account using only the victim's name, email address, and billing address — all found through the public domain. Later, they contacted Amazon, claiming to have lost access to the account and requested to add a new email address, requiring only the name, billing address, and one of the credit card numbers associated with the account. This allowed them to reset the Amazon password using the newly added email.

3. The hackers then contacted Apple Support to access the victim's Apple ID through recovery mechanisms. They only needed to provide the associated email address, billing address, and the last four digits of the credit card linked to the account, which was obtained from the victim's Amazon account. After gaining access to the victim's Apple ID account, the hackers changed the password and logged the account out from all devices. This action prevented the victim from accessing their devices and changing their Apple ID password. Access to the victim's Apple ID account allowed entry to the victim's Apple Mail account.

4. Subsequently, this access enabled the recovery of the victim's Gmail password, granting the hackers access to both email accounts. This allowed them to compromise the victim's Twitter account to post racist and homophobic content, tarnishing the victim's reputation.

Chapter 3

Methodology and Design

3.1 Acquiring Data for Account Access Graphs through a Participant Study

3.1.1 Setting up the Participant Study

A study was conducted to gather authentic data for the generation of account access graphs that could be analysed. This study involved 10 participants spanning an age range from 19 to 66. The benefits of this extensive age range allowed the account access graphs generated to be representative of society as much as possible and aided in capturing the diverse security behaviours exhibited across different age groups [20][21][15]. Ensuring the participant pool was sufficiently large and diverse was beneficial for testing the search space of the Open-Source Intelligence (OSINT) tools. It allowed for assessing their ability to identify the accounts held by individuals with varying internet presence across diverse age groups. This extends to the analysis of the participant's account access graphs, where the adaptability of these graphs could be tested with respect to the varying data being modelled and how they can still be beneficial in identifying weaknesses within the account security setup.

The study comprised three stages. Firstly, approval was obtained from the Informatics Ethics Board to conduct the study, where each participant received an information sheet and consent form for their participation. The participant information sheet can be found in Appendix C, and the participant consent form can be found in Appendix D. The second stage involved using Open-Source Intelligence tools to gather each participant's publicly available data, including credentials like email addresses, phone numbers, and owned accounts. Subsequently, this data was utilised to generate an account access graph. Finally, the third stage entailed collecting data directly from the participants to create an account access graph of their provided data.

3.1.2 Open-Source Intelligence Tools

OSINT tools were utilised to replicate the process by which a malicious actor collects intelligence from public domains such as social media, public records, reports, news articles, and search engines to compromise a victim's security setup. These tools leverage natural language processing and machine learning techniques to facilitate data extraction. A suite of OSINT tools was utilised to ensure comprehensive coverage of each participant's data. The study outlined specific requirements that had to be met for employing an OSINT tool, including the following:

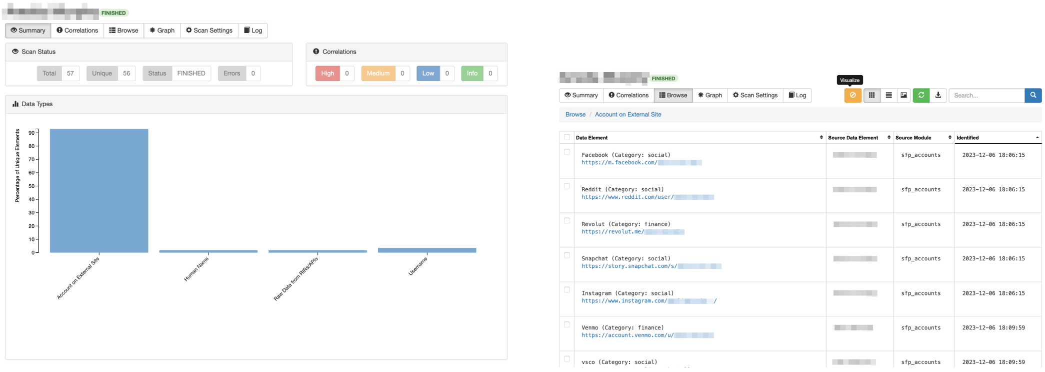
- The search domain should encompass a diverse array of sources.
- Each tool's search domain should include unique sources not covered by other tools.
- The search method employed by each tool should be distinct.
- The data retrieved by the tools should be relevant for generating account access graphs.
- The tool should feature an intuitive user interface or be accompanied by comprehensive documentation for ease of use.
- The data obtained from the tools should be presented in an easily readable and interpretable format.
- The tool should be available for use without requiring a subscription or payment.

The OSINT tools that were employed to extract data for each participant are detailed in the subsequent subsections.

3.1.2.1 Spiderfoot

Spiderfoot is a Python-based reconnaissance tool that integrates over 100 public data sources to gather and analyse information such as email addresses, phone numbers, names, usernames, IP addresses, CIDR ranges, domains, subdomains, and BTC addresses [3]. The tool covers many data sources, including social media platforms, public records databases, email and phone number records, repositories, search engines, data breach databases, and file metadata extraction tools. To maximise *Spiderfoot's* capabilities, optional API keys were obtained for specific modules to broaden the search scope and ensure thorough scans. The tool conducts searches using various combinations and permutations of the target's name to obtain data on each individual, such as their email addresses, phone numbers, usernames and online accounts. For instance, it either inserts a full stop between the target's first and last names or appends an underscore at the end of their first and last names. For example foo.bar@gmail.com or @foobar_ . Additionally, the tool's codebase was expanded to enhance the tool's efficiency and broaden the search scope by incorporating various symbols in different positions within potential usernames and email addresses. Numbers were also introduced into the search space, along with the initials of the participants' first and last names. For example, these credentials could take the form of @foo.bar_, foo.bar3@gmail.com, or f.bar1@yahoo.com. *Spiderfoot* utilises a command-line interface and a web-based GUI for conducting searches [17]. The web-based GUI was primarily utilised in this study due to its ease of interpretability and to mitigate the risk of accidentally overlooking retrieved data, which is more likely to occur when using the command-line interface.

The search target is specified using the GUI, which then displays the acquired data, as show in Figure 3.1.



Total	Unique	Status	Errors
57	56	FINISHED	0

High	Medium	Low	Info
0	0	0	0

Data Element	Source Data Element	Source Module	Identified
Facebook (Category: social) https://m.facebook.com/[redacted]	[redacted]	sfp_accounts	2023-12-06 18:06:15
Reddit (Category: social) https://www.reddit.com/user/[redacted]	[redacted]	sfp_accounts	2023-12-06 18:06:15
Revolut (Category: finance) https://revolut.me/[redacted]	[redacted]	sfp_accounts	2023-12-06 18:06:15
Snapchat (Category: social) https://story.snapchat.com/s/[redacted]	[redacted]	sfp_accounts	2023-12-06 18:06:15
Instagram (Category: social) https://www.instagram.com/[redacted]/	[redacted]	sfp_accounts	2023-12-06 18:06:15
Venmo (Category: finance) https://account.venmo.com/u/[redacted]	[redacted]	sfp_accounts	2023-12-06 18:09:59
vsco (Category: social)	[redacted]	sfp_accounts	2023-12-06 18:09:59

(a) Overview of Data Obtained on Participant

(b) Example of Results Obtained from Search

Figure 3.1: The Presentation of Data Obtained using Spiderfoot in the GUI

3.1.2.2 Maltego

Maltego is a Java program that functions as a data mining tool to extract information from various public sources such as DNS records, Whois Records, search engines, and social networks. This tool allows graphs to be created consisting of names, email addresses, phone numbers, and other identifiable information, such as the usernames of accounts. Once the graph for a participant is created, *transforms* are invoked to carry out a search using the provided data to further derive information on a target [12]. An example of the graph created for each participant is provided in Figure3.2.

For individual users, this program has two available versions: *Maltego CE* and *Maltego Pro*. Each version uses a different number of sources. In this project, the *Maltego CE* version was utilised due to the prohibitive price point of 4999 euros for the Pro version. *Maltego CE* provides limited access to data sources and APIs (which require additional API keys obtained through website registration). However, the APIs that are available for use are restricted based on the program version [22]. This subsequently affected the amount of data that could be collected on each participant using this tool. The *Maltego CE* version can only extract 12 results per API use for a search on an individual, whereas the *Maltego Pro* version can return up to 64,000 results. Additionally, the *Maltego Pro* version has access to commercial data, which the *Maltego CE* version lacks.

3.1.2.3 theHarvester

theHarvester is a Python data scraping tool that utilises over 30 different data sources to obtain emails, subdomains, social media profiles, IPs, and URLs related to a target. This tool utilises search engines such as Google, Bing, Yahoo, and social media platforms like Twitter and Trello, as well as miscellaneous data sources such as DNSdumpster and the Exalead metadata engine[11]. Furthermore, Netcraft Data Mining and the AlienVault Open Threat Exchange are also used to further the tool's goal. To ensure the tool's full utilisation, several API keys were obtained to access specific data sources

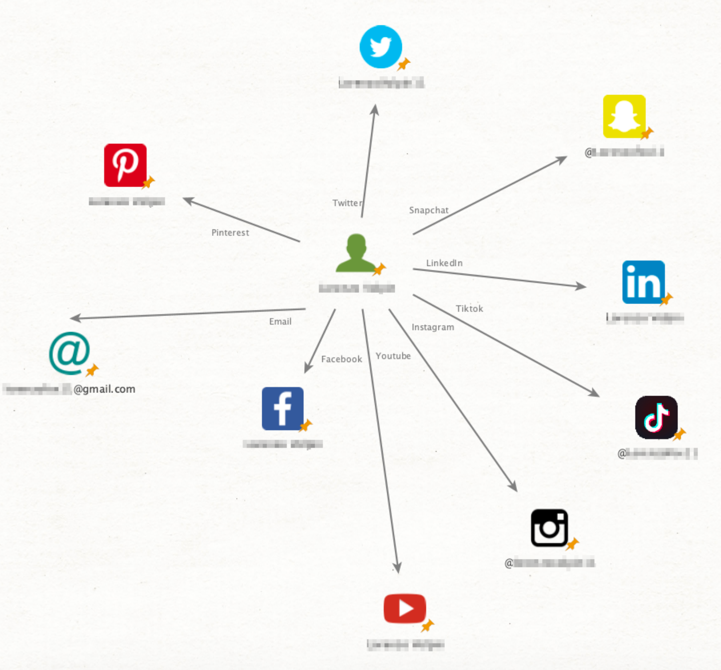
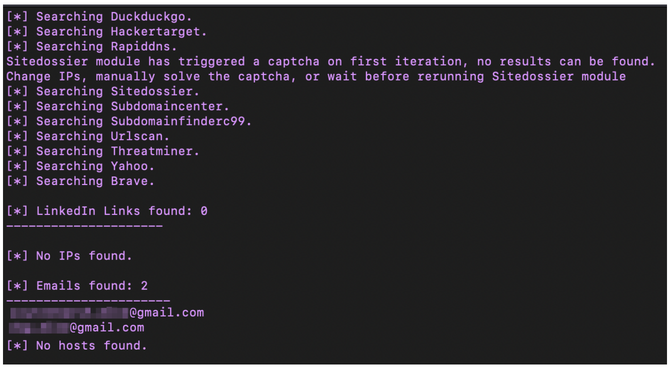


Figure 3.2: Graph Created Using Participant Data to Discover their Phone Number

and maximise the tool's search space. *theHarvester* utilises a command-line interface to conduct searches where results are also displayed. In Figure 3.3, a search was carried out to discover a participant's email address using their name and the queried email domain.

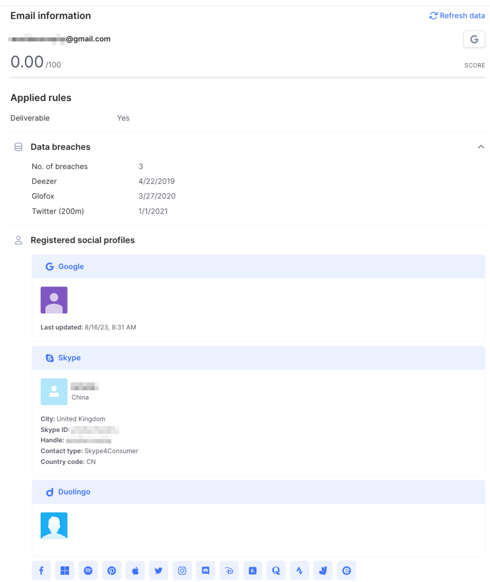


```
[*] Searching Duckduckgo.
[*] Searching Hackertarget.
[*] Searching Rapiddns.
Sitedossier module has triggered a captcha on first iteration, no results can be found.
Change IPs, manually solve the captcha, or wait before rerunning Sitedossier module
[*] Searching Sitedossier.
[*] Searching Subdomaincenter.
[*] Searching Subdomainfinderc99.
[*] Searching Urlscan.
[*] Searching Threatminer.
[*] Searching Yahoo.
[*] Searching Brave.

[*] LinkedIn Links found: 0
---------------------

[*] No IPs found.

[*] Emails found: 2
----------------------
[redacted]@gmail.com
[redacted]@gmail.com
[*] No hosts found.
```

Figure 3.3: Email Addresses Obtained for Participant Following Search

3.1.2.4 SEON Reverse Email/Phone Number Lookup

SEON Reverse Email/Phone Number Lookup is a data enrichment tool that searches public records and databases to find associations and all instances associated with the queried email address or phone number. It retrieves information about the owner of the queried credential, including personal data such as their name, address, and online accounts. In some cases, links to the accounts discovered are provided. This tool operates by being given an email address or phone number from its user, where it scans publicly available records to find matches between the input query and data found in public records. Finally, it presents the matched records. Figure 3.4 illustrates the results of a reverse email lookup conducted using one of the participant's email addresses. For this project, the *SEON Reverse Email address/Phone Number lookup* tool was used, which only required registering an account with them and starting a 30-day free trial [9]. However, finding an appropriate reverse lookup tool required extensive trial and error. This was the case because most reverse email/phone number lookups are based in the US, and so, accordingly, they utilised US databases, which was not helpful for the study's participants. Therefore, no results could be obtained using these lookup tools. Additionally, many reverse lookup tools found through search engines returned results related to the individual's name, address, previous addresses, and public records rather than their credentials and accounts. At the same time, other reverse lookup services obscured their results behind a paywall, with no guarantee that they were genuine. Consequently, these types of reverse lookup tools could not be utilised.



Email information

[redacted]@gmail.com

0.00 /100 SCORE

Applied rules

Deliverable Yes

Data breaches

No. of breaches	3
Deezer	4/22/2019
Glofox	3/27/2020
Twitter (200m)	1/1/2021

Registered social profiles

Google

Last updated: 8/16/23, 8:31 AM

Skype

[redacted]
China

City: United Kingdom
Skype ID: [redacted]
Handle: [redacted]
Contact type: Skype4Consumer
Country code: CN

Duolingo

Figure 3.4: Results Obtained From Reverse Email Lookup Tool

3.1.3 Methodology for Obtaining Participant Data Using OSINT Tools

The process of obtaining data for each participant in the study was approached from the perspective of an attacker. Initially, the only identifiable information used to begin the search for a participant's accounts and credentials was their first and last names. A *TOR* browser was used to ensure the integrity of the data and prevent any bias from previous search results or online interactions. This browser provided anonymity, ensured private browsing sessions, and concealed the source IP address and browsing habits, mimicking the actions of an attacker [19].

The steps of how data was obtained for each participant were as follows and are illustrated in Figure 3.5:

1. The first tool used was *Spiderfoot*, where a search was conducted based on the

participant's name. The accounts and the associated usernames found were then verified to determine whether they belonged to the participant. Additionally, the credentials (email address, phone number) returned from the search were recorded for later verification using the *SEON Reverse Email/Phone Number Lookup* tool.

2. Secondly, *theHarvester* was employed to harvest as much data on the participant as possible. Searches were conducted using the participant's name and possible email domains to find potential email addresses and social media profiles. Furthermore, as a verification mechanism, searches were conducted using the email addresses found using *Spiderfoot* to scrape the names and social media profiles linked to the email addresses.

3. The *Maltego* program was then utilised to compile all the data gathered from the previous tools into a graph. These graphs facilitated the formation of connections between the data, enabling Maltego to derive patterns and trends and obtain additional data such as accounts, email addresses, and phone numbers.

4. Finally, reverse email and phone number lookups were conducted to verify the validity of the email addresses and phone numbers found. Additionally, these searches aimed to discover the accounts registered to these credentials and to cross-reference them with the accounts previously found to confirm their association with the participant in question. Verifying that the accounts and credentials belonged to the participants was crucial for ensuring the accuracy of the generated graphs and that they were truly representative of the credentials and accounts that belonged to each participant. Any errors in identifying whether an account or credential belonged to a participant could impact their entire account access graph, potentially undermining the accuracy of the security setup model and its integrity. Consequently, unverifiable credentials or accounts were discarded during this process and were not included in their respective graph.

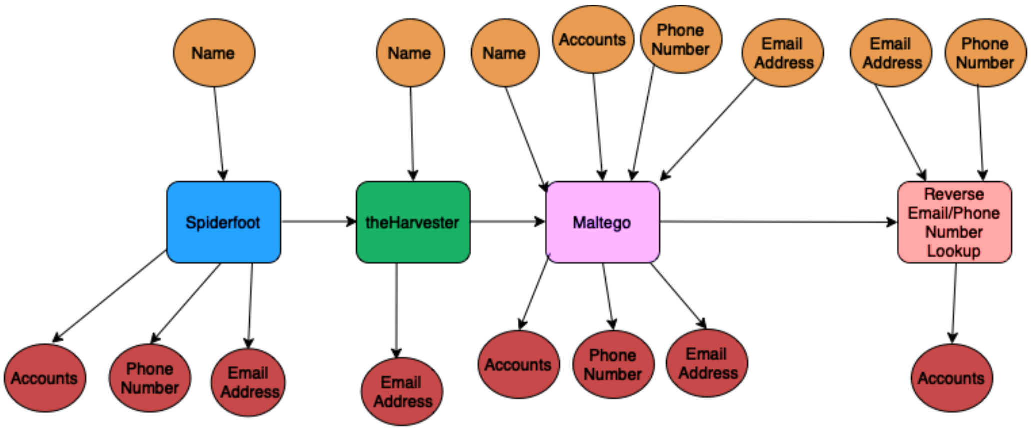


Figure 3.5: Flow chart depicting the sequence of OSINT tools used to obtain data with their respective inputs and outputs

3.1.4 Discarded OSINT Tools Trialed

The process of selecting which OSINT tool to use for the study involved conducting experiments to assess their applicability and determine whether they would extract the data needed to generate the account access graphs. Alternative OSINT tools such as *Recon-ng* and *Mitaka*, which specialise in harvesting data related to IP addresses, domain names and URLs were considered and tested. However, these tools yielded very little to no results for most participants. The little data found on a minority of participants consisted mainly of technical domain information, which was irrelevant to the graphs and, therefore, motivated the omission of these tools from the study. Further tools were tested but, upon review, were found to generate data that is inapplicable to account access graphs and not relevant to the study's requirements. These included *Shodan*, which is a search engine for IoT devices, as well as *searchcode* and *Grep.app*, which are search engines for code. Lastly, tools that scraped dark web archives were considered but not tested. Given the potential ethical implications of obtaining data that was collected illegally and with the consideration of my participants' privacy in mind, the use of these tools was decided against.

3.1.5 Participant Data Collection and Subsequent Account Access Graph Generation

The third stage of the study involved providing each participant with a form to complete at their convenience. This form requested information regarding the accounts and credentials possessed by each participant. Within this form, participants were asked to specify their primary login credentials for each account, including a password identifier to track password sharing across all their accounts. The credentials used as recovery mechanisms for each account were also requested, along with information about the physical devices they owned and utilised for logging into their accounts. Obtaining information about the physical devices used by the participants served the purpose of modelling physical access to a user's security setup. Participants were also asked if they used a password manager. If so, they were asked to indicate which of their passwords had appeared in a data leak to support the analysis of each participant's account access graph. Lastly, to gather participant perceptions regarding the effectiveness of OSINT tools in finding their data, an estimate of the percentage of data they provided that would be discovered using the OSINT tools was obtained.

Once each participant completed their form, they utilised the Java tool I developed specifically for this project's objectives to generate an account access graph using the recorded data. I was present alongside the participant while they were creating their account access graph, offering assistance as needed. After completing the account access graph generation, the participant and I reviewed the graphs generated using data obtained from the OSINT tools and the data provided by the participant to identify any vulnerabilities in their security setup. Based on these graphs, suggestions were made to the participant to strengthen their security measures and enhance their security setup while improving their understanding of online security practices.

3.2 Designing a Java Tool for Generating Account Access Graphs

Following the commencement of the data collection process, a need arose to represent the gathered data effectively. This was achieved using a tool to map the connections between credentials, accounts, and devices based on the inputted data. This tool would then visually represent these connections as an account access graph, aiding in visualising the participant's security setup and enhancing analysis due to its graphical nature. A Java program was developed with a design that aimed to be user-friendly, visually appealing, and inviting, ensuring a satisfying user experience.

3.2.1 Designing the Vertices and Edges

Three distinct types of vertices were incorporated into the tool's design to enhance the clarity and interpretability of the graphs. These vertices could represent an individual's account, credential, or physical device. To make each vertex visually distinct, vertices representing accounts appear as blue rectangles, credentials as red ovals, and physical devices as green trapeziums, as illustrated in Figure 3.6.

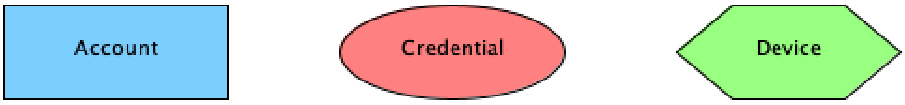


Figure 3.6: Representations of the Different Vertex Types

Following the updated representation of AND and OR connections within the graph definition introduced in [2], edge labels were assigned to each edge to make clear the connections between credentials, accounts and devices and the access methods for a particular vertex. The semantics of the labels local to each target vertex are only relevant for the edges pointing to the same target vertex. A design decision inspired by the original paper was additionally incorporated into the graph's edge design: multi-coloured edges. However, it's important to emphasise that the semantics of using edge colours to signify AND and OR connections within a graph are not applied here. The use of different coloured edges was purely to aid with visualisation and to make the interpretation of which edge labels belong to which edge easier. This design decision was made to account for the size of graphs that could be generated and the number of edges they would subsequently include. It was considered that if all edges were the same colour, it might be challenging to denote which edge label belongs to which edge.

3.2.2 Different Output Format of Graphs

Two different output formats were desired to address varying user needs and increase the tool's functionality. The first output format is a static graph rendered as an image. At the same time, the second is an interactive graph generated using *JavaFX*, enabling users to interact with elements of the graph. With the interactive graph, users can dynamically reposition elements, highlight specific components, and enhance their viewpoint.

The static graph format serves the purpose of quickly creating an image of the graph that can be easily shared and is particularly useful for smaller graphs. On the other hand, the interactive graph format was designed primarily for analysing larger graphs, such as the ones generated in this project. This meant that additional functionality was required to enhance the readability of the graphs that would aid in their analysis. This additional functionality enabled users to customise element positions, highlight elements for analysis, and adjust the viewpoint by zooming in and out and moving the entire graph. The inclusion of zoom-in and zoom-out buttons was aimed at providing users with the flexibility to adjust their preferred viewing size of the graph. Similarly, the select all button enables users to relocate the entire graph to a position that suits their comfort.

3.2.3 Additional Design Decisions Made For Account Access Graphs

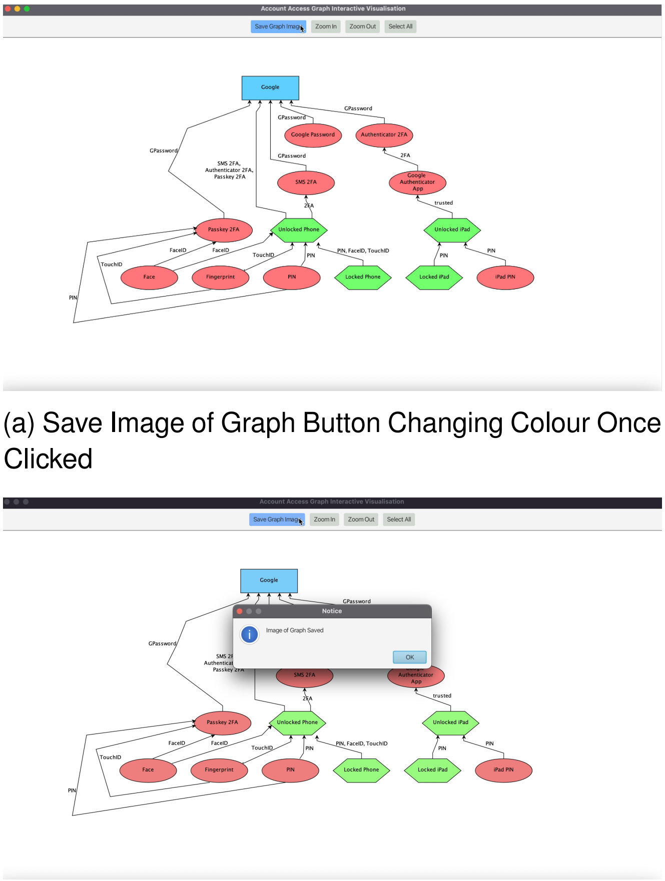
This subsection highlights design decisions made regarding the access graphs generated:

- The addition of states introduced in [2] was not included in the graphs generated in this project. This decision was made because the graphs generated in this project do not model the changes in access resulting from a participant's or malicious actor's actions. Therefore, it was deemed unnecessary.
- Vertices representing password managers were omitted from the account access graphs generated using participant data. This decision was made because all participants reported using a local password manager, such as Apple Keychain, which cannot be accessed online. They can only be unlocked with the exact authentication mechanisms used for their devices. Furthermore, the omission of the password manager was driven by visualisation concerns. The addition of a password manager vertex with its associated edges risked overwhelming the already densely populated participant graphs, reducing the focus on the connections between accounts and credentials.
- Since the graphs generated from OSINT data lacked information regarding the participant's devices, a placeholder vertex representing all of the participant's unlocked devices was used. This allowed the access methods of an account or credential vertex to be included while signifying that this specific data point could not be obtained.
- To reduce the verbosity and the size of the edge labels in the graph, an account's recovery mechanism is indicated by the use of the word *recovery* preceding at least one of the propositions in the logical statement denoting the combination of credentials required to access an account.

3.2.4 Graphical User Interface of the Interactive Graph

When designing the tool's graphical user interface (GUI), emphasis was placed on ensuring an intuitive and consistent interface that aligns with the tool's primary purpose: analysing the account access graph of an individual's security setup. Nielsen's 10

Usability Heuristics for User Interface Design [14] was heavily utilised to optimise the GUI. A minimalist design approach was adopted, aligning with Nielsen's Heuristic of *Aesthetic and Minimalist Design*, to ensure it supports the tool's primary objective. This was further achieved by clearly separating the generated account access graph from the buttons to prevent clutter and make the graph more visually appealing. Furthermore, the design adhered to Nielsen's Heuristic of Visibility of System Status by providing users with appropriate feedback. For instance, when the button responsible for saving an image of the graph is clicked, it changes to blue for a short period to indicate the action taken before returning to the original colour. To further confirm this action, a pop-up message is displayed to confirm the user was successful in saving the image, further reassuring the user of the action taken. The employment of these heuristics is shown in Figure 3.7a 3.7b.



(a) Save Image of Graph Button Changing Colour Once Clicked

Notice

Image of Graph Saved

OK

(b) Pop-Up Message Further Confirming Image was Saved Successfully

Figure 3.7: Process of Saving Image of Graph in GUI

3.2.5 Overall System Design

Overall, it was crucial to ensure that the components of the Java tool incorporated strong coding practices to create a robust system capable of fulfilling the required functionality, delivering the intended output and supporting maintainability and future enhancements.

The coding practices integral to the tool's design included modularity, which involved breaking down components into smaller sub-components with singular purposes, easing maintenance and minimising the requirement for debugging. Encapsulation further mitigated risks by preventing unintended modifications to object states, such as when adding vertices or edges, while promoting code reusability. The tool could be modified without affecting its overall behaviour through maximised decoupling, further enhancing maintainability. Additionally, by adhering to consistent and descriptive naming conventions, the readability and maintainability of the tool was enhanced. The structure and organisation of the tool can be found in Figure 3.8.

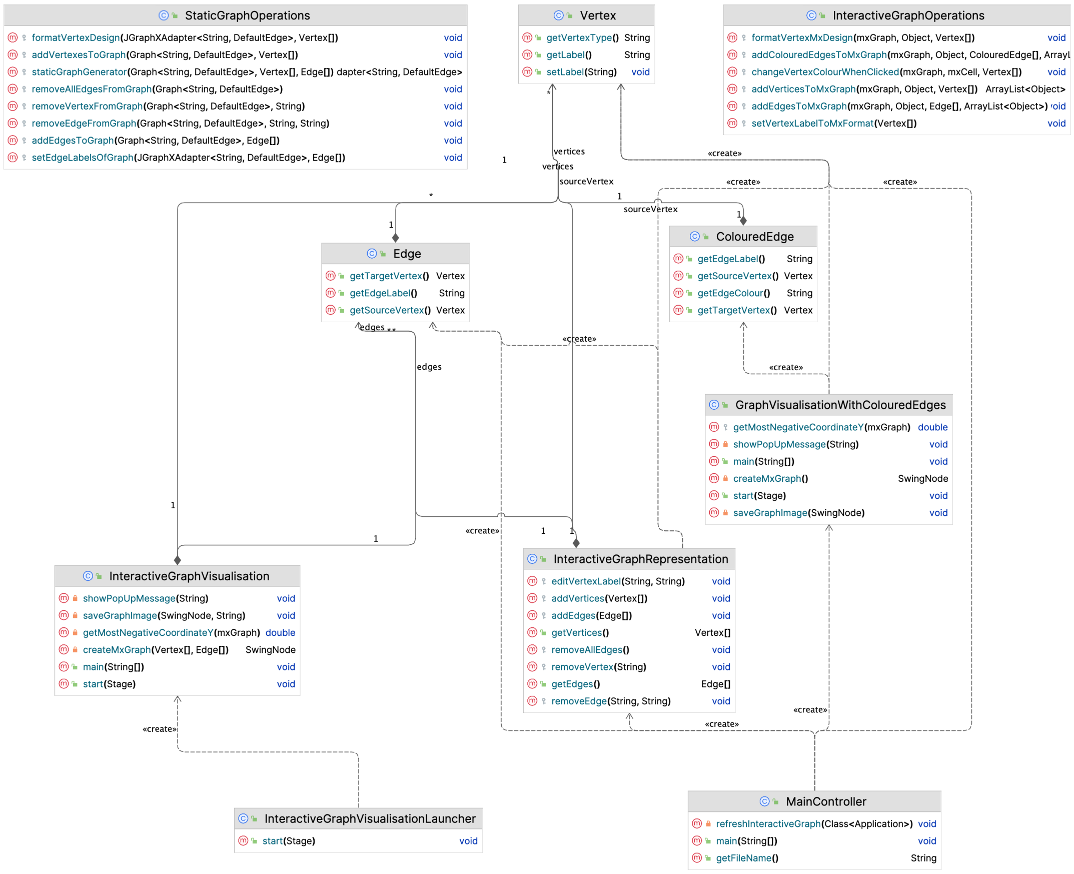

StaticGraphOperations
formatVertexDesign(JGraphXAdapter<String, DefaultEdge>, Vertex[]) void
addVertexesToGraph(Graph<String, DefaultEdge>, Vertex[]) void
staticGraphGenerator(Graph<String, DefaultEdge>, Vertex[], Edge[]) dapter<String, DefaultEdge>
removeAllEdgesFromGraph(Graph<String, DefaultEdge>) void
removeVertexFromGraph(Graph<String, DefaultEdge>, String) void
removeEdgeFromGraph(Graph<String, DefaultEdge>, String, String) void
addEdgesToGraph(Graph<String, DefaultEdge>, Edge[]) void
setEdgeLabelsOfGraph(JGraphXAdapter<String, DefaultEdge>, Edge[]) void
Vertex
getVertexType() String
getLabel() String
setLabel(String) void
InteractiveGraphOperations
formatVertexMxDesign(mxGraph, Object, Vertex[]) void
addColouredEdgesToMxGraph(mxGraph, Object, ColouredEdge[], ArrayL
changeVertexColourWhenClicked(mxGraph, mxCell, Vertex[]) void
addVerticesToMxGraph(mxGraph, Object, Vertex[]) ArrayList<Object>
addEdgesToMxGraph(mxGraph, Object, Edge[], ArrayList<Object>) 'oid
setVertexLabelToMxFormat(Vertex[]) void
*
vertices
vertices
sourceVertex
1
*
1
sourceVertex
«create»
«create»
«create»
1
1
Edge
getTargetVertex() Vertex
getEdgeLabel() String
getSourceVertex() Vertex
edges
edges
«create»
ColouredEdge
getEdgeLabel() String
getSourceVertex() Vertex
getEdgeColour() String
getTargetVertex() Vertex
«create»
GraphVisualisationWithColouredEdges
getMostNegativeCoordinateY(mxGraph) double
showPopUpMessage(String) void
main(String[]) void
createMxGraph() SwingNode
start(Stage) void
saveGraphImage(SwingNode) void
1
1
«create»
1
1
InteractiveGraphVisualisation
showPopUpMessage(String) void
saveGraphImage(SwingNode, String) void
getMostNegativeCoordinateY(mxGraph) double
createMxGraph(Vertex[], Edge[]) SwingNode
main(String[]) void
start(Stage) void
InteractiveGraphRepresentation
editVertexLabel(String, String) void
addVertices(Vertex[]) void
addEdges(Edge[]) void
getVertices() Vertex[]
removeAllEdges() void
removeVertex(String) void
getEdges() Edge[]
removeEdge(String, String) void
«create»
«create»
«create»
InteractiveGraphVisualisationLauncher
start(Stage) void
MainController
refreshInteractiveGraph(Class<Application>) void
main(String[]) void
getFileName() String

Figure 3.8: UML Diagram of Java Tool

Chapter 4

Implementation and Testing

4.1 Implementation of the Java Tool

4.1.1 Initialisation of Graph Structure

Specific classes were created to represent the vertices and edges of an account access graph. Each vertex in the graph is initialised with a label and a type, which can be one of three options: *Account*, *Credential* or *Device*. Moreover, the edges of the account access graph were also represented as a separate class, where each edge is initialised using the source vertex, the target vertex, and the edge label. These class representations provide the benefits of encapsulation and abstraction. For instance, they allow the graphical representation of each vertex to vary based on its type while maintaining a behaviour common to all vertices of the graph. Furthermore, class representations enhance code reusability by allowing behaviors related to creating, editing and interacting with vertices and edges within the graphs to be applied uniformly for different types of vertices and edges. This eliminates the need for alterations and enables efficient code reuse.

4.1.2 Output Format 1: Static Image

The *JGraphT* library was used to construct the directed graph structure for the static graph representation. The utilisation of this library provided benefits in terms of efficiency and reliability. The *StaticGraphOperations* class was developed as a utility class to provide a set of static methods tailored for manipulating the graph data structure. These methods were abstracted from the graph generation logic to enhance code organisation, re-usability, and maintainability. This class encompassed essential functionalities for graph manipulation, including vertex and edge addition and removal and vertex design formatting to provide distinct representations based on the different types of vertices within the graph. Additionally, it overrides the default edge formatting set by the *JGraphT* library, which was *Source : Target* to adhere to the format introduced in [2].

The *JGraphX* library was utilised to visually represent the graph and apply a hierarchical layout. This layout ensured that the orientation of the generated graphs matched the bottom-up topology of those presented initially in [8] and [2]. Additionally, this library allowed the customisation of the graph's general visual design using stylesheets. Finally, the *mxGraph* library was employed to render the graph and generate an image using the *BufferedImage* library.

4.1.3 Output Format 2: Interactive Graph Created using JavaFX

Creating the interactive account access graph required a specialised helper class for graph manipulation. This led to the creation of the *InteractiveGraphRepresentation* class, which stores the graph's vertices and edges and provides the methods responsible for adding and removing vertices and edges and editing vertex labels. The *InteractiveGraphOperation* class implemented methods for formatting vertex design and labels, as well as enabling the highlighting of different elements in the graph. This approach ensured the separation of concerns, with each class responsible for a specific aspect of the application, thereby enhancing code understandability and maintainability.

Unlike generating an image of the graph, creating the interactive graph required a *JavaFX* application class to create and store the graph data. The *start* method of this class serves as the main entry point for the *JavaFX* application. It only accepts the application's primary window as a parameter, which acts as the container for the visual elements of the user interface. To address this limitation, the *InteractiveGraphVisualisation* class, containing the methods responsible for generating the account access graph in the *JavaFX window*, is initialised by passing a list of vertices and edges to its constructor. Subsequently, the vertices and edges can be accessed from the created instance to generate the account access graph in a *JavaFX* window for interactivity.

Displaying the interactive account access graph involved utilising the *SwingNode* library to display content within the *JavaFX* window. The graph structure was constructed using the *MxGraph* library, which enabled the addition and removal of vertices and edges and the editing of vertex labels for anonymisation purposes. After incorporating the desired vertices and edges into the graph and applying consistent formatting, the *MxGraph* object was converted to a *MxGraphComponent* object. This conversion facilitates user interaction with the graph, enabling them to select and drag or highlight vertices and edges within the graph and display the graph bottom-up. Finally, this class implements the functionality necessary for saving the graph state, designing the layout of the *JavaFX* window and adding the desired buttons. These buttons enable saving an image of the current graph state, zooming in/out, and selecting all graph elements.

4.1.4 Main Controller

The *MainController* class is the central component responsible for managing user input, directing user information to the appropriate methods for graph generation, and encapsulating the main code logic that governs the program behaviour. Additionally, it orchestrates the overall flow of the application, including the program loop. User input is obtained through the *Scanner* class. This input includes the user's preferences for generating either a static account access graph as an image or an interactive graph in *JavaFX*, specifying the desired number of vertices and edges and providing the data necessary for creating vertices and edges. Once the initial graphs are generated based on user input and displayed, the program enters a loop that allows dynamicity in both the static and interactive graph generation alternatives. This loop can only be exited by the user. Within this loop, the user is prompted to determine whether they wish to add new vertices or edges to their current graph, remove existing vertices or edges, edit vertex labels for anonymisation, or exit the program.

Input validation is employed for all user inputs to ensure the proper functioning of the program and to prevent unintended crashes or exits due to user error. Input validation mechanisms are invoked in the following situations:

- Inputting numerical values, such as when specifying the type of graph, the number of vertices and edges to add or remove, and choosing the following action after generating the initial graph.
- Addition of a vertex and only allowing the following vertex types to be added:(*Account*, *Credential*, or *Device*)
- Addition of an edge and only allowing an edge to be created between two previously initialised vertices.
- Removal of vertices and edges that must already exist in the graph
- Editing the label of an existing vertex which must already be initialised

Overall, the implementation of the Java tool consisted of 10 classes and 1692 lines of code.

4.2 Testing the Java Tool

Testing was conducted to ensure code quality and error-free generation of account access graphs and to enhance the usability, interaction experience, and maintainability of the Java tool implemented. Testing was carried out by conducting 20 unit tests, each with varying test cases. The tests primarily focused on verifying the functionality responsible for modifying the state of the graph. This included testing the code responsible for adding and removing vertices and edges as well as editing the labels associated with the graph's elements. Due to the distinct graph data structures used for the two potential output formats of the graphs, unique methods were required to execute the same functionalities for both format equivalents. Therefore, unit tests were composed for all methods responsible for altering the graph state. This ensured equal focus was placed on verifying the graph generation's behaviour in both output formats.

Tests were designed with the primary function of each method in mind while also considering edge cases. For example, a method responsible for removing a vertex from a graph would be tested for its primary functionality. Still, it would also consider the edge case where the method is invoked to remove a vertex from the graph that does not exist. Furthermore, it was essential to consider how modifications to the graph state would impact its internal data structure. Therefore, verifying that changes made to the graph's state were accurately reflected in its data structure was imperative. For example, when adding an edge, it was necessary to ensure that the source and target vertices were not unintentionally swapped, as doing so would alter the semantics of the connection between the two vertices. Another consideration was that the consistency of the graph should remain constant throughout the program's execution despite changes made to the graph state. It was imperative to ensure no inconsistencies that would impact the user experience arose. An example scenario would be that removing a vertex should also ensure the removal of all incoming and outgoing edges of the removed vertex.

Usability testing was also conducted with participants as they interacted with the tool to generate their account access graphs. Participants were asked to provide feedback on the tool and suggest improvements in their interaction experience. Based on participant feedback, two areas of improvement were identified in the tool's GUI, which were subsequently addressed.

During the analysis of the account access graph generated using a participant's data, the participant found it challenging to interpret and read the text within the graph. This difficulty arose due to the lack of a zoom-in and zoom-out feature, which heavily influenced the graph's viewpoint. Smaller graphs were found to be easier to read. In contrast, larger ones posed more significant challenges due to their size and the challenges associated with fitting the larger graph within the user's window. To account for the feedback received and use it as an opportunity to improve the tool, a zoom-in and zoom-out feature was implemented using buttons positioned at the top of the application window. This feature allows users to change their scale of view to enhance graph readability and maintain graph visibility irrespective of graph size.

Secondly, while rearranging the vertices of a graph, a participant wanted to shift the entire graph closer to the bottom of the window. This was desired to increase the space between the hierarchies in the graph to enhance their viewing pleasure. However, the participant could only achieve this by individually selecting each element in the graph, which became a tedious process. To address the inconvenience experienced by the participant, the functionality allowing a user to select all elements of the graph was implemented. This enables users to preserve consistent vertex spacing and maintain the relative positions of vertices and edges while allowing the graph to be dragged to a more desirable position.

Chapter 5

Experimental Results and Evaluation

5.1 OSINT Tool Proficiency in Obtaining Data

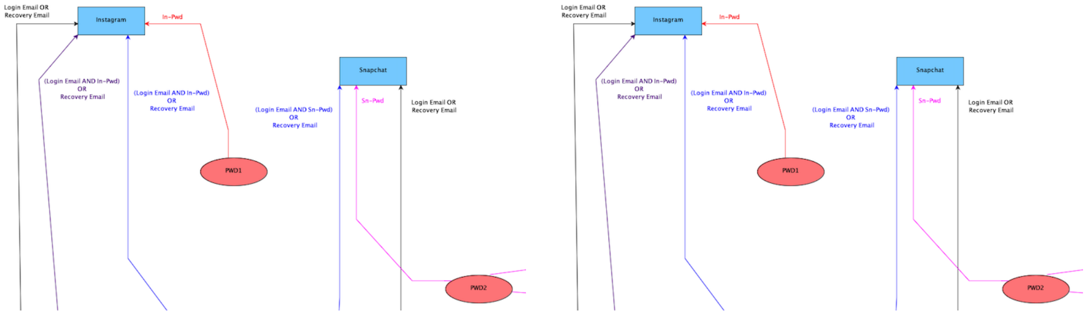
The OSINT tools utilised in the study allowed a substantial amount of data for each participant to be collected, enabling the creation of their account access graph with multiple vertices and edges. The breadth of data obtained was made possible by the collective coverage of the various search domains of these tools. Assuming that the study's participants provided all their email addresses and phone numbers, the OSINT tools demonstrated considerable proficiency in uncovering their credentials, as illustrated in Table5.1. The credentials found served as critical identifiers for their respective accounts and facilitated the connections between the other accounts owned by the participant. Moreover, the tools demonstrated remarkable performance in locating the participants' accounts, as evidenced by the data presented in Table5.1. No previous work was found that measured the effectiveness of OSINT tools in obtaining data on a particular subject.

5.1.1 Effectiveness in Discovering Participant Email Addresses

For each participant, at least one email address was uncovered. Notably, participants 4, 5, 7, 8, and 10 had all of their email addresses successfully identified using the suite of OSINT tools employed. This outcome was expected, given the search strategy of the Spiderfoot tool and the common practice of email addresses incorporating variations of an individual's name through different name arrangements and the use of initials. Notably, participants 5 and 8 had their university email addresses discovered. These email addresses are formatted as sXXXXXXX@ed.ac.uk and do not contain their names. Instead, each *X* represents a number between 0 and 9. This highlights the tools' ability to uncover email addresses that do not rely solely on participant names.

5.1.2 Effectiveness in Discovering Participant Phone Numbers

Finding each participant's phone number proved more challenging, as only half of the participants had their phone numbers discovered. This lower success rate can be attributed to the fact that phone numbers lack identifiable information that can be directly linked to a participant and are often considered more permanent and personal compared to an email address. As a result, individuals are less likely to share their phone numbers publicly, preferring to keep this credential more confidential. The risk of an individual's phone number being leaked or exposed to malicious adversaries is perceived to be more significant than an email address. Therefore, individuals are more inclined to link their email addresses to online accounts than phone numbers. This pattern was consistent across all study participants, where email addresses were more frequently used as primary credentials than phone numbers. This conclusion is further supported by the account access graphs generated for each participant, where the vertex representing the primary email account of each participant possessed a higher number of outgoing edges than the phone number vertex. Furthermore, participant 5's phone number was uncovered using OSINT tools; however, it was found not to be used as a primary login credential for any account. This was later confirmed by the information sheet provided by the participant. The tools also revealed that some participants employed their phone numbers as primary credentials, in addition to their email addresses, despite not disclosing these associations themselves, as illustrated in Figure 5.1. This underscores a lack of awareness among certain participants regarding all authentication methods enabled in their accounts, potentially posing a security risk.



(a) Data Provided by Participant

(b) Data Obtained from OSINT Tools

Figure 5.1: Snippet of Different Account Access Graphs highlighting the different possible authentication methods of the Instagram and Snapchat Accounts of Participant 1

5.1.3 Abundance of Data Discovered Compared to Provided

Overall, for most participants, more than half of their data was found using the OSINT tools employed in the study as illustrated in Table 5.1. Furthermore, for 6 out of the 10 participants, more accounts were discovered than those listed in their information sheets, emphasising the effectiveness of the OSINT tools used. Even in cases where participants provided more accounts than those identified by the tools, the disparity was minimal, with a maximum difference of 6 accounts. Conversely, the most significant increase in the number of accounts found compared to those provided was 13 for Participant 7. Generally, the surplus of additional accounts discovered allowed for the consideration and acknowledgement of previously overlooked accounts in the participants' security setups.

However, it is vital to consider that it is likely that the study participants provided their most important and frequently used accounts rather than all the accounts they possessed. This is evident from the abundance of additional accounts uncovered by the OSINT tools. In some cases, the number of additional accounts found exceeded those provided by participants, indicating that each participant has a larger online footprint than initially realised. Therefore, it's unwise to assume all the credentials and accounts of a participant have been captured from the combination of user-provided data, and the data discovered using OSINT tools. The discovery and inclusion of the additional accounts found within the participant's account access graph highlights the potential to reveal previously unknown connections and vulnerabilities. The additional accounts discovered may possess weaker security measures than the participant's primary accounts, potentially exposing sensitive information to malicious actors. For instance, if a participant inadvertently uses the same password for one of the additional accounts discovered and their primary accounts, a malicious adversary's discovery of this password could allow unauthorised access to both the primary and additionally discovered accounts.

Participant	Data Provided by Participants			Data Obtained Using OSINT Tools		
	Number of Email Addresses	Number of Phone Numbers	Number of Accounts	Number of Email Addresses	Number of Phone Numbers	Number of Accounts
1	2	1	13	1	1	24
2	2	1	16	1	1	21
3	3	1	24	2	1	18
4	1	1	14	1	1	24
5	2	1	16	2	1	27
6	3	1	28	1	0	22
7	1	1	6	1	0	19
8	2	1	26	2	0	24
9	2	1	15	1	0	21
10	1	1	13	1	0	11

Table 5.1: Comparison of Abundance of Data Provided and Obtained using OSINT Tools

5.1.4 Deriving how Connected Credentials are in User Setups

The credentials uncovered using OSINT tools can be examined to determine their interconnectedness within a participant's account ecosystem. This connectedness is determined by the number of outgoing edges from the vertex representing each credential in the participant's account access graph, as demonstrated in Table5.2. The most connected credential found for each participant was their email account, with the highest number of outgoing edges found to be 22 for Participant 6's iCloud Mail account. The significant number of connections a participant's email accounts possess emphasises their criticality within their ecosystem and their overarching influence on the accounts within it. Hence, participants should especially prioritise implementing robust authentication and recovery mechanisms for these accounts. This is crucial due to the potential for these accounts to be uncovered online and the subsequent links that can be made to the participant's other online accounts, potentially putting them at risk.

Conversely, from an adversary's perspective, the same OSINT tools can provide insights into the criticality of the credentials they can uncover within an individual's account ecosystem. Using this knowledge, adversaries can strategically target accounts with high connectivity to more efficiently compromise the user's account ecosystem.

Participant	Credential Uncovered	Number of Outgoing Edges
1	Gmail-1	21
	Phone Number	4
2	Gmail-1	19
	Phone Number	3
3	Gmail-1	7
	Gmail-2	7
	Phone Number	3
4	Hotmail	19
	Phone Number	3
5	Gmail	21
	Outlook	3
	Phone Number	0
6	iCloud Mail	22
7	Gmail	18
8	Yahoo	20
	Outlook	2
9	Gmail	20
10	Yahoo	10

Table 5.2: Connectedness of Participant Credentials

5.1.5 Underestimation of the Data Coverage of the OSINT Tools

The data coverage provided by these OSINT tools exceeded the estimates of all participants except one, as illustrated in Table5.3. This highlights the participants' underestimation of the scope of online data collection using these tools and their effectiveness in uncovering their data. The participants' lack of awareness emphasises the risks associated with malicious actors utilising such tools to determine an individual's account security setup. For instance, participant 7's estimate differed by 50% from the actual data percentage found, illustrating the disparity. The participant who provided a higher estimate than the amount of data found justified their estimate based on their unique name. Their reasoning was well-founded, as no false positives existed among the credentials and accounts found using the OSINT tools.

Participant Number	Participant's Estimate of Data Found (%)	Actual Percentage of Provided Data Found (%)	Number of Additional Accounts Found
1	35	56	17
2	45	63	11
3	50	54	6
4	80	69	15
5	70	79	15
6	20	37.5	11
7	25	75	14
8	30	52	11
9	50	56	12
10	25	27	8

Table 5.3: Participant Perception of Data Coverage of Tools Compared to Reality

5.2 High-Level Analysis of Account Access Graphs

5.2.1 Account Access Graph Generated Using OSINT Data

Comparing the account access graphs generated from OSINT-obtained data with those from the participant-provided data allows one to assess the significance of the email accounts discovered by OSINT tools. Based on the number of outgoing edges to other vertices and by comparing the two graphs, it was determined that the OSINT tools found the primary email accounts of all participants.

In addition, a large number of accounts were consistently uncovered for all participants. These accounts encompassed various categories such as social media, online shopping, streaming services and miscellaneous platforms. Many of these accounts hold significance, as evidenced by the fact that the participants initially provided a substantial portion of the discovered accounts and subsequently incorporated them into their graphs.

Additionally, essential accounts such as Apple ID, Microsoft, and various digital payment apps like PayPal, Revolut, and Venmo were discovered. An Apple ID account is integral for using Apple devices; without it, the functionality of these devices would be severely limited. Furthermore, digital payment apps often store sensitive financial information, including bank account and credit card details, and facilitate money transfers between accounts. All participants who possessed an Apple ID or Microsoft account had these accounts uncovered. Furthermore, participants 4, 6, and 8 had their digital payment accounts discovered and incorporated into their account access graphs.

The ability of OSINT tools to reveal a diverse range of accounts, which can be integrated into an account access graph, offers the advantage of creating a comprehensive model of an individual's security configuration. However, from a malicious perspective, the broad spectrum of uncovered accounts provides numerous potential targets for attackers to exploit across various domains. For instance, if a malicious adversary gains access to a participant's social media account, they could engage in impersonation or spread hateful propaganda. In the case of digital payment accounts, money could be stolen from participants. In contrast, online shopping accounts might contain stored credit card details, which could be exploited for identity theft or used in a recovery mechanism to gain access to an account, as previously illustrated in Mat Honan's account security compromise [10]. Additionally, compromising an Apple ID account could grant unauthorised access to a participant's data, such as photos, contacts, and messages. Overall, the discovery of these accounts poses significant security risks for participants.

The account access graphs generated using data obtained from OSINT tools for all participants are in Appendix A.

5.2.2 Account Access Graphs Generated Using Data From Participants

The utilisation of data provided by participants allowed for the construction of accurate account access graphs depicting their security setup. These graphs are valuable for conducting security analysis and identifying potential weaknesses in the participants' security configurations. While the graphs generated using data obtained from OSINT tools are useful for visualising internet-accessible credentials and accounts, they do not accurately represent overall account security. This limitation arises because the account access graphs created from the data obtained using OSINT tools have been enriched with examples of authentication and recovery mechanisms that are supported by the respective service providers of the accounts, rather than reflecting the actual authentication and recovery mechanisms used by the participants. These additions enhance the practicality of these graphs but only serve as a potential example of what the participant's account access graph could look like. In this project, example authentication credentials could only be included because participant passwords were not searched for online during the OSINT data collection process due to ethical concerns. However, in reality, malicious adversaries are likely to search for leaked passwords online, employing methods such as *RockYou* lists [5] and password cracking techniques to determine passwords.

The account access graphs generated for all participants using the data they provided can be found in Appendix B

5.2.2.1 Lack of 2FA/MFA Adoption and Password Sharing

Overall, the account access graphs of each participant showcased unique intricacies, yet overlapping trends were observed. Across all ten participants, a prevalent pattern was the consistent use of a single email address/phone number and password combination for account logins, spanning various account types, including social media, shopping, and miscellaneous accounts like video streaming or web blogs. Additionally, two-factor authentication (2FA) was sparingly utilised and was primarily used for critical accounts like AppleID, online banking, and digital payment services. However, there were even some cases where participants accessed these critical accounts using only passwords (e.g. AppleID for participants 1, 3, 4, 5, 6, 7). Furthermore, despite the availability of 2FA, many participants relied solely on password authentication for at least one of their email accounts.

Common 2FA methods employed by participants included using One Time Codes (OTCs) via email, text, or Authenticator app, as well as using secret keys. Notably, OTCs were more frequently utilised for account recovery than primary authentication. For instance, Participant 3's Trading212 account can be primarily accessed using a login email address and password. Password recovery for the account is achievable through two methods: confirming the recovery attempt via an authenticator app and receiving a recovery OTC on the participant's phone or using a secret key and receiving a recovery OTC on the participant's phone. The primary and secondary authentication mechanisms of Participant 3's Trading 212 account are illustrated in Figure 5.2.Otherwise, access to participants' accounts was recovered solely through email recovery, a weak security measure susceptible to exploitation by malicious actors if the participant's email account had been compromised.

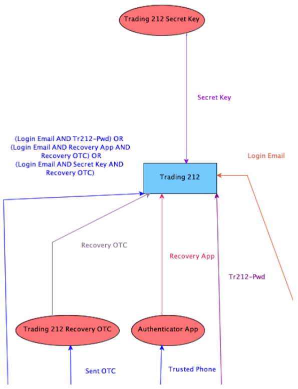


Figure 5.2: Snippet of Trading212 Account Vertex in Participant 3's Account Access Graph

Participants were additionally prone to engage in password sharing between accounts. Some participants were found to use unique passwords for their more critical accounts, such as their AppleID, financial and digital payment accounts and email accounts, while sharing passwords across the rest of their accounts. Others tended to use the same password for their more critical accounts while using different passwords for their less important ones. This case is illustrated in the account access graph of Participant 10 in Figure B.10. Interestingly, a common trend observed among participants was the use of the same password for their university email account and all linked accounts. This trend was observed with participants 3, 5, and 6. Most notably, participant 5 employed a single password shared only between their university-associated accounts.

5.2.2.2 Attitude of Participants

The minimal adoption of 2FA/MFA and the use of password sharing highlights participant attitudes toward account security, suggesting a preference for convenience over security and a streamlined login process over the additional security measures of 2FA/MFA. Furthermore, they might perceive a low risk of unauthorised access to their accounts by malicious actors and consider themselves unlikely targets for cyberattacks.

The reduced adoption of 2FA/MFA and password sharing may ultimately stem from a lack of awareness and understanding of the implications of their security choices.

5.2.2.3 Centrality Score of Participant Account Access Graphs

Deriving the central vertices of the participant's account access graph through centrality scoring offers valuable insight into the vertices with the highest potential security risk if compromised. A higher centrality score indicates a greater criticality of the vertex's compromise to an individual's security setup. This is because a vertex connected to numerous other accounts provides potential access points to a larger network of accounts, thus increasing the coverage of compromised accounts. This project determines the centrality score by the number of outgoing edges from a vertex in the graph. While centrality scores can also be calculated using distance-based scoring schemes, they are not applicable in these graphs due to their size and the arrangement of vertices, which prioritise viewability and interpretability.

The results provided in Table 5.4 highlight that the most central vertex of an account access graph is typically a participant's unlocked phone, email account, or unlocked laptop. This underscores the significance of scenarios where an adversary gains access to a participant's phone or laptop and unlocks it or if they gain access to an email account associated with most other accounts. If the most central vertex is an unlocked device, a local attacker (one who can physically access the device) possesses the potential to compromise a large portion of a participant's security setup. On the other hand, if an email account is the most central vertex, it becomes a more attractive target for remote attackers. The fact that an email account is one of the three most central vertices for each participant emphasises how much damage a malicious remote adversary can do to a participant's security setup.

Participant Number	Most Central Vertex	Second Most Central Vertex	Third Most Central Vertex
1	Unlocked Phone, Unlocked Laptop		Gmail-1
2	Gmail-1	Unlocked Phone	Unlocked Laptop
3	Unlocked Phone	Unlocked Laptop	Gmail-1
4	Unlocked Phone	Unlocked Laptop	Hotmail
5	Unlocked Phone, Gmail		Unlocked Laptop
6	Unlocked Phone	iCloud Mail	Unlocked Laptop
7	Unlocked Phone	Gmail	Unlocked Laptop
8	Unlocked Phone	Yahoo Mail	Unlocked Laptop
9	Unlocked Phone	Gmail, Unlocked Laptop, Unlocked Computer	
10	Unlocked Phone	Unlocked Laptop	Yahoo

Table 5.4: Most Central Vertices in Participant Account Access Graphs

5.3 Focus Point – Presence of a Cycle Between Email Accounts

Analysing intricate patterns in an account access graph can reveal vulnerabilities beyond weak password practices or the lack of 2FA/MFA. Cycles are structural features that indicate potential weaknesses involving multiple accounts that can recover each other directly or indirectly. Vertices found within cycles often have high centrality scores. Cycles were found in several participants' graphs, notably impacting Participant 9's security setup. In their graph, a cycle between their Gmail and AOL email accounts was discovered. Both these accounts are crucial in the individual's security setup, particularly because the participant doesn't use their phone number as a primary credential. Instead, all accounts are linked to the participant's email accounts. Moreover, the *G-Mail* account vertex ranked as the joint second most central vertex in this participant's graph. A cycle is created between these two email accounts, given that both the Gmail and AOL accounts can be used to recover each other. For a malicious actor to gain access to the participant's Gmail account through recovery mechanisms, they would require knowledge of the login Gmail address and access to the AOL email account, and vice versa, as illustrated in Figure 5.3. The AOL email account only requires the account password for access due to the lack of more secure authentication methods.

If a malicious actor were to obtain the password for the AOL account and subsequently gain entry to this account, they could uncover the participant's Gmail account using OSINT tools, as demonstrated during the participant study. Alternatively, the adversary might uncover the participant's Gmail account by exploring the settings of the AOL email account, as the Gmail account is used to recover access to the AOL account. Following gaining access to the AOL email account and learning of the participant's Gmail account, the malicious actor can proceed to trigger the Gmail account's recovery mechanism. An OTC would be sent to the compromised AOL email account, allowing the adversary to reset the participant's Gmail password. Subsequently, the adversary can log out of the Gmail account on all devices and change the associated phone number, effectively blocking the participant's access to the account's other recovery option. To further exacerbate the situation, the weak recovery mechanisms employed by many of the participant's other accounts allow password resets via a recovery link or OTC sent to the participant's Gmail account. Consequently, a malicious actor could access the following accounts: Facebook, Instagram, Snapchat, Spotify, Pinterest, Letterboxd, Reddit, Duolingo, Dropout, AskMyGp, Deliveroo, and Amazon.

5.4 Focus Point – Backdoor Access To A Critical Account

Modelling the recovery mechanisms of each account within a participant's account access graph is critical for identifying potential backdoor access points. Backdoor access implies that an account can be more easily accessed through its recovery mechanism than primary authentication methods, rendering it vulnerable to exploitation by attackers. Participant 2's account access graph analysis revealed a backdoor access point to their

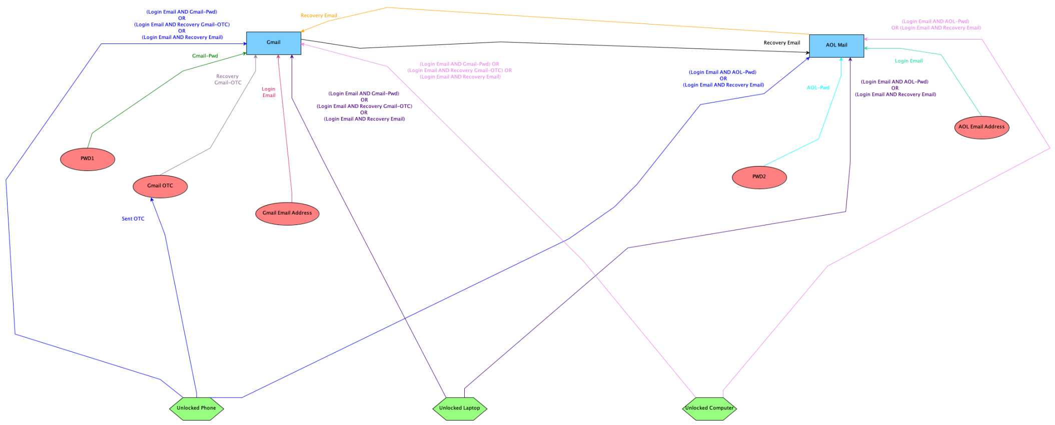

(Login Email AND Gmail-Pwd)
OR
(Login Email AND Recovery Gmail-OTC)
OR
(Login Email AND Recovery Email)
Recovery Email
Gmail
Gmail-Pwd
Recovery
Gmail-OTC
Login
Email
(Login Email AND Gmail-Pwd)
OR
(Login Email AND Recovery Gmail-OTC)
OR
(Login Email AND Recovery Email)
(Login Email AND Gmail-Pwd) OR
(Login Email AND Recovery Gmail-OTC) OR
(Login Email AND Recovery Email)
PWD1
Gmail OTC
Sent OTC
Gmail Email Address
Recovery Email
AOL Mail
(Login Email AND AOL-Pwd)
OR (Login Email AND Recovery Email)
Login Email
(Login Email AND AOL-Pwd)
OR
(Login Email AND Recovery Email)
AOL-Pwd
(Login Email AND AOL-Pwd)
OR
(Login Email AND Recovery Email)
AOL Email Address
PWD2
Unlocked Phone
Unlocked Laptop
Unlocked Computer

Figure 5.3: Partial Account Access Graph Denoting Connections between Participant 9's Email Accounts

primary Gmail account. This vulnerability assumes that the attacker attempting to gain backdoor access is a password attacker capable of compromising passwords that will facilitate gaining unauthorised entry into the account through a backdoor.

Participant 2 has two Gmail addresses with differing levels of primary authentication strength. The main Gmail account *Gmail-1* in B.2 utilises 2FA, requiring the participant's login email address, password, and an OTC. In contrast, their secondary Gmail account *Gmail-2* in B.2 only requires the login email address and password. The primary Gmail account can be recovered using an OTC or the secondary Gmail account as a recovery email. In contrast, the secondary Gmail account can only be recovered using the primary Gmail account. Thus, a password attacker can exploit the weak primary authentication methods of the secondary Gmail account to gain backdoor access to the account.

Through the password attacker's ability to compromise passwords, the attacker can learn *Pwd3* (from the account access graph of the participant in B.2). This password is shared between four accounts, three of which can be accessed using only the login email and password, therefore granting the attacker unauthorised access to the Microsoft, SoundCloud, and the secondary Gmail account of the participant. The participant's failure to adopt 2FA or MFA results in the attacker easily gaining access to the secondary Gmail account. This entry point does not constitute backdoor access, as the attacker logs into the account using the primary authentication method. Furthermore, the attacker can discover the participant's primary Gmail account by investigating the secondary Gmail account's settings or by using OSINT tools, as was the case during the participant study. Subsequently, by triggering the recovery mechanism of the primary Gmail account, a recovery OTC is sent to the now compromised secondary Gmail account, allowing the attacker to create a new password and gain access to the primary Gmail account. In this way, the attacker gains backdoor access to the account through a recovery mechanism which employs weaker authentication methods than the primary authentication method.

After gaining entry to the account, the attacker can change the phone number associated with the account using the newly created account password. This ensures OTCs are no longer sent to the participant and allows the attacker access to the account through the primary authentication mechanism. The attacker's compromise of the participant's two email accounts enables them to exploit the weak recovery mechanisms of the following accounts in their account setup: Depop, Letterboxd, Snapchat, Soundcloud, Instagram, Facebook, Amazon, LinkedIn, eBay, GitHub, and Spotify.

5.5 Focus Point – The Consequences of Leaked Password Sharing

Password sharing is strongly discouraged due to its well-known risks. While creating strong passwords with a combination of characters, numbers, symbols, and non-dictionary words can enhance security [13], it cannot fully mitigate the risks associated with password sharing. These risks are heightened by the absence of both additional primary authentication methods used alongside passwords for account access and robust recovery authentication mechanisms.

Participant 3's account access graph highlights the vulnerabilities and severe consequences of password sharing within their account security setup. The information-forms issued to participants during the study prompted them to record any passwords that had appeared in data leaks, as indicated by their password manager. This process was aimed to emphasise the widespread leakage of passwords and to support the analysis provided by their account access graph. In Participant 3's case, a single compromised password had cascading effects across multiple accounts, further illustrating the significance of this issue. Participant 3 uses *Pwd7* to access three accounts: two Gmail accounts and their EE account, as illustrated in Figure 5.4. This password was flagged as compromised in the participant's information form due to its appearance in a data leak.

In the case where a malicious actor seeks to compromise Participant 3's security setup, they may use OSINT tools to uncover the participant's credentials and accounts. The credentials found by the malicious actor would likely match those discovered for Participant 3 during the study. Subsequently, they could search through databases containing leaked data and use the information they previously obtained through OSINT to find the login email address and password *PWD7* required to access the participant's EE account. Since both Gmail addresses were uncovered using OSINT tools in the study, as shown in Participant 3's account access graph, there is a high likelihood that an attacker would find the same accounts and attempt to exploit password reuse behaviour by trying to access the participant's Gmail accounts using *Pwd7*. With both Gmail accounts requiring only the email address and password for primary authentication, the malicious actor's attempts would grant them access to both accounts.

The consequences of a compromised shared password are magnified by the participant's lack of 2FA/MFA for their email accounts. This is particularly concerning considering the critical role email accounts play in an individual's security setup, as demonstrated by the high centrality score of the *Gmail-1* account vertex in their account access graph. As previously observed, gaining access to the participant's email accounts enables access to other linked accounts via their recovery mechanisms. The compromise of these two Gmail accounts due to their shared password with the EE account could further compromise eight additional accounts. Furthermore, if the attacker discovers the participant's Outlook account, they can recover it using the *Gmail-1* account. By changing the password through this recovery mechanism and altering the authenticated device for Microsoft Authenticator, the attacker can access all four accounts linked to the Outlook account.

The participant cannot have prevented their password from leaking, as data leaks often result from security vulnerabilities and poor security practices on the server provider's side. However, the compromise of most user accounts could have been avoided by refraining from password sharing and enhancing the strength of the primary and recovery authentication methods of their accounts.

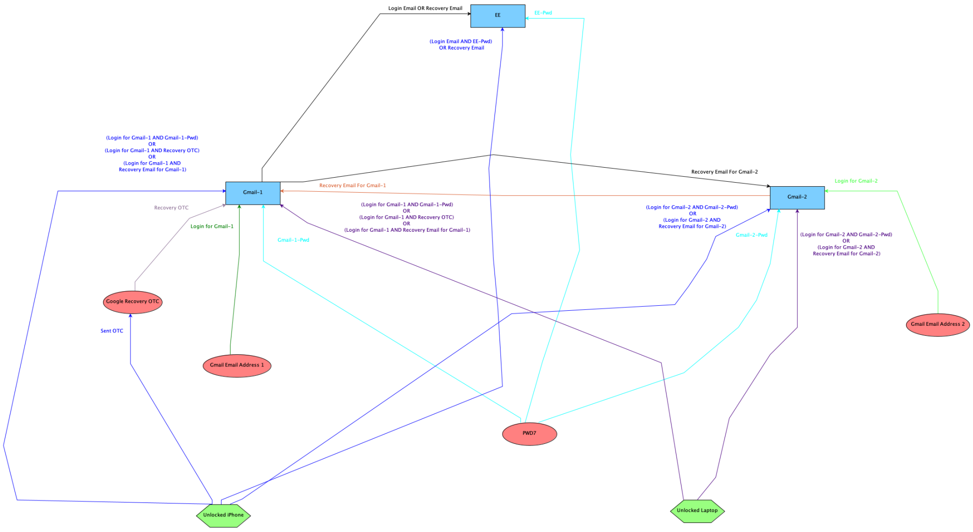

Login Email OR Recovery Email
EE
EE-Pwd
(Login Email AND EE-Pwd)
OR Recovery Email
(Login for Gmail-1 AND Gmail-1-Pwd)
OR
(Login for Gmail-1 AND Recovery OTC)
OR
(Login for Gmail-1 AND
Recovery Email for Gmail-1)
Gmail-1
Recovery Email For Gmail-1
Recovery Email For Gmail-2
Login for Gmail-2
Gmail-2
Recovery OTC
Login for Gmail-1
Gmail-1-Pwd
(Login for Gmail-1 AND Gmail-1-Pwd)
OR
(Login for Gmail-1 AND Recovery OTC)
OR
(Login for Gmail-1 AND Recovery Email for Gmail-1)
(Login for Gmail-2 AND Gmail-2-Pwd)
OR
(Login for Gmail-2 AND
Recovery Email for Gmail-2)
Gmail-2-Pwd
(Login for Gmail-2 AND Gmail-2-Pwd)
OR
(Login for Gmail-2 AND
Recovery Email for Gmail-2)
Google Recovery OTC
Sent OTC
Gmail Email Address 1
Gmail Email Address 2
PWD7
Unlocked iPhone
Unlocked Laptop

Figure 5.4: Partial Account Access Graph Highlighting the Shared Passwords between Accounts

5.6 Challenges Encountered in the Project

5.6.1 Challenges Encountered with the Participant Study

Conducting a participant study provided an opportunity to assess the real-world effectiveness of OSINT tools in gathering data on specific targets, capturing authentic human behaviour and identifying patterns in account security. However, an issue arose with one participant who initially consented to participate in the study and allowed the data collection process using OSINT tools to begin. Unfortunately, the participant ceased communication before providing the necessary data required to generate their account access graph with their actual data. Consequently, their incomplete data rendered it unusable for the project's objectives.

The incident was resolved by replacing the participant with an individual from a list of backup participants initially compiled during the recruitment for the study. This approach ensured that a replacement could be quickly identified if a participant dropped out at the last minute. Moreover, the addition of the replacement participant provided the added benefit of a more diverse participant pool.

Moreover, the return of the information sheets necessary for generating their comparison account access graphs was frequently delayed by participants, often due to forgetfulness or busy schedules. To accommodate their availability and reduce stress, adjustments were made to the study timeline to ensure participants remained willing to participate.

5.6.2 Challenges Encountered During the Implementation of the Java Tool

The initial design of the Java tool was limited to generating account access graph outputs solely as static images. However, testing revealed issues with larger graphs, where overlapping edges and labels made interpretation difficult. Additionally, users were constrained in customising element placement within the graph. To address this, an interactive visualisation was developed using JavaFX. Integrating JavaFX into the existing tool resolved interpretability and visibility issues with the generated graphs while enabling greater customisation. This enhanced user experience and overall tool functionality, resulting in a superior final product. During the implementation phase of the interactive account graph, several challenges were encountered.

Initially, the *JUNG* library was chosen for generating the graph model. However, during implementation, issues arose when the *jung.graph.impl* and *jung.api* packages from the same library conflicted, resulting in runtime errors and rendering the application unusable. Research revealed this was a common, unpatched issue, as the JUNG library is no longer updated. This issue was remedied using the *mxGraph* library. This library was chosen for its suitability to the tool's requirements and seamless integration. It additionally ensured consistency between the graph outputs, whether generated as images or displayed in a JavaFX application window.

Another challenge arose when applying a hierarchical layout to the interactive graph displayed. To ensure the graph was displayed bottom-up to the user, the constant *SwiftConstant.South* was passed to the hierarchical layout function in the *mxGraph* library. However, this resulted in negative y-coordinates for the graph vertices, rendering them invisible in the JavaFX window. Given that no patch was issued for this bug by the library authors, a helper function was implemented to incrementally adjust the y-coordinates of the vertices following the program's initialisation. This made the graph visible and allowed users to move the graph elements to different positions if desired.

The last challenge encountered was that JavaFX applications are limited to a single launch within the same JVM. This obstructed the desired functionality of enabling users to dynamically add or remove vertices and edges to the generated graphs, enhancing usability and user experience. To address this limitation, a helper function employing threading was developed to enable the desired dynamic functionality for the interactive graph.

Chapter 6

Conclusions

6.1 Summary

This project aimed to assess the effectiveness of Open-Source Intelligence Tools in uncovering individuals' credentials and accounts, while also analysing participants' security setups for exploitable weaknesses using account access graphs in a participant study.

The credentials and accounts of each participant were obtained using OSINT tools and provided directly by the participants themselves to create two account access graphs. One account access graph modelled the data obtained using OSINT tools, while the second depicted the participants' actual security setups. Both were generated using a Java tool designed and implemented for this project. Following their generation, analysis was conducted on both graphs. The account access graphs generated using OSINT data highlighted the credentials, accounts, and connections obtained across the internet. This facilitated the analysis of the tools' proficiency and revealed the extent to which an individual's security setup can be uncovered online.

The account access graphs generated using participant-provided data served two main purposes. Firstly, they enabled the evaluation of the accuracy of the data obtained via OSINT tools. Secondly, they facilitated the analysis of each participant's security setup, enabling the identification of vulnerabilities arising from weak security measures of individual accounts and the connections between accounts. The analysis of each participant's security setup was conducted collaboratively to identify vulnerabilities and to educate them on potential improvements that could be made to minimise the security risk of their accounts and overall setup.

Based on the results of the evaluation, the OSINT tools utilised in the participant study exhibited a strong proficiency in uncovering the participants' email accounts, phone numbers, and online accounts. This was highlighted by the discovery of at least one email account for all participants, and in some cases, all email accounts possessed by a participant were found. Furthermore, the phone numbers of half of the participants were uncovered, while a substantial number of accounts were additionally found for each participant. In some cases, more accounts were found online for a participant than those provided by the participants themselves. The proficiency of OSINT tools in uncovering a significant portion of each participant's account security setup underscores the significance of maintaining a robust security setup, including strong authentication methods and reducing the connections between accounts.

The analysis of the account access graphs generated using participant-provided data highlighted common behaviours among the study's participants. Participants were found to rarely employ secure authentication methods such as 2FA/MFA, except for important accounts, instead opting for single-password authentication for most of their accounts. Password sharing and the daisy-chaining of critical accounts within their security setup were also extensively observed.

6.2 Critical Evaluation of Own Work

One limitation of this work that arises is the generalisability of the results obtained due to the skewed sample of participants, primarily consisting of individuals at the extremes of the age range. Time constraints prevented a more diverse participant pool from being acquired, which would have provided a broader representation of all age groups. Additionally, a significant portion of the participants were students whose behaviour regarding account security and awareness may not accurately reflect that of the wider population.

Another limitation was the exclusion of participant passwords in the data collected using OSINT tools. In reality, a malicious actor attempting to determine an individual's security setup would likely search for leaked passwords, use password rules and *RockYou* lists, in addition to their credentials and accounts. This additional data would enable them to construct a more comprehensive model of an individual's entire security setup using data only found online.

One limitation identified in previous work is that the scoring schemes introduced to indicate the security level of an account, do not account for the ease with which a user's credentials or account can be uncovered online. Despite aiming to consider additional factors beyond authentication methods, these scoring schemes overlook this crucial factor. This is an important factor to consider because credentials or accounts that are uncovered online are at a higher risk of compromise once discovered.

6.3 Future Work

A potential area for future work could involve conducting a participant study where participants are taught how to use OSINT tools to obtain data on themselves. This would enable them to search for their own passwords online and determine which of their passwords can be found using these tools. Allowing participants to find and anonymise their data would address the ethical concerns of searching for participant passwords. Subsequently, the email accounts, phone numbers, online accounts, and passwords discovered could be modelled using an account access graph to assess how much a participant's security setup can be discovered across the internet.

Bibliography

[1] Yasemin Acar et al. “Developers need support, too: A survey of security advice for software developers”. In: *2017 IEEE Cybersecurity Development (SecDev)*. IEEE. 2017, pp. 22–26.

[2] Luca Arnaboldi et al. “Tactics for Account Access Graphs”. In: *European Symposium on Research in Computer Security*. Springer. 2023, pp. 452–470.

[3] *Attack Surface Documentation*. `https://intel471.com/attack-surface-documentation`. URL: `https://intel471.com/attack-surface-documentation`.

[4] Just Baker. *What is OSINT Open Source Intelligence? - crowdstrike*. Feb. 2023. URL: `https://www.crowdstrike.com/cybersecurity-101/osint-open-source-intelligence/`.

[5] William J. Burns. *Common Password List (rockyou.txt)*. Jan. 2019. URL: `https://www.kaggle.com/datasets/wjburns/common-password-list-rockyoutxt`.

[6] Ritu Gill. *What is Open-Source Intelligence?* Feb. 2023. URL: `https://www.sans.org/blog/what-is-open-source-intelligence/`.

[7] Sven Hammann et al. “I’m surprised so much is connected”. In: *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 2022, pp. 1–13.

[8] Sven Hammann et al. “User account access graphs”. In: *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 2019, pp. 1405–1422.

[9] Sam Holland. *Reverse email lookup: How it works & how to perform it*. SEON. Aug. 2023. URL: `https://seon.io/resources/reverse-email-lookup/`.

[10] Mat Honan. *How Apple and Amazon Security Flaws Led to My Epic Hacking*. Aug. 2012. URL: `https://www.wired.com/2012/08/apple-amazon-mat-honan-hacking/`.

[11] Laramies. *TheHarvester: E-mails, subdomains and names Harvester - OSINT*. GitHub. 2016. URL: `https://github.com/laramies/theHarvester`.

[12] *Maltego*. Homepage. 2018. URL: `https://www.maltego.com/`.

[13] Microsoft. *Create and Use Strong Passwords*. URL: `https://support.microsoft.com/en-gb/windows/create-and-use-strong-passwords-c5cebb49-8c53-4f5e-2bc4-fe357ca048eb`.

[14] Jakob Nielsen. *10 usability heuristics for user interface design*. Nielsen Norman Group. Feb. 2024. URL: `https://www.nngroup.com/articles/ten-usability-heuristics/`.

[15] Ani Petrosyan. *Global password habits by age group 2017*. Statista. 2023. URL: https://www.statista.com/statistics/803831/password-habits-worldwide-age/.

[16] Daniela Pöhn et al. "A Framework for Analyzing Authentication Risks in Account Networks". In: 2023.

[17] Smicallef. *Smicallef/Spiderfoot: Spiderfoot automates OSINT for threat intelligence and mapping your attack surface*. GitHub. Jan. 2020. URL: https://github.com/smicallef/spiderfoot.

[18] Joanna Stern and Nicole Nguyen. *A Basic iPhone Feature Helps Criminals Steal Your Entire...* Feb. 2023. URL: https://www.wsj.com/tech/personal-tech/apple-iphone-security-theft-passcode-data-privacya-basic-iphone-feature-helps-criminals-steal-your-digital-life-cbf14b1a.

[19] *Tor Project — Anonymity Online*. 2006. URL: https://www.torproject.org/.

[20] Tena Velki and Ksenija Romstein. "User risky behavior and security awareness through lifespan". In: *International journal of electrical and computer engineering systems* 9.2 (2018), pp. 53–60.

[21] Monica Whitty et al. "Individual differences in cyber security behaviors: an examination of who is sharing passwords". In: *Cyberpsychology, Behavior, and Social Networking* 18.1 (2015), pp. 3–7.

[22] wondersmith_rae. *A beginner's guide to OSINT investigation with Maltego*. Medium. Feb. 2020. URL: https://wondersmithrae.medium.com/a-beginners-guide-to-osint-investigation-with-maltego-6b195f7245cc.

[23] Jinghao Zhao et al. "SecureSIM: Rethinking authentication and access control for SIM/eSIM". In: *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*. 2021, pp. 451–464.

Appendix A

Account Access Graphs Generated Using OSINT Data

Due to the number of vertices and edges within the graphs, they are large in size and may require zooming in to increase visibility.

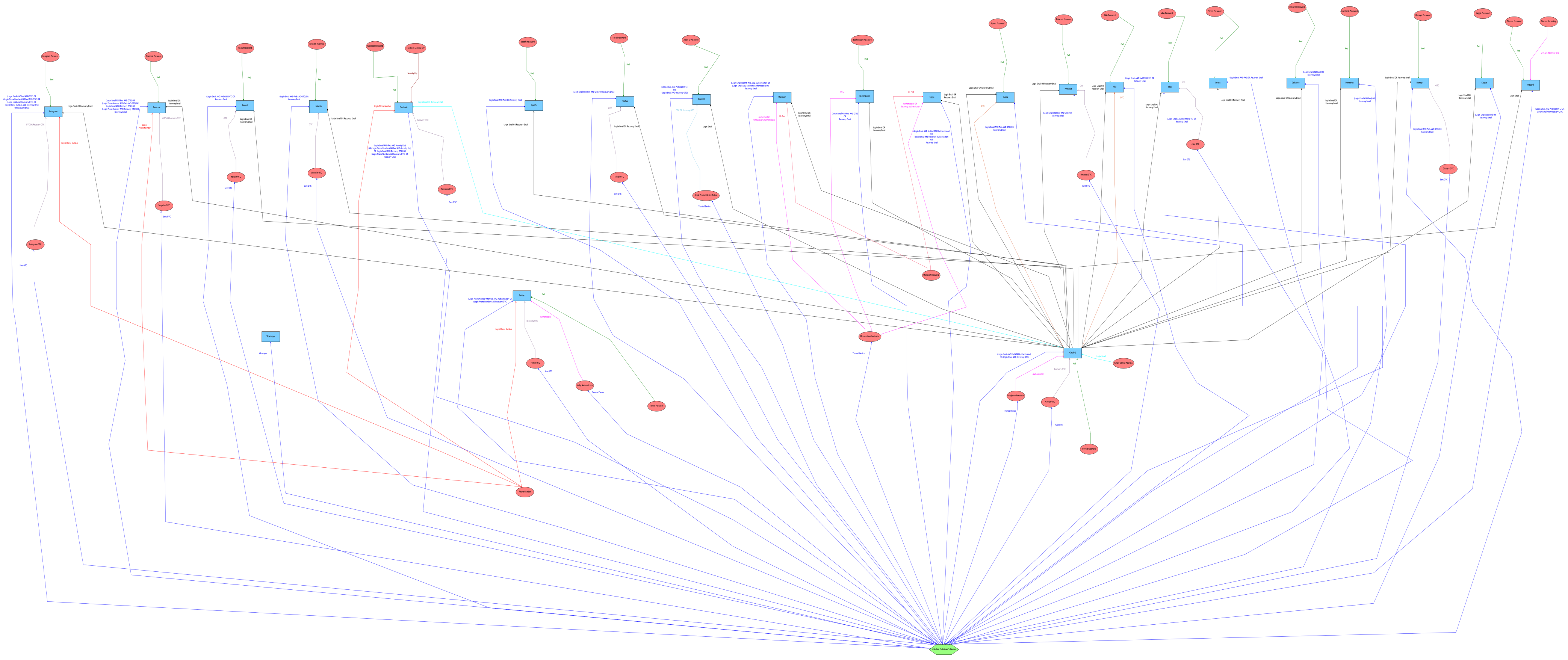

Instagram Password
Pwd
Instagram
(Login Email AND Pwd AND OTC) OR (Login Phone Number AND Pwd AND OTC) OR (Login Email AND Recovery OTC) OR (Login Phone Number AND Recovery OTC) OR Recovery Email
Login Email OR Recovery Email
OTC OR Recovery OTC
Login Phone Number
Instagram OTC
Sent OTC
Snapchat Password
Snapchat
(Login Email AND Pwd AND OTC) OR (Login Phone Number AND Pwd AND OTC) OR (Login Email AND Recovery OTC) OR (Login Phone Number AND Recovery OTC) OR Recovery Email
Login Email OR Recovery Email
Login Phone Number
OTC OR Recovery OTC
Snapchat OTC
Sent OTC
Revolut Password
Revolut
(Login Email AND Pwd AND OTC) OR Recovery Email
OTC
Login Email OR Recovery Email
Revolut OTC
Sent OTC
LinkedIn Password
LinkedIn
(Login Email AND Pwd AND OTC) OR Recovery Email
OTC
Login Email OR Recovery Email
LinkedIn OTC
Sent OTC
Facebook Password
Facebook Security Key
Security Key
Facebook
Login Phone Number
Login Email OR Recovery Email
Recovery OTC
(Login Email AND Pwd AND Security Key) OR (Login Phone Number AND Pwd AND Security Key) OR (Login Email AND Recovery OTC) OR (Login Phone Number AND Recovery OTC) OR Recovery Email
Facebook OTC
Sent OTC
Spotify Password
Spotify
(Login Email AND Pwd) OR Recovery Email
Login Email OR Recovery Email
TikTok Password
TikTok
(Login Email AND Pwd AND OTC) OR Recovery Email
OTC
Login Email OR Recovery Email
TikTok OTC
Sent OTC
Apple ID Password
Apple ID
(Login Email AND Pwd AND OTC) OR (Login Email AND Recovery OTC)
OTC OR Recovery OTC
Login Email
Apple Trusted Device Token
Trusted Device
Microsoft
(Login Email AND Mi-Pwd AND Authenticator) OR (Login Email AND Recovery Authenticator) OR Recovery Email
Authenticator OR Recovery Authenticator
Mi-Pwd
Login Email OR Recovery Email
Microsoft Password
Microsoft Authenticator
Trusted Device
Booking.com Password
Booking.com
OTC
(Login Email AND Pwd AND OTC) OR Recovery Email
Login Email OR Recovery Email
Skype
Sk-Pwd
Authenticator OR Recovery Authenticator
(Login Email AND Sk-Pwd AND Authenticator) OR (Login Email AND Recovery Authenticator) OR Recovery Email
Login Email OR Recovery Email
Quora Password
Quora
Login Email OR Recovery Email
OTC
(Login Email AND Pwd AND OTC) OR Recovery Email
Pinterest Password
Pinterest
Login Email OR Recovery Email
OTC
(Login Email AND Pwd AND OTC) OR Recovery Email
Pinterest OTC
Sent OTC
Nike Password
Nike
Login Email OR Recovery Email
(Login Email AND Pwd AND OTC) OR Recovery Email
OTC
eBay Password
eBay
OTC
Login Email OR Recovery Email
(Login Email AND Pwd AND OTC) OR Recovery Email
eBay OTC
Sent OTC
Strava Password
Strava
(Login Email AND Pwd) OR Recovery Email
Login Email OR Recovery Email
Deliveroo Password
Deliveroo
(Login Email AND Pwd) OR Recovery Email
Login Email OR Recovery Email
Eventbrite Password
Eventbrite
Login Email OR Recovery Email
(Login Email AND Pwd) OR Recovery Email
Disney+ Password
Disney+
Login Email OR Recovery Email
OTC
(Login Email AND Pwd AND OTC) OR Recovery Email
Disney+ OTC
Sent OTC
Kaggle Password
Kaggle
Login Email OR Recovery Email
(Login Email AND Pwd) OR Recovery Email
Discord Password
Discord Secret Key
OTC OR Recovery OTC
Discord
Login Email
(Login Email AND Pwd AND OTC) OR (Login Email AND Recovery OTC)
Twitter
(Login Phone Number AND Pwd AND Authenticator) OR (Login Phone Number AND Recovery OTC)
Pwd
Login Phone Number
Recovery OTC
Authenticator
Twitter OTC
Sent OTC
Authy Authenticator
Trusted Device
Twitter Password
WhatsApp
Whatsapp
Phone Number
Gmail-1
(Login Email AND Pwd AND Authenticator) OR (Login Email AND Recovery OTC)
Login Email
Gmail-1 Email Address
Authenticator
Recovery OTC
Pwd
Google Authenticator
Trusted Device
Google OTC
Sent OTC
Google Password
Unlocked Participant's Devices

Figure A.1: Account Access Graph of Participant 1

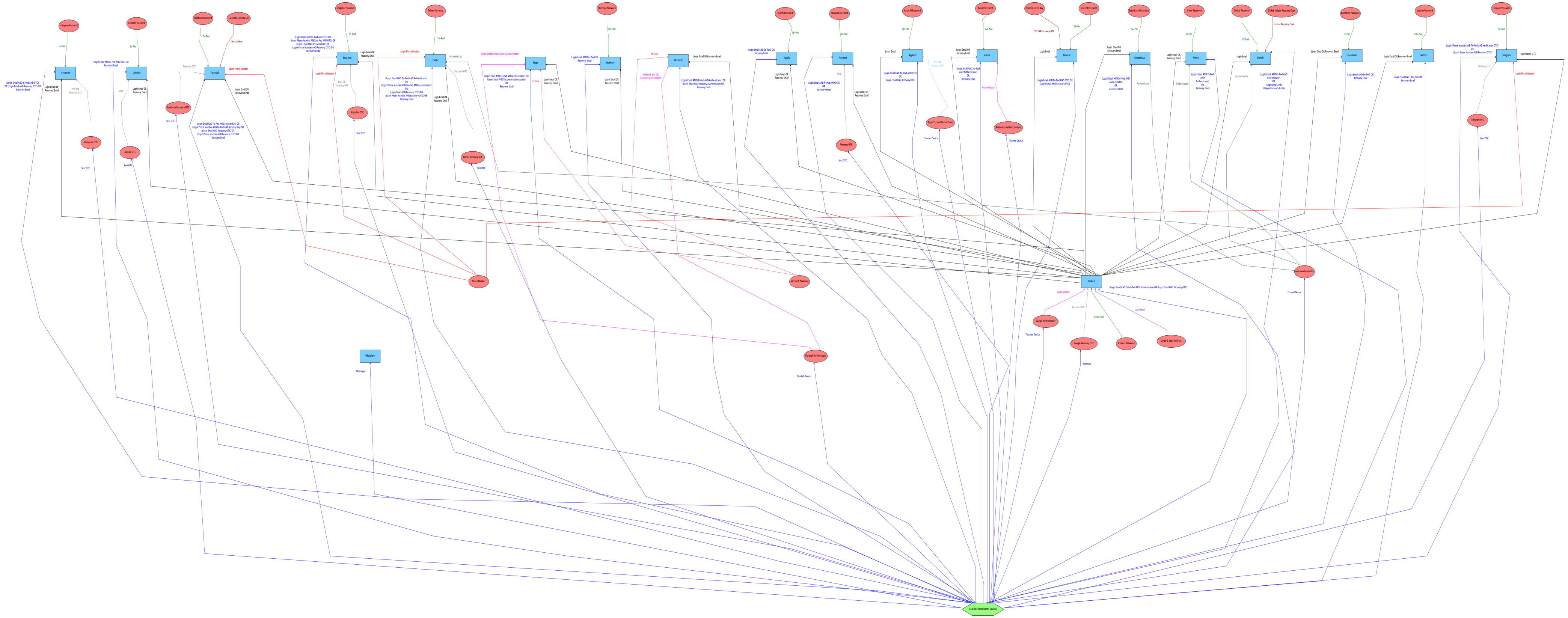

Instagram Password
In-Pwd
Instagram
(Login Email AND In-Pwd AND OTC) OR (Login Email AND Recovery OTC) OR Recovery Email
Login Email OR Recovery Email
OTC OR Recovery OTC
Instagram OTC
Sent OTC
LinkedIn Password
Li-Pwd
LinkedIn
(Login Email AND Li-Pwd AND OTC) OR Recovery Email
Login Email OR Recovery Email
OTC
LinkedIn OTC
Sent OTC
Facebook Password
Facebook Security Key
Fa-Pwd
Security Key
Facebook
Recovery OTC
Login Phone Number
Login Email OR Recovery Email
Facebook Recovery OTC
Sent OTC
(Login Email AND Fa-Pwd AND Security Key) OR (Login Phone Number AND Fa-Pwd AND Security Key) OR (Login Email AND Recovery OTC) OR (Login Phone Number AND Recovery OTC) OR Recovery Email
Snapchat Password
Sn-Pwd
Snapchat
(Login Email AND Sn-Pwd AND OTC) OR (Login Phone Number AND Sn-Pwd AND OTC) OR (Login Email AND Recovery OTC) OR (Login Phone Number AND Recovery OTC) OR Recovery Email
Login Email OR Recovery Email
Login Phone Number
OTC OR Recovery OTC
Snapchat OTC
Sent OTC
Twitter Password
Tw-Pwd
Twitter
Login Phone Number
Authenticator
Recovery OTC
(Login Email AND Tw-Pwd AND Authenticator) OR (Login Phone Number AND Tw-Pwd AND Authenticator) OR (Login Email AND Recovery OTC) OR (Login Phone Number AND Recovery OTC) OR Recovery Email
Login Email OR Recovery Email
Twitter Recovery OTC
Sent OTC
Skype
Authenticator OR Recovery Authenticator
(Login Email AND Sk-Pwd AND Authenticator) OR (Login Email AND Recovery Authenticator) OR Recovery Email
Sk-Pwd
Login Email OR Recovery Email
Duolingo Password
Du-Pwd
Duolingo
(Login Email AND Du-Pwd) OR Recovery Email
Login Email OR Recovery Email
Microsoft
Mi-Pwd
Login Email OR Recovery Email
Authenticator OR Recovery Authenticator
(Login Email AND Mi-Pwd AND Authenticator) OR (Login Email AND Recovery Authenticator) OR Recovery Email
Spotify Password
Sp-Pwd
Spotify
(Login Email AND Sp-Pwd) OR Recovery Email
Login Email OR Recovery Email
Pinterest Password
Pi-Pwd
Pinterest
(Login Email AND Pi-Pwd AND OTC) OR Recovery Email
OTC
Login Email OR Recovery Email
Pinterest OTC
Sent OTC
Apple ID Password
Ap-Pwd
Apple ID
Login Email
(Login Email AND Ap-Pwd AND OTC) OR (Login Email AND Recovery OTC)
OTC OR Recovery OTC
Apple Trusted Device Token
Trusted Device
Adobe Password
Ad-Pwd
Adobe
Login Email OR Recovery Email
(Login Email AND Ad-Pwd AND Authenticator) OR Recovery Email
Authenticator
Adobe Account Access App
Trusted Device
Discord Secret Key
Discord Password
OTC OR Recovery OTC
Di-Pwd
Discord
Login Email
(Login Email AND Di-Pwd AND OTC) OR (Login Email AND Recovery OTC)
Soundcloud Password
So-Pwd
Soundcloud
Login Email OR Recovery Email
(Login Email AND So-Pwd AND Authenticator) OR Recovery Email
Authenticator
Vimeo Password
Vi-Pwd
Vimeo
Login Email OR Recovery Email
(Login Email AND Vi-Pwd AND Authenticator) OR Recovery Email
Authenticator
GitHub Password
GitHub Unique Recovery Code
Gi-Pwd
Unique Recovery Code
GitHub
Login Email
Authenticator
(Login Email AND Gi-Pwd AND Authenticator) OR (Login Email AND Unique Recovery Code)
Eventbrite Password
Ev-Pwd
Eventbrite
Login Email OR Recovery Email
(Login Email AND Ev-Pwd) OR Recovery Email
Last.fm Password
L.fm-Pwd
Last.fm
Login Email OR Recovery Email
(Login Email AND L.fm-Pwd) OR Recovery Email
Telegram Password
Te-Pwd
Telegram
(Login Phone Number AND Te-Pwd AND Verification OTC) OR (Login Phone Number AND Recovery OTC)
Verification OTC
Recovery OTC
Login Phone Number
Telegram OTC
Sent OTC
Phone Number
Microsoft Password
Microsoft Authenticator
Trusted Device
WhatsApp
Whatsapp
Gmail-1
(Login Email AND Gmail-Pwd AND Authenticator) OR (Login Email AND Recovery OTC)
Authenticator
Recovery OTC
Gmail-Pwd
Login Email
Google Authenticator
Trusted Device
Google Recovery OTC
Sent OTC
Gmail-1 Password
Gmail-1 Email Address
Authy Authenticator
Trusted Device
Unlocked Participant's Devices

Figure A.2: Account Access Graph of Participant 2

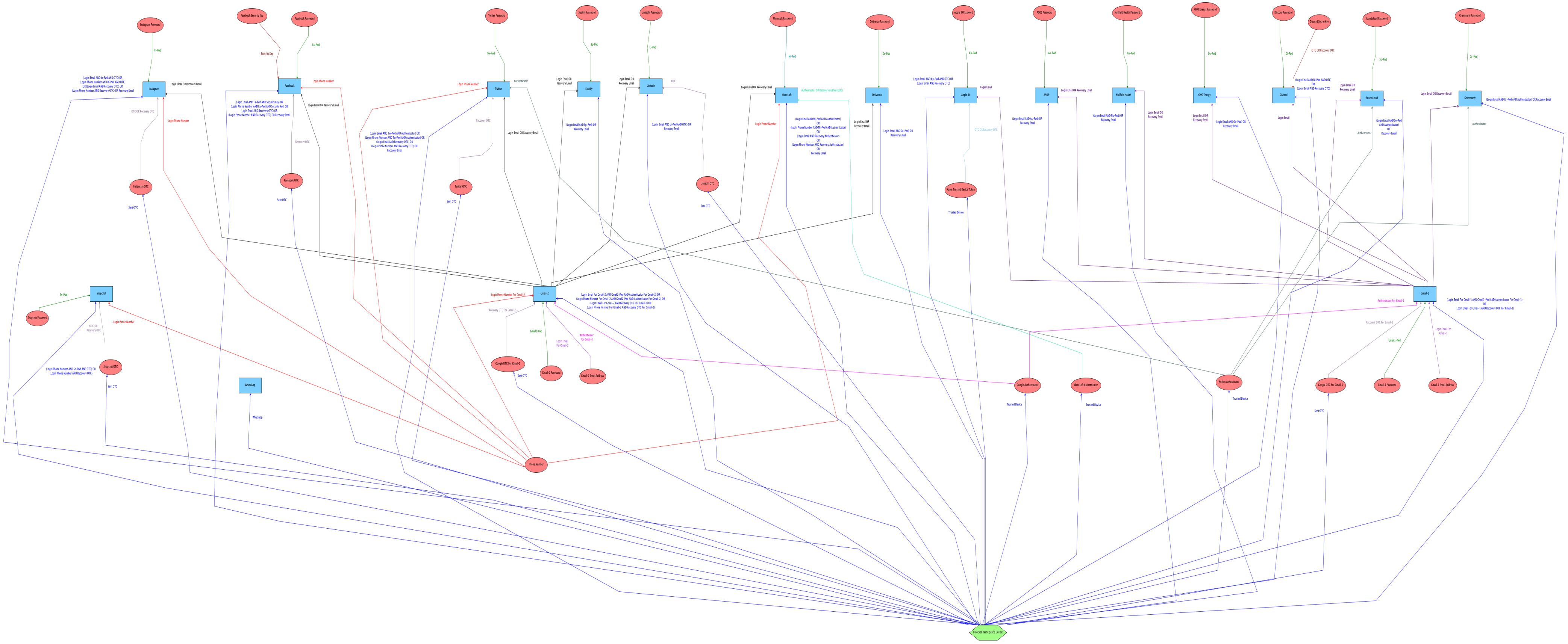

Instagram Password
In-Pwd
(Login Email AND In-Pwd AND OTC) OR
(Login Phone Number AND In-Pwd AND OTC)
OR (Login Email AND Recovery OTC) OR
(Login Phone Number AND Recovery OTC) OR Recovery Email
Instagram
Login Email OR Recovery Email
OTC OR Recovery OTC
Login Phone Number
Instagram OTC
Sent OTC
Facebook Security Key
Facebook Password
Security Key
Fa-Pwd
Facebook
Login Phone Number
(Login Email AND Fa-Pwd AND Security Key) OR
(Login Phone Number AND Fa-Pwd AND Security Key) OR
(Login Email AND Recovery OTC) OR
(Login Phone Number AND Recovery OTC) OR Recovery Email
Login Email OR Recovery Email
Recovery OTC
Facebook OTC
Sent OTC
Twitter Password
Tw-Pwd
Login Phone Number
Twitter
Authenticator
(Login Email AND Tw-Pwd AND Authenticator) OR
(Login Phone Number AND Tw-Pwd AND Authenticator) OR
(Login Email AND Recovery OTC) OR
(Login Phone Number AND Recovery OTC) OR
Recovery Email
Recovery OTC
Login Email OR Recovery Email
Twitter OTC
Sent OTC
Spotify Password
Sp-Pwd
Login Email OR
Recovery Email
Spotify
(Login Email AND Sp-Pwd) OR
Recovery Email
LinkedIn Password
Li-Pwd
Login Email OR
Recovery Email
LinkedIn
OTC
(Login Email AND Li-Pwd AND OTC) OR
Recovery Email
LinkedIn OTC
Sent OTC
Microsoft Password
Mi-Pwd
Login Email OR Recovery Email
Microsoft
Authenticator OR Recovery Authenticator
Login Phone Number
(Login Email AND Mi-Pwd AND Authenticator)
OR
(Login Phone Number AND Mi-Pwd AND Authenticator)
OR
(Login Email AND Recovery Authenticator)
OR
(Login Phone Number AND Recovery Authenticator)
OR
Recovery Email
Deliveroo Password
De-Pwd
Deliveroo
Login Email OR
Recovery Email
(Login Email AND De-Pwd) OR
Recovery Email
Apple ID Password
Ap-Pwd
(Login Email AND Ap-Pwd AND OTC) OR
(Login Email AND Recovery OTC)
Apple ID
Login Email
OTC OR Recovery OTC
Apple Trusted Device Token
Trusted Device
ASOS Password
As-Pwd
ASOS
Login Email OR Recovery Email
(Login Email AND As-Pwd) OR
Recovery Email
Nullfield Health Password
Nu-Pwd
Nullfield Health
(Login Email AND Nu-Pwd) OR
Recovery Email
Login Email OR
Recovery Email
OVO Energy Password
Ov-Pwd
OVO Energy
Login Email OR
Recovery Email
(Login Email AND Ov-Pwd) OR
Recovery Email
Discord Password
Discord Secret Key
Di-Pwd
OTC OR Recovery OTC
Discord
(Login Email AND Di-Pwd AND OTC)
OR
(Login Email AND Recovery OTC)
Login Email
Soundcloud Password
So-Pwd
Login Email OR
Recovery Email
Soundcloud
Authenticator
(Login Email AND So-Pwd
AND Authenticator)
OR
Recovery Email
Grammarly Password
Gr-Pwd
Login Email OR Recovery Email
Grammarly
(Login Email AND Gr-Pwd AND Authenticator) OR Recovery Email
Authenticator
Snapchat Password
Sn-Pwd
Snapchat
OTC OR
Recovery OTC
Login Phone Number
(Login Phone Number AND Sn-Pwd AND OTC) OR
(Login Phone Number AND Recovery OTC)
Snapchat OTC
Sent OTC
WhatsApp
Whatsapp
Login Phone Number For Gmail-2
Gmail-2
(Login Email For Gmail-2 AND Gmail2-Pwd AND Authenticator For Gmail-2) OR
(Login Phone Number For Gmail-2 AND Gmail2-Pwd AND Authenticator For Gmail-2) OR
(Login Email For Gmail-2 AND Recovery OTC For Gmail-2) OR
(Login Phone Number For Gmail-2 AND Recovery OTC For Gmail-2)
Recovery OTC For Gmail-2
Gmail2-Pwd
Login Email
For Gmail-2
Authenticator
For Gmail-2
Google OTC For Gmail-2
Sent OTC
Gmail-2 Password
Gmail-2 Email Address
Phone Number
Gmail-1
Authenticator For Gmail-1
(Login Email For Gmail-1 AND Gmail1-Pwd AND Authenticator For Gmail-1)
OR
(Login Email For Gmail-1 AND Recovery OTC For Gmail-1)
Recovery OTC For Gmail-1
Gmail1-Pwd
Login Email For
Gmail-1
Google Authenticator
Trusted Device
Microsoft Authenticator
Trusted Device
Authy Authenticator
Trusted Device
Google OTC For Gmail-1
Sent OTC
Gmail-1 Password
Gmail-1 Email Address
Unlocked Participant's Devices

Figure A.3: Account Access Graph of Participant 3

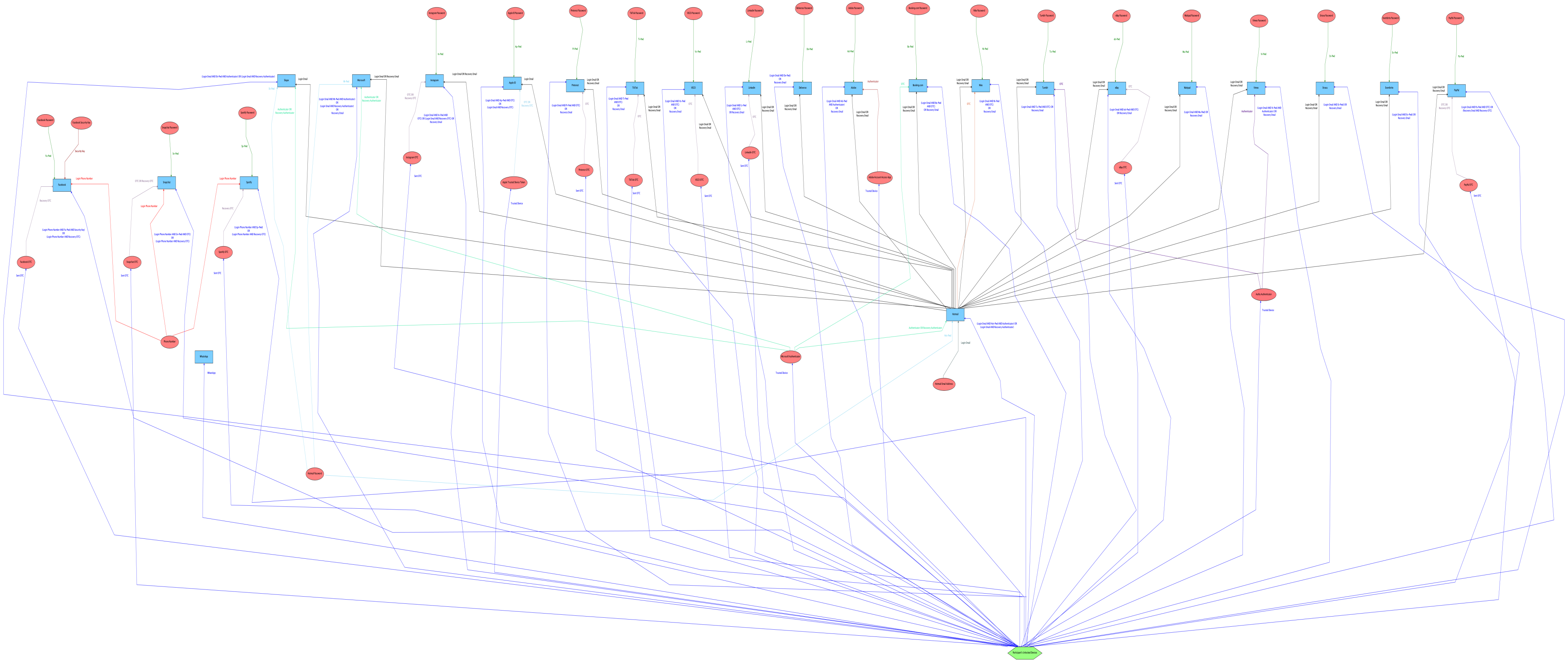

Instagram Password
Apple ID Password
Pinterest Password
TikTok Password
VSCO Password
LinkedIn Password
Deliveroo Password
Adobe Password
Booking.com Password
Nike Password
Tumblr Password
eBay Password
Wattpad Password
Vimeo Password
Strava Password
Eventbrite Password
PayPal Password
Facebook Password
Facebook Security Key
Snapchat Password
Spotify Password
In-Pwd
Ap-Pwd
Pi-Pwd
Ti-Pwd
Vs-Pwd
Li-Pwd
De-Pwd
Ad-Pwd
Bo-Pwd
Ni-Pwd
Tu-Pwd
eb-Pwd
Wa-Pwd
Vi-Pwd
St-Pwd
Ev-Pwd
Pa-Pwd
Fa-Pwd
Security Key
Sn-Pwd
Sp-Pwd
Sk-Pwd
M-Pwd
Hot-Pwd
Skype
Microsoft
Instagram
Apple ID
Pinterest
TikTok
VSCO
LinkedIn
Deliveroo
Adobe
Booking.com
Nike
Tumblr
eBay
Wattpad
Vimeo
Strava
Eventbrite
PayPal
Facebook
Snapchat
Spotify
WhatsApp
Hotmail
(Login Email AND Sk-Pwd AND Authenticator) OR (Login Email AND Recovery Authenticator)
Login Email
Login Email OR Recovery Email
Authenticator OR Recovery Authenticator
(Login Email AND M-Pwd AND Authenticator) OR (Login Email AND Recovery Authenticator) OR Recovery Email
OTC OR Recovery OTC
(Login Email AND In-Pwd AND OTC) OR (Login Email AND Recovery OTC) OR Recovery Email
(Login Email AND Ap-Pwd AND OTC) OR (Login Email AND Recovery OTC)
(Login Email AND Pi-Pwd AND OTC) OR Recovery Email
OTC
(Login Email AND Ti-Pwd AND OTC) OR Recovery Email
(Login Email AND Vs-Pwd AND OTC) OR Recovery Email
(Login Email AND Li-Pwd AND OTC) OR Recovery Email
(Login Email AND De-Pwd) OR Recovery Email
(Login Email AND Ad-Pwd AND Authenticator) OR Recovery Email
Authenticator
(Login Email AND Bo-Pwd AND OTC) OR Recovery Email
(Login Email AND Ni-Pwd AND OTC) OR Recovery Email
(Login Email AND Tu-Pwd AND OTC) OR Recovery Email
(Login Email AND eb-Pwd AND OTC) OR Recovery Email
(Login Email AND Wa-Pwd) OR Recovery Email
(Login Email AND Vi-Pwd AND Authenticator) OR Recovery Email
(Login Email AND St-Pwd) OR Recovery Email
(Login Email AND Ev-Pwd) OR Recovery Email
(Login Email AND Pa-Pwd AND OTC) OR (Recovery Email AND Recovery OTC)
Login Phone Number
Recovery OTC
(Login Phone Number AND Fa-Pwd AND Security Key) OR (Login Phone Number AND Recovery OTC)
(Login Phone Number AND Sn-Pwd AND OTC) OR (Login Phone Number AND Recovery OTC)
(Login Phone Number AND Sp-Pwd) OR (Login Phone Number AND Recovery OTC)
Instagram OTC
Apple Trusted Device Token
Pinterest OTC
TikTok OTC
VSCO OTC
LinkedIn OTC
Adobe Account Access App
eBay OTC
PayPal OTC
Facebook OTC
Snapchat OTC
Spotify OTC
Phone Number
Sent OTC
Trusted Device
WhatsApp
Hotmail Password
Microsoft Authenticator
Authy Authenticator
Hotmail Email Address
(Login Email AND Hot-Pwd AND Authenticator) OR (Login Email AND Recovery Authenticator)
Authenticator OR Recovery Authenticator
Participant's Unlocked Devices

Figure A.4: Account Access Graph of Participant 4

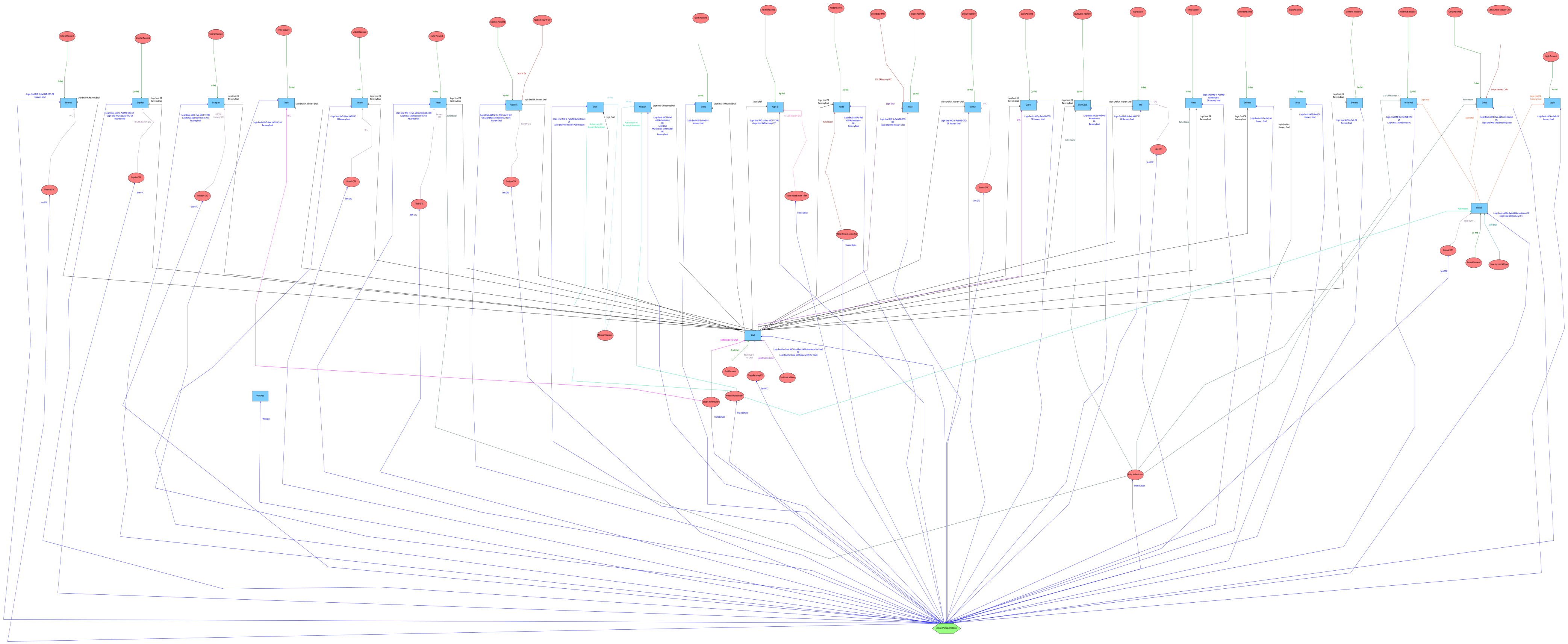

Pinterest Password
Pi-Pwd
(Login Email AND Pi-Pwd AND OTC) OR Recovery Email
Pinterest
Login Email OR Recovery Email
OTC
Pinterest OTC
Sent OTC
Snapchat Password
Sn-Pwd
Snapchat
Login Email OR Recovery Email
(Login Email AND Sn-Pwd AND OTC) OR (Login Email AND Recovery OTC) OR Recovery Email
OTC OR Recovery OTC
Snapchat OTC
Sent OTC
Instagram Password
In-Pwd
Instagram
Login Email OR Recovery Email
(Login Email AND In-Pwd AND OTC) OR (Login Email AND Recovery OTC) OR Recovery Email
OTC OR Recovery OTC
Instagram OTC
Sent OTC
Trello Password
Tr-Pwd
Trello
Login Email OR Recovery Email
OTC
(Login Email AND Tr-Pwd AND OTC) OR Recovery Email
LinkedIn Password
Li-Pwd
LinkedIn
Login Email OR Recovery Email
(Login Email AND Li-Pwd AND OTC) OR Recovery Email
OTC
LinkedIn OTC
Sent OTC
Twitter Password
Tw-Pwd
Twitter
Login Email OR Recovery Email
(Login Email AND Tw-Pwd AND Authenticator) OR (Login Email AND Recovery OTC) OR Recovery Email
Recovery OTC
Authenticator
Twitter OTC
Sent OTC
Facebook Password
Facebook Security Key
Fa-Pwd
Security Key
Facebook
Login Email OR Recovery Email
(Login Email AND Fa-Pwd AND Security Key) OR (Login Email AND Recovery OTC) OR Recovery Email
Recovery OTC
Facebook OTC
Sent OTC
Sk-Pwd
Skype
Login Email
(Login Email AND Sk-Pwd AND Authenticator) OR (Login Email AND Recovery Authenticator)
Authenticator OR Recovery Authenticator
Mi-Pwd
Microsoft
Login Email OR Recovery Email
Authenticator OR Recovery Authenticator
(Login Email AND Mi-Pwd AND Authenticator) OR (Login Email AND Recovery Authenticator) OR Recovery Email
Microsoft Password
Spotify Password
Sp-Pwd
Spotify
Login Email OR Recovery Email
(Login Email AND Sp-Pwd) OR Recovery Email
Apple ID Password
Ap-Pwd
Apple ID
Login Email
(Login Email AND Ap-Pwd AND OTC) OR (Login Email AND Recovery OTC)
OTC OR Recovery OTC
Apple Trusted Device Token
Trusted Device
Adobe Password
Ad-Pwd
Adobe
Login Email OR Recovery Email
Authenticator
(Login Email AND Ad-Pwd AND Authenticator) OR Recovery Email
Adobe Account Access App
Trusted Device
Discord Secret Key
Discord Password
OTC OR Recovery OTC
Di-Pwd
Discord
Login Email
(Login Email AND Di-Pwd AND OTC) OR (Login Email AND Recovery OTC)
Disney+ Password
Di-Pwd
Disney+
Login Email OR Recovery Email
OTC
(Login Email AND Di-Pwd AND OTC) OR Recovery Email
Disney+ OTC
Sent OTC
Quora Password
Qu-Pwd
Quora
Login Email OR Recovery Email
OTC
(Login Email AND Qu-Pwd AND OTC) OR Recovery Email
SoundCloud Password
So-Pwd
SoundCloud
Login Email OR Recovery Email
(Login Email AND So-Pwd AND Authenticator) OR Recovery Email
Authenticator
eBay Password
eb-Pwd
eBay
Login Email OR Recovery Email
OTC
(Login Email AND eb-Pwd AND OTC) OR Recovery Email
eBay OTC
Sent OTC
Vimeo Password
Vi-Pwd
Vimeo
(Login Email AND Vi-Pwd AND Authenticator) OR Recovery Email
Authenticator
Login Email OR Recovery Email
Deliveroo Password
De-Pwd
Deliveroo
Login Email OR Recovery Email
(Login Email AND De-Pwd) OR Recovery Email
Strava Password
St-Pwd
Strava
Login Email OR Recovery Email
(Login Email AND St-Pwd) OR Recovery Email
Eventbrite Password
Ev-Pwd
Eventbrite
Login Email OR Recovery Email
(Login Email AND Ev-Pwd) OR Recovery Email
Docker Hub Password
Do-Pwd
Docker Hub
OTC OR Recovery OTC
Login Email
(Login Email AND Do-Pwd AND OTC) OR (Login Email AND Recovery OTC)
GitHub Password
GitHub Unique Recovery Code
Gi-Pwd
Unique Recovery Code
GitHub
Authenticator
Login Email
(Login Email AND Gi-Pwd AND Authenticator) OR (Login Email AND Unique Recovery Code)
Kaggle Password
Ka-Pwd
Kaggle
Login Email OR Recovery Email
(Login Email AND Ka-Pwd) OR Recovery Email
Outlook
Authenticator
Recovery OTC
Ou-Pwd
Login Email
(Login Email AND Ou-Pwd AND Authenticator) OR (Login Email AND Recovery OTC)
Outlook OTC
Sent OTC
Outlook Password
University Email Address
Gmail
Authenticator For Gmail
Gmail-Pwd
Recovery OTC For Gmail
Login Email For Gmail
(Login Email For Gmail AND Gmail-Pwd AND Authenticator For Gmail) OR (Login Email For Gmail AND Recovery OTC For Gmail)
Gmail Password
Google Recovery OTC
Gmail Email Address
Sent OTC
Google Authenticator
Microsoft Authenticator
Trusted Device
Trusted Device
Authy Authenticator
Trusted Device
Unlocked Participant's Device

Figure A.5: Account Access Graph of Participant 5

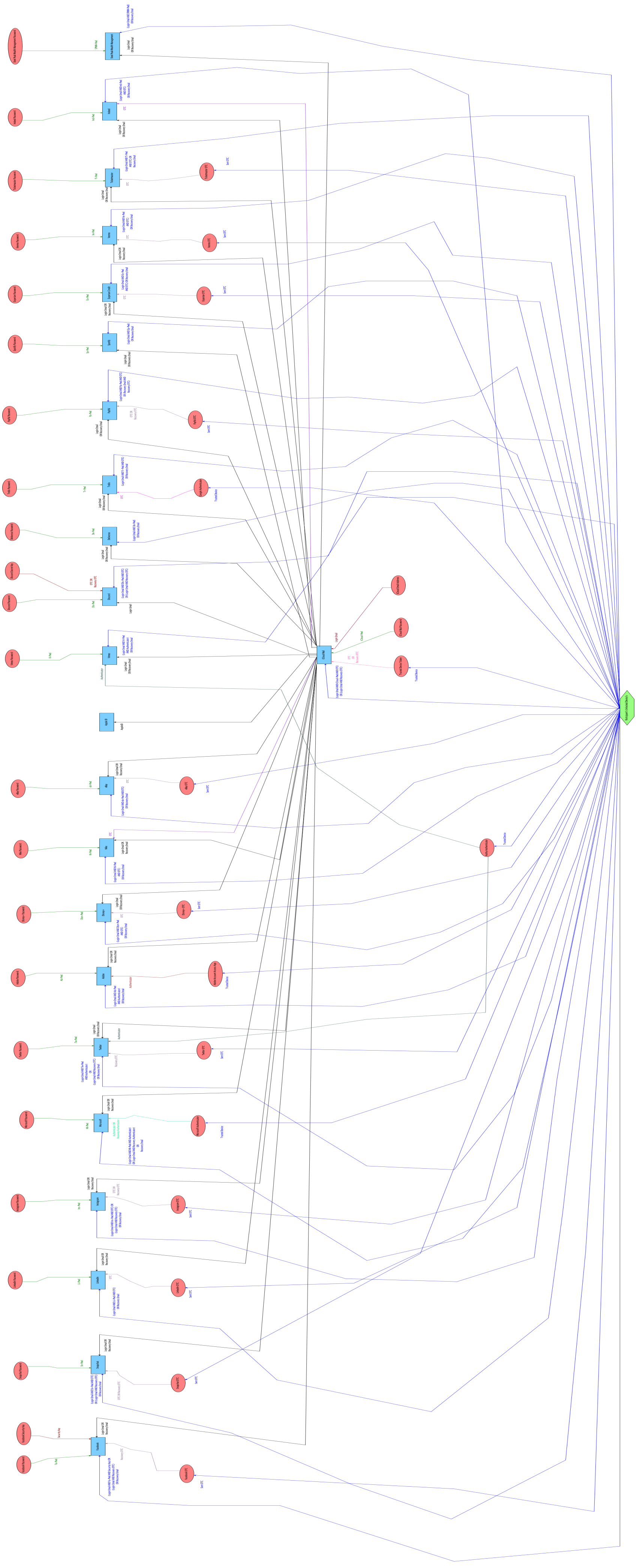


Figure A.6: Account Access Graph of Participant 6

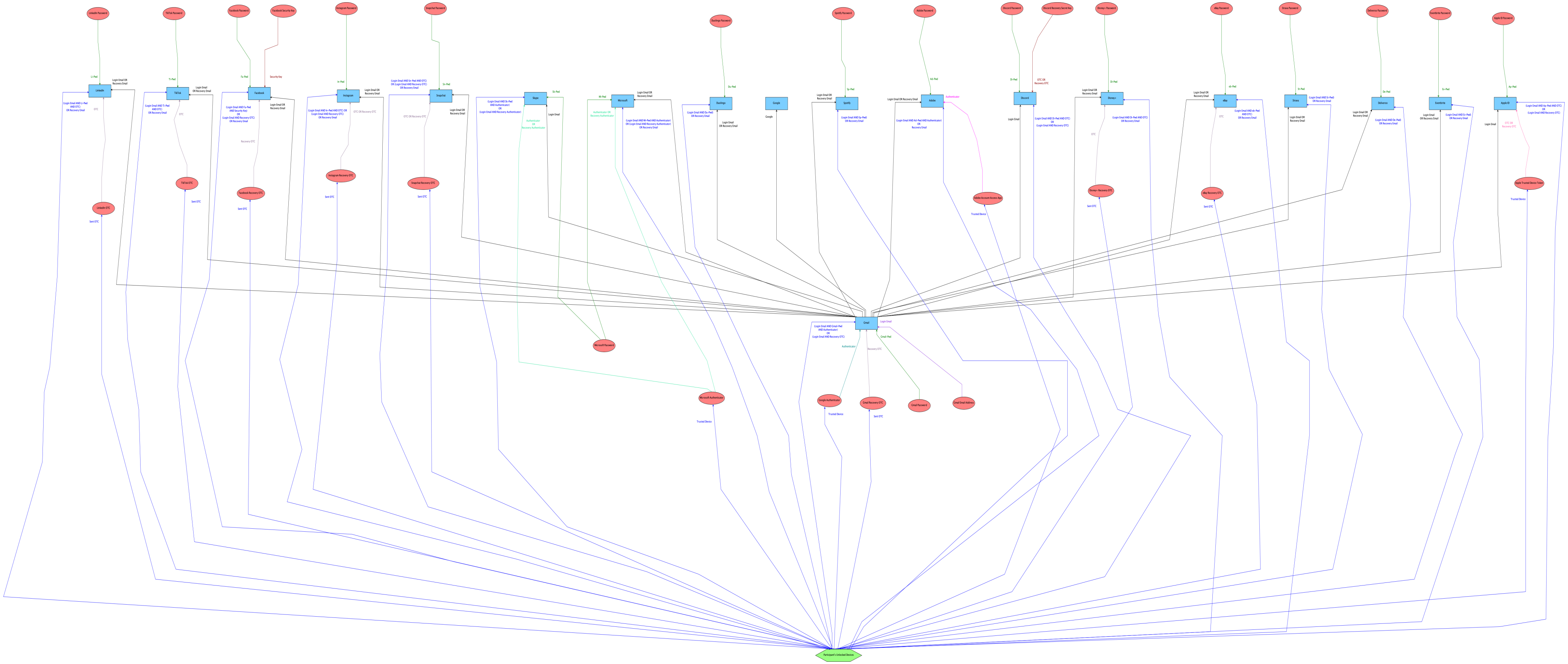

LinkedIn Password
TikTok Password
Facebook Password
Facebook Security Key
Instagram Password
Snapchat Password
Duolingo Password
Spotify Password
Adobe Password
Discord Password
Discord Recovery Secret Key
Disney+ Password
eBay Password
Strava Password
Deliveroo Password
Eventbrite Password
Apple ID Password
Li-Pwd
Ti-Pwd
Fa-Pwd
Security Key
In-Pwd
Sn-Pwd
Sk-Pwd
Mi-Pwd
Du-Pwd
Sp-Pwd
Ad-Pwd
Di-Pwd
OTC OR Recovery OTC
Di-Pwd
eb-Pwd
St-Pwd
De-Pwd
Ev-Pwd
Ap-Pwd
LinkedIn
TikTok
Facebook
Instagram
Snapchat
Skype
Microsoft
Duolingo
Google
Spotify
Adobe
Discord
Disney+
eBay
Strava
Deliveroo
Eventbrite
Apple ID
Login Email OR Recovery Email
(Login Email AND Li-Pwd AND OTC) OR Recovery Email
OTC
Login Email OR Recovery Email
(Login Email AND Ti-Pwd AND OTC) OR Recovery Email
OTC
(Login Email AND Fa-Pwd AND Security Key) OR (Login Email AND Recovery OTC) OR Recovery Email
Login Email OR Recovery Email
Recovery OTC
Login Email OR Recovery Email
(Login Email AND In-Pwd AND OTC) OR (Login Email AND Recovery OTC) OR Recovery Email
OTC OR Recovery OTC
(Login Email AND Sn-Pwd AND OTC) OR (Login Email AND Recovery OTC) OR Recovery Email
OTC OR Recovery OTC
Login Email OR Recovery Email
(Login Email AND Sk-Pwd AND Authenticator) OR (Login Email AND Recovery Authenticator)
Authenticator OR Recovery Authenticator
Login Email
Authenticator OR Recovery Authenticator
Login Email OR Recovery Email
(Login Email AND Mi-Pwd AND Authenticator) OR (Login Email AND Recovery Authenticator) OR Recovery Email
(Login Email AND Du-Pwd) OR Recovery Email
Login Email OR Recovery Email
Google
Login Email OR Recovery Email
(Login Email AND Sp-Pwd) OR Recovery Email
Login Email OR Recovery Email
(Login Email AND Ad-Pwd AND Authenticator) OR Recovery Email
Authenticator
Login Email
(Login Email AND Di-Pwd AND OTC) OR (Login Email AND Recovery OTC)
Login Email OR Recovery Email
(Login Email AND Di-Pwd AND OTC) OR Recovery Email
OTC
Login Email OR Recovery Email
(Login Email AND eb-Pwd AND OTC) OR Recovery Email
OTC
(Login Email AND St-Pwd) OR Recovery Email
Login Email OR Recovery Email
Login Email OR Recovery Email
(Login Email AND De-Pwd) OR Recovery Email
Login Email OR Recovery Email
(Login Email AND Ev-Pwd) OR Recovery Email
Login Email
OTC OR Recovery OTC
(Login Email AND Ap-Pwd AND OTC) OR (Login Email AND Recovery OTC)
LinkedIn OTC
TikTok OTC
Facebook Recovery OTC
Instagram Recovery OTC
Snapchat Recovery OTC
Adobe Account Access App
Disney+ Recovery OTC
eBay Recovery OTC
Apple Trusted Device Token
Sent OTC
Sent OTC
Sent OTC
Sent OTC
Sent OTC
Trusted Device
Sent OTC
Sent OTC
Trusted Device
Microsoft Password
Gmail
Login Email
(Login Email AND Gmail-Pwd AND Authenticator) OR (Login Email AND Recovery OTC)
Authenticator
Recovery OTC
Gmail-Pwd
Microsoft Authenticator
Google Authenticator
Gmail Recovery OTC
Gmail Password
Gmail Email Address
Trusted Device
Trusted Device
Sent OTC
Participant's Unlocked Devices

Figure A.7: Account Access Graph of Participant 7

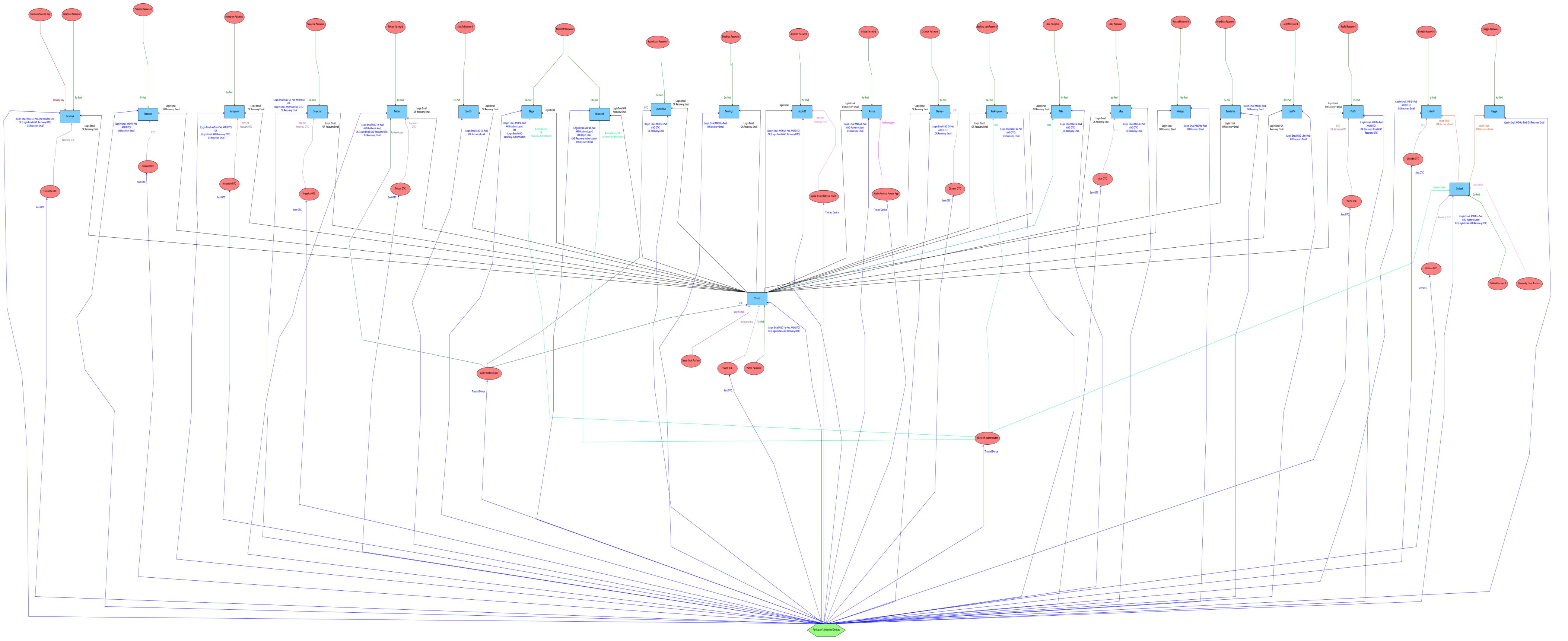


Figure A.8: Account Access Graph of Participant 8

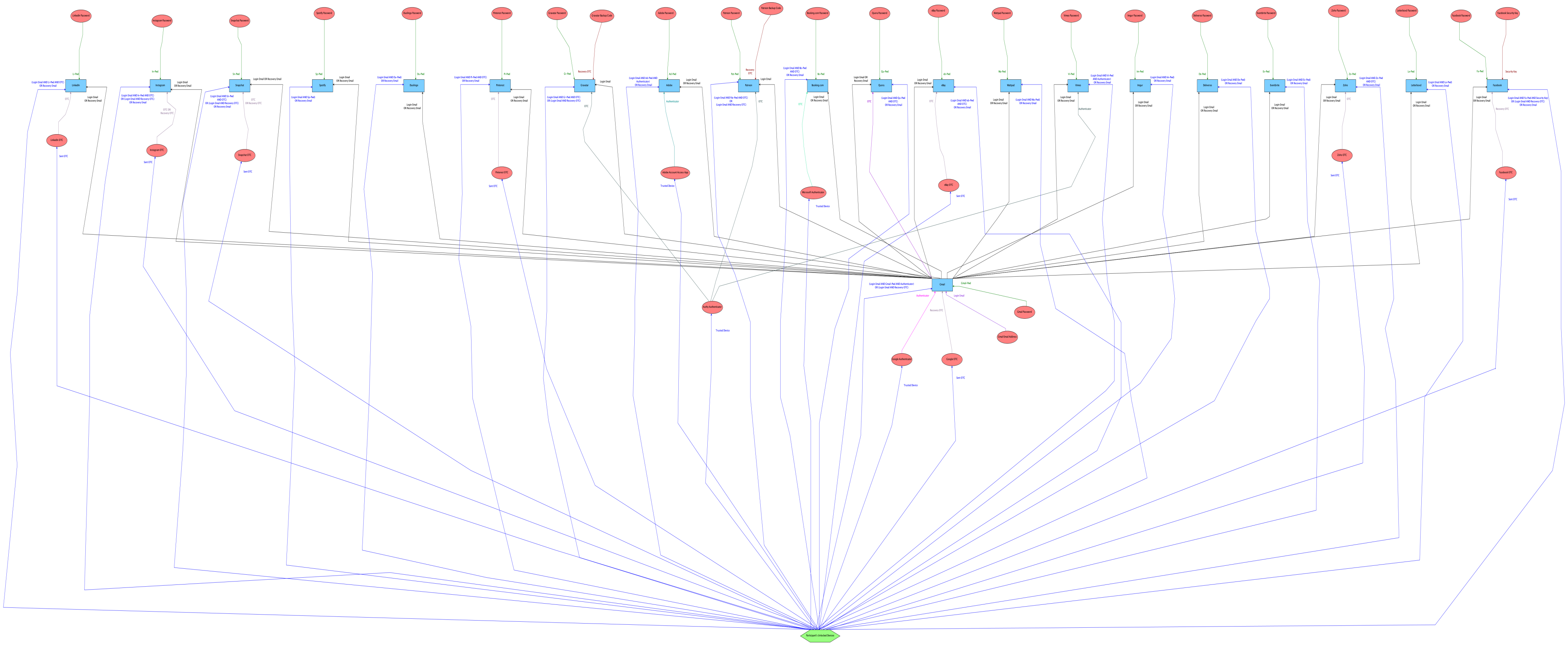


Figure A.9: Account Access Graph of Participant 9

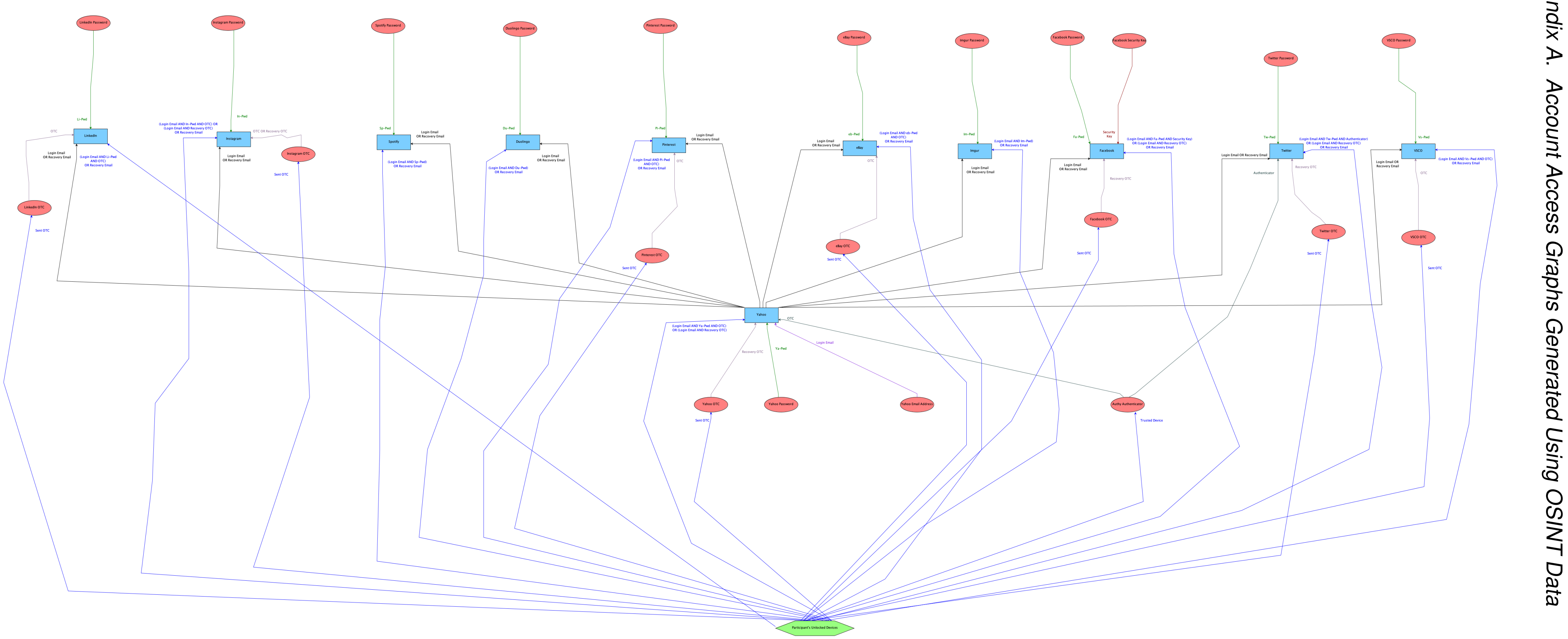

LinkedIn Password
Instagram Password
Spotify Password
Duolingo Password
Pinterest Password
eBay Password
Imgur Password
Facebook Password
Facebook Security Key
Twitter Password
VSCO Password
Li-Pwd
In-Pwd
Sp-Pwd
Du-Pwd
Pi-Pwd
eb-Pwd
Im-Pwd
Fa-Pwd
Security Key
Tw-Pwd
Vs-Pwd
OTC
LinkedIn
Login Email OR Recovery Email
(Login Email AND Li-Pwd AND OTC) OR Recovery Email
LinkedIn OTC
Sent OTC
(Login Email AND In-Pwd AND OTC) OR (Login Email AND Recovery OTC) OR Recovery Email
Instagram
OTC OR Recovery OTC
Login Email OR Recovery Email
Instagram OTC
Sent OTC
Spotify
Login Email OR Recovery Email
(Login Email AND Sp-Pwd) OR Recovery Email
Duolingo
Login Email OR Recovery Email
(Login Email AND Du-Pwd) OR Recovery Email
Pinterest
Login Email OR Recovery Email
(Login Email AND Pi-Pwd AND OTC) OR Recovery Email
OTC
Pinterest OTC
Sent OTC
Login Email OR Recovery Email
eBay
(Login Email AND eb-Pwd AND OTC) OR Recovery Email
OTC
eBay OTC
Sent OTC
Imgur
(Login Email AND Im-Pwd) OR Recovery Email
Login Email OR Recovery Email
Facebook
(Login Email AND Fa-Pwd AND Security Key) OR (Login Email AND Recovery OTC) OR Recovery Email
Login Email OR Recovery Email
Recovery OTC
Facebook OTC
Sent OTC
Login Email OR Recovery Email
Twitter
(Login Email AND Tw-Pwd AND Authenticator) OR (Login Email AND Recovery OTC) OR Recovery Email
Authenticator
Recovery OTC
Twitter OTC
Sent OTC
VSCO
Login Email OR Recovery Email
(Login Email AND Vs-Pwd AND OTC) OR Recovery Email
OTC
VSCO OTC
Sent OTC
Yahoo
OTC
(Login Email AND Ya-Pwd AND OTC) OR (Login Email AND Recovery OTC)
Recovery OTC
Ya-Pwd
Login Email
Yahoo OTC
Sent OTC
Yahoo Password
Yahoo Email Address
Authy Authenticator
Trusted Device
Participant's Unlocked Devices

Figure A.10: Account Access Graph of Participant 10

Appendix B

Account Access Graphs Generated Using Participant Data

Due to the number of vertices and edges within the graphs, they are large in size and may require zooming in to increase visibility.

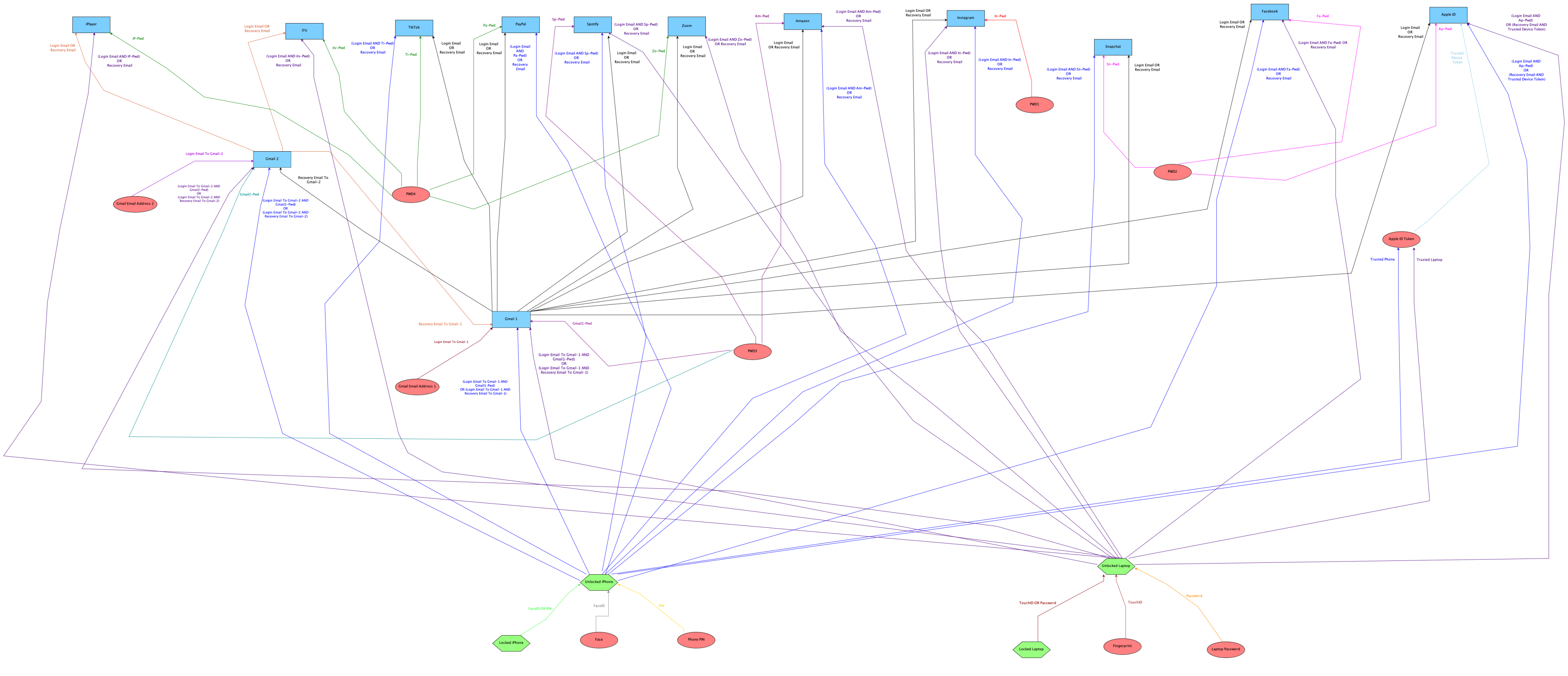


Figure B.1: Account Access Graph of Participant 1

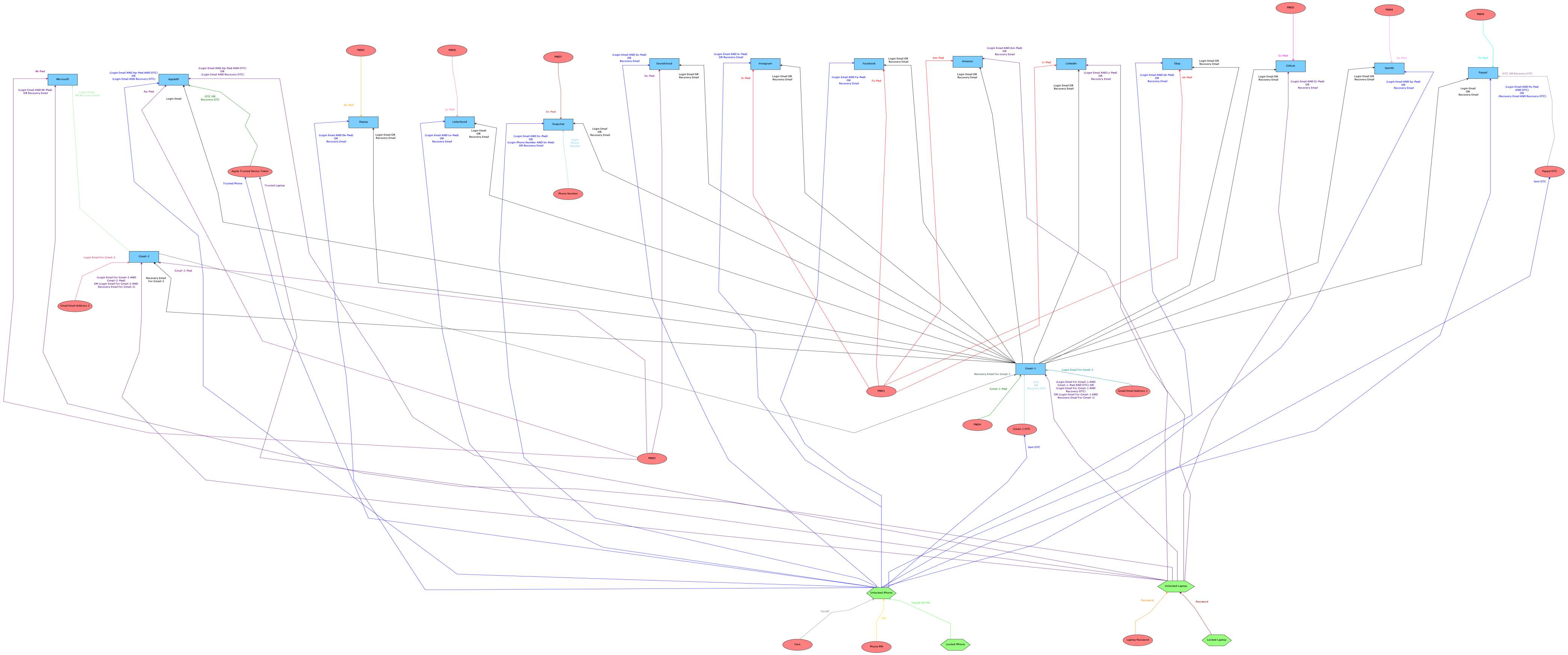


Figure B.2: Account Access Graph of Participant 2

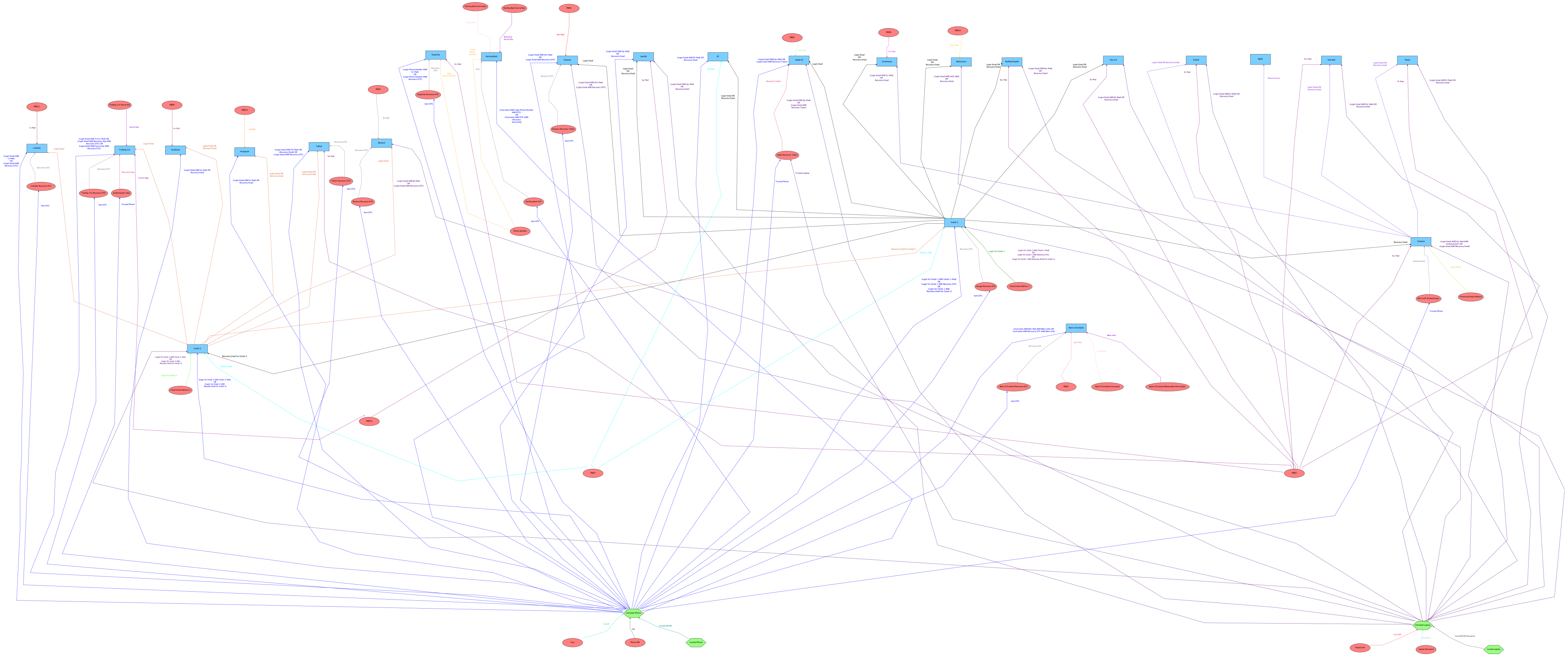


Figure B.3: Account Access Graph of Participant 3

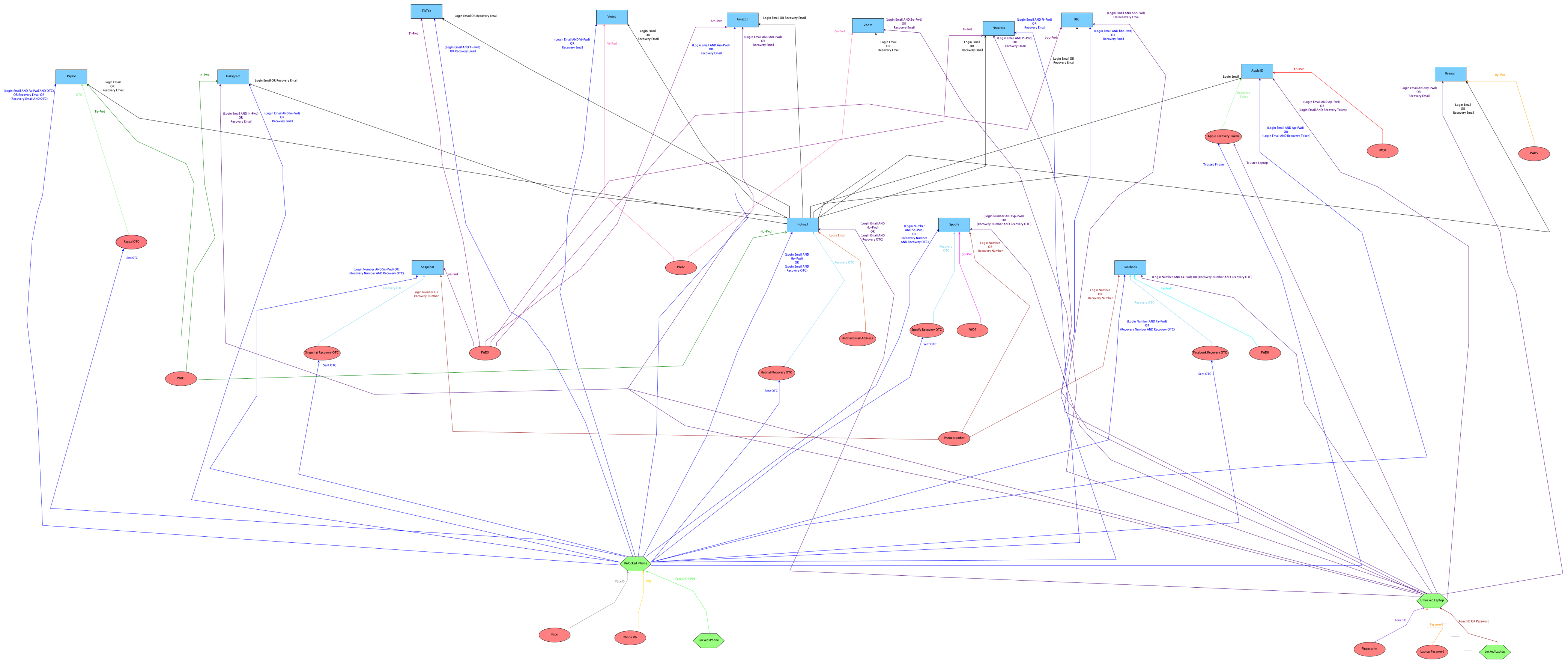


Figure B.4: Account Access Graph of Participant 4

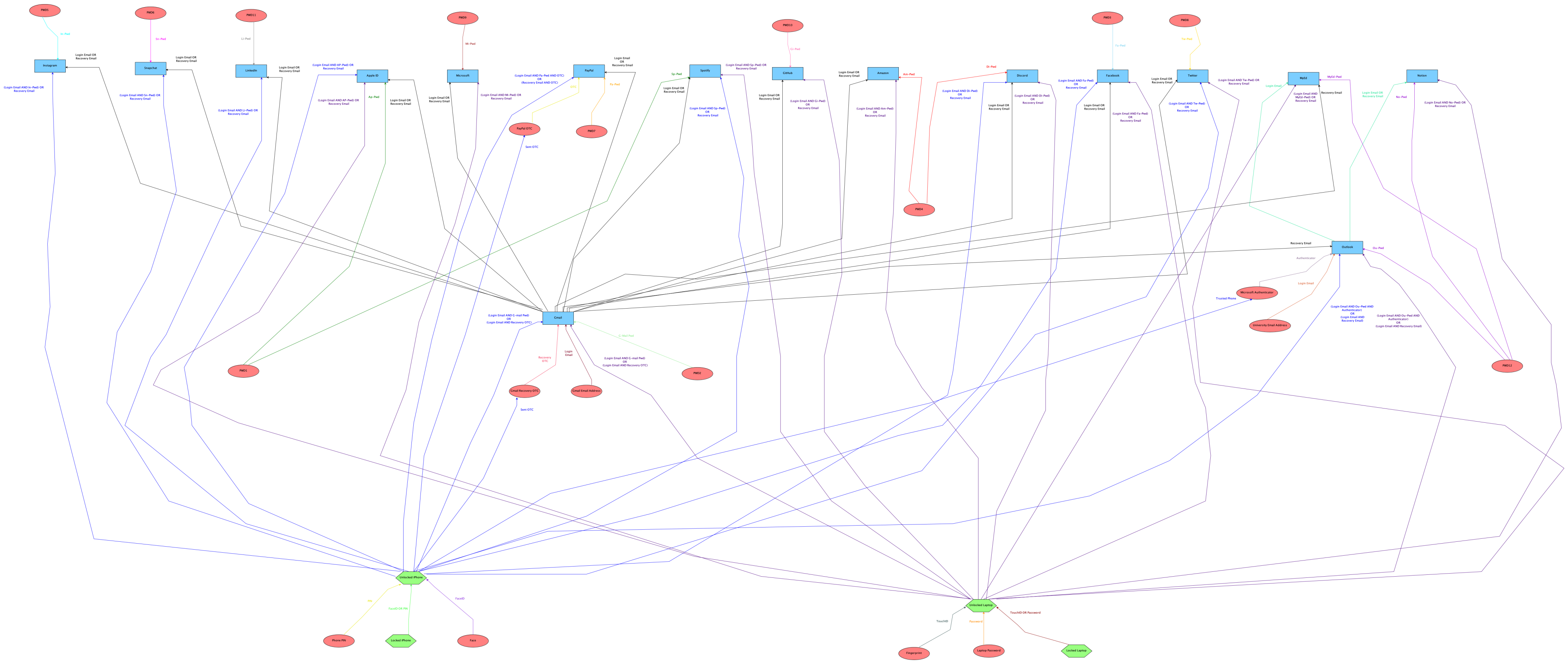

PWD5
In-Pwd
Instagram
Login Email OR Recovery Email
(Login Email AND In-Pwd) OR Recovery Email
PWD6
Sn-Pwd
Snapchat
Login Email OR Recovery Email
(Login Email AND Sn-Pwd) OR Recovery Email
PWD11
Li-Pwd
LinkedIn
Login Email OR Recovery Email
(Login Email AND Li-Pwd) OR Recovery Email
(Login Email AND AP-Pwd) OR Recovery Email
Apple ID
(Login Email AND AP-Pwd) OR Recovery Email
Ap-Pwd
Login Email OR Recovery Email
PWD9
Mi-Pwd
Microsoft
Login Email OR Recovery Email
(Login Email AND Mi-Pwd) OR Recovery Email
(Login Email AND Pa-Pwd AND OTC) OR (Recovery Email AND OTC)
PayPal
Login Email OR Recovery Email
OTC
Pa-Pwd
PayPal OTC
Sent OTC
PWD7
Sp-Pwd
Spotify
(Login Email AND Sp-Pwd) OR Recovery Email
Login Email OR Recovery Email
(Login Email AND Sp-Pwd) OR Recovery Email
PWD10
Gi-Pwd
GitHub
Login Email OR Recovery Email
(Login Email AND Gi-Pwd) OR Recovery Email
Login Email OR Recovery Email
Amazon
Am-Pwd
(Login Email AND Am-Pwd) OR Recovery Email
Di-Pwd
Discord
(Login Email AND Di-Pwd) OR Recovery Email
Login Email OR Recovery Email
(Login Email AND Di-Pwd) OR Recovery Email
PWD4
PWD3
Fa-Pwd
Facebook
(Login Email AND Fa-Pwd) OR Recovery Email
Login Email OR Recovery Email
(Login Email AND Fa-Pwd) OR Recovery Email
PWD8
Tw-Pwd
Twitter
Login Email OR Recovery Email
(Login Email AND Tw-Pwd) OR Recovery Email
(Login Email AND Tw-Pwd) OR Recovery Email
Login Email
MyEd
MyEd-Pwd
(Login Email AND MyEd-Pwd) OR Recovery Email
Recovery Email
Notion
Login Email OR Recovery Email
No-Pwd
(Login Email AND No-Pwd) OR Recovery Email
Gmail
(Login Email AND G-mail Pwd) OR (Login Email AND Recovery OTC)
G-Mail Pwd
Recovery OTC
Login Email
(Login Email AND G-mail Pwd) OR (Login Email AND Recovery OTC)
PWD1
PWD2
Gmail Recovery OTC
Gmail Email Address
Sent OTC
Outlook
Recovery Email
Ou-Pwd
Authenticator
Login Email
Microsoft Authenticator
Trusted Phone
University Email Address
(Login Email AND Ou-Pwd AND Authenticator) OR (Login Email AND Recovery Email)
(Login Email AND Ou-Pwd AND Authenticator) OR (Login Email AND Recovery Email)
PWD12
Unlocked iPhone
PIN
FaceID OR PIN
FaceID
Phone PIN
Locked iPhone
Face
Unlocked Laptop
TouchID
Password
TouchID OR Password
Fingerprint
Laptop Password
Locked Laptop

Figure B.5: Account Access Graph of Participant 5

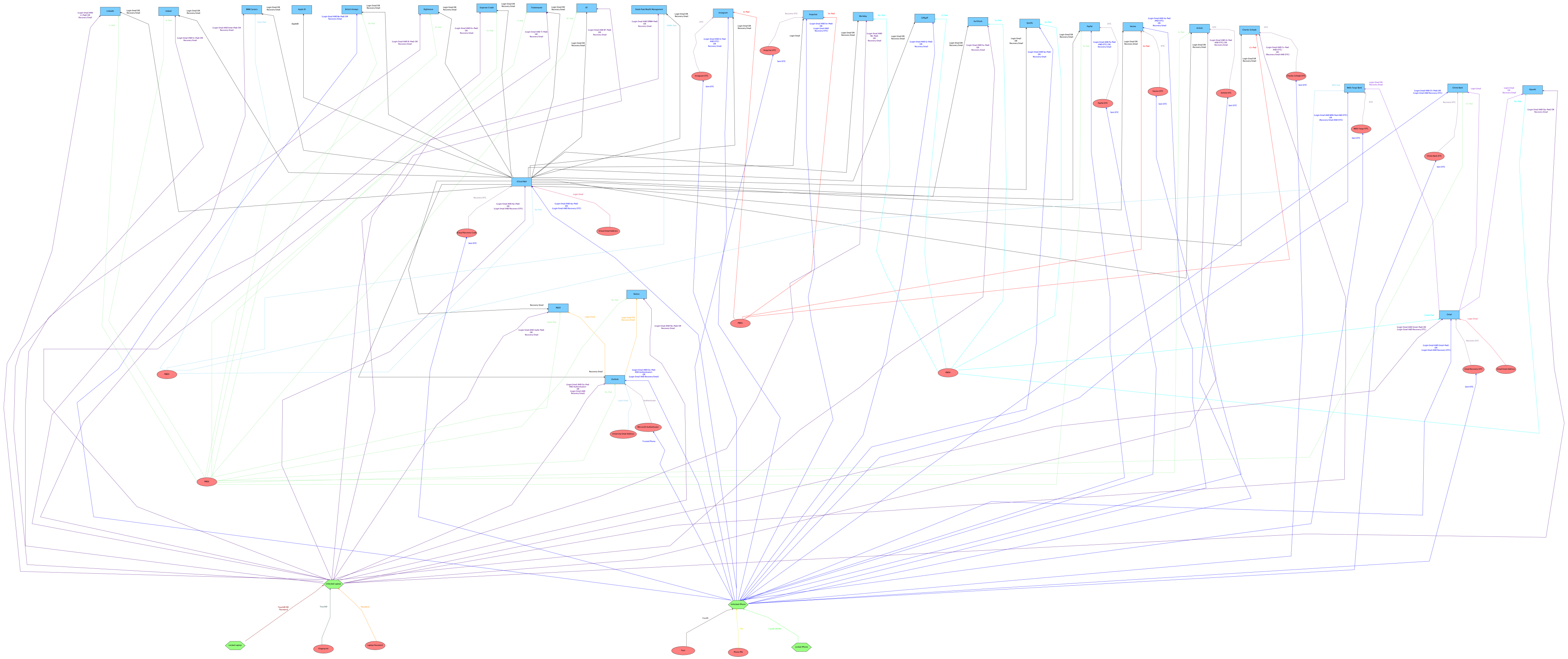

LinkedIn
(Login Email AND Li-Pwd) OR Recovery Email
Li-Pwd
Login Email OR Recovery Email
Indeed
In-Pwd
(Login Email AND In-Pwd) OR Recovery Email
BMW Careers
(Login Email AND bmw-Pwd) OR Recovery Email
bmw-Pwd
Apple ID
AppleID
British Airways
(Login Email AND Ba-Pwd) OR Recovery Email
Ba-Pwd
Rightmove
(Login Email AND Ri-Pwd) OR Recovery Email
Ri-Pwd
Experian Credit
(Login Email AND Ex-Pwd) OR Recovery Email
Ex-Pwd
Ticketmaster
(Login Email AND Ti-Pwd) OR Recovery Email
Ti-Pwd
BT
(Login Email AND BT-Pwd) OR Recovery Email
BT-Pwd
Steele Peak Wealth Management
(Login Email AND SPWM-Pwd) OR Recovery Email
SPWM-Pwd
Instagram
In-Pwd
OTC
(Login Email AND In-Pwd AND OTC) OR Recovery Email
Instagram OTC
Sent OTC
Snapchat
Sn-Pwd
Recovery OTC
Login Email
(Login Email AND Sn-Pwd) OR (Login Email AND Recovery OTC)
Snapchat OTC
Workday
Wo-Pwd
(Login Email AND Wo-Pwd) OR Recovery Email
Giffgaff
Gi-Pwd
(Login Email AND Gi-Pwd) OR Recovery Email
Surfshark
Su-Pwd
(Login Email AND Su-Pwd) OR Recovery Email
Spotify
Sp-Pwd
Login Email OR Recovery Email
(Login Email AND Sp-Pwd) OR Recovery Email
PayPal
Pa-Pwd
(Login Email AND Pa-Pwd AND OTC) OR Recovery Email
PayPal OTC
Venmo
Ve-Pwd
(Login Email AND Ve-Pwd AND OTC) OR Recovery Email
Venmo OTC
Airbnb
Ai-Pwd
(Login Email AND Ai-Pwd AND OTC) OR Recovery Email
Airbnb OTC
Charles Schwab
Cs-Pwd
(Login Email AND Cs-Pwd AND OTC) OR (Recovery Email AND OTC)
Charles Schwab OTC
Wells Fargo Bank
WFB-Pwd
(Login Email AND WFB-Pwd AND OTC) OR (Recovery Email AND OTC)
Wells Fargo OTC
Chime Bank
Ch-Pwd
(Login Email AND Ch-Pwd) OR (Login Email AND Recovery OTC)
Chime Bank OTC
OpenAI
Op-Pwd
(Login Email AND Op-Pwd) OR Recovery Email
iCloud Mail
Recovery OTC
(Login Email AND Ap-Pwd) OR (Login Email AND Recovery OTC)
Ap-Pwd
iCloud Recovery Code
iCloud Email Address
MyEd
Recovery Email
Login Email
myEd-Pwd
(Login Email AND myEd-Pwd) OR Recovery Email
Notion
No-Pwd
Login Email OR Recovery Email
(Login Email AND No-Pwd) OR Recovery Email
Outlook
Ou-Pwd
(Login Email AND Ou-Pwd AND Authenticator) OR (Login Email AND Recovery Email)
Authenticator
University Email Address
Microsoft Authenticator
Trusted Phone
PWD1
PWD2
PWD3
PWD4
Gmail
Gmail-Pwd
(Login Email AND Gmail-Pwd) OR (Login Email AND Recovery OTC)
Gmail Recovery OTC
Gmail Email Address
Unlocked Laptop
TouchID OR Password
TouchID
Password
Locked Laptop
Fingerprint
Laptop Password
Unlocked iPhone
FaceID
PIN
FaceID OR PIN
Face
Phone PIN
Locked iPhone

Figure B.6: Account Access Graph of Participant 6

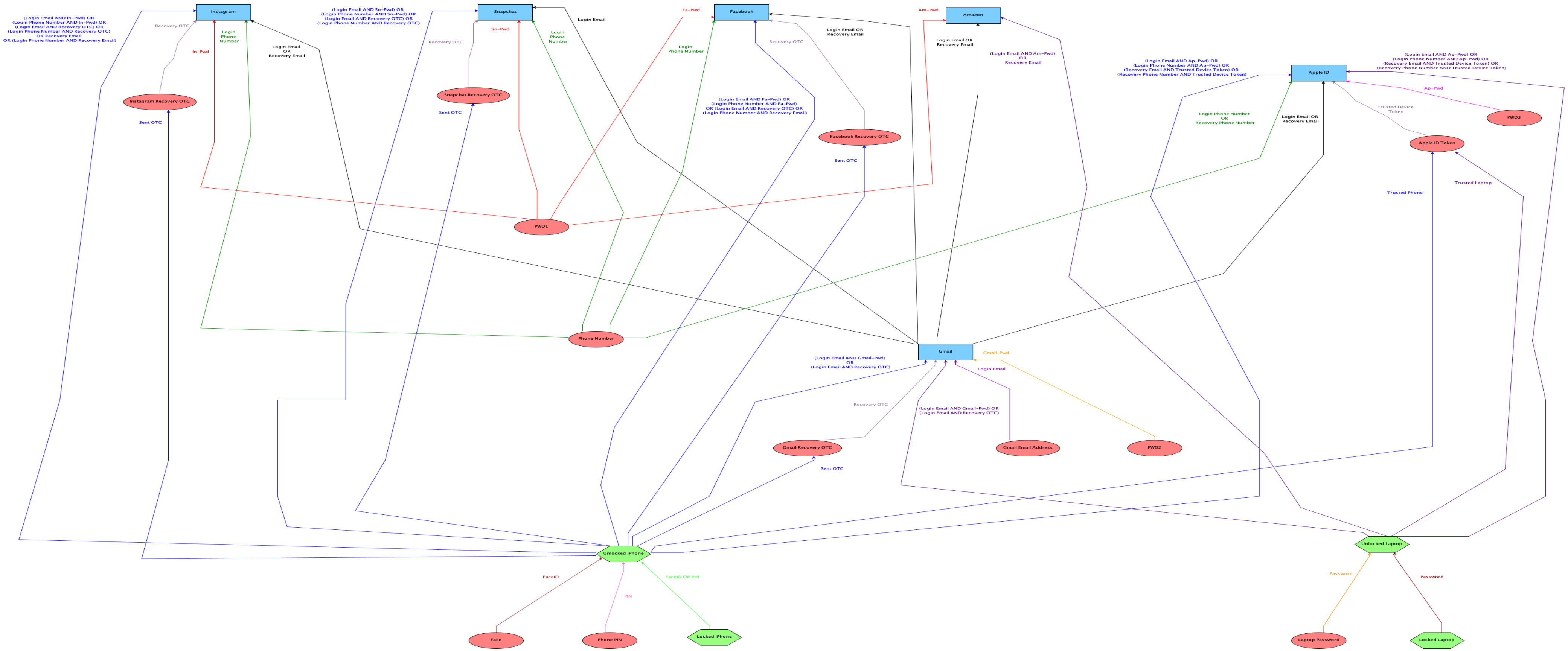

(Login Email AND In–Pwd) OR
(Login Phone Number AND In–Pwd) OR
(Login Email AND Recovery OTC) OR
(Login Phone Number AND Recovery OTC)
OR Recovery Email
OR (Login Phone Number AND Recovery Email)
Instagram
Recovery OTC
In–Pwd
Login Phone Number
Login Email OR Recovery Email
Instagram Recovery OTC
Sent OTC
(Login Email AND Sn–Pwd) OR
(Login Phone Number AND Sn–Pwd) OR
(Login Email AND Recovery OTC) OR
(Login Phone Number AND Recovery OTC)
Snapchat
Recovery OTC
Sn–Pwd
Login Phone Number
Login Email
Snapchat Recovery OTC
Sent OTC
Fa–Pwd
Facebook
Login Phone Number
Recovery OTC
Login Email OR Recovery Email
(Login Email AND Fa–Pwd) OR
(Login Phone Number AND Fa–Pwd)
OR (Login Email AND Recovery OTC) OR
(Login Phone Number AND Recovery Email)
Facebook Recovery OTC
Sent OTC
Am–Pwd
Amazon
Login Email OR Recovery Email
(Login Email AND Am–Pwd) OR Recovery Email
(Login Email AND Ap–Pwd) OR
(Login Phone Number AND Ap–Pwd) OR
(Recovery Email AND Trusted Device Token) OR
(Recovery Phone Number AND Trusted Device Token)
Apple ID
(Login Email AND Ap–Pwd) OR
(Login Phone Number AND Ap–Pwd) OR
(Recovery Email AND Trusted Device Token) OR
(Recovery Phone Number AND Trusted Device Token)
Ap–Pwd
PWD3
Trusted Device Token
Login Phone Number OR Recovery Phone Number
Login Email OR Recovery Email
Apple ID Token
Trusted Phone
Trusted Laptop
PWD1
Phone Number
Gmail
Gmail–Pwd
Login Email
(Login Email AND Gmail–Pwd) OR (Login Email AND Recovery OTC)
Recovery OTC
(Login Email AND Gmail–Pwd) OR
(Login Email AND Recovery OTC)
Gmail Recovery OTC
Sent OTC
Gmail Email Address
PWD2
Unlocked iPhone
FaceID
PIN
FaceID OR PIN
Face
Phone PIN
Locked iPhone
Unlocked Laptop
Password
Password
Laptop Password
Locked Laptop

Figure B.7: Account Access Graph of Participant 7

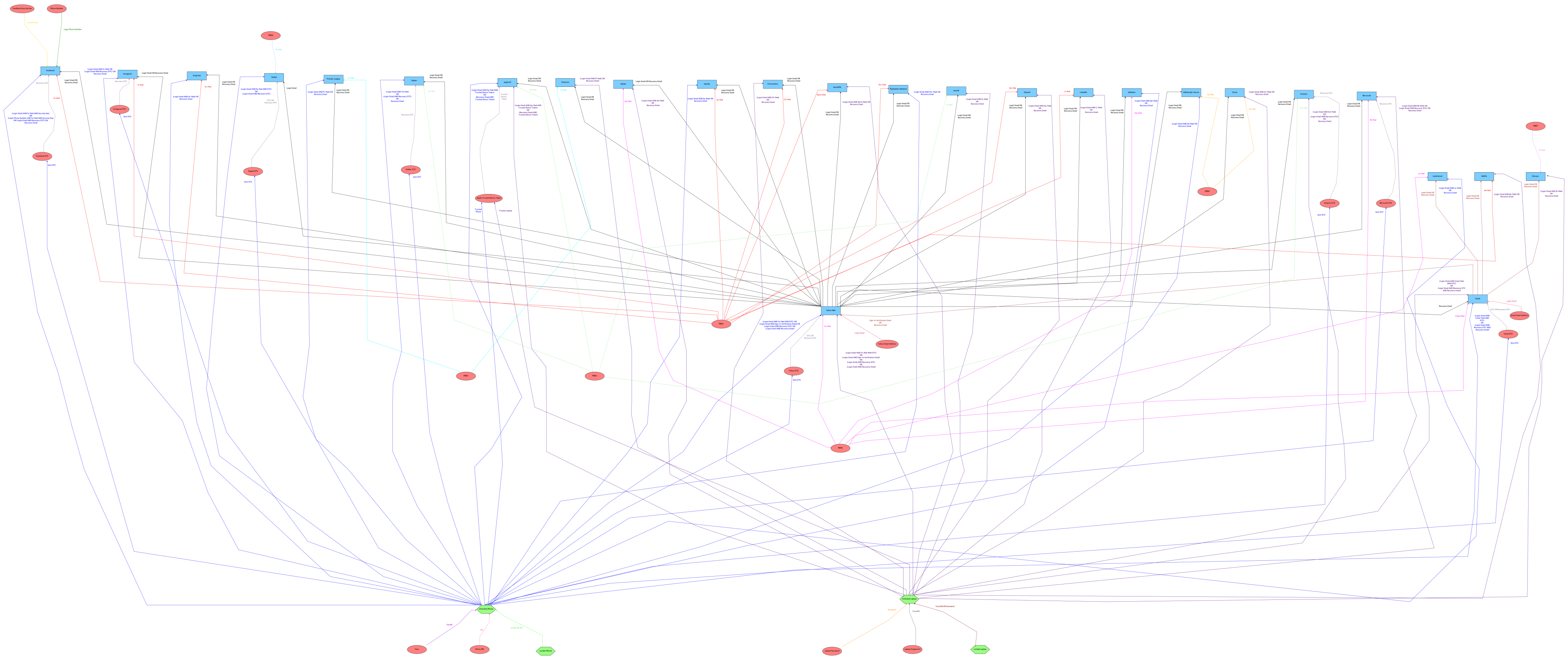


Figure B.8: Account Access Graph of Participant 8

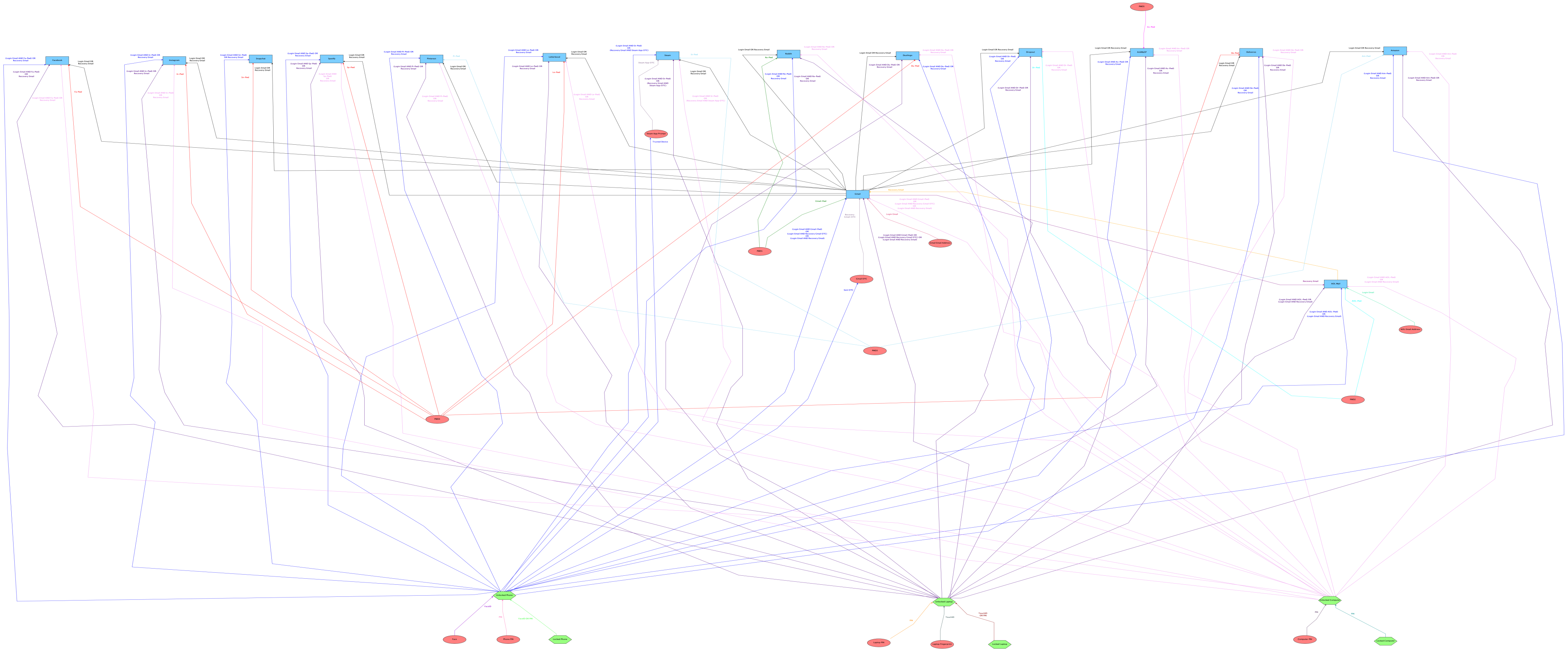

Facebook
Instagram
Snapchat
Spotify
Pinterest
Letterboxd
Steam
Reddit
Duolingo
Dropout
AskMyGP
Deliveroo
Amazon
Gmail
AOL Mail
PWD5
PWD4
PWD1
PWD2
PWD3
Steam App Prompt
Trusted Device
Steam App OTC
Gmail OTC
Sent OTC
Gmail Email Address
AOL Email Address
Unlocked Phone
Unlocked Laptop
Unlocked Computer
Face
Phone PIN
Locked Phone
Laptop PIN
Laptop Fingerprint
Locked Laptop
Computer PIN
Locked Computer
FaceID
PIN
FaceID OR PIN
TouchID
TouchID OR PIN
Login Email OR Recovery Email
(Login Email AND Fa-Pwd) OR Recovery Email
(Login Email AND In-Pwd) OR Recovery Email
(Login Email AND Sn-Pwd) OR Recovery Email
(Login Email AND Sp-Pwd) OR Recovery Email
(Login Email AND Pi-Pwd) OR Recovery Email
(Login Email AND Le-Pwd) OR Recovery Email
(Login Email AND St-Pwd) OR (Recovery Email AND Steam App OTC)
(Login Email AND Re-Pwd) OR Recovery Email
(Login Email AND Du-Pwd) OR Recovery Email
(Login Email AND Dr-Pwd) OR Recovery Email
(Login Email AND As-Pwd) OR Recovery Email
(Login Email AND De-Pwd) OR Recovery Email
(Login Email AND Am-Pwd) OR Recovery Email
Fa-Pwd
In-Pwd
Sn-Pwd
Sp-Pwd
Pi-Pwd
Le-Pwd
St-Pwd
Re-Pwd
Du-Pwd
Dr-Pwd
As-Pwd
De-Pwd
Am-Pwd
Gmail-Pwd
Recovery Gmail-OTC
Recovery Email
Login Email
(Login Email AND Gmail-Pwd) OR (Login Email AND Recovery Gmail OTC) OR (Login Email AND Recovery Email)
AOL-Pwd
(Login Email AND AOL-Pwd) OR (Login Email AND Recovery Email)

Figure B.9: Account Access Graph of Participant 9

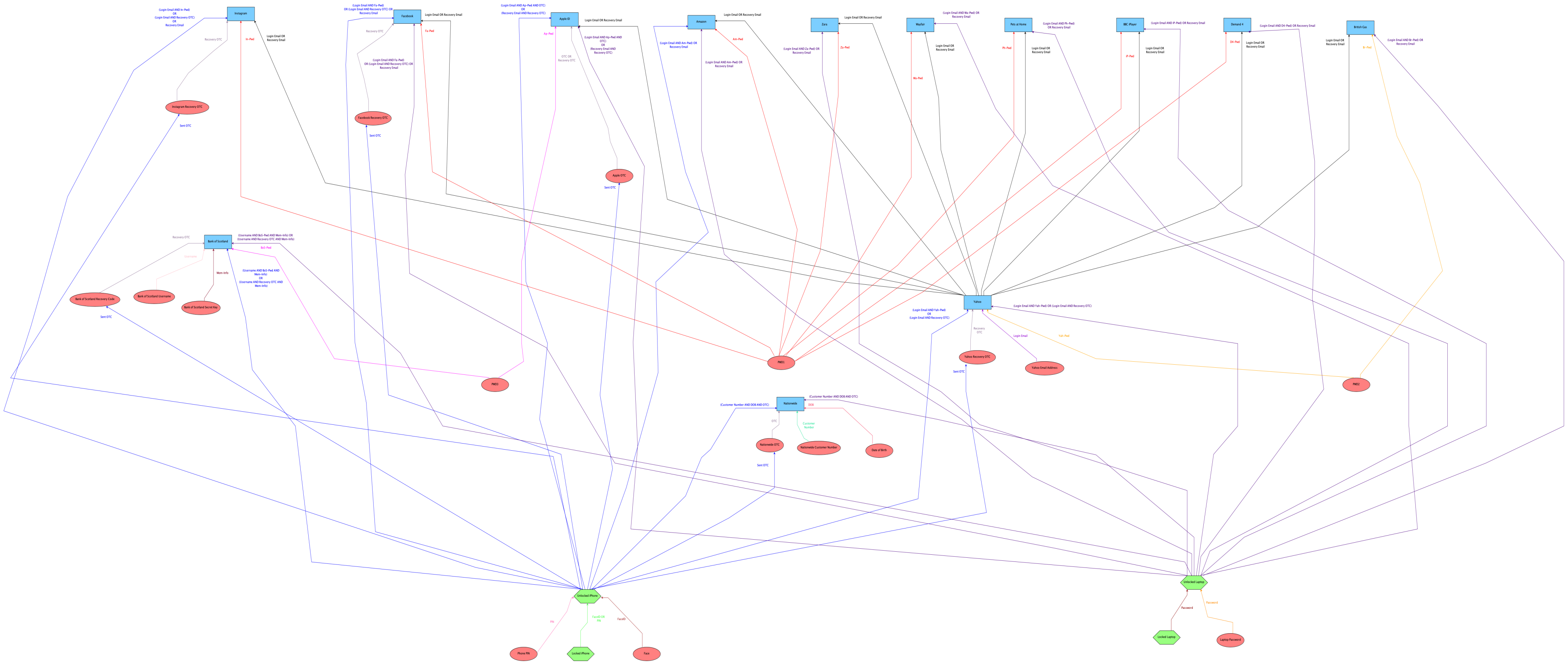


Figure B.10: Account Access Graph of Participant 10

Appendix C

Participants' information sheet

The information sheet issued to participants can be found on the subsequent pages.

Participant Information Sheet

Project title:	Using Open Source Intelligence Tools for the Automation of Account Access Graphs
Principal investigator:	David Aspinall
Researcher collecting data:	Nickon Tajali
Funder (if applicable):	

This study was certified according to the Informatics Research Ethics Process, reference number 724975. Please take time to read the following information carefully. You should keep this page for your records.

Who are the researchers?

The researcher for this project is Nickon Tajali, a 4th Year Artificial Intelligence student. This study will be used in an Informatics Honours project that is being supervised by David Aspinall.

What is the purpose of the study?

The project's purpose is to create an Account Access Graph using your data that is found in the public domain through methodologies such as OpenSource Intelligence tools. The purpose of creating an Account Access Graph is to model your account security setup to analyse how difficult it is for an adversary to gain access to your data, credentials and accounts. In an Account Access Graph, each account or credential will be represented as a node in the graph and edges that form between nodes will be represented as a connection. Each graph will possess multiple nodes and edges that will highlight the connections in your overall security setup.

Why have I been asked to take part?

You have been included in this study because we are seeking a diverse group of participants that may not have necessarily the same type of data publicly available. This diversity is critical for obtaining a variety of Account Access Graphs and for making our findings more relevant and inclusive

Do I have to take part?

No – participation in this study is entirely up to you. You can withdraw from the study at any time, up until December 2024 without giving a reason. After this point, personal data will be deleted, and the remaining anonymised data will be aggregated to ensure the impossibility of extracting individual information from both the graph and its analysis. Your rights will not be affected. If you wish to withdraw, contact the PI. We will keep copies of your original consent, and of your withdrawal request.

What will happen if I decide to take part?

The project is scheduled to run from September 2023 to April 2024. To ensure ethical and legal practices, I will firstly seek your consent to obtain publicly available data using open-source methods. The data I will be looking for includes personally identifiable information such as email addresses, usernames etc, but I want to stress that I will not engage in any illegal or unethical data acquisition methods, including hacking. Each participant will also receive a form in which they will document information pertaining to the various credentials they employ and specify the accounts to which these credentials are applied. Each participant will be able to take this form away with them to complete in their own time. The completion of the form should not exceed 30 to 60 minutes, and participants would be required to return the form within two weeks. This information will be used to create a comparison Account Access Graph. Finally, I will conduct an interview to present the Account Access Graph generated from publicly available information about you and compare it with the graph created from the information you provided (15 minutes). This presentation serves the purpose of sharing what publicly available information is known about you. I will anonymise your data before analysing it and incorporating the graphs into my report. I value your privacy and will ensure that your data is treated with the utmost care and confidentiality throughout the project. I am committed to upholding the highest ethical standards in data collection and use.

The interview will take place in person and will be an individual interview with the researcher. The interview will take place in a location without the presence of any third party or observer for privacy purposes and to create an environment where the participant will feel most comfortable.

Are there any risks associated with taking part?

There are no significant risks associated with participation. Potential disadvantages of participating in this project include the possibility of discovering that more of your personal information is publicly accessible than initially realised. While all necessary precautions will be taken to protect your data, it's important to note that no system is completely immune to security breaches or the potential theft or loss of devices where your data is stored. Rest assured that I am committed to minimising these risks to the best of my abilities.

Are there any benefits associated with taking part?

It is hoped that this work will help you gain a better understanding of how much of your personal information is publicly available, potentially resulting in improved online security practices. Your participation may contribute to research on data privacy and online security practices and allow more secure practices to be implemented.

What will happen to the results of this study?

The results of this study may be summarised in published articles, reports and presentations. Quotes or key findings will be anonymized: We will remove any information that could, in our assessment, allow anyone to identify you. Your anonymised data may be archived for a maximum of 4 years. All potentially identifiable data will be deleted within this timeframe if it has not already been deleted as part of anonymisation.

Data protection and confidentiality.

Your data will be processed in accordance with Data Protection Law. All information collected about you will be kept strictly confidential. Your data will be referred to by a unique participant number rather than by name or username. Your data will only be viewed by the researcher Nickon Tajali and David Aspinall.

All electronic data will be stored on a password-protected encrypted computer, on the School of Informatics' secure file servers, or on the University's secure encrypted cloud storage services (DataShare, ownCloud, or Sharepoint) and all paper records will be stored in a locked filing cabinet in the PI's office. Your consent information will be kept separately from your responses in order to minimise risk.

What are my data protection rights?

The University of Edinburgh is a Data Controller for the information you provide. You have the right to access information held about you. Your right of access can be exercised in accordance Data Protection Law. You also have other rights including rights of correction, erasure and objection. For more details, including the right to lodge a complaint with the Information Commissioner's Office, please visit www.ico.org.uk. Questions, comments and requests about your personal data can also be sent to the University Data Protection Officer at dpo@ed.ac.uk.

Who can I contact?

If you have any further questions about the study, please contact the lead researcher, Nickon Tajali (s2063346@ed.ac.uk).

If you wish to make a complaint about the study, please contact inf-ethics@inf.ed.ac.uk. When you contact us, please provide the study title and detail the nature of your complaint.

Alternative formats.

To request this document in an alternative format, such as large print or on coloured paper, please contact Nickon Tajali (s2063346@ed.ac.uk).

General information.

For general information about how we use your data, go to: edin.ac/privacy-research

Appendix D

Participants' consent form

The consent form signed by the participants of the study can be found on the next page.

Participant number:________________________

Participant Consent Form

Project title:	Using Open Source Intelligence Tools for the Automation of Account Access Graphs
Principal investigator (PI):	David Aspinall
Researcher:	Nickon Tajali
PI contact details:	David.Aspinall@ed.ac.uk

By participating in the study you agree that:

- I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.
- My participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data will be stored for the duration outlined in the Participant Information Sheet.

Please tick yes or no for each of these statements.

1. I allow my data to be used in future ethically approved research. □ **Yes** □ **No**

2. I agree to take part in this study. □ **Yes** □ **No**

Name of person giving consent	Date dd/mm/yy	Signature
Name of person taking consent	Date dd/mm/yy	Signature

THE UNIVERSITY *of* EDINBURGH
informatics