Application of Formal Modelling to Mitigate Metadata Driven Privacy Violations

Marine Eviette

A dissertation submitted in partial fulfilment of the requirements of the degree of Doctor of Philosophy

Department of Computer Science
University of Oxford
United Kingdom
Trinity Term 2023

Abstract

In this current Data Economy, the increasing value of data, in addition to the lack of ownership afforded to users, has brought about a plethora of digital privacy issues. These issues can be seen as a result of the rate at which data generated by users absconds to exist in the databases of data aggregators and brokers, often occurring without user awareness, conscious input, or reasoned consent. As it relates to passive data, which is mostly in the form of metadata, such data collection risks exposing users to inferential predictions made by sophisticated data algorithms on these aggregated data sets that reside in companies' databases.

In this dissertation, we have dissected the various challenging and, at times, conflicting issues in this area leading us to draft consequent subsidiary research questions to address these issues. From this, in this dissertation we consider the following question: "How can we leverage a decentralised approach to data ownership, together with formal models of data access, to aid in the protection of data subjects' privacy so as to mitigate inference-driven identity exposures from metadata collection?"

Thus, our contributions are as follows. First, we explicitly outline our understanding of privacy. We then explore a formal model for the Solid ecosystem, which serves to provide a decentralised web architecture offering user-determined access control. Whereupon, we decided to build upon this decentralisation structure via the dynamic Category-Based Access Control (CBAC) framework.

Finally, we present a formal model that has the potential to underpin models to mitigate inference-driven identity exposures from metadata collection through compounding the prior research contributions in a cohesive manner that meets our intended privacy objectives.

Acknowledgements

I am forever indebted to my supervisor, Andrew Clive Simpson, for guiding me through a challenging DPhil course, which was interrupted by a global pandemic and health issues. This dissertation would likely not have been possible without his realistic outlook and continuous support in reviewing, commenting, and correcting my work.

I must thank all my friends and colleagues in Oxford for making these years hilarious, enjoyable, and emotional and for continuously propping me up and encouraging me to fulfill the wholly exorbitant expectations they had set for me.

I am obliged to extend appreciation to my family: my father for the belief, my mother for her continued support and harsh truths, and my siblings for essentially strong-arming me into finishing; I wouldn't have been able to complete this without you all.

I also want to take the time to thank my lifelong friend Mannie for holding down the fort and therefore, granting me the space needed to continue on with my research. Then, I must simultaneously appreciate Mannie for imparting joyful moments whenever I was afforded time away.

And finally, my love and appreciation to my good friend Evie, who exposed the true limits of my intelligence and never stopped facilitating my intellectual evolution and reinforcing motivation.

Glossary of Terms

General Terms

Metadata: structured, descriptive data that informs and expands on other data, such that its main task is to aid in discovery, organisation and dissemination of information online.

Personal data: information relating to an identifiable person

Personal attributes/traits: information that pertains to characteristics of an identifiable person

Identity element: a characteristic or trait indicative of an identity

Formal Mathematical and Logical Terms

Δ: Delta (to represent a change in state)

$\mathbb{F}$: to represent a finite set

$\mathbb{F}_1$: to represent a non-empty finite set that has at least 1 element in it.

$\mathbb{P}$: to represent a power set

(x)?: to represent a free variable

$\mathbb{N}$: to represent natural numbers

$\cup$: to represent the logical union symbol, which depicts the addition relation between two sets.

$\vee$: to represent the logical OR symbol, which depicts the OR relation between two objects

$\cap$: to represent the strict intersection of objects in differing sets

$\wedge$: to represent the logical AND symbol, which depicts the AND relation between two objects.

$\in$: to represent set membership

$\notin$: to represent set non-membership

Contents

Chapter 1

Introduction

This chapter presents an overview of the dissertation. In delivering this, we first note our motivations, followed by a presentation of the research objectives. We then move to present a discussion of our research contributions before finally describing the outline of the components within this dissertation.

1.1 Research Motivation

The current ubiquitous era has given rise to monumental levels of data, with some sources suggesting that 2.5 quintillion bytes of data are being created on a daily basis[1]. Despite the perceived benefits of the technological advancements due to these great stores of data, coined *Big Data*, it would be naive to discard the drawbacks of the large volumes collected from the populous. This generated sea of information is currently being fed to data mining algorithms [1], which aim not only to foster prediction but also to establish connections between data instances and relate these to a given person or a network of connected people

[1] `https://www.domo.com/learn/data-never-sleeps-5`

[2].

Another, and potentially more serious, unobserved consequence results from the existence of metadata. Metadata has been described, generically, as 'data about data' [3] and serves as an event record of an interaction. Just as the Internet has become ingrained into everyday life, so too has metadata [4].

Whilst the definition of online metadata has remained largely the same throughout the years, the encompassed information is becoming far more complex, the quantity — far greater — and the specificity — far more concerning [5, 6].

In this thesis, metadata should be understood as, in explicit terms, structured, descriptive data that informs and expands on other data, such that its main task is to aid in discovery, organisation and dissemination of information online. As part of this definition, we mean for metadata to include entities such as author, date, time, location, sensor positioning etc.

Furthermore, despite current concerns [7, 8] over data management with reference to privacy and human rights, the area of metadata has remained relatively unexplored. In fact, online metadata is routinely overlooked as it is generated without users' explicit consent or, at times, awareness. As a result of this, there is often unregulated collection [9] of such metadata, which can have disastrous implications [4, 10] for an individual's privacy.

In this dissertation, there is a discussion into the convoluted and evolving notion that is privacy so that we are able to convey our notion of privacy to experts and non-experts alike. Privacy, though difficult to define due to the sheer breadth of the concept, has been recognised as a right since 1948 as a result of the Universal Declaration of Human Rights [11]. Resulting infringements on privacy are seen to arise due to the sensitivity of metadata as it relates to personal data [9].

In the General Data Protection Regulation (GDPR) [12], a privacy regulation intro-

duced in the EU, personal data is denoted as 'any information relating to an identified or identifiable natural person'. In relation to metadata, the identifiability of a single instance is not always immediately obvious due to the fact that single non-identifying, and therefore, supposedly "harmless" items can become extremely identifiable when aggregated with others. These aggregation models can be used to make extremely detailed inferences about a wide range of personal attributes and thus, go much further than linking actions to a specific user device.

In fact, prior research has already begun to expose the sensitivity of such personal metadata in relation to re-identification and privacy violations. Deduction of one's identity has been thoroughly explored by the likes of Hinds and Joinson [13], who investigated the extent to which our online identity identifies our offline selves by exploring demographic inferences deduced from the digital traces left behind during browsing sessions. Personality researchers [14] have also found that online users leave behind "behavioural residue" that can be used objectively to determine identity. The resulting privacy violations that occur may involve typing patterns inferring sexual orientation or health conditions and can extend much further, leading to discrimination [15] or arrests [16].

Despite these worrying revelations, methods of controlling the metadata information flow are seen to be in direct contention with the Internet's current Data Economy model, and so have little chance of success. The Data Economy, as described by Surblytė-Namavičienė, is meant in terms of the economy, where companies do business on the basis of employing algorithms and using data as an input for such algorithms [17]. From this, the Data Economy may be viewed as an emergent economy in which an organisation's performance relies largely on its ability to leverage Big Data and analytics in order to increase efficiency and meet customers' expectations. Consequently, it has previously been claimed by several (including [18,19]) that data is the new online currency, which may explain the

profound collection currently being accumulated.

Given that this problem is only set to worsen as technology becomes further intertwined with individuals' lives with, for example, the growing IoT industry, the generated metadata agglomeration will only further expand. Therefore, regulation and access control mechanisms are much needed in this area to divert privacy violations that threaten human rights and to ensure the metadata collected isn't readily exploitable. To this end, through delivering this work, we aim to provide regulation via means of access control; more specifically, an investigation into a method of access control that limits the degree of exposure that individuals are faced with.

In line with this, we will be exploring the Solid [20] distributed data decentralisation project, which aims to preserve personal privacy via Access Control Lists. The Solid proposal boasts 'true data ownership' [21], which means that users are given full ownership and thereby, control of their own data. With such user-specified control, granted by the Solid proposal, we can then reason about access control and, therefore, may begin tentative steps to resolving our specific problem.

Consequently, we consider approaches to methods of drafting and implementing access control policies, which leads us to formal methods. Formal methods in the access control space have the potential to support the specification and management of access control policies due to the complex and dynamic requirements of modern systems. In addition, formal methods grants the opportunity of greater confidence and, with it, assurance of an access control policy as well as privacy requirements. To approach an issue such as ours, we must be able to verify that our intended goals and thus, requirements are adequately implemented.

In fact, with respect to our privacy aims, it has been said that '[all] the machinery of the formal methods community can help us gain a more rigorous understanding of privacy

rights, threats, and violations' [22], which supported our intent to achieve formalisation of our privacy objectives. In our work, the aim is for these eventual privacy objectives to be implemented atop of an access control framework such that we will have the ability to grant measurable and actionable privacy initiatives.

1.2 Research Objectives

The previous subsection discusses the motivation for our work and, in doing so, paints a picture of the current pitfalls of a centralised internet, where the trade of implicitly generated user data is commonplace. The trade occurs without much user awareness as to the potential harms of collection as it relates to inferences. In our work, we seek to frame an objective that recognises these problems and can best make use of the tools available to rectify them.

Thus, to address these issues resulting from lack of privacy as it relates to metadata's role in Big Data and user personas, paired with an understanding of the centralisation on the current Internet, the following overarching objective was set:

To investigate how we can leverage a decentralised approach to data ownership together with formal models of data access to aid in the protection of data subjects' privacy so as to mitigate inference-driven identity exposures from metadata collection.

Given that this objective is fairly broad, it was noted that it would serve best to interrogate the objective through division into differing subsidiary research objectives so that we could build towards our larger objective through a culmination of smaller investigations and hence, we frame the following subsidiary research objectives:

RQ1: How can we formalise a model for privacy to tackle implicit inferential data from consideration of identifiable privacy harms?

This subsidiary objective focuses not only on which privacy requirements should be drafted but also looks into how one may define privacy in this context since privacy has enjoyed many differing definitions in the past. In light of our leveraging the Solid proposal, we shall take full advantage of their data ownership approach so that we may consider solutions to personal data management that make use of access control. By assuming users own their data, we may begin to focus on contending the implicit nature of metadata collection, such that our eventual privacy model must consider the dynamic, contextual nature of privacy as it relates to aggregates of metadata. With regards to our specification, we must ensure that prior to approaching the drafting of any access control model, we have clearly defined privacy requirements that we may feed into an eventual formal privacy model.

RQ2: How can a formal model of the Solid approach aid in reasoning about privacy in a user-focused data ownership system?

This question seeks to formally explore the proposed Solid project. Here, we look to formal models as a means of ensuring ease of understanding and reproducibility. The use of formal methods has been noted as providing a suitable abstraction [23] that can increase accessibility and aid in the verification of goals. The Solid proposal [20] allows us to reason about access control for user data through means of a decentralised web. In order to achieve this aim, Solid utilises Web Access Control Lists and personal online data stores (pods) to manage user data. By modelling an instantiation of the Solid approach, we may gain useful insights into the workings of these WAC Lists that will enable us to investigate how a possible extension may be built to resolve the issues presented with regard to metadata inferences.

RQ3: How can a formal model for the Category-Based Access Control framework help to facilitate a privacy-centric system with dynamic privacy requirements?

This subsidiary objective focuses on the exploration of the Category-Based Access Control (CBAC) [24] framework by means of translating to a formal model to aid in preventing the unintentional exposure of personal data. Due to the nature of implicit data collection, it is evident that we would require an access control mechanism that readily adapts to the given situation. Hence, we intended to utilise an access control framework that is sensitive to the dynamic changes that may occur in the system. Therefore, we investigate the dynamic flexibility afforded by the CBAC framework so that we may avoid the pitfalls accompanying static access control models.

RQ4: Is it possible to build a formal model that supports our privacy requirements through extension of an integrated Solid and Category-Based Access Control model?

Following from the privacy properties determined by RQ1, we built a formal model that may aid in privacy-preserving metadata collection. The objective of such an endeavour would be to provide a useful abstraction that lends itself to translation to other modelling platforms. It has previously been seen that a framework with logical foundations enables verification of policy properties. By investigating its use in this particular case, where policy properties are reliant on privacy-preserving notions, it is important to ensure that these notions, as it applies to our problem, can be logically expressed and upheld.

Models have also proven extremely useful in guiding development cycles, and the precedent research question, RQ1, utilises such models as it exposes the privacy aims that shall

hold in this dissertation. In its construction, we see a translation of Solove's [25] privacy harms to objectives and then to a formal model that handles implicit inferential data. Then, in RQ2, we see an interpretation of the necessary decentralisation mechanism, as provided by the Solid approach, to reason about data ownership as well as access inheritance. Then, by extending the Category-Based Access Control model described in RQ3 through the application of our defined, metadata-focused privacy requirements, we aim to provide a formally verifiable model that can mitigate privacy exposures from metadata aggregates.

1.3 Outline of dissertation

We now present an overall outline of our dissertation, as portrayed in Figure 1.1. In this dissertation, we begin in Chapter 2 with a detailed foray into the background and motivation behind this dissertation so that we will explore the key topics of data on the online web, privacy as it relates to metadata, and the role of access control in providing a solution. In Chapter 3, we describe the preliminary articles to aid understanding, such that we present items of our work that inform and introduce the research to follow.

Then, in Chapter 4, we provide a detailed picture on our method of defining not only requirements for our model but also an explanation for our inclusion decision process, given privacy's vast array of interpretations. Here, we sought to formalise our objectives for clarity and verification purposes, utilising the Solove taxonomy, in order to gain a formal translation of defined privacy harms in line with the motivations of subsidiary Research Question 1. We move, in Chapter 5, to discuss the Solid Protocol, which will be an instrumental part of our work, as the decentralisation and use of granular personal online data stores (pods) provide an invaluable foundation upon which we may extrapolate to meet our larger objective. Thus, through an investigation of this proposal, we primarily

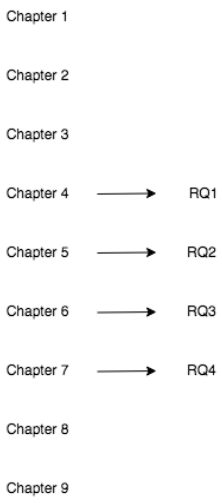


Figure 1.1: Mapping of Chapters to RQs

seek to address Research Question 2.

Before, in Chapter 6, we provide an insight into the workings of the Category-Based Access framework, which we were keen to adopt due to its flexibility and autonomous reactions; this work will set the foundations necessary to encourage dynamic requirements as per Research Question 3, whilst also providing the basis to address Research Question 4. Moving to Chapter 7, which pertains directly to Research Question 4, we lay out our eventual model that we aim to implement to mitigate personal exposure from metadata alone. Exploring Chapter 8 reveals an evaluation of our model's methods to corroborate the crafted objectives to ensure our model satisfies the goals set out concerning individual privacy protection. Finally, in Chapter 9, we outline a brief discussion in addition to our concluding thoughts.

1.4 Summary

With our brief foray into the background of our work, in addition to our outline of our research objective accompanied by the provision of three subsidiary research objectives, which we have framed as research questions, we will now move to detail a more extensive background and motivation behind our work so that one can gain an understanding of the problem we discovered before we move to discuss steps taken towards our solution.

Chapter 2

Background and Motivation

In this chapter, we discuss the background and motivation for our work. In Section 2.1 we detail the background of the data agglomerate and look at our motivation with regards to data generation and collection, and in Section 2.2 we explore the background of privacy and frame our motivation around privacy preservation. In doing so, we hope to lay down the motivations and foundations for our work involving the first of our subsidiary research questions, namely : *How can we formalise a model for privacy to tackle implicit inferential data from consideration of identifiable privacy harms?*

We then move in Section 2.3 to examine the platform we will be utilising to help us achieve our goals of user privacy through decentralisation and data stores. We detail our reasons behind decentralisation as a necessary means to create a viable solution to our over arching research question. Atop of said decentralisation platform, in Section 2.4, we describe an access control mechanism that will aid in the autonomous changes needed when considering our focus of metadata, which will pave the way to address the reasoning behind the subsidiary Research Question 3: *How can a formal model for the Category-Based Access Control framework help to facilitate a privacy-centric system with dynamic*

privacy requirements? Finally, in Section 2.5, due to the large part access control will play in our work, we delve into the background of access control as it relates to formal modelling as formal models will allow us to fulfill our main research objective to investigate *how we can leverage a decentralised approach to data ownership together with formal models of data access to aid in the protection of data subjects' privacy, so as to mitigate inference-driven identity exposures from metadata collection* .

2.1 Big data and metadata: the landscape and challenges

In this section, we explore data as it exists online, with an interest in user-generated data that exists in the databases of data aggregators and brokers. Hence, in order to explore the online data in this ubiquitous age, we delve into the depths of Big Data. Following, we narrow our focus of this collated data to the area of metadata so as to shed light on our reasons for investigating metadata as a primary area of interest.

2.1.1 An Exploration of Big Data

We begin this subsection with a definition of Big Data; therefore, we define Big Data as a term used to describe the high volume of data generated and collected from the vast array of interconnected devices, including, but not limited to, smartphones, PCs and the growing Internet of Things (IoT).

In our increasingly connected world, Big Data has become much more encompassing as well as varied [26, 27]; as such, Big Data can be viewed as an amalgamation of diverse data types: including voice, text, photos etc, where it is used to gain deeper insights, identify patterns and divulge correlations. Wherein some researchers have claimed that '[b]ig data variety is arguably the most important defining Big Data characteristics' [27]. Furthermore, Big Data has collectively been used to build clearer models for trends and

generate user predictions [28, 29]. For instance, the use of Big Data in healthcare has led to an opportunity to 'develop more thorough and insightful diagnoses and treatments, resulting [...] in higher quality care at lower costs and in better outcomes overall' [30].

With regards to the online space, the value of data [7] has since been realised by many leading to the current climate of the *Data Economy*. This Data Economy [31] has resulted in the widespread trade of data amongst data brokers and data aggregators. Much of this revenue is derived from the tailored advertising that is afforded due to the complex data algorithms that can extrapolate and predict user preferences. This situation has led to a recurring loop whereby the Data Economy has encouraged not only collection of data but also encouraged the growth in stronger analytics in order to make profit.

Furthermore, this shift in data's value is also epitomised by the reordering of the world's most valued firms: where previously (pre-mid-2000s) oil and energy companies dominated lists of the world's most valued firms; such lists are now dominated by firms such as Apple, Facebook, Amazon and Microsoft [32]. This shift is reflective of the current Data Economy, where 'many firms . . . are valued primarily for the data they have accumulated' [31].

As it pertains to Big Data, the considerable depth and breadth of available data has led to a wealth of opportunities for data aggregators to capitalise on the monitoring and predicting of user preferences. Initially, this data exchange may seem mutually beneficial to both the data subject and the data holder. However, despite the supposed benefits that users may glean from the resulting personalisation, it is widely recognised that the scale and diversity of collated data can considerably impact personal privacy [33].

This impact on privacy has been sighted as the fourth V [34], to accompany the three Vs of Big Data; namely, *volume*, *velocity*, and *variety*. This fourth V, *veracity*, is concerned with the difficulties of ensuring that an individual's data cannot be used to re-identify themselves. Re-identification of a person is a serious problem that has led to several

problems as it relates to the release of research that uses anonymised datasets; here, we note the research undertaken by Netflix [35] was subject to scrutiny following re-identification of individuals, which jeopardised their privacy.

As such, the current levels of data collection and widespread sharing present an ongoing challenge for internet users who have no means of identifying or controlling the data amassed about them or discovering where their data is going. Tracking in the online space renders users powerless to ascertain the scale of the third-party profiles, or digital personas, made about them, where it has been suggested that 'more than 80% of users gets in touch [with] the first tracker within 1 second after starting navigating' [36].

Furthermore, the cloaked veil surrounding data collection and sharing has led some researchers to become concerned about the secondary purposes for which the data may be used for. Research undertaken in [37] found that 'consolidated personal data are processed and used for numerous purposes about which [users] typically know nothing, let alone consent to it', elucidating the lack of control afforded to users.

Until now, we have discussed Big Data as it exists in the stores of data aggregators and have mentioned its variety, though we have yet to recognise the general types of data that are being collected. By this, we define the types collected as: active, the data that requires explicit user input such as a tweet or like; in comparison with passive, the data collected without explicit user input, since it is generated by the computer system or automatically recorded by web pages and may include data such as metadata. To distinguish these further, we may view active data as explicitly collected data versus passive data, which is implicitly collected.

Digital footprints, passive or otherwise, can be thought of as identifying features that are either used to infer identity elements or are themselves identity elements. These identity elements are defined as single pieces of information that are indicative of an identity [38].

Furthermore, some researchers [39] have likened digital footprints to 'digital DNA that consumers share on technological platforms including social media' [39].

As it relates to the average internet user, their data, of both the active and passive nature, is being collected in order to build a detailed online persona. This conversion of data into a personal profile for the user is termed as *datafication*. Datafication [40] is prevalent in the online arena due to, in part, the rising data generation from users and the consequent aggregation and analysis of this data. That said, in our work, we place greater focus on the implicitly collected data, or metadata, as this is often collected without users' awareness despite having disastrous implications for their personal privacy.

2.1.2 An Examination of Implicitly Collected Metadata

In the past, metadata has been described as 'data about data', which, although true, is a vague definition that lends itself to user unawareness and misinterpretation as to its complexity and prevalence [9].

Metadata's usefulness can be found in the organisation of data online, through use in e-commerce for filtering or search engine optimisation, and therefore is collected implicitly, as it is required for function and operation. In line with this, for numerous years, metadata has been considered harmless, prematurely likened to a form of bookkeeping, which has long been an integral part of record-keeping for libraries and book collections. However, as it relates to the internet, where instead, this metadata links to a digital persona, i.e. a user, it is far more concerning [5, 6].

This implicitly collated data, such as those gathered by trackers and other online organisations, allows organisations to update and refine the digital personas they build from an individual's data without the user having to purposefully interact with the online space. This leads to concerns over online surveillance, or *dataveillance*. Dataveillance [41] — the

systematic and continuous monitoring of persons as they traverse the web via data collection — has become increasingly prevalent in recent years due to the capabilities afforded by modern devices in the current era of pervasive computing.

Pervasive (or ubiquitous) computing was birthed from Weiser and colleagues in efforts 'to conceive a new way of thinking about computers' [10], where they strived for a world with technology that could 'weave themselves into the fabric of everyday life until they are indistinguishable from it' [10]. Conversely, they also began to theorise the privacy implications, noting an emergent need for different ownership mechanisms.

The resulting data agglomeration formed from the dataveillance of individuals is under the ownership, and thus, control, of data aggregators and brokers, which significantly hampers individuals' ability to control the information held about themselves. This situation is worsened due to the unconscious input that is created for users via passive digital footprints, largely in the form of metadata.

As a result, despite being an integral part of the web's infrastructure, the extraneous supplementary purposes of metadata considerably degrade user privacy by stripping users of their control as it relates to the collection and subsequent sharing of these passive digital footprints. This lack of control is further evidenced by users' lack of awareness of passive data collection and the resulting inferences that are not shared with the users.

As discussed earlier, a simple piece of harmless metadata can become much more useful in uncovering personality traits and actions when aggregated with others. Furthermore, feeding this data into machine learning models can create detailed and increasingly accurate inferences about users. In [42], the authors exposed the scale of third-party trackers on apps and revealed how it allows companies to subsequently identify users by profiling. Profiling by companies is used to identify consumers with particular susceptibility to offers of goods or services and hence can result in inferences on behaviour as well as interests [2].

Inferences, particularly those that arise from implicitly collected metadata, are a significant threat to users' privacy, autonomy and identity, as they can be used to divulge sensitive information without users' awareness. Furthermore, due to modern capabilities afforded by smartphones, wearables and the like, the range of inferential, implicitly collated data is largely unmanageable. For instance, researchers [43] have investigated how readings from the smartphone's gyroscope alone can be used to infer a user's activity at any given time, similarly feasible where inferences of users' demographics (age and gender) based solely on their daily mobile communication patterns [44]. These are inferences that individuals would be unlikely to predict or thwart as the value of metadata is often insignificant without aggregation and consequent analysis.

The use of persistent metadata aggregates may well have challenged current understanding and conceptualisation of privacy as the current online environment presents a hostile landscape in which the line between public and private information is continually blurred through the pervasive practices on the Web. This is further exemplified by revelations from the likes of Google claiming to 'more or less know what [users are] thinking about' [45].

The nature of these inferences has been found to be decidedly pervasive whilst being extremely varied. As an example, the covert use of mobile sensors, for instance, related to advert impressions, can help to further isolate an individual. Through use of the proximity sensor, it is possible to verify whether an individual is present at the time of an advert delivery and through use of the camera, it is also possible to record user reaction and gauge emotions, helping to enrich not only the advertising profile but helps to refine users' profile to predict preferences and interests [46].

Whilst these inferences cause serious privacy harms, we note that not all exposures necessarily cause harm - for instance, metadata use in authentication systems and timestamps.

Though, we move to consider the harm of incorrect inferences. Given these inferences are not shared or checked with the user generating the inferential data, these can lead to a host of problems with respect to validity. Research in [47] speaks to the issues of inferential distortion by explaining that individuals currently have no means of correcting false information about them, which is particularly troubling as this information is used to make a range of decisions about them.

With regard to inferences, we find current privacy measures fail to account for the nature of metadata and its role in the inference of sensitive articles. While the GDPR delivers a promising start for reasoning about the management of identifiable information, the shortcomings of the regulation have already been recognised by some researchers (see, for example, [47]), particularly as it relates to these inferences where it's been noted that: 'the GDPR . . . provides insufficient protection against sensitive inferences (Article 9) or remedies to challenge inferences or important decisions based on them (Article 22(3))'.

To this end, due to the implicit collection of metadata, users are left contending with a hostile environment that readily exploits their privacy. In this respect, the widespread collection and trade of metadata unsettles data privacy expectations as users are unable to limit access to personal information with respect to collection, trade and subsequent inferences and analysis. Further, its implicit nature leaves users with a grave knowledge gap as to the information attained or inferred.

2.2 The Privacy Challenge

In this section, we strive to investigate the breadth of privacy as a multidisciplinary notion that continues to evolve with the times. In doing so, we note the varying definitions of privacy and divulge the changes that have occurred throughout time, before we move to discuss privacy in relation to metadata and access control.

2.2.1 Privacy Definitions and Hinderances

Privacy is a comprehensive yet convoluted concept with diverse definitions, solutions, focuses and aims. Furthermore, the definition of privacy is an ever-encompassing, fluid notion that relies on culture, law and experience to frame. Such a description is highly reflective of opinions from a range of scholars, including Arthur Miller, who described privacy as 'difficult to define because it is exasperatingly vague and evanescent' [48].

Delving deeper, we find the journey to elucidate a definition of privacy as it exists as an opposing feature to *the public* has been long and arduous, with refinements that have taken place due to its obscure and contending elements. Nonetheless, we may briefly explore earlier thoughts relating to privacy as it pertains to its counter, the public space. Here, we observe a slight distinction marked by Aristotle [49], which, although subtle, speaks of the *polis*, or public, as a communal sphere, in contrast to the *oikos*, which refers to one's household and property.

In the 1690s, John Locke made an attempt to discuss this elusive notion in his discussions on the Law of Nature [50]. Here, we look to his distinction of man as a property of himself, which we may extrapolate as man's private sphere in contrast to what he framed as nature, upon which he decided no single individual can make a claim of property, i.e. it is a communal space.

One of the first well-known entries of privacy into law was achieved due to the Warren and Brandeis article [51] discussing the infringement caused by the, then, novel camera and instated a posit referring to a 'right to be let alone'. From this, there was an influx of discussions surrounding the issue from the likes of [52, 53]. It had clearly set a precedent of expectations with regard to privacy and was further extended as a result of public backlash following the Robertson vs. Rochester case [54], in which a young teenager had a portrait of hers used for advertising without her consent. Following this and the surrounding

controversy, there was an introduction in privacy rights in the New York Civil Rights Law [55] that protected the likeness of oneself from being used for purposes of trade without permission.

Following, we observe that in spite of the differences in conceptualising privacy, it has still been viewed as an important factor in human life as it is noted that '[over] 130 countries have constitutional statements regarding the protection of privacy' [56]. Furthermore, privacy has held a role as a fundamental human right as per Article 12 of the 1948 Universal Declaration of Human Rights [11] and has seen recent adjustments to adhere to privacy issues of the modern world [57].

Subsequently, we see the evolution of privacy in Western life, which has largely been precipitated by the advent of technological advances; for instance, we may appreciate the notable change with respect to privacy expectations following the adoption of landlines and the introduction of the personal computer with the Electronic Communication Privacy Act of 1986 [58].

Several of these changes have led many to re-evaluate the definition of privacy and its role in self-development as well as self-expression [59]. In fact, it has been found that privacy lends itself to autonomy as to preserve personal privacy, individuals uphold the right to disclose and conceal as one sees fit, which in and of itself speaks not of concealing harm but speaks to possessing control over which aspects of self one chooses to reveal and amongst who.

This element of control with respect to privacy was believed to be an inherent part of the online self; whereupon users may choose to present themselves however they see fit in a variety of different online environments.

However, clear contradictions to this notion were found, where it was discovered that a unique persona could be derived regardless of the differing personas created by a user [60].

Therefore, if users have no control over which aspects they wish to share and withhold, it stands to question whether privacy exists within the online realm and what has been done to protect it.

As evidenced by earlier examples, it is, perhaps, unsurprising that with technological progression as it relates to the growing capabilities and advent of the internet, discussions related to privacy have entered the online sphere. Privacy in the online sphere was consequently rebranded as data or *information privacy* since it largely related to information users share, and companies collect. Information privacy [61] refers to the privacy of personal information as it relates to personal data on computer systems.

Information or data privacy has been a huge topic of concern due to the extent to which data is collected, shared and aggregated online. Due to the resulting challenges with regard to this collection, standardised online privacy has proved difficult. As it relates to data, as we discuss later, users do not own their own data and are mostly unaware of the true extent of sharing; this, along with behavioural patterns of use, are amongst reasons a unique persona can be deduced.

Looking to current research in the area of privacy, we may focus on the multidisciplinary vantage points that have since been investigated. From a more philosophical standpoint, as we look to modern-day considerations of privacy and big data we find the likes of the Extended Mind [62]. The Extended Mind theory is based on 'the idea that the use of various tools, artifacts, or aspects of the environment can facilitate, enhance, or even constitute cognition' [63].

Such reasoning speaks to an additional layer of privacy protection needed as it feeds into items such as a smartphone being a core part of an individual's mind. In our work, as we try to limit access to self it makes a good argument that this should also include articles separate from a given person that is capable of exposing an individual's traits and

personality.

Clark's [62] investigation into how epistemic actions performed within the brain as part of a person's thoughts are as tangible a cognitive process as those performed using a device, leads one to reason about the state of predictive machines that may be performing epistemic actions on a user's behalf to predict their behaviour, which is decidedly worrisome for privacy as well as individual autonomy.

Further, we find from looking at work undertaken by Joinson and Hinds [64] that in evaluating approaches to personality determination, it was discovered that there remains a distinct lack of multi-disciplinary research in resolving personality determination insights. Joinson and Hinds' two-part study using human and technological-based determinations of personality demonstrated that 'personality can be predicted from digital traces' [64].

Furthermore, Joinson and Hinds found that there exists a convergence between human perception of personality and those from computer-based prediction. All of which gives credence to our work in upholding privacy through the limitation of access to user-generated data, particularly those of a passive nature, since it is generated without conscious input and thus may be primarily personality-based.

Then, as we look to interrogate data collection's role in identity or personality determination, we may focus on the part metadata plays. Here, we see others in the area who talk of the data fragments that 'collectively and passively create a hidden identity built up from metadata' [6]. Furthermore, the role of metadata in identity determination is expressed through the posit of the question: 'Might the passive digital footprint more accurately reveal the individual's genuine or authentical self than the individual realises?' This supports our positioning in delineating the online projected self from a larger, more complete and perhaps truthful presentation of the self.

Finally, turning our attention to the economic realm, we may discover that the current

threat to user privacy has also been realised by economists in the space, giving rise to scholars [65, 66] categorising it as a new form of capitalism dubbed by some [66] "techno feudalism". This is derived from technology companies capitalising from user data through recommendations and presentation of differing items based on personal user history. Techno feudalism works on the basis that technology companies harvesting user data are able to utilise predictive algorithms to exploit the data for financial gain.

Using their various platforms as a marketplace they breed popularity through personalisation and thus both users and producers are forced to buy into their services. This leads to a situation where these companies have unprecedented control of the market similar to the state of affairs in Medieval England. Whereupon the current economic state can be likened to the next level of capitalism with regressive characteristics of a state before capitalism but with technology companies as acting lords of the land.

That said, critics of techno feudalism (e.g [67]) have noted the dramatic and hyperbolic nature of the term and are quick to scrutinise its use as a means of garnering media attention and thus describe it as inherent 'intellectual laziness' [67]. Though it seems while the semantics of the term may draw critique, it should be noted that despite the 'definitional ambiguities' [67] concerning the state of the economy due to the exploitation of user data by technological corporations, there still exists a current landscape of exploitation by way of the attention economy, predictive algorithms and personalisation [15] which companies are profiting [68] from.

With these revelations, we may be certain that as the technology has evolved, so too has the range of possible privacy exploitations and the requirements for privacy maintenance as characterised by, for example, the latest GDPR Regulations [12], which mention not only the breadth of personal data but also touches upon the exploitable metadata found online, yet without well-defined methods in place to reason about the protection of privacy as it

relates to metadata; proposed solutions will likely fail to account for all possible threats and situations.

That said, before we formally define privacy as it applies to this dissertation, we must first consider hindrances to privacy that degrade user intention. We briefly mention the term, the *privacy paradox* [69], which is a phenomenon whereby an individual's attitude towards privacy is in sharp contrast with their behaviour. Here, it has been observed that despite individuals claiming to have concerns for their privacy, they do not behave accordingly. This has been studied throughout the online space [70, 71] and, as a result, it is evident that the path to privacy is not as straightforward as users might have hoped. Instead, it is fair to reason that there are many hindrances to achieving privacy for an everyday user, which has been explored by many.

One of these hindrances stems from the utility gained by users through data sharing as well as that gained by data collectors. In the online realm, privacy is often treated as a good for trade: users can get a better reward from services, provided they give up information that they may have previously been unwilling to share. With regard to data collectors, 'the utility of a data source lies in its ability to disclose data' [35], and thus, inherently, privacy considerations may considerably harm utility. In fact, it has previously been claimed that whilst an app may be free, the product is actually the user data. This has been supported by research [36] that concluded that free mobile apps requested data to a greater extent than paid mobile apps. Furthermore, the data that is collected and harnessed for profit may be extrapolated for purposes beyond its initial intention, thereby making it a potential danger to privacy.

Similarly, another interesting stance that has been studied in relation to privacy hindrance is that of privacy fatigue. It has been noted that '[over] time it has become burdensome for users to maintain their privacy' [72] due to a lack of awareness, an underes-

timation of risk and the overwhelming choices that are difficult, for an everyday user, to understand. Privacy fatigue describes a state of weariness towards privacy management whereby individuals are met with a grave "lack of control" feeling, such that an attempt to adhere to privacy best practices seems futile, given the frequency of data breaches as well as privacy-averse design dark patterns [73, 74] and long-winded terms of use or privacy policies [71, 75].

This brings us to another privacy hindrance, which is that of dark patterns. These dark patterns have been researched in reference to user manipulation. It was found that aggressive dark patterns can be described as user interfaces that are designed 'to manipulate the consumer into doing something that is inconsistent with her preferences' [76], whereupon the question of legality arises as 'many dark patterns appear to violate federal and state laws restricting the use of unfair and deceptive practices in trade' [76].

Another hindrance stems from the nature of data ownership; the state of the internet means that a considerable amount of the data is generated by each user, but this does not translate to ownership. This data that forms an individual's digital trail and the lack of ownership results in increasing difficulty for users to keep track of their data, specifically as it relates to the analytics later performed on aggregates to make predictions.

2.2.2 Privacy in Context

Following our exploration of privacy, it is evident that there is a pressing need to carve out a definition for privacy for the context of this dissertation, given its subjective nature. Therefore, we now move to define privacy as it relates to our work. As such, in this dissertation, we define privacy as *'a limitation of access to self'*. A definition that is not only reminiscent of earlier philosophic musings but is highly appropriate as we move to the online sphere.

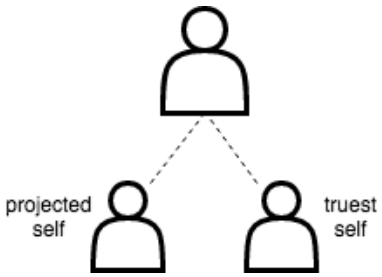


Figure 2.1: Identity profiles for an individual

In this dissertation, we construct a view of privacy in terms of two identity profiles for a single individual. As portrayed in Figure 2.1, we may view each individual as having two identities: an identity that the individual chooses to present in the online realm and then a separate identity that is the individual's *'truest'* self, which includes their thoughts, feelings and personal attributes as well as sensitive articles that they may not have shared with any other individual either online or offline.

The identity that is built from projection on the online space can be seen as the aggregates of all the separate selves a user may present on the online space. This aggregation serves to reason about company aggregation models and interrogates the nature of information that can be found online for a given user. The reduction of the online identities for a given user into a singular *'projected identity'* allows for consolidation of data, allowing users to gauge which parts of themselves form part of their online public persona.

Privacy, here, seeks to protect this latter, *'truest'* identity profile from being unwillingly or unknowingly exposed through collection of metadata. By modelling privacy as akin to an access control problem concerning identity profiles, we can define privacy, in this context, to be a limitation of access to the *'truest'* identity profile, and seek to build access control models that combat inference-driven identity exposure that threatens this profile. We aim to achieve this through clearly defined policy rules that handle aggregation of metadata and consequent inferences.

In line with these aims for clearly defined policy rules, we first move to formalise our privacy definition. We understand that formalisation of this definition of privacy will later allow for ease in drafting and verifying the specification so that we may provide one clear semantic interpretation of privacy as it relates to our work.

Thus, in order to propagate our objectives, it is imperative that we formally explore privacy, as defined in our work, as it pertains to identity profiles and the eventual restriction of access to data objects. This linkage of data objects with identity has led the way for online identity profiles to be built from *datafication* of personal information [76]. Hence, it is our aim that through modelling the identity profiles as a collection of data objects, we may be able to begin reasoning about the formation of metadata-driven identity profiles created by companies.

We will later move, in Chapter 4, to formalise our conception of privacy so that we can clarify the scope of privacy that we are interested in addressing, though we first may begin with a description of privacy as it should apply in our work. As aforementioned, we take privacy to refer to a *'limitation on access to self'*, or more specifically, given our interest in the division of an individual's identity as it exists in the online space, we may further refine our privacy definition to be '*a limitation of access to the "truest" self identity profile'*, where each individual has two identities, a projected identity profile and a *"truest"* identity profile. That said, due to the level of protection we seek to provide for this latter identity profile, we shall instead refer to the 'truest' profile as the *protected* profile for the remainder of this dissertation. Figure 2.2 depicts this notion.

Moving to formally describe this notion, we may consider the projected self as a collection of identifier data objects, which are the result of explicit user entry or active digital footprints. For instance, a Facebook "like" on an advert generates an active digital footprint, which describes that the user in question likes the particular product being ad-

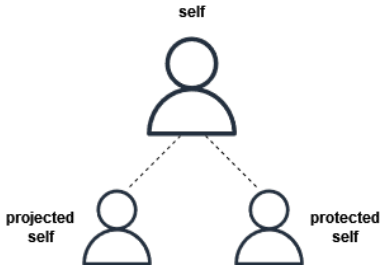


Figure 2.2: Identity profiles for an individual

vertised [77]. With respect to our proposed model, this would form part of the projected self, as the user is explicitly expressing this positive engagement. For Facebook and the companies that it shares this data with, the positive appraisal for this product would have been acquired; hence, it may form part of a "known" set of identifier data objects. This is in contrast to a company that Facebook does not share its data with; said company would not have the aforementioned positive appraisal in their set of "known" identifier data objects, since the user has not made this known to this particular company.

As a result of this, in our proposed model, we view the set of data objects, that is collected by/shared with a company, as a "known" set of potential identifier objects for a particular individual. In this way, this known set will vary from company to company, dependent on the data the company has acquired. Finally, we may consider attributes that the user has not shared; if a user does not explicitly share their age anywhere across the online space, a data object pertaining to age can be considered to be part of a private set of data objects. This private set would make up the protected self and thus would be the objects that we aim to protect, or keep private. As a result of this, we aim to provide a means to ensure that the collection of metadata does not lead to an inference of any particular private data object.

2.3 The Solid Approach to Data Ownership

We have explored the pitfalls of the Data Economy and Big Data, which were heavily impacted by the lack of control granted to users. Here, we discovered that a particular focus was needed in the area as it relates to data management and user privacy. Hence, we now aim to discuss a proposal that offers to bestow control unto users through decentralisation and data ownership. This proposal follows that of many in the area, e.g. [78], that have looked to decentralisation as a means to revert to the original vision of the web, as described in Barlow's manifesto [79].

In spite of the vision for the web centring around decentralisation and inherent user control, the current web is plagued with centralisation, which leads to an online arena with disadvantages that are very similar to the pitfalls of corrupt central entities, such as corrupt governments and companies; whereupon we find the web today ripe with surveillance, data breaches, privacy loss, and manipulation. As a result, we see that a 'few large companies now own important junctures of the web, and consequently a lot of the data created on [it]' [78] — so that, whilst individuals largely remain the source of their data, this does not translate into automatic ownership.

The case for ownership becomes increasingly complex when we consider the analysis done on data aggregates. Here, companies are utilising complex predictive algorithms to analyse user-generated data with aims of identifying behavioural patterns and generating subsequent inferences. At times, these processes generate a further set of data that is often as valuable as the original data. Yet, this analysis information is automatically owned by the parties performing these analyses.

In our work, we reason about data collection and access control from the perspective of users, which is decidedly a very challenging task, as users do not typically control access to their collated data. Therefore, we understood the need to investigate projects allowing

users to regain ownership, and consequent control of their data, through the introduction of decentralised architectures that can be implemented atop the current internet. One such work is the decentralisation project, Solid [20], created by Web inventor, Sir Tim Berners-Lee, and colleagues.

The web, as previously mentioned, was created with the intent to facilitate easy data sharing between researchers across the internet in a largely decentralised manner. However, the web today, as we have alluded to, tends towards centralisation as data is controlled and processed by a few large companies. The Solid project aims to rectify the resultant privacy dilemmas that grew from such monopolisation. Here, Berners-Lee and colleagues began work on the Solid proposal — a distributed data decentralisation project that seeks to enable information sharing in a privacy-preserving manner.

In the Solid approach, data generated by each user is written to their own individual "pods": an abbreviation of "personal online data stores". These pods may be hosted wherever the user chooses, and the user can then authorise granular access of the pod to other parties as they please. As a result, authenticated applications are allowed to request data given the user has given that particular application permission.

The manner in which generated data is dispersed across pods is determined by the user's preferences. Therefore, a user may have separate pods containing differing information; for instance, the user may have a pod for financial information and another for medical information, and so forth. This separation lends itself to a more restricted set of information available to companies, where, for instance, a user — utilising an authenticated social networking application — need only grant permission to certain information in a specific pod whilst restricting access to others. Consequently, this restriction results in the user retaining full, granular ownership and control of their data.

For managing said control over data, Solid utilises Web Access Control (WAC) Lists.

These Web Access Control Lists are very similar to the Access Control Lists generally used in Discretionary Access Control [80] policies, but instead, both users and groups are attached to URIs or WebIDs. In comparison, the resources are identifiable by assigned URLs that may refer to web documents or resources. We observe that to handle the permissions, these latter resources are accompanied by a set of Authorisation statements that describe:

- which agents have access to the resource; and
- what type, or mode, of access the given agent has.

This structure supports the aims of flexibility and grants users, as described by Solid researchers, "true" granular ownership. To conclude, we see that user ownership of data translates not only to control but also the Solid Proposal's use of separate granular pods, with varying purposes as dictated by the user, encourages ease of control, which is a much-needed feature given everyday users are not access control experts.

2.4 Category-Based Access Control

Having briefly described the non-expert roles users will play in a decentralised system, we move to discuss a method of access control that encourages autonomous changes, which will decidedly aid users as they attempt to maintain their privacy. Here, we introduce an access control framework that aims for ease of use and clarity by providing a more 'fundamental' form of access control. Therefore, we move to describe the Category-Based Access Control [24], which has previously been labelled as a more primitive form of access control, by this Barker was attempting to elucidate that this form of access control was intended as a foundation for access control frameworks to be built upon and specialised from.

In our work, we gravitated towards Category-Based Access Control due to the degree of adaptability and consequent flexibility afforded. Here, we noted that Category-Based Access Control (CBAC) presents a foundational conception of access control by considering notions of categories to provide flexible access control concepts. Subsequently, these concepts can be specialised for particular scenarios or requirements.

Concerning our problem of interest, CBAC has the potential to provide the foundations for a model that allows one to reason about autonomous changes to permissions. In CBAC, we find that permissions are associated with categories, which can be defined based on, for example, roles and resources, as well as attributes and geographical constraints. Hence, permissions can change when a user attribute or geographical location changes without intervention from an administrator. This functionality would be decidedly beneficial in our interest area regarding, for example, thresholds for data aggregation and inferences, whereupon a permission can readily be withdrawn.

This form of access control — CBAC — was introduced by Steve Barker, who noted the varying introductions of novel access control policies to solve specific bespoke access control problems. Instead, he envisaged a framework that could easily be adapted for individual and unique needs to encourage a reproducible base upon which varying solutions could be built. In line with this, we felt drawn to the logical foundations proposed in this framework that may be refined for our particular needs.

Said variability can be seen in its adoption by a range of different researchers, including Asad et al. [81]. Here, we find that in pursuit of a new hybrid approach to policy enforcement, Asad et al. looked to CBAC. Thus, in line with this hybrid approach of using both "static" and "dynamic" access control concepts, the researchers employed the CBAC framework to extend the roles of their prior RBAC solution through definitions of static categories, as well as to introduce much-needed dynamic categories. In fact, in earlier

research, they discovered that 'in the context of Object-Oriented programs [...], it is very difficult to know by statically analysing code which user, with which of their associated static access rights, can execute any method call found in the code' [81]. Hence, these additional dynamic categories would likely be able to rectify this situation.

Another example of adoption can be found in the use of CBAC in privacy-enhancing data collection from IoT devices by Fernandez et al. Here, researchers highlighted the need 'to develop new mechanisms to control which/when data is transmitted from IoT devices' [82], describing how it was crucial that 'this control should take place before the data is transmitted' [82]. Said motivation to enable users to specify fine-grained access control mechanisms can effortlessly be achieved via the CBAC framework.

Furthermore, it has been demonstrated [24, 83, 84] that CBAC can be used to model a range of access control policies, such as Discretionary Access Control (DAC) as well as Role Based Access Control (RBAC), providing a robust argument for its use as a foundational access control framework with the flexibility to be refined based on an individual's needs and objectives. To this end, we not only hope to use the CBAC framework as a means to demonstrate a solution for our problem, but we aim to use it to build upon the decentralised access proposal of Solid, whose use of Web Access Control introduces many similarities with that of DAC, which, as aforementioned, CBAC has been used to model.

2.5 Formal Modelling for Access Control

In this section, we detail not only the importance of access control as it relates to our work but also examine the role of formal methods in the presentation of access control schemas and policies. We have previously defined our definition of privacy and pointed towards access control being a crucial part of our work; however, we now move to explicitly explore our use of access control while simultaneously shedding light on the benefits it can provide

in this space.

Our preliminary steps in this area involve the use of formal methods as a means of expressing access control policies. As we outlined earlier, we have delved into the formalisation of our objectives as it relates to privacy, which we had offered to ensure the clarity and verifiability of our privacy objectives. Although, we shall also note the importance of formal methods regarding more general access control.

We shall begin our discussion with an acknowledgment of our definition of access control. To this end, Access Control can be described as a process that places constraints upon what can be achieved in a system by each party. Its primary purpose usually serves 'to limit the actions or operations that a legitimate user of a computer system can perform' [80], though may extend much wider based on the policies of any given system. Generally, Access Control is concerned with interactions of four key elements: subjects, objects, operations, and a resource monitor. Here, "subjects" refer to the users or processes of a system; "objects" describe resources in the system such as files, servers, databases, and so forth; "operations" pertain to permissions and abilities of the subjects in the system — in reference with its objects; whilst, the reference monitor enforces the policies bounding the system.

In its entirety, an Access Control System can typically be described as providing four primary services, which are namely: Identification, Authentication (I&A), Authorisation, and Accountability. Jointly, Identification and Authentication is a two-step process working to determine who can use the system — first, a user must identify themselves before completing the authentication step, which verifies a user's claimed identity. Authorisation defines and determines the allowed actions of an authenticated user, which typically includes the likes of read, write, and execute as found on computer operating systems. Whereas Accountability determines an audit trail associating users with performed ac-

tions, helping to ensure the integrity of the system is upheld.

The consequent development of such Access Control Systems describes a three-point categorisation consisting of policy, model, and mechanism.

- The Access Control Policy: explicit rules that govern the system— correlates to the authorisation step and can be thought of as authorisation policies.
- The Access Control Model: a clear representation of the access control policy, broadly fit under Mandatory, Discretionary, or Role-Based.
- The Access Control Mechanism: defines the low-level functions enforcing this policy such that the access control mechanism must work as a reference monitor for the system.

In this dissertation, we primarily focus on the authorisation stage, and with it, we describe the corresponding access control policies and the consequent modelling. In this space, we note that these Access Control Policies have seen substantial growth and evolution accompanied by several ad-hoc modelling solutions as well as mechanisms. Though we shall attempt to pursue a more fundamental approach to access control due to our interest being more focused on privacy concerns and restriction to data as opposed to the habitual case for security concerns over data access; therefore, we are more concerned with the notion of access control meta-modelling.

Notably, researchers [85] found that 'a large number of security breaches are caused by policy misconfigurations', which suggests that although the importance of encompassing policies has long been understood, drafting of policy specification, at times, presents a unique challenge as the resultant specifications are not always necessarily in line with policy objectives. In fact, comparisons have been drawn between the state of policy specifications and that of software-hardware developments before the introduction of formal verification

techniques [85, 86].

Furthermore, it has been found [87] that a multitude of papers describing policies are somewhat informal or present access control mechanisms through examples — without any formalisation of policy concepts, which may aid in understanding a particular access control system's workings but does little to help with implementation in a separate context.

Further, due to the distributed, complex, and larger-scaled systems that we are faced with in the modern day, we have witnessed development of a multitude of differing models for access control. It has been noted [88] that a major difficulty concerning access control 'lies in the interpretation of, often complex and sometimes ambiguous, real world security policies and in their translation in well defined and unambiguous rules enforceable by a computer system'. As a solution to this, methods of simplifying rules, including operations on rules such as regulations for rule compositions, have been proposed; however, a more tangible method to bridge the gap between policy and mechanism has been found in utilising a more formal modelling approach.

Though, even while utilising Formal Methods as a tool for access control, the types of existing policies that may be formally modelled are not all-encompassing as exemplified through the introduction of numerous novel access control methods, e.g. [89–91]. The paper, [24], addresses this problem through introduction of a general interpretation of access control using a 'unifying meta-model', which may provide a basis upon which our research augments as portrayed in a similar paper concerning privacy and general data collection on smartphones [82]. Thus, we may discover that a primitive, as described by Barker [24], approach to access control facilitates reasoning about privacy concerning implicit metadata collection and inferences.

The strength of formal methods as a means of abstracting unnecessary complexities will be much appreciated by those in the field of privacy such as [22, 92], so that we can

make quantifiable differences in the exposure individuals are faced with as it relates to the implicit collection of their metadata and the consequent inferences that can be made about them.

In our work, we pursued formal modelling through the use of the Z notation [93], which is based on axiomatic set theory and first-order predicate logic. Using Z notation to model access control has provided a way to model an abstraction that can be checked for inconsistencies and policy conflicts. As well as providing the necessary formal foundations, Z schema languages can be analysed using the ProZ animator, which allows for mechanical validation of a given specification [94, 95].

2.6 Summary

In our background and motivation, we have explored the issues of data collection and sharing on the internet, primarily as it relates to user data management and, consequently, user privacy. As part of this, we have considered the evolution of the concept of privacy to aid in the understanding of the current privacy harms that online users contend with. Subsequently, we have explored the implicit nature of metadata and exposed its effect on privacy with respect to digital personas and the economy of privacy for trade.

To this end, we have delivered an exploration of online privacy with an insight into a more focused privacy definition that we would be taking forward in our work. Following, we introduced the Solid proposal to portray how a decentralised architecture with aims of providing user ownership aligned with our intention to offer user-modulated data management and consequent privacy. Here, we discussed its use as a platform upon which we could begin to visualise a solution to our problem.

We then move to outline the access control framework of Category-Based Access Control (CBAC), which we saw as a potential access control solution due to the autonomous changes

that the system could perform. CBAC, we decided, would likely prove highly beneficial when protecting non-expert users from unwanted inferences. Building on the usefulness of the CBAC framework, we then highlighted the advantages gleaned from formal methods and access control paired with data ownership in providing a solution to these issues. Here, we briefly discussed the benefits of formalisation concerning clarity and ease of translation such that we are in a position to begin demonstrating how these elements comprise our solution to the aforementioned overarching research objective: *To investigate how we can leverage a decentralised approach to data ownership together with formal models of data access to aid in the protection of data subjects' privacy, so as to mitigate inference-driven identity exposures from metadata collection.*

Chapter 3

Preliminaries

This chapter serves to inform on the necessary basis upon which our solution has been developed. Due to the numerous components that has been layered atop one another to form our proposed solution there is a need to preliminary investigate aspects of the individual components to further aid understanding. To this end, we shall begin by exploring the formal requirements behind the CBAC model and the Solid proposal. Thus, in this chapter, we seek to explain not only the mechanism of the Category-Based Access Control (CBAC) framework but we shall also explore the reasons behind the crafting of such a framework. Following, we shall then explore the basic tenets of the Solid proposal to aid in the formal interpretation of the Solid proposal, as described in Chapter 5.

We note that the delivery of these preliminaries provides the necessary background behind our resolution of the main research objective, which includes a description of a Solid-style approach to data ownership together with formal models of access, via means of the CBAC framework. In part of addressing this research, we have demonstrated, in Chapter 2, the need for a decentralisation platform to base our solution on, whereupon we selected the Solid proposal — drafting a subsidiary research question that looks to explore:

how a formal model of the Solid approach aids in reasoning about privacy in a user focused data ownership system . Thus, in this chapter, we deliver upon the workings of Solid to aid in the comprehension of the latter Z formal model presented in Chapter 5.

As it relates to the CBAC framework, we take a similar stance, wherein we explained the need for a dynamic autonomous access control framework when handling the privacy implications arising from metadata. Therefore, as part of the preliminaries, we work to explain the necessary background and propositional axioms behind the framework to help guide our resolution of the third subsidiary research question: *how can a formal model for the Category-Based Access Control framework help to facilitate a privacy-centric system with dynamic privacy requirements?* Thus, we present these preliminaries with the hope that the articles outlined in this chapter will later aid in understanding of the various components, sets and schemas that we have chosen to include in the Z formal model found in Chapter 6.

3.1 Understanding Category-Based Access Control

A key factor in the development of CBAC [24] was that Barker sought to address the multiple access control mechanisms that were being readily created to solve each novel problem by exploring a more foundational approach, seeking to deliver a basis for access control he came to describe the CBAC meta-model [24].

Although RBAC had speculatively been thought to provide a basis for access control [96, 97], this had been challenged by the likes of Barker, who proposed an alternative Category-Based Access Control meta-model, which he determined could be used to describe most access control models, including RBAC [24, 83, 84].

We present Figure 3.1 as means to express the main tenets of CBAC as an interaction between users, which Barker describes as principals, as well as categories and permissions,

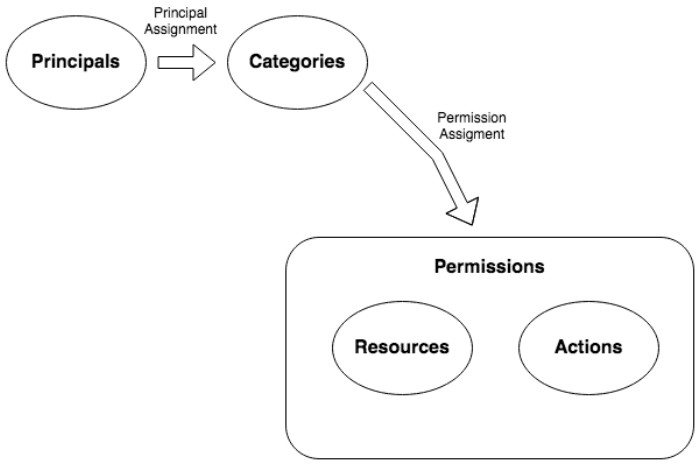


Figure 3.1: The CBAC framework

where permissions are described as a relation between actions and resources. Figure 3.1 describes the core concept of CBAC in an accessible manner to aid with understanding as the subsequent mathematical notation is introduced.

With regard to our specific problem, CBAC may prove extremely useful when reasoning about autonomous changes to permissions. In Category-Based Access Control, permissions are assigned to categories of users as opposed to single users; this allows for permissions to be linked with categories. Categories can be defined on the basis of roles and resources as well as attributes and geographical constraints so that permissions can change when a user attribute or geographical location changes without intervention from an administrator. This, we noted, could prove decidedly beneficial in our case regarding thresholds for data aggregation and inferences, whereupon, through a change in categories, a permission can readily be withdrawn.

In dealing with the complex issue of privacy as it relates to passive data aggregation and selective access to inference-driven attributes, a flexible and autonomous meta-model such as CBAC would likely prove much needed in this area as in the case of privacy protection

against metadata aggregation and inferences, there will need to be ever-evolving access rights dependent on the current state of the system.

Having discussed our interest in this meta-model, we now move to describe the main notions of the Category-Based Meta-Model. In doing so, we note that the meta-model works on interactions between the following sets, as described in [24]:

- A countable set C of categories, where c_0, c_1, ... are used to denote arbitrary category identifiers.
- A countable set P of principals, where p_0, p_1, ... are used to identify principals.
- A countable set A of named atomic actions, where a_0, a_1, ... are used to denote arbitrary action identifiers.
- A countable set R of resource identifiers, where r_0, r_1, ... denote arbitrary resources.
- A countable set S of situational identifiers, where s_0, s_1,... describe possible situations that may occur in the system.
- A countable set E of event identifiers, where e_0, e_1,.. describe possible events that may happen in the system.

Accordingly, in line with pre-requisites for access control involving permissions and authorisation, it is noted that:

- A permission is a pair (a, r) of an action $a \in A$ and a resource, $r \in R$ to reflect an action that may be performed on a given resource.
- An authorisation is a triple (p, a, r) that associates a permission with a principal $p \in P$, reflecting that a principal may perform a given action on a given resource.

Following, the meta-model details the following relations between these sets:

- Principal-category assignment: $PCA \subseteq P \times C$, where $(p, c) \in PCA$ if, and only if, the principal $p \in P$ is assigned to the category $c \in C$.
- Permission-category assignment: $ARCA \subseteq A \times R \times C$, where $(a, r, c) \in ARCA$ if, and only if, the action $a \in A$ on resource $r \in R$ can be performed by the principals assigned to the category $c \in C$.
- Authorisations: $PAR \subseteq P \times A \times R$, where $(p, a, r) \in PAR$ if, and only if, the principal $p \in P$ can perform the action $a \in A$ on the resource $r \in R$.

From this, Barker determines that the set of $par(p, a, r)$ facts that hold with respect to the specification of an access control policy π may be expressed in first order terms, such that:

$$\forall\, p \in P, \forall\, a \in A, \forall\, r \in R\ \exists\, c \in C$$

$$[pca(p, c) \wedge arca(a, r, c) \Rightarrow par(p, a, r)]$$

While a further relationship ρ may be expressed if there exists a relationship, such as inclusion, between categories, where:

$$\forall\, p \in P, \forall\, a \in A, \forall\, r \in R, \exists\, c \in C, \exists\, c' \in C'$$

$$[pca(p, c) \wedge \rho(c, c') \wedge arca(a, r, c') \Rightarrow par(p, a, r)]$$

Although the meta-model does not go on to consider sessions, delegations, denial of permissions or conflict resolution strategies — it is noted that from this initial basis, these notions can naturally be accommodated for [24]. Such accommodation is proven true by contributions in [82], which shall provide ample influence in our research due to the similarities of the problem they have addressed. Nonetheless, we shall first begin by detailing an

example presented in the foundational paper by Barker [24], noting that the rule language in use is expressed in terms of Clark's completion [88], such that the meta-model of access control, $\mathcal{M}$, is based on the axiom below, derived from the first order expression above, such that:

Axiom 1: $par(P, A, R) \leftarrow pca(P, C),\ contains(C, C'),\ arca(A, R, C')$

To aid in further understanding, we present the following example, which considers the discussion of Section 2.1 as part of our background and motivation. Hence, we define the following policy requirements.

EXAMPLE 1.

*Principals are assigned to the **closefriend** category if they are categorised as being **friends** and their current linked list items are greater than 15 (which causes them to be categorised as members of the **activelinks** category)*

Here, we consider the linked lists present in the Solid system and create categories based on these lists. As such, this requirements policy can be defined in terms of the rule language with the domain-specific elements, *closefriends*, *friends* and *activelinks*, in the general class of categories and the binary predicate *links*, which takes a principal and an integer.

$pca(P, closefriend) \leftarrow pca(P, friends),\ pca(P, activelinks).$

$pca(P, activelinks) \leftarrow links(P, X),\ X \geq 15.$

Following, we turn to [82], which is reminiscent of the work in [83], where researchers take a similar approach but effortlessly extend Barker's foundations by the inclusion of denial, delegations and conflict resolution strategies.

In this paper, their similar focus on data collection has led them to consider additional sets that will be necessary in the CBAC meta-model. These added sets focus on the answers that may be given to a request for permission such that they revolve around interactions between undetermined results (UNDET) and prohibitions. In this way, they have described the sets BAR and BARCA as antitheses to sets PAR and ARCA, whereby BAR is the set of prohibitions of users, and BARCA is the set of category prohibitions. They note that these additional relations obey the following axioms [82]:

$a2$: $\forall\, p \in P, \forall\, a \in A, \forall\, r \in R,\ ((\exists\, c \in C, \exists\, c' \in C$

$$[(p, c) \in PCA \wedge c' \subseteq c \wedge (a, r, c') \in BARCA) \Leftrightarrow (p, a, r) \in BAR)]$$

$a3$: $\forall\, p \in P, \forall\, a \in A, \forall\, r \in R$

$$[(p, a, r) \notin PAR \wedge (p, a, r) \notin BAR \Leftrightarrow (p, a, r) \in UNDET]$$

$a4$: $PAR \wedge BAR = \emptyset$

Where it should be noted that in Axiom 3, $a3$, category containment leads to the inheritance of prohibitions from the senior category, in contrast to junior inheritance for permissions. Additionally, we note that the axiom '$a1$' is not included due to it being a repetition of the aforementioned axiom: Axiom 1.

This reasoning led to their extension of CBAC to define a Category-Based Access Control policy that specifies prohibitions in addition to the authorisations, where they have defined [82]:

(CBAC policy) A Category-Based Access Control (CBAC) policy is a tuple:

$$(\mathcal{E}, PCA, ARCA, BARCA, PAR, BAR, UNDET)$$

where $\mathcal{E} = (P, C, A, R, S, \subseteq)$, such that the axioms are satisfied.

By utilising a similar approach to the permissions and prohibitions in this paper, we have found an integral basis that can easily be verified through formal modelling tools. Said verification has been found in the past to be a crucial component of access control systems; here, it has been noted [88] that a significant difficulty concerning access control 'lies in the interpretation of, often complex and sometimes ambiguous, real-world security policies and in their translation in well defined and unambiguous rules enforceable by a computer system'. As a solution, methods of simplifying rules, such as regulations for rule compositions, have been proposed. However, a more tangible approach to bridge the gap between policy and mechanism has been found by utilising a more formal methods approach in the access modelling stage.

As such, with a view to incorporate formal methods as a means to describe the CBAC meta-model, we set the following requirement: *To build a formal model that meets the specifications described in [24] so that we have the basis upon which we can leverage and extend as we move forward* . To this end, we wish to formally describe the CBAC meta-model that we shall be using as a basis for which our latterly defined privacy rules will be implemented atop. Presenting a formal model for this meta-model will encourage ease of understanding and provide clarity — as is the case for many other formally described access control models [98, 99].

Following, in the next section, it is our aim to introduce the objectives behind as well as the various components of the Solid system to allow for a comprehensive view of the platform that CBAC will be layered unto.

3.2 Outline of the Solid System

To discuss the objectives of the Solid system, we must first appreciate the issues faced due to users' lack of control or autonomy over their data. One of the more worrisome consequences of the current online climate results from the persistent data generated as users interact with the web. As discussed, users must contend with trackers and privacy policies that do not favour them. This constant data generation and collection can be seen as a form of surveillance of user activity as they interact in this space. Hence, to constrain the level of surveillance, it is imperative that users are afforded more control of their data.

Though, it easily becomes redundant to discuss user autonomy or control with ownership of the data placed in the hands of large companies, who benefit from monetary gains from analysing user data and consequently will have limited interest in protecting individuals' privacy. This sentiment was realised by researchers who understood the challenges faced by everyday users regarding data privacy and dark practices such as prejudicial advertising [100, 101] or even user manipulation [68]. Therefore, we move to interrogate a proposal that offers users 'true data ownership' by the provision of a decentralised web architecture that gives users control over access to data.

The method Solid employs is based on individuals' ownership of their data through the use of personalised online data stores — or pods. As discussed in Chapter 2, these pods allow users to grant granular access control rights for their data. The arrangement of data in these pods is decided by the user, with capabilities that allow for the grouping of like items together into the same pod. This grouping facilitates granular access, such that an application may only be permitted access to resources in a particular pod.

Given individuals have complete ownership of their data, there was a need for a structure that would allow interactions of data from different parties, for instance, a possibility for a User A to leave a comment on another User B's photo. To facilitate this, Linked

Data [102] was utilised. Linked Data, a term coined by Berners-Lee's team, describes structured data that is interlinked with other data for use through semantic queries. This notion came about as part of the Semantic Web [103] vision for the Internet to become a machine-readable global data space, such that Linked Data would make use of standard Web technology including HTTP, URIs and RDF [104] for information sharing. That said, Linked Data can more simply be described as typed links that enable explicit connections to be made when necessary.

Concerning Solid's approach to data ownership in a social networking application, there may be a need to retain existing links between different users' data. To maintain ownership, a method to connect data in different pods is necessary so that, given a User A's comments on User B's photo. Here, the comment will be stored in one of User A's data pods, whilst the photo is stored in User B's data pod. Solid's implementation of Linked Data list fulfills these through each data resource getting its own HTTP URL with reference to others.

When considering these Linked Data Lists, in the Solid Approach, we see that for a comment, stored at https://usrApod.solid/comments/67843, for a photo, stored at https://usrBpod.solid/photos/cinema254, the comment will link back to the photo URL. This can be described explicitly on the linked list as:

```
https://usrApod.solid/comments/67843
http://www.w3.org/ns/oa#hasTarget
https://usrBpod.solid/photos/cinema254
```

This format makes use of existing link types, from the World Wide Web Consortium (W3C) Ontology 1, which can be reused by several clients. For instance, 'oa' makes use of the link type Web Annotation Ontology [53]. Berners-Lee et al.'s implementation of Linked Data in this manner serves to allow URIs to be names for objects, HTTP URIs for users

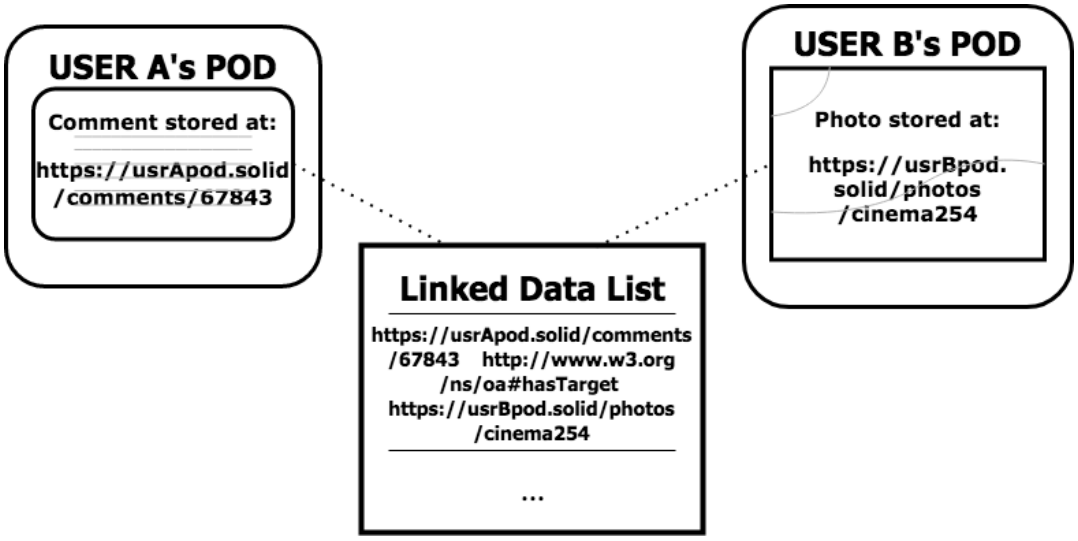


Figure 3.2: Linked Data for Social interactions

to look up these names and inclusion of links at these IURIs to allow further discovery. In its entirety, Solid is an approach to web decentralisation that has been described as 'both simple and extraordinarily powerful' [54].

This interaction via Linked Data can be portrayed by Figure 3.2.

With a view to investigate the system in its entirety, we will now move to describe various components that contribute to the Solid Ecosystem. From this, we may then narrow our focus to the components involved with Resource Management and Access Control, as these will be vital in the discovery of our solution. Thus, we begin our foray into the Solid Ecosystem through a description of the following components through an investigation of the modular specifications, including the 'Protocol' specification and the 'Web Access Control' specification. Hence, we present an examination of the articles that we decided were the core components of the proposal.

Data Pods. As we have previously described, these pods provide a secure space for users to store their information, with the benefit of the pods being decentralised and given the capability of defining fine-grained access control rights.

Solid app. Defined as an application that interfaces with the Solid platform, can be viewed as an agent, along with several other entities, including a client or collection of users etc., such that these are given a WebId or URI for identification purposes.

Uniform Resource Identifier (URI). These URIs serve to identify the different components of the Solid ecosystem, such as resources and container resources.

Resource. A resource may be described as any data held within the pod such that it is the target of an HTTP request, identified by URLs. In line with the fine-grained access control promoted by Solid, resources can have access rights defined specifically on them, though given there are no orphan resources, upon creation, resources inherit access rights from their parent container.

Container Resource. These container resources provide a means of hierarchical collection. Here, these containers are collections of other resources, including other container resources.

Root Container. A root container is the top-level container resource, at the highest level of the collection hierarchy, such that all other resources are contained within, in a hierarchical fashion, i.e. a root container may contain a container resource, which contains a resource.

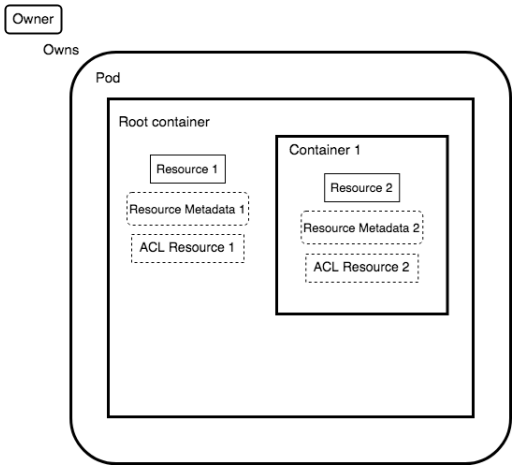


Figure 3.3: Outline of Resources within Pod

Resource metadata Resource metadata describes all the data about given resources, which are stored in the form of RDF (resource description framework) statements.

Owner An owner is taken to be any person or social entity that has ownership rights to the pod, such that they have all the rights and responsibilities on the data in storage.

Figure 3.3 acts as a visualisation of the relationship between these described articles.

Following, we move to outline the following articles of the specification that are more so aligned with the access control aspects of Solid. As such, we have:

Agent. An agent encapsulates any person, social entity or software that interacts with the system, identified by a URI, and thus may have authoritative rights to perform actions, such as read and/or write.

Access Mode. The access mode is used to describe which action is being performed. In this case, the types of access mode refer to the read, write, append and control operations.

Authorisation. As is the case amongst many access control systems, an authorisation describes the access privileges that an agent is permitted on a given resource.

ACL resource. An ACL resource describes any RDF document which details the authorisations of any given agent.

3.3 Summary

We remind ourselves of the goals of this chapter, which was to serve as a means to introduce the key components that will be incorporated in our eventual model to aid in the protection of user privacy in the face of metadata exploitation. To this end, we have been successful in exploring the CBAC meta-model and the likely extensions that will be included in our model. We have also successfully elucidated the workings of the Solid proposal to help facilitate understanding of the presentation of our later model, which shall be presented utilising Z.

Having described these elements with relation to the models that shall inform and compound to our eventual model, we understand that prior to our presentation of these models, we must first consider the privacy requirements that should hold in order to counteract the collection of implicit inferential data that threatens users' privacy.

Chapter 4

Formal Privacy Objectives

Having discussed the scope of privacy as it relates to data, particularly that of a passive nature, we are in a position to describe objectives that we should hope to achieve to combat the issues currently facing users. Therefore, in this chapter, we begin by examining how a taxonomy for privacy can aid in refining our objectives into actionable entities. From this taxonomy, we narrow the focus to include privacy concerns, and further, we hope to draft consequent objectives that are within our scope of passive data collection.

As part of this, we use the resulting condensed taxonomy to define our privacy objectives. Though, as aforementioned, we choose to formalise our privacy objectives so as to glean the resulting benefits that include the likes of clarity and verifiability. As such, in this chapter, we give consideration to the following question: *RQ1: How can we formalise a model for privacy to tackle implicit inferential data from consideration of identifiable privacy harms?*

4.1 Chosen Taxonomy for Privacy

During exploration of privacy through the lens of access control, it becomes evident that we will need a means to dissect and interrogate privacy in a more discrete manner. Here, there is appreciation that a more discrete interpretation is more aligned with the drafting of formal objectives. Therefore, with adherence to the likes of many in the fields of requirement engineering and legal space, favour may be found with the use of taxonomies, which has proven to be a beneficial tool for the consideration of privacy harms, as in [105].

In fact, it has previously been noted that the use of taxonomies for privacy 'in both the legal and requirements engineering communities suggests that both fields have found value in considering privacy as a plurality of discrete elements rather than a single uniform concept' [106]. Said plurality of discrete elements is of particular importance as we move to consider passive data, as the applicability of many privacy evaluations fails to account for this somewhat elusive manner of data extraction and its effects on privacy.

Hence, in the research, a choice of the Solove taxonomy [25] was made due to linkage found from many of the privacy harms, which although not explicitly sighted as resulting from passive data collection, were very much aligned with the harms found in the passive data arena. In its entirety, the Solove taxonomy [25] can be described as a comprehensive, concrete embodiment of privacy, particularly information privacy in the digital age. Further, although the taxonomy was drafted to aid in legal considerations of privacy, its interdisciplinary use is a direct consequence of its broad applicability [105, 107, 108].

Despite this, some (e.g. [109] and [110]) have been critical of the Solove taxonomy for reasons ranging from its failure to provide 'compelling reasons' why the legal system should care [109] to its lack of a rule of recognition for privacy harms [110]. Nonetheless, we followed the path of researchers such as [111], who found great favour in the Solove taxonomy. Whereupon, we, too, saw the benefits and envisioned it would more than

suffice, as while it may be necessary to expose the consequences of privacy violation in a 'very visceral, dramatic way' [109] to aid in understanding the social need for privacy, when translating to actionable privacy requirements for formal modelling, abstraction necessarily rids of these intricacies.

With regards to [110], we agreed that several of Calo's critiques were justifiable, particularly as it relates to the need for a limiting principle to identify privacy harms; though, we decided the privacy taxonomy still presents an instrumental basis upon which meaningful objectives can be derived. Solove's reliance on the law and societal recognition of privacy violations isn't necessarily a negative as it strengthens the argument for inclusion in such a taxonomy. Furthermore, much like privacy is in constant evolution, it is often the case that the law, too, adapts to these changes. Similarly, it should be the case that the taxonomies to describe privacy harms and our model, too, should adapt to accommodate these changes to privacy, a viewpoint likely shared by others who have utilised Solove's taxonomy in their understanding of privacy [107, 112].

In the current climate of Big Data analytics, although metadata was not at the root of these privacy harms expressed by Solove, its applicability is undoubtedly great due to the widespread use and collection of metadata. Metadata, which was once thought of as non-sensitive data, has been proved by many to be a fuel for the Data Economy [68]. Thus, it is fair to reason that discussions about the privacy harms from data processes will also include privacy harms from metadata processes, particularly as this data is more implicit in nature and therefore lacks appropriate regulation [113].

Therefore, in reasoning about privacy objectives for metadata collection, we decided it would be helpful to explore the Solove Privacy Taxonomy [25]. In our consideration, we may expose that the Solove taxonomy conveys four distinct groups of privacy threats: Information Collection, Information Processing, Information Dissemination, and Invasion.

Table 4.1: Privacy Vulnerability List

Vulnerability List	
Privacy Vulnerability	Literature Citations
Surveillance	[114], [115], [116], [117]
Aggregation	[9]
Association	[9], [10]
Identification	[59], [118]
Insecurity	[119], [47]
Secondary Use	[16]
Exclusion	[47], [120], [121]

For the purposes of our goal, Information Collection and Information Processing were identified as the most appropriate areas to focus on; this is due to the capabilities afforded to us through a user data ownership model.

In an arena where users have ownership of their data, they can grant granular access as they see fit, which will ultimately impact data collection; however, while we intend to influence the processing stage through limitations on collection, once the data has left users' possession, they once more have limited control over their data, for that reason we focus on the two areas of the Solove taxonomy that we can meaningfully impact with our privacy objectives and access control.

Consequently, to focus on the vulnerabilities that apply when the focus is metadata, we summate the vulnerability areas — derived from the key areas of Solove Taxonomy, along with relevant citations that relate the areas to issues surrounding metadata collection — in the Table 4.1.

We now shall further explore each of these vulnerability areas.

4.1.1 Surveillance

In his taxonomy, Solove identifies two forms of harm resulting from Information Collection: that of *Surveillance*, as well as *Interrogation*. Given the nature of passive data collection, we argue that the latter — Interrogation — is not applicable to the scope of our concern, i.e. implicitly collated metadata. However, the former is of concern to us.

It is evident that surveillance will form a portion of the privacy vulnerabilities due to the lack of distinction between public and private in the online space. Solove explains that while surveillance in public spaces is deemed by the vast majority as acceptable for the purposes of preventing crime and so forth, surveillance in private spaces is seen to be a significant infringement that may lead to harms such as 'self-censorship and inhibition' [25].

Solove discusses the handling of issues caused by surveillance in the law through cases such as 'Kyllo vs. United States' and considerations of the Fourth Amendment [122]. We can, in addition, readily find several other evaluations from the literature that relate more specifically to metadata, such as [114], [115], [116] and [117]. Schneier [114] explicitly describes metadata as surveillance data, explaining that 'collecting metadata on people means putting them under surveillance'. From this, one may reason that since metadata collection occurs regardless of whether persons are in public or private spaces, surveillance effortlessly breaches private spaces. This point is further epitomised in [117], which gives reference to the mobile, involuntary nature of surveillance in the digital age of 'hyper-surveillance'.

4.1.2 Aggregation and Association

As well as Information Collection, Solove gives consideration to another group of categories: those associated with *Information Processing*. These categories are described as:

(*Aggregation*, *Identification*, *Insecurity*, *Secondary Use* and *Exclusion*), several of which are of interest to us.

The privacy threat of aggregation applies to the metadata space since it forms the basis upon which inferences pertaining to undisclosed attributes are made. Solove argues that aggregation can cause dignitary harms as it 'unsettles expectations' [25]. Here, we see that the scope and power of aggregation disrupts user expectation of limits on known facts; this is as a result of aggregation combining data in 'new, potentially unanticipated ways to reveal facts about a person that are not readily known' [25]. This viewpoint is supported by similar literature concerning metadata aggregates, such as in [123].

Although Solove is not explicit about the association threat, in addition to aggregation from personally created metadata, we must consider individuals as part of a social network; hence, we see the distinction between the threat of association from that of aggregation. In recent years, association has been referenced by others, including Ferra et al. [124], who named the threat as "networked privacy harms". Concerning association, we see that aggregation of data from multiple persons can be damaging for one particular individual; for instance, it was found in [10] that sexual orientation could be inferred from a person's social network. Furthermore, associational information has also been highlighted as 'inherently expressive [and] capable of directly exposing intimate details of an individual's life' [123].

4.1.3 Identification

Solove focuses on identification as the means of linking an individual to their digital persona, which is primarily constructed from records of their online activities. Though a valid privacy concern, this linkage (even if occurring with metadata alone [118]) may well be out of the scope of our focus, as presently, there are sophisticated methods used to fingerprint users as they traverse the web — without the need for previously relied-on cookies [125].

Moreover, many of the mechanisms in place for identifying users are used to facilitate authentication and to ensure the accessor of the site is not partaking in malicious activity such as the deployment of botnets or identity fraud.

Rather than working to diminish the identification linkage, we shall instead seek to limit the privacy exposure individuals are faced with as a result of this linkage. To do so, our focus is the restriction of identifying attributes so as to mitigate reputation and dignitary harms that may arise from linkage to a digital persona that has certain attributes. To this end, we wish to limit the degree to which an individual's identity attributes are exposed by way of their digital persona. By approaching the privacy harm in this manner, we are more in line with Altman's conceptualisation of privacy [59], seeking to provide selective control of access to self attributes and thereby selective control of access to self.

4.1.4 Insecurity

In considering insecurity, Solove addresses the issues arising from inadequate data protection. Although data protection and database security are out of scope, the threats arising from such issues — including identity theft and distortion — clearly crossover into the metadata realm. Viewing insecurity in the manner Solove subsequently describes — in consideration of the court case Board of Education v. Earls [126], as 'the injury of being placed in a weakened state, of being made more vulnerable to a range of future harms' [25] — then we may find insecurity more fitting to issues stemming from metadata aggregation.

Here, we see that aggregation of metadata and the subsequent inferences directly feed into insecurity due to the resulting threats from these, including those of identity theft, stalking, distortion and so forth. This is due to the fact that metadata can be used as a tool to infer travel patterns and user activity, facilitating stalking, and the linkage of data from multiple sources facilitates nefarious verification of identity by malicious parties.

Then, in the face of inadequate protection that has been cited by [119] with regards to lower-end IoT devices, these risks are further exacerbated.

The risk of distortion — the dissemination of false information about an individual — is quite a significant harm with regards to metadata inferences as erroneous inferences can lead to damage to reputation, financial repercussions or even arrests [16]. Research in [47] speaks to the issues of inferential distortion by explaining that individuals currently have no means of correcting false information about them, particularly troubling as this information is used to make a range of decisions about them.

4.1.5 Secondary Use

The relevance of secondary use with regard to metadata inferences is decidedly impactful, as inferences are rarely stated as the intended use for metadata collection. Instead, companies take advantage of privacy policies that are too lengthy or otherwise difficult for the average user to understand [75]. This, paired with dark patterns that make it difficult for users to opt out of privacy-averse settings, propagates the problem of secondary use, particularly as it relates to metadata, where consent is oftentimes an implicit action [71].

Furthermore, metadata collection may inherently have a secondary use problem due to the nature of Big Data and analytics. With the climate of the web as it relates to continuous monitoring, a surveillance-like environment is naturally produced, where the use of connected devices along with their generated metadata inherently facilitates secondary use, with a range of inferences being made possible due to increasing analytical capabilities.

4.1.6 Exclusion

Exclusion is the last of the privacy harms collated in Solove's Information Processing category. In consideration of the literature, we found that this harm seems to crossover into the

metadata realm. Solove explains that exclusion refers to 'the failure to provide individuals with notice and input about their records' [25]. This is directly applicable to the metadata space, as the collection and processing in this context are done without conscious user input, with individuals being powerless in any quest to discover the information companies hold about them — much less to make any amendments.

Although many researchers do not explicitly state this phenomenon as exclusion, they allude to this concept concerning metadata collection and inference (e.g. [120, 121]). In [120], the authors describe that 'consolidated personal data are processed and used for numerous purposes about which [users] typically know nothing, let alone consent to it'; elucidating that exclusion is very much a privacy harm relating to implicitly collated metadata, as metadata forms part of the consolidate data profile for individuals that individuals are largely powerless to find out about. Similarly, in [121], the authors discuss exclusion by way of an absence of transparency, which, they explain, 'means that unreasonable and even simply wrong decisions cannot be detected, and correction, recourse and restitution are all but impossible'.

4.1.7 Summary

With these considerations, we have a basis upon which we might develop our privacy objectives. Although it may be argued that several other of Solove's taxonomy categories have relations to metadata, since we are working on the basis of data ownership and access control, the scope of collection and processing are as far as our research should reasonably extend. To this end, as it applies to processing, we recognise that access control cannot intervene with companies' data processing; however, by making adjustments to methods of collection, we intend to impact the resulting processing that can be achieved with the data so as to prevent unintended inferences from being made at later stages.

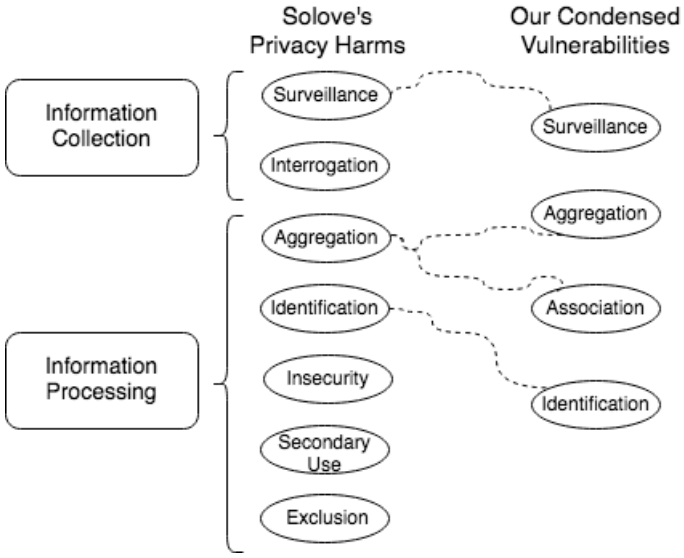


Figure 4.1: Condensed Privacy Vulnerabilities

In summary, then, Figure 4.1 presents our vulnerabilities of focus, derived from Solove's vulnerability taxonomy with an appreciation of the limits of the scope of our research. Here, we note that the latter two sections within the information processing group were not included due to their nature; with them occurring on the collector side, they are very much out of the scope of our research.

4.2 Defining Objectives for Privacy

Following our presentation of the concise set of applicable privacy harms that are within the scope of our consideration of metadata collection and processing, we are now equipped with the necessary prerequisites to begin our consideration of tangible privacy objectives that our model should support. Before we begin our discussion, it is worth first addressing the areas of Solove's taxonomy that, although relevant to the area of metadata, are not included in our condensed list of vulnerabilities: *exclusion*, *insecurity* and *secondary use*.

We shall begin by discussing the necessary exclusions in our condensed taxonomy. Both of these exclusions were in the group of privacy harms related to Information Processing. First, we have the privacy harm of exclusion. Exclusion, while not considered explicitly, will be tackled implicitly as a result of our focus on decentralisation. Moreover, the privacy harm of exclusion helps motivate our decision to build upon the work of the Solid project, as the ability of users to 'own' their data via personalised online data stores (pods) ensures that they have the means to observe the data that they are divulging to companies — and, as such, potentially have greater awareness as to the data these companies are collecting about them, as well as immediate control over the data that is being divulged.

With regards to the second, insecurity and secondary use, we reason that although these may well be relevant to metadata, they are decidedly out of the scope of our research, as we will not be looking at methods to secure databases or control secondary use of data. Once data has left the user's pods, there is very little we can do to mitigate secondary use. It is the responsibility of legislation, guidelines, licensing agreements and the like to both prescribe and proscribe.

We now move to discuss the objectives derived from the areas that are of interest and within scope.

4.2.1 Surveillance

With respect to surveillance, Solove's taxonomy highlights the ostensible need for a separation of public and private spaces. This separation is well established within privacy law. For example, the Fourth Amendment to the US Constitution expresses 'that surveillance in "private" places implicates a reasonable expectation of privacy' [25]. As another example, Article 8 of the UN's Human Rights Convention [11] stresses the 'right to respect for private life'.

Given metadata collection occurs regardless of positioning, the first of our objectives is to delineate the boundary of the private space so that we may explicitly separate the private sphere from the public arena. The use of Solid's decentralisation project gives a foundation upon which we can reason about when and where to share data with companies and, following, can reason about public and private spheres. Therefore, we define our objective as: ensure surveillance does not breach private spaces.

Additionally, as it pertains to inferences, it may prove beneficial to regard the state of covert surveillance in public space as covert surveillance is an integral part of inferences that are made about persons [127]. Whilst surveillance devices implemented by larger government bodies are often well accepted, surveillance through mobile devices and wearables may be seen to infringe on individuals' expectations; hence, we look to reduce the extent of extraneous surveillance (or dataveillance). User data ownership will decidedly impact the arena of dataveillance, as ownership ensures we shall be able to curtail surveillance through limitations of access as the user sees fit. As such, we define our subsequent objective within the space of surveillance as: limit the extent of surveillance possible through dataveillance.

4.2.2 Aggregation

As we have previously alluded to, the value of metadata is decidedly linked to the ability of aggregation to lead to inferences, which explains why metadata, at first glance, is often overlooked as being "harmless". The aggregation of digital footprints users leave behind whilst browsing, or even by simply carrying a connected device can paint a detailed picture of their life. Consequently, we understand that managing the aggregation of metadata is a key component to upholding users' privacy expectations. Furthermore, since we will be utilising Solid's use of pods, we envision that we will be in the position to manage the volume and rate at which data is shared. Thus, in order to combat the resulting issues

that appear with metadata aggregation, we define our objective as: limit the aggregation of passive digital footprints.

In line with this, we appreciate that metadata aggregations can be used to infer personality attributes, as detailed by [13]. To this end, we shall define a second objective in this space that focuses more specifically on protecting against aggregates leading to personality attributes; as such, we define another objective within the consideration of aggregation as: limit the exposure of personal attributes that result from aggregation.

4.2.3 Association

Although Solove does not explicitly divide association from aggregation, we saw it may seem fitting as the causal landscape is quite disparate, where methods of managing the risks from association will likely require a more comprehensive consideration of individuals as part of a network. Much like the problem of averages, where disclosure from other parties in the group risks unintentional disclosure of unwilling members, the exposure that individuals are faced with due to their network of associations requires a different approach than that used with individual aggregates.

Solid's use of Linked Data Lists to enable explicit connections between different users' pods allows easy identification of peers in the network as well as identification of the information — about the user — being shared through associates. All of this gives us a basis from which we may consider leakage of sensitive information through associative metadata. To this end, we describe an objective in this space as: limit the degree of unwanted, unintentional exposure arising from associative connections.

4.2.4 Identification

As we have previously stated, instead of focusing on the linkage of individuals to a digital persona, we instead want to focus on limiting the exposure resulting from such linkage. Hence, we aim to attain a digital persona that does not inherently expose what the individual considers to be sensitive information. We approach this with consideration of the sociologist Altman's conceptualisation [59] of privacy as a restriction of access to self, which we may appropriate in terms of access control and identity in cyber-space so that we look at defining privacy as restriction of access to elements/attributes indicative of self. To this end, we define an objective in the space of identification to be: provide means for individuals to control access to elements of self.

Furthermore, despite debates [128] on what constitutes an identity, it is understood by many (e.g. [129, 130]) that identity can be expressed in a variety of ways dependent on the context. Though, due to the practices of covert surveillance, aggregation and inferences, the online space as it relates to metadata largely lacks said context. To rectify the exposure caused by this, we seek to define a strict digital context for identity. Therefore, we describe a second objective in the space of identification to be: provide a contextual identity for the online space.

4.2.5 Summary

To summarise, our objectives can be stated thus:

OS1: Ensure surveillance does not breach private spaces

OS2: Limit the extent of surveillance possible through dataveillance

OA3: Limit the aggregation of passive digital footprints

OA4: Limit the exposure of personal attributes that result from aggregation

OC5: Limit the degree of unwanted, unintentional exposure arising from associative connections

OI6: Provide means for individuals to control access to elements of self so they may have a contextual identity for the online space

4.3 A Formal Privacy Model

In an effort to efficiently capture our objectives, we look to formal methods as a means to ensure reproducibility and validation. The applicability of formal methods in the privacy arena has been highlighted by researchers such as [22], who noted that 'formal methods can, and should be applied to privacy'. Following this sentiment, we strive to formally model our privacy requirements through predicate logic, which has previously been used within software engineering for ensuring objectives are translated into a more verifiable form.

It is hoped that by defining privacy in a formal manner, we will be able to formally specify access control policies that should hold in our eventual modelling. We intend that, in our extension of the Category-Based meta-model, these formally defined privacy notions will be interpreted by the addition of situational and event identifiers, as described in the original paper [24], as well as in our previous paper [131], so that we may change access rights with inclusion of prohibitory sets similar to those described in [82]. But first, we move to explicitly describe these privacy notions.

Therefore, we approach our reasoning of privacy by considering the existence of two identity profiles for each individual so that we may have a profile that a user chooses to project to the online space and another profile that may be more indicative of a user's identity which shall be a protected profile. As aforementioned, we take privacy to refer

to a 'limitation on access to self' — or, more specifically — a limitation on access to the protected identity profile. Thus, privacy in our work seeks to protect this latter protected identity profile from being unwillingly or unknowingly exposed through collection of metadata.

As it pertains to these identity profiles and the eventual restriction of access to data objects, it is imperative that we further propagate our objectives through consideration of the linkage of data objects to identity. These linkage practices have led the way for online identity profiles to be built from datafication of personal information [132]; hence, it is our aim that through modelling the identity profiles as a collection of data objects, we may be able to begin reasoning about the formation of metadata-driven identity profiles created by companies.

To motivate this further, we consider the following formalisation:

Actively created data objects are defined as d_{ai}, where i represents an integer 1..n.

Metadata data objects are defined as d_{mi}, where i represents an integer 1...j.

Identifier data objects are defined as q_i, where q_i are data objects that combine a trait with a data object, such as a metadata object.

The set P of data objects is indicative of the projected self,

The set T of data objects is indicative of self,

whilst the set K is the set of "known" or exposed identifier data objects.

Membership of q_i data items, indicates that this data object gives rise to/infers a particular trait.

We further split the set T into two disjoint subsets:

- the set T_u of identifier data objects, that is generated for the user, by considering

metadata inferences

- and the set T_s of identifier data objects that the user explicitly marks as sensitive.

For all these sets, we have that:

$K \subseteq P$

$T_s \cap P = \emptyset$

From this definition and given our intention to limit access to the protected self, we may deduce that: any object q_i of T_s is an object we are seeking to protect for privacy to be upheld.

Given this examination into these necessary articles, we therefore may begin our synthesis of the formal model, utilising Z, through introduction of the following basic type, free types and abbreviations:

$$[User, Company, Active\_Data, Metadata, San\_Metadata, ID, Trait]$$

$$Time ::= Q \mid M \mid B$$

$$Data\_Object ::= active\_data\langle\langle Active\_Data\rangle\rangle \mid metadata\langle\langle Metadata\rangle\rangle \mid smetadata\langle\langle San\_Metadata\rangle\rangle$$

$$Identifying\_Object ::= data\_ob\langle\langle Data\_Object\rangle\rangle \mid input\_data\langle\langle ID\rangle\rangle \mid NULL$$

$$Identifier\_Data == Trait \times Identifying\_Object$$

$$Data ::= data\_object\langle\langle Data\_Object\rangle\rangle \mid id\_data\langle\langle Identifier\_Data\rangle\rangle$$

The basic type definitions begin with an introduction to the types, *User* and *Company*, where *User* reflects the set of all individuals whose data we aim to protect and *Company* represents the set of all companies collecting data from individuals.

We then define the following types:

Active_Data representing active data,

Metadata representing metadata,

San_Metadata representing sanitised metadata,

ID representing input data from the user,

Trait representing a user's trait or personal attribute.

Following this, we introduce the free type *Time*, which takes the values Q, M and B to represent the time periods of 'end', 'mid' and 'beginning' respectively, as may be found in a representation of a countdown.

We define the free types as *Data_Object*, *Identifying_Object* and *Data* each making uses of injective constructors to define themselves using existing types.

Thus, to this end we present the privacy model *PrivModel*, which shall be used with Z style conjunctions to address each of the vulnerability areas that we have previously highlighted.

PrivModel

$Companies : \mathbb{P}_1\ Company$

$Users : \mathbb{P}_1\ User$

$P : \mathbb{P}\ Data$

$T : \mathbb{P}\ Data$

$K : \mathbb{P}\ Data$

$Kc\_Data : Company \leftrightarrow Data$

$T\_s, T\_u : \mathbb{P}\ Data$

$Dataset : \mathbb{P}\ Data$

$\text{ran}(Kc\_Data) \subseteq K$

$\forall\, q : Identifier\_Data \mid id\_data(q) \in T \bullet$

$\quad id\_data(q) \in \text{ran}\ Kc\_Data \Rightarrow id\_data(q) \in P$

$K \subseteq P$

$T \subseteq Dataset$

$P \subseteq T\_u$

$T\_s \cup T\_u = T$

$T\_s \cap T\_u = \emptyset$

This *PrivModel* schema should be regarded as our attempt to tangibly formalise data privacy — in that respects, we have separated data into different sets with differing privacy expectations, so that we have the protected set T for all generated data, that we must evaluate, this set is made as a union of disjoint sets T_u and T_s. The set T_u is the automatic set that generated data will reside unless latterly defined constraints force it into a more secret set, T_s Therefore, the set for the generated most-sensitive data that a

user wants to retain secrecy of, is granted by way of T_s.

Then, as we evaluate this schema further, we see that we are introduced to the power set *Companies* of type $\mathbb{P}_1$ *Company*, as well as the power set *Users* of type *User*, before we observe the familiar, previously defined sets P, T and K, which are the projected, protected and known sets, respectively. In addition, we see the relation Kc_Data, which describes a relation between the types *Company* and *Data* to note that a company, c, has collected a data object, d.

We, then, are reintroduced to the sets: T_s and T_u. Followed by the power set *Datset* of type *Data*, which serves as a means to record all of the generated data objects, which will be particularly important later as we consider supplementary items that are not up for public consumption.

Acting on this schema, we, then, have the following constraints.

- The first constraint reminds us that all data objects in the range of $Kc_D ata$ are in the known set, K.

- The second constraint speaks to concern about an identifier data object, q, such that we have in the case q is in the protected set if q is in the range of Kc_Data then it is the case that q is in P, the projected set.

- Following, we have the earlier outlined set interactions, e.g. the known set is a subset of P, the projected set is a subset of the protected set, etc.

- To conclude, we have the constraint that the protected set is a subset of or equal to the *Dataset*.

Having offered a presentation of our formal model for privacy, with the differing sets that shall allow us to reason about data generation and the differing private and public spheres,

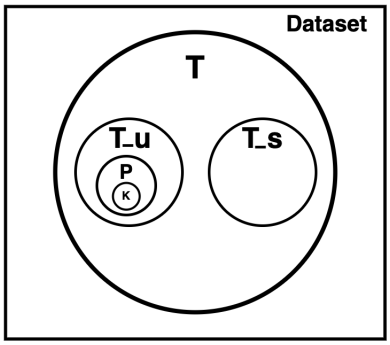


Figure 4.2: Outline of the relation between the sets

we now move to examine the amendments made to handle the privacy vulnerabilities through addressing our defined privacy objectives.

4.3.1 Identification

As has been alluded to, we seek to attain privacy through the limitation of access to self. Further, in order to satisfy the objective **OI6**, we have created the protected identity profile, T, as well as the projected identity profile, P, where objects must first be screened before permitted to be contained in this profile space. To handle these intricacies, we first remind ourselves of the objective in the space of Identification : **OI6 : Provide means for individuals to control access to elements of self so they may have a contextual identity for the online space**.

Then, we move to introduce the following functions that should help us address said objective:

- $E(d_1, q_1)$ indicates a data object d_1 exposes an identifier data object q_1.
- $Sk(d_{m1}, q_1)$ represents that the metadata object d_{m1} positively skews towards inference of an identifier data object q_1.

- $San(d_{m1})$ which describes an arbitrary sanitisation method taking place on the metadata object $d_m i$, in sanitising we would seek to maximise privacy without compromising utility. Although not explicitly described here, sanitisation may take the form of one of the several utility/privacy filters (e.g. [133, 134])

Utilising this, we move to present the following schema, *Identification*

Identification

$$P : \mathbb{P}\, Data$$

$$E : Data\_Object \leftrightarrow Identifier\_Data$$

$$T : \mathbb{P}\, Data$$

$$Dataset : \mathbb{P}\, Data$$

$$T\_s, T\_u : \mathbb{P}\, Data$$

$$Sk : Metadata \leftrightarrow Identifier\_Data$$

$$T \subseteq Dataset$$

$$P \subseteq T$$

$$T\_s \cup T\_u = T$$

$$T\_s \cap T\_u = \emptyset$$

$$T\_s \cap P = \emptyset$$

$$\forall\, d : Data\_Object;\ i : Identifier\_Data \mid i \in T\_s \bullet d \mapsto i \in E \Rightarrow data\_object(d) \notin P$$

$$\forall\, a : Active\_Data;\ i : Identifier\_Data \bullet$$

$$active\_data(a) \mapsto i \in E \Rightarrow data\_object(active\_data(a)) \notin P$$

$$\lor\ id\_data(i) \notin T\_s$$

$$\forall\, i : Identifier\_Data;\ m : Metadata \mid i \in T\_s \bullet$$

$$m \mapsto i \in Sk \Rightarrow data\_object(metadata(m)) \notin P$$

Observing the Identification schema, we move to outline the reasoning behind the additions of the newer constraints.

- First, we begin with the sixth constraint which expresses that for all identifiers marked as sensitive it should not be the case that a data object exposing the sensitive identifier should be in P.

- We then look to the seventh constraint concerning active data, which although not our main area of interest shall be integral in preventing accidental exposure, therefore this constraint works to ensure that in the event an active data object, a, exposes an identifier data object, i — noted through membership in E — then it should indicate that a is not a member of P or that the identifier data object i is not in T_s.

- Following, the eighth constraint serves to ensure that for identifier data marked as sensitive, expressed through expressed through membership of the set T_s, it should not be the case that a metadata object that skews towards inference of the sensitive identifier should be in P.

Finally, with the addition of a sanitisation function, *San*, we are able to conclude the exploration of the previously defined functions necessary to combat the Identification privacy harms:

$$San : Metadata \pfun San\_Metadata$$

Following, we note that the *Identification* schema can be used with the original privacy model *PrivModel* due to the following conjunction.

$$PrivModelWithIdentification \mathrel{\widehat=} Identification \land PrivModel$$

4.3.2 Aggregation

In the space of aggregation we have two objectives, the first, **OA3: to limit the aggregation of passive digital footprints**, whilst the latter is **OA4: to limit the exposure of personal attributes through aggregation**. Although seemingly similar, the first objective is concerned with aggregation as it pertains to surveillance whilst the second is concerned with aggregations leading to inferences.

The necessity of both of these is largely due to the fact that it is likely implausible to accurately predict all possible inferences that can be made from aggregation. Therefore, the former objective, **OA3**, is outlined to prevent undiscovered linkages from harming an individual — thus, with this objective, we seek to limit the overall aggregative data a particular company is able to obtain. While the second of the objectives is concerned with predictable inferences from linkage of passive digital footprints particularly as it relates to personal identifiers and personal attributes.

Thus, the following schema, *Aggregation* has been drafted to handle these specific objectives.

Aggregation

$P : \mathbb{P}\, Data$

$T : \mathbb{P}\, Data$

$K : \mathbb{P}\, Data$

$L : \mathbb{P}\, Company$

$Kc\_id : \mathbb{P}_1\, Data \leftrightarrow Identifier\_Data$

$Kc\_Data : Company \leftrightarrow Data$

$Ac : Data\_Object \nrightarrow Kc\_Id$

$T\_s, T\_u : \mathbb{P}\, Data$

$Dataset : \mathbb{P}\, Data$

$\text{ran}(Kc\_Data) \subseteq K$

$\forall\, q : Identifier\_Data \mid id\_data(q) \in T \bullet$

$\quad id\_data(q) \in \text{ran}\ Kc\_Data \Rightarrow id\_data(q) \in P$

$\forall\, q : Identifier\_Data;\ c\_k : Kc\_id;\ d\_m : Metadata$

$\quad \mid id\_data(q) \in T\_s \land second\ c\_k = q \bullet$

$\quad\quad metadata(d\_m) \mapsto c\_k \in Ac \Rightarrow data\_object(metadata(d\_m)) \notin P$

$\forall\, d : Data;\ c\_k : Kc\_id \mid d \in first\ c\_k \bullet d \in K$

$L \subseteq \text{dom}(Kc\_Data)$

$K \subseteq P$

From this we see a similar set repetitions found in the original privacy model schema, *PrivModel*, and so here we focus on the additions of the following sets, relations and functions.

- First we note the addition of set L, which is a power set of type *Company*.

- We then observe the inclusion of the relation Kc_id expressing a relation between a power set of type *Data* and articles of type *Identifier_Data* to describe that a set of data items is linked to a specific identifier data object.

- Following, the relation *Kc_Data* exposes a relation between the set *Company* and *Data* to acknowledge the data collected by a specific company.

- We then have the partial function Ac between articles of type *Data_Object* and that of the previously defined Kc_Id.

From these, we have the newly defined constraints:

- We have the first constraint expressing that all items, k, in the range of *Kc_Data* are in the set K, which explains that all data collected by companies can be found in the "known" set.

- Then, we have the second constraint, which entails that given an identifier data object, q, is in T_s, if the addition of a metadata object, d_m, in combination with a set of data objects, c_k, skews towards q then this metadata object, d_m, should not be in P, i.e. it should not be shared.

- Next we see a constraint that handles the case that an identifier data object is in T_s and this same object is the second of the order pair c_k of type Kc_id then given there is a metadata object, d_m, that maps to $c\_k \in Ac$ then it implies that the metadata object, d_m, is not in the projected set, P.

- Following the fourth constraint notes that for all data objects and power sets of data, given a data object, d, is in a power set, c_k, then it entails that d is in the set K.

- Finally, the last addition constrains the newly defined set L, such that L is restricted to being a subset of the domain of Kc_Data, expressing that the companies found in L are a subset of companies that have collected data. In fact this set, L, is concerned with the tracking of companies who have reached their aggregative limit.

- While we see the familiar constriction expressing that $K \subseteq P$.

To facilitate the necessary amendments that should occur once this aggregative limit has been reached we, first must consider the conjunction of the *PrivModel* with that of *Aggregration*, which can be described as follows:

$$PrivModelWithAggregation \mathrel{\widehat{=}} Aggregation \land PrivModel$$

With this, we have the following schema, *Limit*, that serves to enact the objective specified as **OA3**.

Limit

$\Delta PrivModelWithAggregation$

$c? : Company$

$m, n : \mathbb{P}\, Data$

$c? \in \mathrm{dom}(Kc\_Data)$

$c? \in L$

$m = \{md : Metadata;\ kc : Kc\_Data \mid$

$\quad data\_object(metadata(md)) = second\ kc \land first\ kc = c? \bullet$

$\quad\quad data\_object(metadata(md))\}$

$n = \{nd : Metadata;\ kc : Kc\_Data \mid$

$\quad data\_object(metadata(nd)) = second\ kc \land first\ kc \neq c? \bullet$

$\quad\quad data\_object(metadata(nd))\}$

$Kc\_Data' = (Kc\_Data \rhd m) \cup (\{c?\} \lhd Kc\_Data)$

$L' = L \setminus \{c?\}$

$P' = (P \setminus m) \cup n$

$T\_s' = (T\_s \cup m) \setminus (m \cap n)$

$T\_u' = (T\_u \setminus m) \cup (m \cap n)$

$K' = (K \setminus m) \cup (m \cap n)$

$Kc\_id' = Kc\_id$

$Dataset' = Dataset$

$Ac' = Ac$

$Users' = Users$

$Companies' = Companies$

This schema takes a company, $c?$ and two power sets, m and n of type $Data$, and performs the following.

- We begin by establishing that the chosen company, $c?$ is in the domain of Kc_Data a check that the company, has collected data.

- We then observe a check to see that the company has reached their aggregative limit $c? \in L$.

- The third constraint involves forming a set of the metadata data objects, m that have been collected by the company, $c?$, which is the data that we wish for the company in question to "forget", therefore a set of all the metadata collected by the company.

- Whilst the fourth constraint involves formation of a set of metadata data objects, n that have been collected by other companies besides $c?$, so that we check for overlap in members of m and n so it can be forgotten by the company in question without being forgotten by other companies that have also collected the data.

- We then remove the company from both Kc_Data and L, to note that the collated metadata has been deleted, which will likely have to be enforced by legislation.

- Followed by the removal of the metadata set, m from the public set, P and then addition of the set n to account for overlapping data.

- We then see the addition of the set, m, to the set T_s.

- In keeping with the disjoint sets T_s and T_u, there is a removal of the set m from T_u.

- Due to the deletion of the collated data performed by $c?$, we state that m should no longer be in the known set apart from the data items found that intersect with those in set n.

- Finally, we then see a maintenance of states in the succeeding sets: *Kc_id*, *Dataset*, *Ac*, *Users* and *Companies*.

4.3.3 Association

With regards to the objective **OC5** in the space of association, we reason about associative exposure from the vantage of the collective associates as well as the individual associate. In order to do this, we look to the case of unintentional exposure from an individual associate through consideration of sensitive data objects within a particular time period.

Furthermore, as has been studied in the past [135] we assume associates have a vested interest in protecting each other's privacy therefore a linked data object marked as sensitive by one would lead to an inability for either party to share without consolidation.
Such that we have the following schema:

Association

$Users : \mathbb{P}_1\ User$

$T\_s, T\_u : \mathbb{P}\ Data$

$T : \mathbb{P}\ Data$

$P : \mathbb{P}\ Data$

$A : Data\_Object \leftrightarrow \mathbb{P}\ User$

$EA : \mathbb{P}(Metadata \times Metadata)$

$Sk : Metadata \leftrightarrow Identifier\_Data$

$Tr : \mathbb{F}\ Metadata$

$\forall\, u1, u2 : Users;\ m : Metadata \mid u1 \neq u2 \bullet$

$metadata(m) \mapsto \{u1, u2\} \in A \land m \in Tr \Rightarrow$

$data\_object(metadata(m)) \in T\_s$

$\forall\, m1, m2 : Metadata;\ u1, u2 : Users \mid u1 \neq u2 \bullet$

$m1 \mapsto m2 \in EA \land metadata(m1) \mapsto \{u1, u2\} \in A \land$

$data\_object(metadata(m1)) \in T\_s \Rightarrow$

$data\_object(metadata(m2)) \in T\_s$

$\#Users > 1$

This schema sees the introduction of a set of users, aptly named *Users*.

We also see the introduction once more of the two sets T_s and T_u as well as the set T and P.

We then have the addition of the set A, which is a relation between articles of type *Data_Object* and power set of *User*.

The set EA describes a power set of a relation between two metadata articles.

We then see the relation Sk that exists between a metadata article hand an article of type *Identifier_data*.

Finally, the set Tr is a finite set of type *Metadata*.

With regards to the objective **OC5** in the space of association, we reason about associative exposure from the vantage of the collective associates as well as the individual associate. In order to do so, we have the introduction of the following constraints acting on these previously defined sets and relations:

- The first constraint specifies that for any pair of users where there exists a metadata data object that associates the two, e.g. record of a call, noted by the metadata object m being linked to the two users, u_1, u_2, in A, i.e $(m, u_1, u_2) \in A$ as well as it holding true that this metadata object, m, is in the set Tr then the metadata data object, m, is restricted to membership in T_s for the user u_2 where the metadata object is in user u_1's T_s which owes to its inclusion in Tr. The aim of this constraint is to protect the metadata article via temporal means, such that the linked data object is protected for a time period, i.e is kept in T_s, this is in anticipation of issues that may arise from activities from u_2 that may inadvertently leak information pertaining to u_1.

- We then have the second constraint that seeks to keep metadata protected from second-hand accidental exposure so that for any pair of metadata objects and pair of users, given that, for the sake of first order predicate logic, u_1 is not equal to u_2 and that the two metadata objects are linked in the set EA and also that the metadata object is linked to both of the users in the set A and finally that the metadata object, m_1, is in T_s then the secondary metadata data object, m_2, should be restricted to T_s.

- We see the association schema takes into account multiple users and hence, has a restriction that the number of users in set *Users* is greater than 1.

4.3.4 Surveillance

In the space of surveillance, we have two main objectives **OS1** and **OS2**. To reason about surveillance as it pertains to these defined objectives, we introduce the following additional set of basics:

$$Mode ::= On \mid Off$$
$$Env ::= Priv \mid Public$$

The type *Mode* serves as a means to decipher whether sanitisation mode is on or off, to allow for sanitised data to be collected when it is necessary for the user to interact with services that require monitoring, such as an interactive pedometer. Whereas the type *Priv* describes whether the user is in a private or public setting, so that environment specific amendments may be made to access rights and consequent data collection. From this, we then explore the following schema:

Surveillance

$Dataset : \mathbb{P}\, Data$

$T\_s : \mathbb{P}\, Data$

$t : Time$

$Ti : \mathbb{F}\, Metadata$

$\forall\, d\_m : Metadata;\ e : Env;\ m : Mode \mid data\_object(metadata(d\_m)) \in Dataset \land$

$e = Priv \bullet$

$m = Off \Rightarrow$

$data\_object(metadata(d\_m)) \in T\_s$

$\forall\, d\_m : Metadata;\ d\_s : San\_Metadata;\ e : Env;\ m : Mode \mid e = Priv$

$\land\ d\_s = San(d\_m) \bullet$

$m = On \Rightarrow$

$(data\_object(metadata(d\_m)) \in T\_s \land data\_object(smetadata(d\_s)) \notin T\_s)$

$\forall\, d\_m;\ Metadata;\ e : Env;\ m : Mode \mid e = Priv \land m = On \bullet$

$d\_m \in Ti \Rightarrow$

$d\_m \in T_s$

Here, we introduce the set *Dataset* as a power set of data, in addition to the set Ti as a finite set of *Metadata*

We then have the following two constraints that have been introduced to handle the delineation of private spaces:

The first constraint expresses that for any metadata object where the metadata object, m is in the set *Dataset* and the environment, given by the type *Env*, is set to *Priv* then if the sanitisation mode, given by the type *mode* is *Off* , then the metadata data object is

in T_s.

Whereas the second constraint is concerned with the case when sanitisation is set to on, which allows sanitised metadata to be collected while the user is in the home environment, useful for example when utilising interactive fitness apps. And so, we have:

For any metadata object where the metadata object, m is in the set *Dataset* and the environment, given by the type *Env*, is set to *Priv* then if the sanitisation mode, given by the type *mode*, is *On* then the metadata data object is in T_s

Following, we observed the conjunction that

$$PrivModelWithSurveillance \mathrel{\widehat=} Surveillance \land PrivModel$$

4.4 Summary

The work presented in this chapter, attempted to solve the subsidiary research question, given by **RQ1**, which can be described as follows:

How can we formalise a model for privacy to tackle implicit inferential data from consideration of identifiable privacy harms?

In the resolution of such a question, we found we were successfully able to craft a privacy model to meet the requirements set out by our eventual condensed privacy objectives. These privacy objectives as has been demonstrated have been formed from an evaluation of the Solove taxonomy and an understanding of the applicability to the passive data arena and the scope informed by means of decentralised user-owned data storages. Therefore through the construction of the *PrivModel* schema and the accompanying schemas and conjunctions in the Z schema language, we were able to perform testing in the form of type checking and model checking, via the animator, which returned a successful run as no errors or violations were found.

Our consideration of this privacy model that attempts to meet the privacy requirements to be implemented in our eventual system leads us to consider the other components of our eventual model, which will include the likes of decentralisation provided by the Solid proposal as well as the CBAC model where the flexible framework will be extended to include these requirements.

Chapter 5

An Exploration of the Solid Ecosystem

In this chapter, we delve into the depths of the Solid Proposal through discussions of the mechanisms for which user data ownership is granted. Therefore, this chapter serves to answer the second of our subsidiary research questions, which is defined as follows: **RQ2: How can a formal model of the Solid approach aid in reasoning about privacy in a user-focused data ownership system?**

As part of this, we look to describe the various components of the system relating to resource management and access control, paying particular attention to the components of the proposal that allow for an enhancement of the level of privacy that may be achieved. Before, we move once more to present models for the system, making use of the Z schema language to provide a mathematically based, easily translatable model. We have separated the formal model into two distinct parts to allow for a comprehensive view of the workings of resource management within the system and then that of the access control mechanisms in use. We note, the latter of which shall be found in Appendix C. Although, here we

set the following requirement for our work on the Solid proposal: *To expose the resource management workings of the Solid proposal through use of formal modelling.* Following, we now move to describe the Solid system as a whole, including the motivations that drove the aforementioned project.

Having introduced the various components relating to resource management, as well as access control, in Chapter 3, which delivered a look into the preliminary aspects of this proposal. Thereby, in the following sections, we move to present our formal model for the resource management portion of Solid to have the structure upon which we may frame our privacy-enhanced access control model.

5.1 A Formal Model for Resource Management within Solid

To begin, we shall be defining a core model that will be used for both the Solid Resources and the Solid Access Control – though the schema will be adjusted into two sub-schemas to individually handle the two facets: Solid Resources and Solid Access Control. Referring, first, to the Solid Resources, we see that Solid has chosen to manage the data and information generated by users in terms of various resources, to create hierarchical links between resources and to enable grouping by relevance or subject matter. Within the consideration of the Solid Protocol's Resources, we handle the following definitions and axioms, which handle the intricate interactions between different sets and their types.

Therefore, we begin by defining the following seven basic types:

$$[owner, pod, container\_resource, resource, resource\_metadata,$$
$$root\_container, uri, acl\_rsrc]$$

As previously discussed, within Chapter 2, the Solid protocol handles data from the

vantage of decentralised ownership, with users, which we may describe as owners, able to store their data in differing personalised online data stores, or pods. This feature of granular data stores grants users a decided privacy advantage as data in the pods can be grouped based on sensitivities and thus have restricted or refined access rights defined on them, as seen fit.

Following the introduction of owners and pods, we see the inclusion of both *owner* and *pod* in the list of types. Subsequently, the type *owner* and *pod* represent, respectively, the previously defined *Owner* and *Data Pod* . As it relates to the differing resources, we include the types of various resources that may be stored in the pods, such that we denote the following types: *container_resource*, *resource_metadata*, *resource* and *root_container*.

The type *container_resource* may be defined as a container that has the ability to hold other resources, including those of type *container_resource*, as aforementioned. Furthermore, resources of the type *container_resource* may be linked with an *acl_rsrc* so that it has the ability to define access control rights for the resources contained within. This is in line with Solid's view of hierarchical access control bestowment, where access rights are inherited from the parent resource. Hence, if the container does not have access control rights defined, then the container above, be it the root container or another container that it is held within, will define the access rights for its constituents.

In line with this, we have the type *root_container*, representing *Root Containers* , that are assigned to each pod as the overarching container that initially contains every defined resource, or these resources may be created in more specific containers, which are still contained by this top-level container. Whereas, the type *resource* is used to describe the information or data that owners choose to store, along with the type *resource_metadata*, which contains data about a given *resource*, described in Solid, using RDF statements.

Finally, we have the inclusion of the type *uri*, to be used for the definitions of URIs

due to Solid's use of URIs as a means to identify the various resources as well as agents (i.e. a person, Solid application or social entity).

From this, we move to define the type *container*, which will be integral when considering relationships such as ownership and inheritance between resources and containers, particularly as we begin to reflect on containment. Since, as we mentioned, those of the type *container* will contain, or hold, all the resources such that there will be no resource existing outside of a container. This encompassing type of *container* will include both types of containers: namely, the *container_resource* as well as the *root_container*, such that we define:

$$container ::= con\langle\langle container\_resource\rangle\rangle \mid root\_con\langle\langle root\_container\rangle\rangle$$

As aforementioned, these containers facilitate the inheritance of access rights such that the access control resource associated with a given container will also be associated with the resources held within the container, except in the case that it is another container or the owner chooses to specify an access control resource on a given resource. A portrayal of this can be found in Figure 5.1.

In line with this, we move to wholly consider all the differing types of resources, not limited to the priorly defined type *resource*, which can be viewed as the resources created intentionally by the owner of the pod. In the Solid ecosystem, we may view many components of the system as resources such that we define the overarching type *Resource* as follows:

$$\begin{aligned} Resource ::= \; & rsrc\langle\langle resource\rangle\rangle \mid rsrcmeta\langle\langle resource\_metadata\rangle\rangle \\ & \mid cresource\langle\langle container\rangle\rangle \mid aclrsrc\langle\langle acl\_rsrc\rangle\rangle \end{aligned}$$

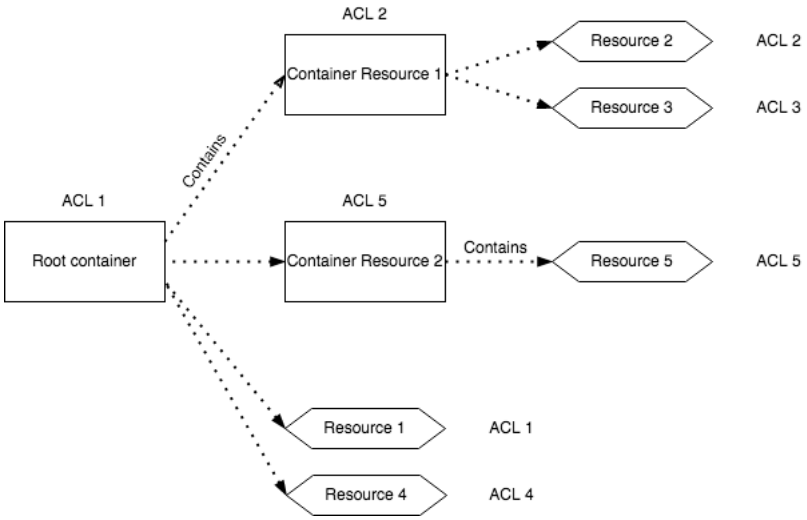


Figure 5.1: Outline of Resource Containment within Pod

As can be gleaned from this definition, we see that the Resource definition encompasses the formerly defined containers of type *container_resource* and *root_container* as well as the formerly defined *resource*, *resource_metadata* and *acl_rsrc*; the grouping of which will be instrumental as we begin to reason about relationships between different components as it relates to pod containment (i.e. a member of *Resource* is found in a particular pod).

Given this encapsulating definition of *Resource* and its value in interactions and comparisons, we define the following two functions, which serve as a means to promote resources of type *resource* or *resource_metadata* to that of type *Resource*. Thus, we have:

$$returnRsrc : resource \pfun Resource$$

$$\forall\, r : Resource;\ rd : resource \mid r = rsrc(rd) \bullet returnRsrc\,(rd) = r$$

This axiomatic definition describes the function *returnRsrc*, which encompasses the transformation of a resource of type *resource* being returned as a resource of type *Resource*

via the constructor function *rsrc*.

Similarly, we have:

$$returnRsrcMeta : resource\_metadata \pfun Resource$$

$$\forall\, r : Resource;\ rd : resource\_metadata \mid r = rsrcmeta\,(rd) \bullet$$
$$returnRsrcMeta\,(rd) = r$$

Here, the axiomatic definition describes the function *returnRsrcMeta*, which takes a resource of type *resource_metadata* and returns that of type *Resource*, utilising the appropriate function *rsrcmeta*.

5.1.1 The Solid Resource Management schema

Following the definitions of these various components, we now present the schema that we have written to handle the interactions between the various Solid resources, as such is aptly named *Solid_Resources*.

Solid_Resources

$Owners : \mathbb{F}_1\ owner$

$Resourceset : \mathbb{P}\ Resource$

$Dataset : \mathbb{P}\ Resource$

$Owned\_By : Resource \pfun owner$

$Pod\_owner : pod \pfun owner$

$Data\_pod : Resource \pfun pod$

$Contains : container \rel Resource$

$Nest : container \rel container$

$Meta : resource \pinj resource\_metadata$

$Ident : Resource \pinj uri$

$ACL\_Link : acl\_rsrc \pfun Resource$

$inheritACL : \mathbb{F}\ Resource$

$Dataset \subseteq Resourceset$

$InheritACL \subseteq Resourceset$

$\mathrm{ran}\ ACL\_Link \subseteq Resourceset$

$\mathrm{ran}\ Owned\_By \subseteq \mathrm{ran}\ Pod\_owner$

$\mathrm{dom}\ Owned\_By \subseteq Resourceset$

$\mathrm{ran}\ Pod\_owner \subseteq Owners$

$\mathrm{ran}\ Data\_pod \subseteq \mathrm{dom}\ Pod\_owner$

$\mathrm{dom}\ Data\_pod \subseteq \mathrm{dom}\ Owned\_By$

$\mathrm{ran}\ Contains \subseteq Resourceset$

$\forall\, r : resource \bullet rsrc(r) \in \mathrm{dom}\ Data\_pod \Rightarrow rsrc(r) \in Dataset$

$\forall\, c : Contains;\ r : root\_container \bullet cresource(root\_con(r)) \neq second\ c$

$\vdots$

Upon observation, we may find that the *Solid_Resource* schema provides us with four sets: *Owner*, which describes a finite set of type owner, in addition to the sets: *Resourceset* and *Dataset*, which are both finite sets of *Resources*. Then, we have *InheritACL*, which is a finite set of *Resources*, which, although related to the Access Control Workings, proves imperative as we describe inheritance between containers and resources. Accompanying these sets, we have also defined the following seven relations: *Owns*, *Pod_owner*, *Data_pod*, *Contains*, *ACL_Link*, *Meta* and *Ident*.

- *Pod_owner* relays ordered pairs (o, p), where inclusion is based on an owner o owning a pod p.

- The more general relation, *Owns* defines an ordered pair (o, r), which describes that an owner o owns a Resource r.

- The relation *Data_pod* involves the relation (p, r) between a pod p and a resource r describing the pod p contains the Resource r.

- Contains is the relation that occurs between a container c and a Resource r, so that (c, r) describes that a container c contains a given Resource r.

- Nest describes an irreflexive, transitive relation that occurs between two containers, c_1, c_2, where a mapping between them suggests containment, i.e. $c_1 \mapsto c_2$ relays that c_1 contains c_2.

- The relation *Meta* describes a relationship between a resource, *resource*, and its associated metadata, or *resource_metadata*.

- *Ident* portrays a relation (r, i) between a resource, r, of type *Resource* and the uri, i, used in its identification.

- The relation *ACL_Link* specifies the access control resource, a, that is linked to the resource, r of type *Resource*.

On these sets and relations, we define the following eleven constraints:

- The first constraint serves to constrict the set *Dataset* to being a subset of the set *Resourceset*.

- The second constraint describes that the domain of *Owns* is restricted to the set of *Owners*, which means a potential owner who is not part of a given instance cannot be entered in the domain of *Owns*.

- The third constraint constricts the range of *Owns* to be a subset of *Resourceset*.

- Next, The fourth constraint details that the domain of *Data_pod* is a subset of the range of *Pod_owner*, such that any pod p that is in the domain of *Data_pod* must be in the range of *Pod_owner*.

- The fifth constraint constricts the range of *Data_pod* to a subset of *Resourceset*, so that every resource r found in the range of *Data_pod* must be contained in the set *Resourceset*.

- The sixth constraint constricts the range of *Data_pod* to a subset of *Resourceset*, entailing that every resource r in a pod, must be a member of the set Resourceset.

- The seventh constraint is a constraint detailing that if the second of the pair, of owner to resource, is the same for two members of the set, since a single resource cannot have dual ownership, then it stands to reason that the owner is the same and hence it is the same member of the set.

- Following, we have the eighth constraint, which ensures all resources in the set *Resourceset* are also contained in the more specific subset of the set *Resourceset*, to account for user-generated data.

- The ninth constraint specifies that a *root_container* cannot be in the range of *Contains*, which stands to ensure the root container is not contained by any other container resources.

- The tenth constraint serves to ensure that all generated resources, r, of type *resource* where the associated resource, $rsrc(r)$ of type *Resource* is in the domain of *Data_pod*, then it implies that the resource $rsrc(r)$ can be found in the set *Dataset*.

- Finally, the eleventh constraint describes the invariant expressing that containers cannot contain root containers, which is directly equivalent to the root container not being in the range of *Contains*.

Following, we look to define the *Init* schema, which can be viewed as the initial state of the system. The formalisation of Z works on the idea of a "before" state as well as an "after" state. By defining the schema *Init*, we are describing the 'after' state of the system — a state in which components have been identified and, thus, operations may be performed on the system.

Init

$Solid\_Resources$

$o? : \mathbb{F}_1\ owner$

$Owners = o?$

$Dataset = \emptyset$

$Resourceset = \emptyset$

$Owns = \emptyset$

$Pod\_owner = \emptyset$

$Data\_pod = \emptyset$

$Contains = \emptyset$

$Nest = \emptyset$

$Ident = \emptyset$

$Meta = \emptyset$

$inheritACL = \emptyset$

$ACL\_Link = \emptyset$

We may view that there is an exclusion of the set *Owners*, which stands to ensure the set of owners is deterministic, and so can be variably chosen with each initialisation of the system.

5.2 Operations on Pods

Following this schema definition, we may begin to explore the various schemas that are involved with the interactions and actions of resources as well as the subsequent management of these resources. First, we begin by considering resource creation, which, as we

have outlined, must take place in the owner's pod. Therefore, we look to schemas involved with operations on Pods. The first of which is a modular schema, *Assign*, that shall also be utilised once more when considering the Access Control workings within the Solid system. Yet here, its use will be instrumental in the delegations of pods.

Assign

$\Delta Solid\_Resources$

$Owners' = Owners$

$Dataset' = Dataset$

$Contains' = Contains$

$Nest' = Nest$

$Meta' = Meta$

$Ident' = Ident$

$InheritACL' = InheritACL$

Here, the *Assign* schema allows for stagnant modelling so that schemas that incorporate the *Assign* schema in their definition may keep these entities described within constant.

Subsequently, the following *AssignPods* schema looks to assign pods to owners; in doing so, it ensures that there is no accidental joint ownership as well as assigning a root container, which, as outlined earlier, is the top-level container that houses all resources and containers belonging to the pod.

AssignPods

$\Delta Solid\_Resources$

$Assign$

$u? : owner$

$p? : pod$

$rc? : root\_container$

$a : acl\_rsrc$

$u : url$

$a \notin \text{dom } ACL\_Link$

$u \notin \text{ran } Ident$

$cresource(root\_con(rc?)) \notin \text{dom } Owned\_By$

$u? \in Owners$

$p? \notin \text{dom } Pod\_owner$

$Ident' = Ident \cup \{cresource(root\_con(rc?)) \mapsto u\}$

$Data\_pod' = Data\_pod \cup \{cresource(root\_con(rc?)) \mapsto p?\}$

$Pod\_owner' = Pod\_owner \cup \{p? \mapsto u?\}$

$Owned\_By' = Owned\_By \cup \{cresource(root\_con(rc?)) \mapsto u?\}$

$Resourceset' = Resourceset \cup \{cresource(root\_con(rc?))\}$

$ACL\_Link' = ACL\_Link \cup \{a \mapsto cresource(root\_con(rc?))\}$

Here, we note that the schema performs the following checks:

- A check to ensure the acl resource, a, is not in the domain of ACL_Link to uphold uniqueness.

- A check to ensure the uri u is not already in the range of $Ident$, serving as a means

to uphold adequate identification so that no two resources are given the same URI for identification.

- A check ensuring there is not a re-use of a root container by checking if the given root container $rc?$ is in the domain of $Owned\_By$.

- A check to ensure the owner $u?$ is in the set of Owners, as this set is decided upon initialisation, hence is subject to change.

- A check to ensure that the pod $p?$ has not already been assigned to an owner.

Then, for the sake of theorem provers we have the following schema to ensure that there are all local variables are assigned:

$$VerPodAssign \mathrel{\widehat=} \exists\, a : acl\_rsrc;\ u : url \bullet AssignPods$$

Finally, the assignment of the pod is rendered complete with the addition of the pod and owner relation to the set $Owned\_By$.

5.3 Operations for Resources

Following, we note that in consideration of data storage, as it relates to resource management, we must focus on resources that will be housed in these pods. First, we look to the type *resource* denoting the resources intentionally created by a user, though within the confines of resource management, as it relates to containment and inheritance, we must also consider the creation of resources such as container resources of type *container_resource*. To this end, we may make use of a modular schema *GenResource* which is once again used to retain the states of entities not involved in the operations for resources.

GenResource

$\Delta Solid\_Resources$

$Owners' = Owners$

$Pod\_owner' = Pod\_owner$

Following, we focus on the creation of resources of type *resource*, which can be achieved via the following schema, *CreateResource*

CreateResource

$\Delta Solid\_Resources$

$GenResource$

$o? : owner$

$r? : resource$

$m? : resource\_metadata$

$a : acl\_rsrc$

$c? : container$

$u : url$

$p? : pod$

$u \notin \text{ran } Ident$

$a \mapsto cresource(c?) \in ACL\_Link$

$p? \mapsto o? \in Pod\_owner$

$cresource(c?) \mapsto o? \in Owned\_By$

$rsrc(r?) \notin Dataset$

$Ident' = Ident \cup \{rsrc(r?) \mapsto u\}$

$Dataset' = Dataset \cup \{rsrc(r?), rsrcmeta(m?)\}$

$Resourceset' = Resourceset \cup \{rsrc(r?), rsrcmeta(m?)\}$

$Owned\_By' = Owned\_By \cup \{rsrc(r?) \mapsto o?\} \cup \{rsrcmeta(m?) \mapsto o?\}$

$Data\_pod' = Data\_pod \cup \{rsrc(r?) \mapsto p?\} \cup \{rsrcmeta(m?) \mapsto p?\}$

$inheritACL' = inheritACL \cup \{rsrc(r?)\}$

$Contains' = (Contains \cup \{(c?) \mapsto rsrc(r?)\}) \cup \{(c?) \mapsto rsrcmeta(m?)\}$

$Meta' = Meta \cup \{r? \mapsto m?\}$

$\vdots$

As part of this resource creation schema, we observe that we have similar checks where we can see a check to ensure the URI is not already being assigned to another resource, in addition to a series of checks related to the creation of the new resource, such as a check to ensure the resource does not already exist in the set *Dataset*.

We also see that in addition to the creation of this new resource, we also have the creation of a corresponding metadata item for the resource, which is then also added to the set, *Dataset* and to the pod via the *Data_pod* relation.

We then have the following schema, which is concerned with the creation of a container of type *container_resource*. Such that with this schema, we are able to propagate the organisation and fine-grained access control proposed by the Solid proposal [20].

New_Container

$GenResource$

$u : owner$

$a? : acl\_rsrc$

$rc : root\_container$

$c? : container\_resource$

$p? : pod$

$a? \notin \text{dom } ACL\_Link$

$p? \mapsto u \in Pod\_owner$

$cresource(root\_con(rc)) \mapsto u \in Owned\_By$

$cresource(con(c?)) \notin Resourceset$

$Ident' = Ident$

$Dataset' = Dataset$

$Resourceset' = Resourceset \cup \{cresource(con(c?))\}$

$Owned\_By' = Owned\_By \cup \{cresource(con(c?)) \mapsto u\}$

$Data\_pod' = Data\_pod \cup \{cresource(con(c?)) \mapsto p?\}$

$ACL\_Link' = ACL\_Link \cup \{a? \mapsto cresource(con(c?))\}$

$inheritACL' = inheritACL$

$Contains' = Contains \cup \{root\_con(rc) \mapsto cresource(con(c?))\}$

$Nest' = Nest \cup \{root\_con(rc) \mapsto con(c?)\}$

$Meta' = Meta$

5.3.1 Operations on Containers

For operations on these containers, we present the modular schema *ManageCon*, which serves to retain elements of the system that are not directly concerned with container management.

ManageCon

$\Delta Solid\_Resources$

$Owners' = Owners$

$Pod\_owner' = Pod\_owner$

$Owned\_By' = Owned\_By$

$Dataset' = Dataset$

$Data\_pod' = Data\_pod$

$Ident' = Ident$

$Meta' = Meta$

$Access\_Rights' = Access\_Rights$

$Authenticated' = Authenticated$

$Ag\_Credentials' = Ag\_Credentials$

$ACL\_Link' = ACL\_Link$

$inheritACL' = inheritACL$

$Resourceset' = Resourceset$

Following, we have a nesting schema that allows containers to contain other containers, making use of the aforementioned modular schema *ManageCon*.

Nest_In_Container

$\Delta Solid\_Resources$

$ManageCon$

$u : owner$

$c?, c_n? : container$

$r : Resource$

$p : pod$

$r = cresource(c_n?)$

$c? \mapsto r \notin Contains$

$(Nest^+) \cap ((Nest^+)^\sim) = \emptyset$

$p \mapsto u \in Pod\_owner$

$r \neq cresource(c?)$

$r \mapsto p \in Data\_pod$

$cresource(c?) \mapsto u \in Owned\_By$

$Resourceset' = Resourceset$

$Contains' = ((Contains \rhd \{r\})) \cup \{c? \mapsto r\}$

$Nest' = (Nest \rhd \{c_n?\}) \cup \{c? \mapsto c_n?\}$

This schema allows containers to contain other containers, given the container is not already nested in the container it wishes to contain, which has been achieved through the inclusion of the condition: $(Nest^+) \cap ((Nest^+)^\sim) = \emptyset$. Here, the intersection of the transitive closure of Nest with its inverse being the empty set serves to help prevent cyclic containment so that an ancestor container cannot be nested in its descendant.

5.4 Summary

In this chapter, we have been successful in our attempts to elucidate the resource management workings in Solid in a manner that is machine-readable as well as transferable, which is in line with the objective set out in Section 5.1. In our efforts, we have also been able to expose the resource management portions of the Solid Ecosystem, which will play an integral role as we move to consider the access control mechanisms granted by Category-Based Access Control, which will be implemented atop this structure.

This integration of Solid and Category-Based Access Control will be compounded, in addition with elements from our privacy model, to grant us an eventual model to help resolve our overarching research question. Furthermore, the inclusion of the Solid proposal lends the basis upon which we may begin reasoning about access control and, therefore, underpins the eventual model that has been constructed.

In closing, regarding our work in crafting a formal model for the Solid proposal, we utilised the ProZ animator. Such that, for type checking, we utilised the ProZ model checker, which returned no errors leading to confidence in the predicate logical basis of our model. Then, we ran the animator to traverse through the differing operations so that we could have confidence that our model was working in the expected. Similarly to the case when type checking, no errors or violations were found in this traversal.

Chapter 6

Modelling the Category-Based Access Control Framework

We now move to begin framing our solution to the problems raised in the previous chapter. As such, we lean into the flexibility afforded by one of the access control mechanisms whose elements shall be incorporated in our overarching solution; namely, that of Category-Based Access Control (CBAC), which presents a flexible framework that can be adapted to suit implementation requirements. Therefore, our investigation in this chapter serves to offer credence to the research question, *RQ3: How can a formal model for the Category-Based Access Control framework help to facilitate a privacy-centric system with dynamic privacy requirements?* Here, we hope to support this question through the provisioning of a formal model for the CBAC framework.

6.1 Constructing a Formal Model for CBAC

In this section, we now move to present a formal model for Category-Based Access Control; it is our intention to provide the meta-model that was incorporated in our paper, [131],

published in the 15th IFIP Summer School on Privacy and Identity Management. As such, we now outline our interpretation of the CBAC meta-model in the axiomatic-based Z [136] schema language. Our choice of the Z notation [93, 136] stemmed from its use in formal specification as a succinct, axiomatic-based tool that can be used to model Access Control Policies for proof checking and uncovering inconsistencies [96, 137].

Accordingly, we were able to describe the meta-model $\mathcal{M}$ [24] in Z, which we present using the schematic language to help promote good specification style. In this particular case, the use of schemas ensures we may retain the flexibility intended by the meta-model through the addition or removal of optional elements such as those of the situational and event identifiers set variety. By presenting the core meta-model, we hope to be able to use this as a foundation for later extension and modelling as it pertains to our problem.

Our interpretation of the model, $\mathcal{M}$, attempts to attain the key components of the original presentation such that any differences may attributed to a preference for strict type checking or are otherwise syntactical. We see that the original paper lends itself to a model consisting of six basic types: C, the set of categories; P, the set of principals/users; A, the set of actions; R, the set of resources; S, the situational identifier set; and E, the event identifier set — where S and E are optional components.

All these types can be introduced by the following declaration:

$$[C, P, A, R, S, E]$$

From these types, as described in the paper [24], we may obtain the required permissions and authorisations sets which were defined as follows:

- A permission is a pair (a, r) of an action $a \in A$ and a resource, $r \in R$
- An authorisation is a triple (p, a, r) that associates a permission with a principal

$p \in P$.

Following, we were able to define these in Z such that *Perm* would be an abbreviation for a Permission, while *Auth* would abbreviate authorisation, such that we have sets of the two relations as:

$$Perm == A \times R$$
$$Auth == P \times Perm$$

It should be noted that as opposed to defining *par* and *arca* as a triple, as it is defined in the original paper, we decide it would suffice to define as a couple, as it would best fit our purposes whilst still upholding the semantic meaning of the declaration. With this consideration in mind, we may continue our extrapolation from the paper [24], which describes the following relations between the four primary sets:

- Principal-category assignment: $PCA \subseteq P \times C$,
- Permision-category assignment: $ARCA \subseteq A \times R \times C$,
- Authorisations: $PAR \subseteq P \times A \times R$

We note that while we have included S and E in the set definition, we did not see a need to include these as a primary set as they work with environmental and situational issues and, as such, are highly dependent on implementation requirements. As we mentioned earlier, these requirements will be included as we later come to appreciate the translation of our privacy requirements to schematic-based language.

Due to our choice of couples in place of triples, we may rewrite, and now present the relations above as:

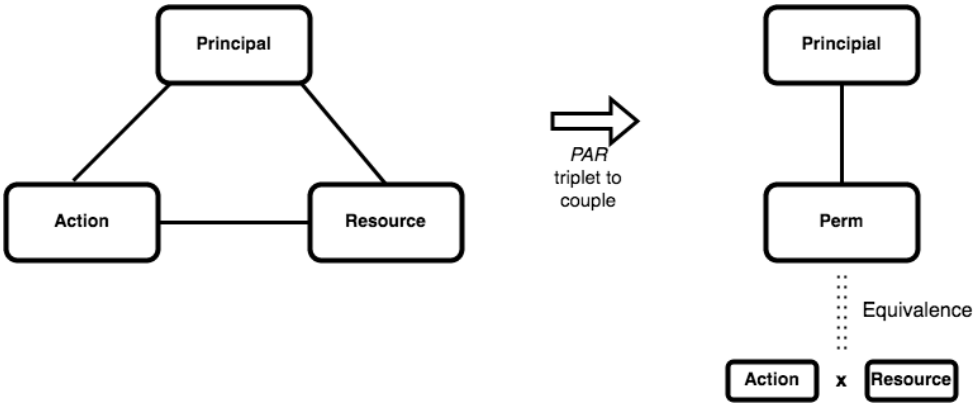


Figure 6.1: Depiction of relation PAR

Relations 1

- *Principal-category assignment:* $PCA \subseteq P \times C$,
- *Permission-category assignment:* $ARCA \subseteq C \times Perm$,
- *Authorisations:* $PAR \subseteq P \times Perm$

In line with this, we present Figure 6.1 as a means to portray the rewriting conversion from the presentation of relation in terms of triples to that of doubles.

From these and the addition of a set of Permissions, we may synthesise the *MModel* schema. Though, in order to perform model checking on our synthesis, the *MModel* schema is accompanied by the schema, Instance, which allows for specification and tracking of the "current" categories, principles and permissions of the system. This inclusion ensures that different instances can have differing category sets, principle sets and permissions sets; hence, we have:

Instance

$permset : \mathbb{F}\ Perm$

$principalset : \mathbb{F}\ P$

$catset : \mathbb{F}\ C$

Following, we now move to define the *MModel* schema, as an interpretation of the model defined in [24]:

MModel

Instance

$par : P \leftrightarrow Perm$

$arca : C \leftrightarrow Perm$

$pca : P \leftrightarrow C$

$\text{dom}\ par \subseteq principalset$

$\text{ran}\ par \subseteq permset$

$\text{dom}\ arca \subseteq catset$

$\text{ran}\ arca \subseteq permset$

$\text{dom}\ pca \subseteq principalset$

$\text{ran}\ pca \subseteq catset$

$par = pca \fatsemi arca$

As a means to describe the *MModel* schema above, we note that the placement of the permset, principalset and catset into the schema, by way of inclusion via the *Instance* schema, allows different instances of the meta-model to operate on different Permissions,

Principles and Categories. Whereas, the sets *par*, *arca* and *pca* denote the sets concerned with Authorisations, Category Permissions Assignment and Principal Category Assignments in the current system.

Regarding the constraints within the*MModel* schema, these are direct derivatives of the rules placed upon the sets defined in the original model. As described, the first six constraints are based on the noted relations between the four primary sets, which have been described above as Relations 1. Whilst the final constraint, '$par = pca \fatsemi arca$', is based on the general case of the axiom placed upon the original model:

Axiom 1: $par(P, A, R) \leftarrow pca(P, C),\ contains(C, C'),\ arca(A, R, C')$

Whereupon, in the general case, the axiom need not include a category containment, such that we have:

$$par(P, A, R) \leftarrow pca(P, C),\ arca(A, R, C)$$

Which is semantically equivalent to:

$$par = pca \fatsemi arca$$

Following, we consider that in our model, we are utilising the Pro-B toolkit, which includes support for Z, and therefore, we utilise an initialisation schema to be associated with the "after" state of the *MModel* schema, as has been described below:

Init

$MModel$

$permset = \emptyset$

$par = \emptyset$

$arca = \emptyset$

$pca = \emptyset$

In order to encourage understanding and to properly interrogate our model, we have chosen to present our model in groupings to enable a description of our model by way of differing topics. Therefore, we have:

Schemas for stagnant modelling

As part of our modelling, it is often the case that we wish to work on some 'entities' of the model while retaining others; hence, for the benefit of reproducibility and thereby to avoid recurrent stagnant assignments, we have the following two schemas. The first of which is concerned with the attributes described in the *Instance* schema such that we may define:

SetupRetained

$\Delta MModel$

$catset' = catset$

$principalset' = principalset$

$permset' = permset$

The *SetupRetained* schema is to be used in operations that do not need to update the set of categories, principles or permissions of the system whilst allowing for updates to be made on the relations *arca*, *par* or *pca*.

Whilst the latter schema, *ModelRetained*, is to be used in operations that may need to update the set of categories, principles or permissions of the system whilst retaining the existing relations between these constants such that *arca*, *par* and *pca* remain stagnant.

ModelRetained

$\Delta MModel$

$par' = par$

$arca' = arca$

$pca' = pca$

6.2 Operations for Permissions

In Category-Based Access Control (CBAC), we recall that permissions are not assigned to individual users but are instead assigned to categories of users. These categories are readily changed, such that permissions can change in an autonomous way, with clearly defined rules to cause changes based on the current system state. In this group of functions, which work with permissions in the system, we are specifically concerned with explicit changes to the set of permissions as opposed to the latter changes made by category or principal assignments.

Thus, we begin by considering, once more, the benefit of schema reproducibility; hence, we make use of the Z model's schemas as declarations so that we define the schema *UpdatePerms*, which we can use in consequent schemas whenever the only change to the *MModel* is occurring with regards to the set of permissions, defined as *permset*.

UpdatePerms

$$ModelRetained$$

$$catset' = catset$$

$$principalset' = principalset$$

In addition to this, we notice that for any given instance of the model, we want to be able to define a starting set of permissions on the set of actions and resources. Hence, we introduce the schema *AllocatePerms*, which will allow us to allocate a chosen set of permissions to a particular instance:

AllocatePerms

$$UpdatePerms$$

$$allocation? : \mathbb{F}_1\ Perm$$

$$permset = \emptyset$$

$$permset' = allocation?$$

That said, to allow for increased flexibility, we have also added the schema *AddPerms* that allows the set of permissions to be augmented, which can be seen below:

AddPerms

$$UpdatePerms$$

$$allocation? : \mathbb{F}_1\ Perm$$

$$permset \neq \emptyset$$

$$permset \cap allocation? = \emptyset$$

$$permset' = permset \cup allocation?$$

Likewise, we have also added the schema *RemovePerms* that allows for permissions to be removed from the set of existing permissions within the system, which is as described:

RemovePerms

$$\Delta MModel$$

$$allocation? : \mathbb{F}_1\ Perm$$

$$permset \neq \emptyset$$

$$allocation? \subseteq permset$$

$$permset' = permset \setminus allocation?$$

$$par' = par \rhd allocation?$$

$$pca' = pca$$

$$arca' = arca \rhd allocation?$$

$$principalset' = principalset$$

$$catset' = catset$$

It should be noted that the schema *RemovePerms* may affect the set of authorisations

as well as the permissions given to particular categories; as such, we can see that the sets *par* and *arca* are modified in addition to the modification of *permset*.

6.3 Operations for Principals

In our system, we make use of a set — *principalset* — to determine the principals in the system; as such, *principalset* is declared as a finite set of P (Principals). With this group of functions, we are now concerned with the operations primarily involving principals; here, we may want to make changes such as adding to or subtracting from the existing principal set and so utilise the following schemas to do such things.

The first of these schemas, *AddPrinc*, can be used to add a principle to the set of current principals, where the current *principalset* has been decided during the initialisation stages. Nonetheless, *AddPrinc* can be realised as:

AddPrinc

$$ModelRetained$$

$$p? : \mathbb{F}_1\ P$$

$$p? \cap principalset = \emptyset$$

$$permset' = permset$$

$$catset' = catset$$

$$principalset' = principalset \cup p?$$

Whilst the following schema, *RemPrinc*, may be used to remove a principal from the *principalset*, this handles a multitude of cases, including removal of a principal who has yet to be allocated to a category, removal of a principal that has previously been assigned

to a category and removal of a principal that has a set of permissions allocated to them, which has been defined as:

RemPrinc

$\Delta MModel$

$p? : P$

$au : \mathbb{F}\ Auth$

$au = \textbf{if}\ p? \in \text{dom}\ par\ \textbf{then}\ \{m : permset \mid p? \mapsto m \in par \bullet (p?, m)\}\ \textbf{else}\ \emptyset$

$pca' = \{p?\} \lhd pca$

$arca' = arca$

$par' = par \setminus au$

$permset' = permset$

$principalset' = principalset \setminus \{p?\}$

$catset' = catset$

As discussed, in the CBAC framework, principles belong to categories, which define the set of permissions that a principal can have. Principals may also belong to multiple categories, which entails that their permissions may come from a multitude of categories; this results in operations on principals often involving the set(s) *pca* and/or *par*.

Similar to the case above for the schema, *UpdatePerms*, we have defined a schema, *UpdatePCA*, which we may use in the operations for principals that are solely concerned with changes to the *pca* set. The schema *UpdatePCA* is described as follows:

UpdatePCA

$$SetupRetained$$

$$par' = par$$

$$arca' = arca$$

Accordingly, in order to assign a principle to a particular category, we consider two cases that we may be faced with. The first case is where a principal is being assigned to a category that does not have any assigned permissions, i.e. for a category c, $c \notin \text{dom}\, arca$, whilst the second case of principal category assignment is concerned with a principal being assigned to a category that already has assigned permissions: i.e. for a category c, $c \in \text{dom}\, arca$. As a result of this difference, we can see that the schema contains an **if** clause on the set of authorisations added to *par*

AllocatePrincCat

$$SetupRetained$$

$$p? : P$$

$$c? : C$$

$$v : \mathbb{F}\, Auth$$

$$v = \textbf{if } c? \in \text{dom } arca \textbf{ then } \{m : arca \mid first\, m = c? \bullet (p?, second\, m)\} \textbf{ else } \emptyset$$

$$p? \mapsto c? \notin pca$$

$$pca' = pca \cup \{p? \mapsto c?\}$$

$$par' = par \cup v$$

$$arca' = arca$$

Following, to handle the opposing case: removal of a principal from a given category, we have defined the following schemas. The first of which handles the simplest case, the removal of a principal from a category that has no permissions assigned to it, such that the only change needed to be made is to the set *pca* with the removal of the principal mapping to the category:

*RemovePrincCat*0

$$
\begin{array}{l}
UpdatePCA \\
c? : C \\
p? : P \\
\hline
p? \mapsto c? \in pca \\
c? \notin \mathrm{dom}(arca) \\
pca' = pca \setminus \{p? \mapsto c?\}
\end{array}
$$

The second of the principal removal schemas handles the removal of a principal from a category, which has permissions assigned to it such that the category c_1, $c_1 \in \mathrm{dom}\, arca$. Expanding upon this case, we see that the removal of principals may result in the need to change the authorisation set, *par*, and in doing so, we must take care to consider removal of a principal, particularly if the authorisations of the principal come from inclusion in multiple categories. In the case of removal of a principle p_1 from a category c_1 with permissions that grant the action a_1 to be performed on resource r_1 where the principal p_1 also belongs to a category c_2 which also includes the permission granting action a_1 on resource r_1, removal from category c_1 should not remove the authorisation of action a_1 on resource r_1 for p_1 in *par* due to p_1's inclusion in category c_2. These considerations have been reflected in the following schema:

*RemovePrincCat*1

$SetupRetained$

$p? : P$

$c? : C$

$d : \mathbb{F}\, C$

$v : \mathbb{P}\, Auth$

$pe : Perm$

$d = \{n : pca \mid first\ n = p? \land second\ n \neq c? \bullet second\ n\}$

$v = \{m : arca \mid first\ m = c? \land second\ m \notin \mathrm{ran}(d \lhd arca) \bullet (p?, second\ m)\}$

$p? \mapsto c? \in pca$

$c? \mapsto pe \in arca$

$par' = \textbf{if}\ (pe \in \mathrm{ran}(d \lhd arca) \land v = \emptyset)\ \textbf{then}\ par\ \textbf{else}\ par \setminus v$

$arca' = arca$

$pca' = pca \setminus \{p? \mapsto c?\}$

6.4 Operations for Categories

Categories are the core concepts in the CBAC meta-model, where each category can be likened to a class of entities that share some property. The set of categories, *catset* enlists the categories that exist in the current system. To this set, similar to the *principalset*, we define functions for modification so that we are able to add and remove categories from the system with the use of two schemas. The first of these, *AddCat*, may be used to add categories to the existing set, which we have defined as:

AddCat

$$ModelRetained$$

$$c? : \mathbb{F}_1\ C$$

$$c? \cap catset = \emptyset$$

$$permset' = permset$$

$$principalset' = principalset$$

$$catset' = catset \cup c?$$

The second of the schemas, *RemCat*, handles removal of categories from the current system. Category removal needs to handle the cases where: the category has principals previously assigned; the category has no assigned principals; and the category has permissions assigned to it and, as such, is found in *arca*. All of these cases can, then, be handled by the following schema:

RemCat

$\Delta MModel$

$c? : C$

$pe : \mathbb{F}\ Perm$

$c? \in catset$

$pe = \textbf{if}\ c? \in \text{dom}\ arca\ \textbf{then}\ \{m : permset \mid c \mapsto m \in arca\}\ \textbf{else}\ \emptyset$

$pca' = pca \rhd \{c?\}$

$arca' = \{c?\} \lhd arca$

$par' = par \rhd pe$

$permset' = permset$

$principalset' = principalset$

$catset' = catset \setminus \{c?\}$

Within a given category, members are given permissions that have been assigned to the category so that operations on categories often involve the sets *arca* and *par*.

With this in mind, we may define a schema, *UpdateCatPerms*, which we may use in operations on categories that are solely concerned with changes to *arca* and/or *par*:

UpdateCatPerms

SetupRetained

$pca' = pca$

Following, we may consider the assignment of permissions to categories, found in the

set *arca*, where we consider how assignment of permissions applies to a category that does not have any assigned principals, i.e. for a category c, where $c \notin \operatorname{ran} pca$. Here, we reason that the only set to be updated is that of *arca* and so $v = \emptyset$. Whereas, we may also consider the handling of category permission assignment when the category already has assigned principals, i.e. for a category c, $c \in \operatorname{ran} pca$. Such that, *arca* is updated as above but also the set of authorisations: *par* must change to reflect an assignment of permissions to principals found in the given category.

*CategoryPerms*0

$$
\begin{array}{l}
UpdateCatPerms \\
c? : C \\
pe? : Perm \\
v : \mathbb{P}\, Auth \\
\hline
v = \{m : pca \mid second\ m = c? \bullet (first\ m, pe?)\} \\
c? \mapsto pe? \notin arca \\
arca' = arca \cup \{c? \mapsto pe?\} \\
par' = par \cup v
\end{array}
$$

The next schema provides a method to remove permissions from a category and, as a result, simultaneously update the authorisation set, *par*. In the first case, we may deal with the removal of permissions from a category that does not have any principals assigned to it, and as a result, the only set to change is that of *arca*. This is in contrast to the case where the category has principals assigned to it. This difference is reflected in the need to update the authorisation set *par*; if these permissions are the sole result of membership of the category in question, i.e. for a category c_1, we have that $\forall\, pe \in arca(c_1) \bullet pe \notin$

$\text{ran}(c_1 \lhd arca)$:

*RemoveCatPerm*0

$$
\begin{array}{l}
\textit{UpdateCatPerms} \\
c? : C \\
pe? : \mathbb{F}\, Perm \\
v : \mathbb{F}\, Auth \\
\hline
v = \textbf{if } c? \in \text{ran}(pca) \\
\quad \textbf{then } \{p : pca;\ n : arca \mid \\
\quad\quad \textit{second}\, p = c? \land \textit{first}\, n = c? \bullet \\
\quad\quad\quad (\textit{first}\, p, \textit{second}\, n)\} \\
\quad \textbf{else } \emptyset \\
pe? \subseteq permset \\
\forall p : pe? \bullet c? \mapsto p \in arca \\
arca' = arca \setminus \{q : pe? \bullet c? \mapsto q\} \\
par' = par \setminus v
\end{array}
$$

The following schema encapsulates an update or 'swap' on the permissions found within a category, such that, for a category c_1 with $c_1 \mapsto perm_1 \in arca$, if, instead, we wish to 'swap' the permission $perm_1$ with $perm_2$ so that, we will have $c_1 \mapsto perm_2 \in arca$, we can define this, for the case for a category with and without principals. In the case for a category without principals, we will see changes to solely *arca*, and for the latter case, we will see changes to the sets: *arca* and *par*:

*UpdateCatPerms*0

$$
\begin{array}{l}
UpdateCatPerms \\
c? : C \\
pe?, qe? : Perm \\
v, d, w : \mathbb{P}\, Auth \\
\hline
v = \{m : pca \mid second\ m = c? \bullet (first\ m, qe?)\} \\
d = \{n : v;\ s : pca;\ t : arca \mid \\
\qquad first\ n = first\ s \land second\ s = first\ t \land second\ s \neq c? \bullet \\
\qquad\quad (first\ n, second\ t)\} \\
w = \{m : pca \mid second\ m = c? \bullet (first\ m, pe?)\} \\
c? \mapsto qe? \in arca \\
c? \mapsto pe? \notin arca \\
arca' = (arca \setminus \{c? \mapsto qe?\}) \cup \{c? \mapsto pe?\} \\
par' = ((par \setminus v) \cup d) \cup w
\end{array}
$$

And the following schema is provided for theorem provers to eliminate free variables:

$$Ver \mathrel{\widehat=} \exists\, d, v, w : \mathbb{P}\, Auth \bullet UpdateCatPerms0$$

Then, in closing, we present the succeeding output operations in order to provide a method to quickly receive feedback concerning the current state of the system, which would boast fruitful results as the state of the system becomes more complex. Therefore, we recognise the need to verify the permissions held by a given principal achieved through the following schema:

PrincipalsToPermissions

$\Xi MModel$

$u? : P$

$P! : \mathbb{P}\, Perm$

$P! = \{p : par \mid first(p) = (u?) \bullet second(p)\}$

Similarly, in the case where we wish to inspect which of the categories have a particular permission, we have the schema:

PermissionsToCategories

$\Xi MModel$

$p? : Perm$

$C! : \mathbb{P}\, C$

$C! = \{a : arca \mid second(a) = (p?) \bullet first(a)\}$

6.5 Summary

In this chapter, we have successfully presented a formal model for the CBAC model as described in the paper [24], which is very much in line with the previously defined requirement for our work in CBAC. To this end, we have presented a formal model for the CBAC framework in the axiomatic-based language of Z, which has previously been used to describe other access control models due to its mathematical basis, which promotes clarity and transferability.

Similar to the earlier cases, we utilised ProZ for type checking. Then, we ran the ProZ

animator to step through random operations to encourage confidence in our model. Here, we found no errors or invariant violations in this animation stage or in type checking.

Chapter 7

A Privacy-Enhanced Solid integrated with Category-Based Access Control Formal Model

By investigating the use of formal models in this particular case, where policy properties are reliant on privacy-preserving notions, we aim to ensure that these notions — as they apply to our problem — can be logically expressed and upheld. Models have also proven extremely useful in guiding development cycles [85, 138], and the precedent research that we have outlined forms the necessary basis for which we may synthesise a model that aims to address the subsidiary research question: *Is it possible to build a formal model that supports our privacy requirements through extension of an integrated Solid and Category-Based Access Control model?*

In order to achieve such an aim, we have drafted a model that acts as an extended amalgamation of the previous models. As part of this, we incorporated basal portions of the Solid proposal with elements of Category-Based Access Control augmented by our privacy

objectives. To this end, we, once again, present our formal model in the mathematically-based predicate logic language of Z.

7.1 Forming a Compounded Model

We have previously discussed the privacy objectives we seek to achieve in our quest to aid in the mitigation of personal information exposure through metadata collection and analysis. In this chapter, we plan to actively incorporate these earlier defined objectives in a decentralised and autonomous manner.

With regard to decentralisation, the Solid proposal has granted the opportunity for a decentralised platform that places data ownership in the control of users. This granting of ownership can be thought to be the first stepping stone in the journey to aid in mitigating the privacy violations resulting from unencumbered data generation and collection. Though, due to the nature of metadata, it was immediately evident that Solid alone would not be equipped to handle issues arising from the later stages of analysis and inferences.

Hence, without the use of a dynamic, autonomous access control framework, any attempts to halt personal information exposure or leakage through metadata will likely be thwarted. With this requirement, we investigated and modelled the Category-Based Access Control Framework that we felt would be able to grant the required access control flexibility and autonomous change.

7.2 Basics of the Model

Through an examination of the various models that shall be compounded, we note that we may extract the following type definitions as the rudimentary elements for this extended formal model. Thus, we begin by considering the original model for the CBAC framework,

where we had the following six basic types:

$$[C, P, A, R, S, E]$$

Whilst these still exist in the extended model, we have broken down the types further to handle the intricacies of our research question. As such, we see the definitions of the following basic types in our extended model.

$$[C, A, \mathit{active\_object}, \mathit{metadata\_object}, \mathit{sanmetadata\_object}, \mathit{user}, \mathit{company}, \mathit{private}, \mathit{public}, \mathit{ID}, \mathit{Trait}, \mathit{link\_descrptn}]$$

From this, it is immediately noticeable that both the set P and the set R are missing, which will be accounted for in the succeeding deliverables. Though we are greeted with the sets, C denoting the categories and A denoting the actions that may be performed, which are reminiscent of the earlier CBAC type definitions. The use of the set S was found redundant due to our incorporation of the logical privacy requirements, which would address different situational changes. In addition to the two types C, A, we have the basic types: *active_object*, to denote data objects that are actively created, i.e. intentional user creation; *metadata_object*, used to denote metadata objects, as well as the types *user* which shall be used to denote the users of the system; and *company*, which is used to describe the collecting companies and aggregators.

In line with these basic definitions, we also preemptively discuss that when there is later a need to have a permission for an action on a resource, we may have the opposing feature to express a forbidding of the action of the resource. In line with this, we define:

$$[F]$$

Then, we have the following function that links each action to a forbidding of the action, so that we have:

$$fbd : A \rightarrowtail F$$

Though we note that the definition of a forbiddance shall come later when resources have been defined, so at present for an action, a_1, i.e. read, we have a forbidding of the action, f_1, i.e. forbid read. So that, in comparison, a forbiddance would be defined on a resource, in a similar way a permission is defined on a resource, i.e. permitted to read a resource compared with forbidden from reading a resource.

In addition to these CBAC-inspired basics, we have also included the basic types of the Solid model that are primarily concerned with resource management since we will be implementing our own access control, utilising elements of CBAC in conjunction with elements of our formal privacy model. Therefore, we begin with an introduction to the following articles of the Solid Resource Management:

$$[owner, pod, container\_resource, root\_container, uri, acl\_rsrc]$$

Here, we are met with the type *owner*, which shall describe the user that we shall focus on, such that this user will have the defined privacy sets and ownership of the pods. We then are met with the type *pod* to represent the pods that will be used as a structure where user-generated data will be stored. Following, we are reintroduced to the type *container_resource*, which shall serve as a means to separate components within the pod such that these containers may have a different set of access control rights assigned to them.

Then, we have the type *root_container*, which shall be used as the master container

within the pod, from which access rights will be inherited upon creation of new resources. Subsequently, we have the type *acl_rsrc*, which denotes the access control resource, the mechanism by which resource access rights are granted in the Solid system.

We then see the grouping of the different container types through the succeeding zed definition. So that we have the type *container*, which may be of type *container_resource* or of type *root_container*; as portrayed below:

$$container ::= con\langle\langle container\_resource\rangle\rangle \mid root\_con\langle\langle root\_container\rangle\rangle$$

Further, we have the following schemas that relate these new Solid additions to the sets found in the CBAC meta-model as part of our amalgamation. Therefore, we have the definitions for the set P and also that for the set R.

For the set P denoting the principals of the system, we have the following definition that exposes the variability in the principal set P.

$$P ::= upr\langle\langle user\rangle\rangle \mid cpr\langle\langle company\rangle\rangle \mid opr\langle\langle owner\rangle\rangle$$

Here, we see that this definition describes that principals can be of type *user*, type, *company* or type, *owner*.

Next, in our attempts to reason about the set R, we must first consider the articles from our privacy model, which looked at identifying data and identifiable data objects, so that we see our first privacy contribution.

First, we have an identifying object, which can be thought of as a data object that can be interpreted to expose identifying personal information; to this end, we have the following type definition of *Identifying_Object*:

$$Identifying\_Object ::= active\_ob\langle\langle active\_object\rangle\rangle \mid m\_ob\langle\langle metadata\_object\rangle\rangle \mid$$
$$input\_data\langle\langle ID\rangle\rangle \mid NULL$$

In this type definition, we see that articles of type *Identifying_Object* can be those that are generated through interaction with the web, such as active objects, *active_object*, or metadata objects, of type *metadata_object*, or an article of type *Identifying_Object* may be of type input data given by *ID* or *NULL*, this serves to allow the owner to explicitly provide input to define an identifying object or define it as *NULL*, so they are able to protect an identity trait without specifying an entry for the trait.

We then observe the type definition of *Identifier_Data*, which is described as a cartesian product between elements of type *Trait* and that of type *Identifying_Object*.

$$Identifier\_Data == Trait \times Identifying\_Object$$

From these supplementary definitions, we may now look to consider the set R of resources, and we see that this is a type that appears all-encompassing for the different objects that can be thought of as resources.

$$R ::= arsrc\langle\langle active\_object\rangle\rangle \mid mrsrc\langle\langle metadata\_object\rangle\rangle \mid$$
$$cresource\langle\langle container\rangle\rangle \mid aclrsrc\langle\langle acl\_rsrc\rangle\rangle \mid$$
$$id\_data\langle\langle Identifier\_Data\rangle\rangle \mid smrsrc\langle\langle san\_metadata\rangle\rangle \mid$$

In the type definition for R, we see an inclusion of items of type *active_object* and *metadata_object* as data items generated by a user. As well as items of type *container* and *acl_rsrc*, which is in line with the Solid specification and outline for resources. In addition to these types, we see the inclusion of the type *Identifier_Data*, which shall allow for us to

later reason about inferences and data skews.

Following, we are introduced to the type P denoting all the principals of the system; here, we see principals of the type: *user* and *company*.

$$P ::= upr\langle\langle user\rangle\rangle \mid cpr\langle\langle company\rangle\rangle$$

We then define the following type as a relation between two companies that are associated with one another to allow us to later reason about possible data sharing between parties so that we have the following zed definition, which keeps track of which companies are associated with one another:

$$Assoc\_comp == company \times company$$

We then move to express the boolean true, false and undetermined we make use of the following syntax definition;

$$bool ::= t \mid f \mid u$$

$$Cat\_Types ::= priv \mid pub \mid undet \mid rstr$$

Here, we describe the different category types so that we are able to have different operations based on the type of categories; for example, given a company reaches the collection threshold — the point at which they have exhausted their aggregative limit, they can be put in a category that has the assigned category type of "restricted" given by *rstr* so they are unable to collect further data.

We now move to consider the basics of the individual models that have formed a part

of our amalgamated privacy-enhanced Solid integrated with CBAC model.

7.2.1 Basics Motivated By the CBAC model

We have introduced the CPARSE types, and in this subsection, we shall expose more of the basic type definitions that were influenced by our drafting of the CBAC model. Thus, we begin with the following type definition:

$$Perm == A \times R$$

Here, we have the familiar definition of a permission such that the preceding zed definition exists to allow permissions to be considered as a relation between actions and resources so that *par* may be described as a couple between a principal and a permission.

Then, as an antithesis, we have the opposition notion, so that we have the following type definition:

$$Forb == F \times R$$

In this definition, we have the introduction of a forbiddance, such that the forbidding of an action on a resource may be expressed.

Following, we now use the previously defined *Perm* definition to complete the syntactically modified *par* relation so that we have an authorisation defined as a Cartesian product between principals and permissions. Similarly, we make use of the previously defined *Forb* to express a denial of rights, defined by a Cartesian product between principals and forbiddances.

$$Auth == P \times Perm$$

$$Denl == P \times Forb$$

7.2.2 Basics Motivated By the Solid Proposal

In the arena of the basics pertaining to our drafting of a Formal Model for the Solid Proposal, we may begin by defining the technicalities that must hold in our use of the pod architecture. To this end, we look at the interactions between the generated resources and that of differing holding containers.

Thus, we observe the following schema:

Solid_Additions

$$Resourceset : \mathbb{P}\, R$$

$$Owners : \mathbb{F}\, user$$

$$Owns : user \leftrightarrow R$$

$$Pod\_owner : user \leftrightarrow pod$$

$$Data\_pod : pod \leftrightarrow R$$

$$Contains : container \leftrightarrow R$$

$$ACL\_Link : R \nrightarrow acl\_rsrc$$

$$ACL\_CatLink : acl\_rsrc \leftrightarrow C$$

$$podtoC : pod \nrightarrow C$$

$$assoc\_data : active\_object \rightarrowtail metadata\_object$$

$$Owners \subseteq \text{dom}\ Owns$$

$$\text{ran}\ Owns \subseteq Resourceset$$

$$\text{dom}\ Pod\_owner \subseteq Owners$$

$$\text{dom}\ Data\_pod \subseteq \text{ran}\ Pod\_owner$$

$$\text{ran}\ Data\_pod \subseteq Resourceset$$

$$\text{ran}\ Contains \subseteq Resourceset$$

$$\text{dom}\ ACL\_Link \subseteq Resourceset$$

Here, we see the *Solid_Additions* schema, influenced by our earlier *Solid_Resource* schema in our formal model for the Solid Proposal. Thus, in this schema, we begin with the introduction of the various sets, relations, and functions that will enable us to make full use of the Solid decentralised architecture and resource management framework.
First, we observe the *Resourceset* set, which is a power set of resources of type R, used

to denote the existing resources in a current instance of the system. We then see the introduction of the set *Owners*, defined as a finite set of type *user*, denoting users in ownership positions.

We then consider the relations and functions in the schema so that we have: The relation *Owns* between a user and a resource expresses that a user, u, owns a specific resource. We see the relation *Pod_owner* between a pod and a user, signifying ownership of the pod by the user.

Then, we observe the relation *Data_pod* between an element of type *pod* and that of type R, which describes that a pod holds a given resource.

The subsequent relation denotes a relationship between a container of type *container* and a resource of type R, denoting that a container, c, of type *resource_container* or *root_container*, contains a resource, r, which could be of any of the various resources types, e.g. *active_object*, *metadata_object* and so forth.

After, we have the function *ACL_Link* between an object of type R and that of type *acl_rsrc*, which is in line with our interpretation of the Solid proposal where a single resource links to a sole access control list resource.

In contrast, the following definition of *ACL_CatLink* defines a relation between an access control list resource, *acl_resource*, and a category, C, which stands to allow an access control resource to have multiple categories that it is related to, so that one may have different access rights dependent on the category of the requester. Finally, we end our functions and relations with a function' *assoc_data* defined between an active data object and a metadata object that entails association.

Following our exploration of these sets, relations, and functions, we now look to the constraints placed upon them to ensure the model performs as intended. To this end, we have subsequent constraints.

- The first constraint highlights that every member of the *Owners* set is found in the domain of *Owns*, i.e. every owner, owns a resource.

- The next constraint also looks at ownership, where we find that the range of *Owns* is a subset of the set *Resourceset* of currently generated resources.

- Following, we observe constraints related to the pods, where we see the restriction on the domain of *Pod_owner* being a subset of Owners.

- We then see the constraint entailing that the domain of *Data_pod* is restricted to a subset of the range of *Pod_owner*. This serves to describe that pods containing generated data are owned by pod owners.

- Whereas, the data in these pods is restricted by the subsequent constraint, which explains that the range of *Data_pod* is a subset of *Resourceset*, i.e. all resources in pods are found in the larger set of resources generated in the current run of the system.

- Similarly, we see a constraint on the range of *Contains*, which expresses that the range of *Contains* is a subset of *Resourceset* or that items contained in containers are items found in *Resourceset*.

- Finally, we have the last constraint, which again deals with subsets of *Resourceset*, so that we have a restriction of the domain of *ACL_Link* as a subset of *Resourceset*, expressing that all resources that have an access control link are in the set of resources found in the system —*Resourceset*.

From the inclusion of these Solid basics, we hoped to draft a consequent model that would provide an integral basis upon which our larger objectives could be realised.

7.2.3 Basics Motivated by Our Defined Privacy Model

Following our decision on the objectives that we wished to take forward into an eventual model that could be used to reason about privacy-driven metadata collection in a decentralised manner, we now shall explore the basal privacy components of this model. Therefore, we move to present the basic schemas and type definitions that will enable us to achieve the said goal of building a formal model that supports our privacy requirements through extension of an integrated Solid and Category-Based Access Control model.

General Privacy Notions

We begin with a look at the schema *Priv_Sets*, which is wholly based on the earlier defined *Priv_Model* of Chapter 4 and shall provide the necessary privacy sets for our augmented model.

Priv_Sets

$T\_s : \mathbb{F}\, R$

$T\_u : \mathbb{F}\, R$

$T : \mathbb{F}\, R$

$Pr : \mathbb{F}\, R$

$K : \mathbb{F}\, R$

$Li : \mathbb{F}\, company$

$Kc\_Data : company \leftrightarrow R$

$\text{ran}(Kc\_Data) \subseteq K$

$K \subseteq Pr$

$Pr \subseteq T$

$T\_s \cup T\_u = T$

$T\_s \cap T\_u = \emptyset$

$T\_s \cap Pr = \emptyset$

This schema serves as provision of the articles related to our formal privacy model presented in Chapter 4. Here, we are reintroduced to several earlier defined sets: for instance, the set T, which, as previously described, will relate to the protected set. Therefore, such articles will be vetted before being shared. Then, we see the highly protected set T_s, where members of this set will not form part of the data collected. Due to said protection, any trait placed here shall be granted a high level protection — and hence, no company may easily infer this trait from analysis or manipulation of collated data.

We then see the set T_u in which data is automatically placed for further consideration — either a user can move articles from here to the more protected set T_s or can allow it

to be projected onto the public sphere, so it can then be collected, and therefore found in the "projected" set Pr.

Following this, we see the definition for the set Pr, which — as alluded to — is a finite set of resources that shall make up the "projected" set.

We then see the introduction of sets that are directly related to the companies collating the generated data so that we may begin with the set K, a finite set of resources that will make up the set of resources that have been collected by companies, or, in other words, the known 'set. Following, we have the finite set Li, which exists to keep track of the companies that have reached their aggregative limits in place of L, described in Section 4.3.2. Then, we have the familiar set Kc_data, which relates the collated data to the company that has collected it.

We then examine the constraints placed upon these sets.

- We start with the first constraint wherein we find the relation between the set Kc_Data and the set K such that we see for any item in the range of Kc_data it is found in K

- We then see the constraint describing that the set K is a subset or equal to the set Pr, which describes that the known set is a subset or equal to the projected set

- The next constraint details that the set T is a union of the sets T_u and T_s

- Following, we see the constraint that informs that the projected set Pr to be a subset or equal to the set T, which essentially describes that the projected set is a subset of all generated data, seeing as generated data is placed in one of the sets T_u or T_s which makes up the set T

- Next, we may observe the constraint describing the relation between T_s AND T_u, where we see that their intersection is equal to the empty set, expressing that they

do not share any items, i.e. items in the set T_s are not in the set T_u

- Finally, the last constraint describes a similar situation to the preceding constraint, detailing that the sets T_s and Pr are mutually exclusive such that the intersection of the two sets is equal to the empty set

As part of this introduction to the formal general privacy notions, we may begin by interrogating the basics relating to the overarching sectors of interest derived from the Solove Privacy Taxonomy.

Within this, we see a need to keep record of entities such as degree of skew or time, such that to fulfill this, we have introduced the following type definition and corresponding functions:

$$Values ::= H \mid M \mid L \mid N$$

Here we define the type *Values* and assign corresponding values of H, M, L, or N, denoting high, medium, low, and nil, respectively; utilising these quantifiers, we felt they would be appropriate for measuring items such as skew and time restraints.

Given our use of these quantifiers, we understood the need to allow comparisons; therefore, we defined relations *lessthan* and *morethan*. Subsequently, we see the axiom definition for *lessthan* as a pair in which the first of the couple would be seen as less than the second. Further, we note that the axiom definition of *morethan* can be found in Appendix Chapter E.

$$lessthan : Values \leftrightarrow Values$$

$$lessthan = \{(N, L), (N, M), (L, M), (N, H), (L, H), (M, H)\}$$

We then defined functions that could be used to modify the values in an incremental or decremental fashion such that we have defined the two modifier functions. The former of which is described as *increaseValue*, while the latter may be found in Appendix E.

First, we have the function *increase_Value*:

$$increase\_Value : Values \pfun Values$$

$$increase\_Value\,(L) = M$$

$$increase\_Value\,(M) = H$$

With these definitions, we are poised to begin consideration of each of the privacy sectors in our metadata-driven condensed list of privacy harms.

Identification

In the arena of Identification, we may first regard our ambition to ensure individuals retain control of the shared identifying features, and hence, we must integrate the notion of skewing. In our work, we strove to ensure that any traits that are described as hidden, through membership in T_s, must be protected in such a way so that we may measure how far collective and individual data objects skew towards the inference of a particular trait.

The collective case is monitored through the use of the *Skew* schema:

Skew

$comp : company$

$usr : user$

$tr : Trait$

$skewValue : Values$

$ctr : \mathbb{N}$

$ctr \leq 9$

In this schema, we see the inclusion of the variable *comp*, which is of type *company*, to keep track of the company collecting the data that may skew towards an identifying trait. We then have the variable *usr* of type' *user*, which denotes the user with the identifying 'trait' that may be skewed towards. Following, we observe the inclusion of the trait in question via the variable *tr* of type *Trait*. Then, we have the variable *skewValue*, of type *Values*, which records the inference skew level towards the identifying trait, *tr*. We then use the *ctr* variable as a means to count exposures affecting the skew so that after a specific number of collated data inferring the given trait, we may increase the skew value such that we may go from L to M based on aggregative exposure. Using these variables with means to record inference skew levels, we can ascertain which traits have most likely been inferred and, therefore, are in the public sphere, Pr.

We may then interrogate the individual case, which investigates the inference skew level of particular resources with respect to the identifying trait. Here, we may appreciate the schema, *RsrcSkewScore*, that allows us to reason about the degree to which an individual resource skews to the inference of a particular trait. Therefore, we present the following schema.

RsrcSkewScore

$rsrc : R$

$tr : Trait$

$score : \mathbb{N}$

$shared : bool$

$score \leq 9$

Here, we may observe the *RsrcSkewScore* schema, which relates a resource with a trait and a natural number; this number will later go on to influence the aforementioned *skewValue* for a particular 'trait' as recorded in the *Skew* schema. The link between natural numbers and skew allows for our solution to comprehend and handle aggregative inferences so that the jump from 'low inference' likelihood to high inference likelihood does not remain entirely trivial. Therefore, in this schema, we are introduced to the variable *rsrc* of type R, which shall be the resource in question whose inference score shall be decided. We note that the mathematics behind this inference score shall have to be attained through use of predictive algorithmic methods, thus out of scope, though we may still reason about the effects of the inference score on data management and collection.

We then are introduced to *tr* of type *Trait*, which is a variable to ascertain the given trait we are considering. Then, the variable *score* relates to the degree to which the resource, *rsrc*, skews towards the trait, *tr*. Finally, we have the variable *shared*, which is in use to record whether the resource has been collected by any of the companies. This variable is defined so we may ascertain the result that sharing would have on the discovery of a particular trait.

Then, upon these variable definitions, we find a constraint that restricts the value of

score to less than or equal to 9, which is also the maximum value that the skew counter given by the variable *ctr* can take. We note that we have defined the 'values' to have a maximum of 9 for the sake of this abstraction. However, we expect these values to be much higher in real-world circumstances.

Following these definitions, we now may present the *Identification* schema, which recognises some of these definitions so that we have:

Identification

$$
\begin{array}{l}
Priv\_Sets \\
E : R \leftrightarrow Identifier\_Data \\
Sk : \mathbb{F}\ RsrcSkewScore \\
\hline
\forall\, a : active\_object;\ i : Identifier\_Data \bullet \\
\quad arsrc(a) \mapsto i \in E \Rightarrow \\
\quad\quad arsrc(a) \notin Pr \lor id\_data(i) \notin T\_s \\
\forall\, i : Identifier\_Data;\ m : metadata\_object;\ rs : RsrcSkewScore \mid \\
\quad id\_data(i) \in T\_s \land rs.r = mrsrc(m) \bullet \\
\quad rs.tr = first\ i \land rs.score \geq 4 \Rightarrow mrsrc(m) \notin Pr
\end{array}
$$

We remind ourselves of the earlier defined objective in the area of Identification under the named objective: **OI6**. The stated goal was to **provide means for individuals to control access to elements of self so they may have a contextual identity for the online space**. In line with this, we may appreciate this schema, which aims to limit unwanted exposure of identity elements through the relation E as well as limit inference capabilities through a record of skew ratings for different resources recorded in the finite set Sk.

Through examining this *Identification* schema, we see that we begin with the earlier defined *Priv_Sets* so that we may meaningfully interact with these existing sets and place restrictions on the allowed states and operations that may exist. We then are introduced to the function E, which relates a resource, r of type R, with an object i of 'type' *Identifier_Data* as a means to relay the exposure of an identifying feature so that a couple (r, i) conveys that the resource i exposes the identifier data object i. With this introduction, we may regard the following constraints.

The first constraint informs us that for any active object and any identifier data object, given a maps to i in E, then a is not in the public set Pr or the resource for i is not in the set T_s, which is a means to prevent active objects sharing any identifying information that a user wishes to keep private.

Whereas the second constraint is concerned with the earlier defined *RsrcSkewScore*, such that it aims to protect collected metadata objects from skewing in favour of any private identifying trait. Thus, in this respect, it describes the following situation: for any identifier data object, i, and any record rs of type *ResourceSkewScore* given i is in T_s and that the resource recorded in $rs.r$ is equal to the metadata resource $mrsrc(m)$ then we have that the if the resource score in $rs.score$ is greater than 4, then the resource, $mrsrc(m)$, cannot be in the projected set, Pr.

Aggregation

We now regard the second of our condensed privacy vulnerability, which is the area of aggregation. We recall the harms of this area as it relates to identity elements being discovered in a similar fashion to jigsaw re-identification [139], as well as the accumulation resulting in the discovery of patterns that can be used to deduce identifying elements, such as traits and preferences.

AcExposure

$Priv\_Sets$

$r : R$

$ri : \mathbb{F}\, R$

$i : Identifier\_Data$

$exprisk : bool$

$ri \subseteq K$

$r \notin K$

$exprisk = t \lor exprisk = f$

$second\, i \in \{id : ID \bullet input\_data(id)\}$

Here, this schema is concerned with the desire to track accumulation exposure in line with the objective **OA4**. Hence, if a particular resource, given by r in this case, was due to cause exposure of an identity attribute, i, when considered in combination with data, ri, found in the known set' then the exposure risk, *exprisk* is set to true.

All of this allows us to reason about the case where, for instance, the identifier data object, i, is in the protected set T_s, then there should be a restriction on where r can be placed.

Aggregation

$Priv\_Sets$

$Li : \mathbb{F}\ company$

$Kc\_Data : company \leftrightarrow R$

$Ac : \mathbb{F}\ AcExposure$

$\text{ran}(Kc\_Data) \subseteq K$

$\forall\ q : Identifier\_Data;\ d : \mathbb{P}_1\ R;\ d\_m : metadata\_object;\ a : Ac \mid$

$id\_data(q) \in T\_s \land d \subseteq K \land d = a.ri \bullet$

$(mrsrc(d\_m) = a.r \land a.i = id\_data(q) \land a.exprisk = t)$

$mrsrc(d\_m) \notin Pr$

Similar to the *Identification* case, we begin with an inclusion of *Priv_Sets*; before, we are introduced to the set Li — defined as a finite set of companies. Li, here, shall be used to record any company that has reached their aggregative limit, as per our objective $OA3$, which is described as: **limit the aggregation of passive digital footprints**. Next, we see the relation Kc_Data between a company, c, and a resource, r, with membership denoting that the company, c, has collected the resource, r. Finally, we introduce the finite set Ac, which is defined as a set of *Ac_Exposure* articles. Thereby, Ac is concerned with the accumulative effect of a resource on the items already in the known set.

In accordance with these articles, we have the following constraints: The first constraint exposes the relation between the set Kc_Data and the set K such that we see for any item in the range of Kc_data it is found in K

We then have the explanation that for any metadata item d_m that maps to a mapping, d, q, between a finite set of data items, d, which is a subset of the known set, K, and an

identifier data object q in Ac — which denotes exposure of the identifying trait in the q — then given q is in T_s then the metadata item should not be permitted to be in Pr.

Association

In the sector of association, we began by making full use of Solid's linked data list, in conjunction with the privacy concerns that arise from associative settings, enabling us to define the relevant schemas that would help prevent identity exposure from associative connections.

Therefore, we began by defining the schema *DataLink* so that we could keep track of linked metadata articles from different users.

DataLink

$m_1, m_2 : metadata\_object$

$l : link\_descrptn$

$sens : bool$

$tr : bool$

$m_1 \neq m_2$

Therefore, in the *Data_Link* schema, we see that two metadata object variables, m_1 and m_2, are defined, in addition to a link description variable, l, of type *link_descrptn* to represent the Solid link types from the World Wide Web Consortium (W3C) Ontology. Then, we have the boolean variable *sens*, which marks the correspondence data as sensitive. As well as these variables, we are introduced to the variable *tr* of type *bool*, which serves to denote whether the given time restriction has passed, in line with the objective **OC5: limit the degree of unwanted, unintentional exposure arising from associative**

connections.

Thus, one of the ways we choose to fulfill this objective is by preventing associative temporal aspects from influencing a user's inferred online persona. Then, we have included the constraint denoting that m_1 is not equal to m_2 to ensure we are talking of two distinctive pieces of metadata that are linked to each other. Finally, we are introduced to the constraint, which explains the case where an item is marked as sensitive by one of the interacting parties, and as a result, the time restriction becomes permanent.

As part of this look into associative data, we note that upon creation of these linked metadata objects, they are placed in each user's respective T_s sets, where it remains for a period of time unless our principle principal of interest decides to allocate otherwise or if it exposes an identity element and therefore will be caught by the earlier skew constraint in the *Identification* schema or the variable *sens*.

Surveillance

In this space, we have defined two objectives, the first being **OS1**, which aims to **ensure surveillance does not breach private spaces**, and the second, $OS2$, which is described as: **limit the extent of surveillance possible through dataveillance**. In efforts to fulfill these objectives, we begin with the following definitions.

$$Env ::= Priv \mid Pub$$

This first type definition is to allow for the environment, given here by Env, to be set so that, in line with the privacy objective **OS1**, we may delineate the private and public spheres — where such delineation is given by the setting of either $Priv$ and Pub, respectively.

$$Mode ::= On \mid Off$$

This second definition of the type *Mode* is to allow the user to set sanitisation mode to be on or off, which will change which data is shared in the public sphere, in line with **OS2**; so that only sanitised data is shared in the public spaced if sanitisation mode is on.

Surveillance

$$
\begin{array}{l}
Priv\_Sets \\
t : Timer \\
\hline
\forall\, d\_m : metadata\_object;\ d\_s : san\_metadata;\ e : Env;\ m : Mode \mid \\
\quad e = Priv \land d\_s = San(d\_m) \bullet \\
\quad\quad m = On \Rightarrow (mrsrc(d\_m) \in T\_s \land (smrsrc(d\_s)) \in Resourceset) \\
\forall\, d\_m : metadata\_object;\ d\_s : san\_metadata;\ e : Env;\ m : Mode \mid \\
\quad e = Priv \land m = On \bullet \\
\quad\quad d\_m \land d\_s = San(d\_m) \in Ti \Rightarrow \\
\quad\quad\quad \{mrsrc(d\_m), smrsrc(d\_s)\} \in T\_s
\end{array}
$$

In the Surveillance schema, we are presented with the familiar Priv_Sets in addition to the variable *Timer*, which is used to reduce the frequency of metadata outputs when the environment has been set to private, given by *Priv*. To accompany these variables, we then have the following constraints:

The first details that for any metadata d_m in any environment, then, given the sanitisation mode is on, the metadata object is kept in T_s whereas the sanitised version of the metadata is included in T_u. The constraint is defined in this manner to enable only sanitised versions to be amongst the articles permitted to be shared.

After, the second constraint explores the case for any metadata d_m where the environment is set as *Priv*, denoting a private environment, and the sanitisation mode is on,

then we see that given the metadata object is in the timed interval set Ti then it implies that the metadata object and its sanitised version are both found in T_s.

7.3 The P-E S.CBAC Model

Having discussed the tenets that make up the fundamental parts of the model, we move to present the primary model, which shall incorporate several of these previously defined articles.

We begin by first including an example of a decomposition of our model, which exists to enable ease of understanding while simultaneously ensuring we are able to present the main tenets of the model here. Therefore, we have the first schema.

Principal_Sets

$$Companies : \mathbb{P}_1\ company$$

$$Users : \mathbb{P}_1\ user$$

Then we may look into attributes that make up an instance of the model for inclusion into the latter PE_SCBAC model.

Instance

$Principal\_Sets$

$permset : \mathbb{F}\ Perm$

$principalset : \mathbb{F}\ P$

$catset : \mathbb{F}\ C$

$assoc\_princ : \mathbb{F}\ Assoc\_comp$

$cat\_link : \mathbb{F}\ Link\_Cat$

$closeFriend : \mathbb{F}\ P$

$friend : \mathbb{F}\ P$

$acquaintance : \mathbb{F}\ P$

$principalOI : assgndOwner$

$group : \mathbb{N}$

$group \leq 3$

$closeFriend \subseteq principalset$

$friend \subseteq principalset$

$acquaintance \subseteq principalset$

$closeFriend \cap friend = \emptyset$

$friend \cap acquaintance = \emptyset$

$acquaintance \cap closeFriend = \emptyset$

$\forall\, u : Users \bullet upr(u) \in principalset$

$\forall\, c : Companies \bullet cpr(c) \in principalset$

$\forall\, p : principalset \bullet p \in \{u : Users \bullet upr(u)\} \lor p \in \{c : Companies \bullet cpr(c)\}$

In the *Instance* schema, we are met with the same sets found in the *Instance* schema

of the CBAC model of Section 1.4. We then see a set to keep track of associated principles — namely, companies in *Assoc_Princ* relation. Following, we see a set that keeps track of linked categories. Subsequently, we have the introduction of the sets: *closeFriend*, *friend*, and *acquaintance*, which are sets of principals that will allow the owner to assign groups of users to categories. Finally, we are introduced to the *principalOI* variable, which shall keep track of the assigned owner, whose data the *Priv_Sets* schema will record.

Having presented these schemas, we shall now describe the primary model.

PE_SCBAC

$Instance$

$Identification$

$\vdots$

$Priv\_Sets$

$Solid\_Addition$

$Principal\_Sets$

$skewn : \mathbb{P}\, Skew$

$Linked\_Data : \mathbb{F}\, DataLink$

$Assocs\_Sets : \mathbb{F}\, assgndPrivSets$

$Threshold : company \leftrightarrow \mathbb{N}$

$par : P \leftrightarrow Perm$

$ForbiddenCategorySet : P \leftrightarrow \mathbb{F}\, C$

$\text{dom}\ par \subseteq principalset$

$\text{ran}\ par \subseteq permset$

$\vdots$

$\text{dom}\ Owns \subseteq Users$

$\forall\, a : assoc\_princ \bullet first\ a \neq second\ a$

$\forall\, a : assoc\_princ \bullet cpr(first\ a) \in principalset \land cpr(second\ a) \in principalset$

$\forall\, p : acquaintance \bullet p \in \{u : Users \bullet upr(u)\}$

$\forall\, c1, c2 : C \bullet \{c1 \mapsto c2\} \in cat\_link \Rightarrow c1 \neq c2$

$\forall\, i, j : Skew \mid i \in skewn \land j \in skewn \bullet$

$$(i.comp = j.comp \land i.usr = j.usr \land i.tr = j.tr) \Rightarrow i = j$$

$\vdots$

As part of the definition for our primary schema, we see the inclusion of the various earlier defined schemas — namely that of Instance, Identification, Aggregation, Association, Surveillance, *Priv_Sets*, and *Solid_Addition*. However, we note that not all of these articles are presented in this condensed version of the model. Following these schema inclusions, we find a series of sets that will enable us to keep track of persons interacting with the system, including that of *Users* and *Companies* as well as the sets keeping track of attributes such as skew, via *skewn* or linkage, as portrayed through the finite set of *Linked_Data*. In addition, we have the set recording the privacy sets of the associates of the main principal by way of the *assgndPrivSets* schema.

From an observation of the condensed list of constraints presented here, we see we begin with those concerned with entities from the CBAC model, such as *par*, and may rightfully reason the others are included in the full version of the model. Then we see restrictions relating to the Solid model; for example, we see a restriction on the set *Owns*.

Following, we see a constraint referring to ownership so that the domain of *Owns* is restricted to a subset of *Users*. Then, we have constraints on associated companies such that one constraint looks at associated principals and determines that the first principal in the pairing cannot be the same as the second principal in the pairing, to mean that a principal cannot be associated with itself in the *assoc_princ* pairing.
Next, we see the other constraint in the area, which details that both principals in the paring are found in *principalset*

After, we have a constraint that looks at items of the set *acquaintance* so that we have that all acquaintances can be found in the set *Users*
Similarly, the same constraints are placed on the sets *friends* and *closefriends* though these are not demonstrated in this condensed schema.

As it relates to the following constraint, we focus on linkage between categories, where we see that any pairing indicating a category linkage — given by inclusion in $cat - link$ — can't have a category linked to itself. Finally, we have a constraint, which looks at records forming a *Skew* article. Here, we see that given two records share the same company, same user, and same traits, then it implies they are identical records and, thus, are the same singular record.

It is hoped that from this primary model, we may begin to frame the necessary logically based requirements to resolve our defined condensed privacy vulnerabilities while leveraging the dynamic access control influenced by the Category-Based Access Control meta-model and the privacy-inducing decentralisation granted by the Solid proposal. In light of this, we shall subsequently describe the operations incorporated in this model to achieve our earlier decided-upon objectives.

7.4 Operations Pertaining to the Solid Protocol and Category-Based Access Control

Due to the inclusion of these components as essential elements in our model, we see the inclusion of schemas relating to both the Solid Protocol workings as well as the Category-Based Access Control framework. For example, we have the reoccurrence of schemas such as *CreateResource*, *RemovePerms*.

Then, in addition to these schemas, we also see schemas in this area that have been created to deal with the interactions existing between these two components of our model. Thus, for the integration of Solid and Category-Based Access Control, we have defined a method for linking access control resources to the existing categories.

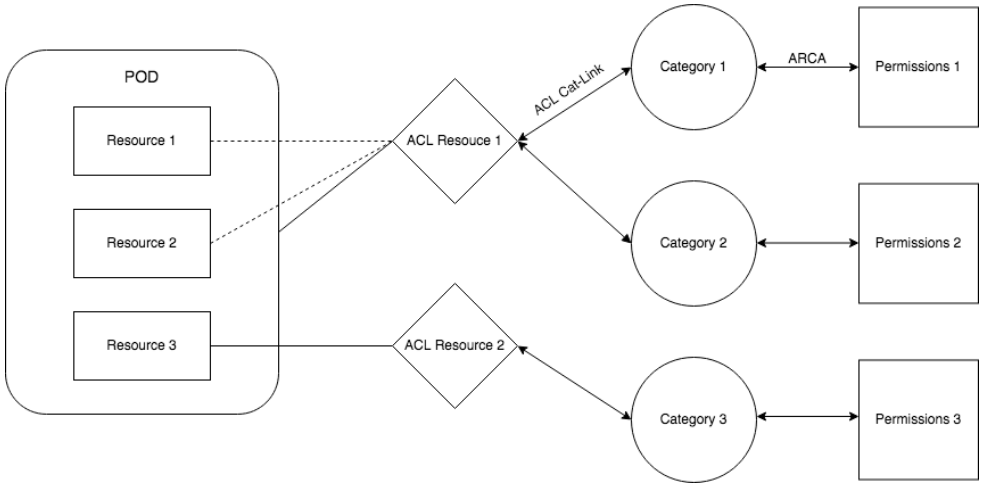


Figure 7.1: Linkage of access control resource to categories

With respect to this linkage, we describe Figure 7.1, which demonstrates how resources may attain their permissions hierarchically while simultaneously incorporating categories. In this diagram, we see that resources inherit their access control resources from their containing pod or container — as is indicated by the dotted lines. This access control resource is then linked to differing categories with permissions linkages.

Following such a description, we now move to present the schema that handles this linkage.

$$
\begin{array}{|l}
\hline
ACLtoCat \\
SetupRetained \\
\vdots \\
a? : acl\_rsrc \\
c? : C \\
\hline
a? \mapsto c? \notin ACL\_CatLink \\
a? \in \operatorname{ran} ACL\_Link \\
c? \in catset \\
Pod\_owner' = Pod\_owner \\
Owns' = Owns \\
Data\_pod' = Data\_pod \\
ACL\_CatLink' = ACL\_CatLink \cup \{a? \mapsto c?\} \\
\vdots \\
\hline
\end{array}
$$

Here we take an access control resource, $a?$, that is in the range of ACL_Link, to signify that it is linked to a parent container, as well as a category, $c?$, in the set of categories in the current run of the system, given by *catset* and we add a relation between the access control resource, $a?$, and the category, $c?$, to $ACL\_CatLink$.

Having considered this integration, we now move to interrogate a selection of the schemas that may run in an Instance of the system.

7.5 Operations Working on an Instance of the Model

In this section, we explore the operations at play during an instance of the system so that we may expose the workings and interactions at play within an instance of the system.

We may begin this foray through the introduction of a primary component of any such system — the creation of resources. We note that while we have previously taken time to consider the active creation of resources, we shall now appreciate the generation of metadata. As the generation of metadata occurs without conscious input, we saw it fitting to define a schema that is concerned with sole metadata generation without any associated active data, such that we may have metadata generation that may occur as a user interacts with devices, apps, or background activities, i.e. step tracker — hence, we describe the schema: *GeneratedMetadata*.

GeneratedMetadata

$SetupRetained$

$GenResource$

$\vdots$

$o : user$

$m? : metadata\_object$

$a : acl\_rsrc$

$c?container$

$p : pod$

$cresource(c?) \mapsto a \in ACL\_Link$

$o \mapsto p \in Pod\_owner$

$mrsrc(m?) \notin Resourceset$

$Resourceset' = Resourceset \cup \{mrsrc(m?)\}$

$Owns' = Owns \cup \{o \mapsto mrsrc(m)\}$

$Data\_pod' = Data\_pod \cup \{p \mapsto mrsrc(m)\}$

$ACL\_Link' = ACL\_Link \cup \{mrsrc(m?) \mapsto a\}$

$Contains' = Contains \cup \{(c?) \mapsto mrsrc(m?)\}$

$\vdots$

As part of this schema, we notice the necessary inclusion of stagnancy operations. Then, we are met with the inputs $m?$, denoting a metadata object, and $c?$, denoting a container object, as well as the local variables o, a, and p denoting, respectively, a user, an access control resource and a pod.

Amongst the constraints, we see the addition of m to the *Resourceset* and the in-

heritance of the access control resource linked to the parent container via $ACL\_Link' = ACL\_Link \cup \{mrsrc(m?) \mapsto a\}$.

Although not present in this condensed presentation of the schema, we have checks to ensure the container is owned by the user in question, i.e. $o \mapsto cresource(c?) \in Owns$ and that the container, $c?$ is in the pod p.

Then, as is usual for resource generation, we see that this metadata object $m?$ is added to the sets $Owns$, $Data\_pod$, $Contains$, and ACL_Link.

We may then view a schema that involves an outside party interacting with resources so as to gain insights into the data interaction from a company, amassing data through use of their services or applications. Thus, we introduce the *CompanyInteraction* schema.

CompanyInteraction

$SetupRetained$

$\vdots$

$c_1? : company$

$c2 : \mathbb{F}\ company$

$r? : R$

$a? : A$

$p : Perm$

$n : \mathbb{N}$

$c : C$

$Environment.e = Public$

$\vdots$

$p = (a? \mapsto r?)$

$cpr(c_1?) \mapsto p \in par$

$n = \textbf{if}\ c_1? \in \mathrm{dom}\ Threshold\ \textbf{then}\ Threshold(c_1?) + 1\ \textbf{else}\ 1$

$r? \in \{s : skscores \bullet s.rsrc\}$

$c2 = \{a : company \mid c_1? \mapsto a \in assoc\_princ \bullet a\} \cup$

$\quad \{b : company \mid b \mapsto c_1? \in asssoc\_princ \bullet b\}$

$\vdots$

$Kc\_Data' = \textbf{if}\ c2 \neq \emptyset\ \textbf{then}\ Kc\_Data \cup \{(c_1?, r?)\} \cup \{c : c2 \bullet (c, r?)\}$

$\qquad\qquad \textbf{else}\ Kc\_Data \cup \{(c_1?, r?)\}$

$K' = K \cup \{r?\}$

$Threshold' = \{c_1?\} \lhd Threshold \cup \{(c_1?, n)\}$

$\vdots$

In this schema, we see a check to ensure the company interacting with the resource has the authorisation to do so, given by $cpr(c_1?) \mapsto p \in par$. Following, we observe a check for the aggregation threshold and then an increment in the company's threshold count.

As expected from our privacy motivations, we see a check to see that the resource's skew scores have been recorded. We then define the finite set $c2$, which include the companies associated with the company, c_1, by relation in $assoc\_princ$, and hence, we assume the data is shared amongst them.

Finally, we see the addition of the resource, $r?$, to the projected and known sets of Kc_Data, K, and Pr.

Further, we note that these schemas handled the case when the environment had no conditional settings. Thus, we may now consider when the environment variable is set to $Priv$, or sanitisation mode is set to On, and following, the case where both conditional settings are set.

Thus, we move to regard one of the more privacy-focused areas of an instance so that we may introduce the schema $PrivSanDataInteraction$.

PrivSanDataInteraction

$SetupRetained$

$\vdots$

$c_1? : company$

$r?, sr : R$

$a? : A$

$c2 : \mathbb{F}\ company$

$p, sp : Perm$

$n : \mathbb{N}$

$c : C$

$Environment.e = Priv$

$Environment.m = On$

$P\_Timer.b = t$

$sr = smrsrc(san(returnmeta(r?)))$

$\vdots$

$c2 = \{a : company \mid c_1? \mapsto a \in assoc\_princ \bullet a\} \cup$

$\quad \{b : company \mid b \mapsto c_1? \in asssoc\_princ \bullet b\}$

$(cpr(c_1?) \mapsto c) \in pca$

$\vdots$

$T\_u' = T\_u \cup \{sr\}$

$Kc\_Data' = \textbf{if } c2 \neq \emptyset \textbf{ then } Kc\_Data \cup \{(c_1?, sr)\} \cup \{c : c2 \bullet (c, sr)\}$

$\qquad\qquad \textbf{else } Kc\_Data \cup \{(c_1?, sr)\}$

$Threshold' = \textbf{if } (c2) \neq \emptyset$

$\qquad\qquad\qquad \textbf{then } (\{c_1?\} \cup c2) \lhd Threshold \cup \{(c_1?, n)\} \cup \{c : c2 \bullet (c, n)\}$

$\qquad\qquad \textbf{else } Threshold \cup \{(c_1?, n)\}$

This schema is concerned with the most private environment that data collection can be occurring; such that, it has checks to ascertain that the P_Timer boolean variable is set to true denoting enough time has passed since the last metadata object was sent. In addition to a check that the environment variable is set to $Priv$ and the sanitsation mode set to On.

Then we see the collection of the sanitised version of the metadata so that we have an addition to Kc_Data that sees sr added to the collecting company's data set and its associated companies. We would also see additions to the set K and the relevant privacy sets, but this is not included in this condensed presentation of the schema.

We may then go on to consider aggregate exposure, and the mechanisms in place to prevent this, as such we look to the threshold variable that we defined earlier, to keep track of the metadata currently collected. Once, a threshold has been reached the following schema is invoked:

$$c? : company$$
$$InvokeLimitReached \mathrel{\widehat=} AddtoL(c?) \fatsemi ForbAgent(c?)$$

This $InvokeLimitReached$ schema ensures that appropriate schemas are called to handle the case that the aggregation threshold for metadata objects has been reached. The schema $AddtoL$ performs the necessary action of adding a company to the list Li which is used to account for companies that have reached their aggregative.

Following this addition, we may interrogate the succeeding schema, $ForbAgent$ defined in $InvokeLimitReached$, which seeks to replace the current category permissions assigned to the company with corresponding forbiddances instead:

ForbAgent

$SetupRetained$

$ModelRelRetained$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$PrivSetsRet$

$c? : company$

$f : \mathbb{F}\ Forb$

$cb : P \leftrightarrow \mathbb{F}\ C$

$ca? : C$

$f = \{x : par \mid first\ x = cpr(c?) \bullet fbd(first(second\ x)) \mapsto second(second\ x)\}$

$ForbiddenCategorySet' = ForbiddenCategorySet \cup$

$\{cpr(c?) \mapsto \{pc : pca \mid first\ pc = cpr(c?) \bullet second\ pc\}\}$

$ca?.category\_type = rstr$

$pca' = \{cpr(c?)\} \lhd pca \cup \{cpr(c?) \mapsto ca?\}$

$par' = \{cpr(c?)\} \lhd par$

$pfar' = pfar \cup$

$\{b : par \mid first\ b = cpr(c?) \bullet$

$first\ b \mapsto (fbd(first(second\ b)), second(second\ b))\}$

$arca' = arca$

$farca' = farca \cup \{fo : f \bullet ca? \mapsto fo\}$

Here we retain a list, using the global variable *ForbiddenCategorySet* of the past catergories the company belonged to. We also observe a change of all current permissions to their forbiddance version, via the function *fbd* that is performed on each action in the companies previous authorisations to result in the set $f?$.

Upon reception of the companies' metadata deletion, which will have to be enforced through legislative changes, so that we can have evidence via mechanisms of a cryptographic commitment, zero-knowledge proof or any other multi-party zero trust proof method, that does not expose the companies' data, we have the following schema that shall be invoked:

$$c? : company$$

$$InvokeLimitResolved \mathrel{\widehat=} LimitResolve(c?) \fatsemi ReinstatePerms(c?)$$

We note that the schema for *ReinstatePerms* is presented in Appendix Chapter E.

LimitResolve

$SetupRetained$

$ModelRelRetained$

...

$AssocDataRet$

$c? : company$

$m, n, o : \mathbb{P}\, R$

$c? \in Li$

$m = \{d : metadata\_object;\ kc : Kc\_Data \mid (mrsrc(d)) = second\, kc \land$

$first\, kc = c? \bullet (mrsrc(d))\}$

$n = \{n : metadata\_object;\ kc : Kc\_Data \mid (mrsrc(n)) = second\, kc \land$

$first\, kc \neq c? \bullet (mrsrc(n))\}$

$o = m \cap n$

$Kc\_Data' = Kc\_Data \setminus \{m_0 : m \bullet (c?, m_0)\}$

$Li' = Li \setminus \{c?\}$

$Pr' = Pr \setminus m \cup n$

$T\_s' = T\_s \cup (m \setminus o)$

$T\_u' = T\_u \setminus (m \setminus o)$

$K' = K \setminus m \cup n$

$E' = E$

$Sk' = Sk$

$skewn' = skewn$

$skscores' = skscores$

$Threshold' = Threshold \setminus \{a : Threshold \mid first\, a = c?\}$

7.6 Summary

Through our description of our formal model we have been able to demonstrate how an amalgamation of the various models would be framed. Here, we were able to portray the areas of the model that pertained to the various sub-models, which enabled the interactions between these models to be portray as per the *CatACL_Link* of subsection 7.4.

In addition, in this model we were able to explicitly see the differences due to the level of privacy enhancement wherein through the pursuit for this final model we were able to demonstrate our success in building a privacy enhanced Category-Based access control model atop of our model for Solid.

Furthermore, given these privacy enhancements were resolutions to our objectives, we have been successful at creating a model that adheres to the goals of our subsidiary Research Question 4 : *Is it possible to build a formal model that supports our privacy requirements through extension of an integrated Solid and Category-Based Access Control model?*

To this end, we were able to build said formal model and test it utilising the ProZ animator. In this space, we were able to perform necessary type checking, which returned no errors as well as model checking, which performed as expected during a random traversal of operations.

Chapter 8

Addressing Our Privacy Objectives

In the preceding chapter, we presented the formal model that we envisioned would aid in mitigating privacy violations caused by the widespread collection of metadata, which we felt was bordering on exploitation, given the rate of metadata collection.

As a result of our presentation of such a model, we have determined it best to investigate the success and effectiveness of our model regarding the overarching objectives set. Consequently, we aim to expose the resolution of these individual objectives and grant examples of operations in our model that enabled us to meet the privacy requirements defined in Chapter 4.

Our discussion will begin with a brief reminder of the context that led to the formation of the objective before we describe the consequent operations formed. Then, we conclude by describing how these given operations satisfy the conditions set in the goals. To this end, we have chosen an objective from each of the defined privacy sectors so that we may interrogate our resolution through operations.

8.1 Metadata-Driven Privacy Objectives

From our work with the Solove taxonomy as recounted in Section 4.1, we ascertained a concise view of the sectors that would be appropriate for our investigation. This condensed view of privacy harms was informed by consideration of the capabilities granted through use of the decentralised Solid system.

Then, for these condensed privacy harms of Identification, Aggregation, Association, and Surveillance, we began drafting privacy objectives to address the harms discovered in these areas.

From our determined privacy objectives, found in Section 4.2.5, we selected the following objectives for further investigation in this chapter. Thus, our selection was as follows.

- Identification

  OI6: Provide means for individuals to control access to elements of self so they may have a contextual identity for the online space

- Aggregation

  OA4: Limit the exposure of personal attributes that result from aggregation

- Association

  OC5: Limit the degree of unwanted, unintentional exposure arising from associative connections

- Surveillance

  OS2: Limit the extent of surveillance possible through dataveillance

Therefore, focusing on this list of objectives, we move to interrogate each of these objectives so as to grant an understanding of the mechanisms we have taken in our resolution of these objectives.

8.2 Addressing the Identification Objective

8.2.1 Context

In determinations of what constitutes identity, there are theories [140] rejecting the concept of identity as a static or singular object where, instead, a more fluid identity is described — differing based on context and audience. That said, in the current web environment, there is little differentiation in context or audience.

We must also evaluate identity as it relates to devices owned by an individual. In this respect, we must consider extraneous instances such as sensors. Sensors have the advantage of being a dual-use covert method of data collection so that while the accelerometer serves to detect screen orientation and, thus, heighten usability, it may also be used to fingerprint devices.

Therefore, in crafting solutions, we must ignore the extraneous identification of unique devices and, instead, focus on limiting an individual's identity exposure. Whereupon we understood the need to work on granting context to the unbounded, distrustful terrain that is the online space [141].

These considerations led to the following objective in this space.

OI6: Provide means for individuals to control access to elements of self so they may have a contextual identity for the online space

8.2.2 Proposed Solution

In our efforts to combat some of the issues faced with identity exposure in the hostile online environment, we, as described in Section 2.2, regarded an individual's persona in terms of data items, as is the case in the datafication of individuals. Then, we separated this data into differing sets with different protection levels, with the T_s being the most protected

set whose articles would be kept away from the public sphere of the web.

Following this, we introduced methods to allow users to define traits to keep hidden. Then, any data objects that skew towards that attribute would be added to the most protected set T_s to prevent companies from collecting data objects that could be used to infer the attribute.

Therefore, we present the following schema, which will handle the hiding of a trait or attribute.

HideTrait

$$
\begin{array}{l}
SetupRetained \\
\vdots \\
t? : Trait \\
i : \mathbb{F}\, Identifier\_Data \\
\hline
i \neq \emptyset \\
\{i_1 : i \bullet id\_data(i_1)\} \cap T\_s = \emptyset \\
\forall\, id : i \bullet first\ id = t? \\
Kc\_Data' = Kc\_Data \\
T\_s' = T\_s \cup \{i0 : i \mid second\ i0 = NULL \bullet id\_data(i0)\} \\
T' = T \cup \{i0 : i \bullet id\_data(i0)\} \\
\vdots
\end{array}
$$

Following this addition of the trait into the private set would influence any subsequent generated data so that if there was a non-negligible skew, the data could be added to T_s, as indicated in the full model found in Appendix E.

8.2.3 Discussion

The use of the schema *HideTrait* grants users fine-grained control of access to self. More so, through the ability to calculate the degree of skew for a particular resource, as seen in Section 7.2.3. From the objective outlined above, we are content with the tentative steps made in this area and are appreciative of the control users would be granted as a result of the formal operations in place to restrict access to sensitive articles of data.

8.3 Addressing the Aggregation Objective

8.3.1 Context

The aggregation of online data, as discussed in Section 4.2.2, poses a considerable threat to user's privacy, particularly as it relates to the discovery of identifiable traits. These current aggregation models can be used to form extremely detailed inferences about a wide range of personal attributes, as found in [13].

Furthermore, the expectation of limits on discoverable or known facts is disrupted by the power and scope of aggregation as it combines data in 'new, potentially unanticipated ways to reveal facts about a person that are not readily known' [25].

To handle such, we understood the need to define concise objectives that would enable construction of methods to combat these pressing concerns. Thus, we have the following objective.

OA4: Limit the exposure of personal attributes that result from aggregation

8.3.2 Proposed Solution

Amongst the schemas used to handle the issue of aggregation as it relates to the potential of recombination, we present one of these schemas that works to restrict the inference of

identity attributes that have been marked for the most protection. Thus, the schema we choose to present is the schema that handles this exposure risk as a result of recombination.

AcExposure

$$Priv\_Sets$$

$$r : R$$

$$ri : \mathbb{F}\, R$$

$$i : Identifier\_Data$$

$$exprisk : bool$$

$$ri \subseteq K$$

$$ri \neq \emptyset$$

$$r \notin K$$

$$id\_data(i) \in T\_s \Rightarrow exprisk = t$$

$$exprisk = t \Rightarrow r \notin Pr$$

In the *AcExposure* schema we see a structure involving four elements: r, of type R; ri, of type $\mathbb{F}\, R$; i, of type *Identifier_Data* and *exprisk* of type *bool* for boolean.

Upon these variables, we have constraints that serve to achieve the objective set out, such that we have the following two constraints.

The first describes the following situation: given an identifier data object is found in the set T_s, then the exposure risk, *exprisk*, should be true, t.

Then, we finish with the restriction where, given the exposure risk is true, then the resource r, which causes exposure, should not be found in the projected set Pr.

From this, we now consider the schema that handles the resources found in the set keeping track of potential exposures, *Aggregative_Exp*.

PositiveAgExp

$SetupRetained$

$\vdots$

$a : AcExposure$

$i? : Identifier\_Data$

$id\_data(i?) \in T\_s$

$a \in Aggregative\_Exp$

$a.r = r?$

$T = \textbf{if } first\ a.i = first\ i? \wedge exprisk = t \textbf{ then } T \cup \{a.r\} \textbf{ else } T$

$T\_s = \textbf{if } first\ a.i = first\ i? \wedge a.exprisk = t \textbf{ then } T\_s \cup \{a.r\} \textbf{ else } T\_s$

$\vdots$

8.3.3 Discussion

We note that we have previously exposed aggregation with regards to the objective **OA3** and thresholds in Section 7.5; hence, in this chapter, we have taken the objective **OA4** to explore. To this end, we began tentative steps in this area, relating to the need to assess the addition of a data object to the existing resources in the K set and whether its addition to the projected sphere would lead to an inference of an item in the protected set, T_s.

We appreciate that there shall have to be more work in this area relating to predictive analysis to calculate the exposure risk faced by a single metadata item. However, we note that as an abstraction, this schema fulfills the objective set out. Through the use of this schema, we have ensured that any item found to cause recombinational exposure is not in the realm of items to be collected and shared. Thereby granting users protection over the accidental leakage of identity attributes through aggregative data recombination.

8.4 Addressing the Association Objective

8.4.1 Context

In the provisioning of an objective in the sector of Association, we understood the unique challenges faced in the space, and hence, alienating it from the pairing with Aggregation as found in the Solove taxonomy [25]. With regard to Association, researchers have begun looking into 'privacy infringements that happen not on an individual, but on a multi-party level' [142].

In this space, an individual's privacy was affected and, thus, reliant on associates' data-sharing decisions. Therefore, it was evident there was a need to ensure online communications with associates were protected to prevent leakage of extraneous data on either side. In doing so, a wariness of temporal leakage via vanity checks had to be considered. Thus, in pursuit of addressing these issues, we framed the following objective:

OC5: Limit the degree of unwanted, unintentional exposure arising from associative connections

8.4.2 Proposed Solution

To grant users greater control of their associative data, we have the following schemas, which work to limit the leakage of data from associative connections.

Therefore, we begin with the *DataLink* schema as introduced in Section 7.2.3, which works to place a temporal restriction on the share of either data items.

DataLink

$m_1, m_2 : metadata\_object$

$l : link\_descrptn$

$sens : bool$

$tr : bool$

$m_1 \neq m_2$

Here, the *Data_Link* structure is added to the *Linked_Data* set to keep track of the metadata generation from associative interactions. As well as the temporal restriction by way of tr, we see that if the *sens* boolean value is set to true, then the metadata objects cannot be shared and must remain in the sets T_s.

Further, we may consider a constraint found in the *PESCBAC* schema:

PE_SCBAC

$\vdots$

$Linked\_Data : \mathbb{F}\ DataLink$

$\vdots$

$\vdots$

$\forall\, d : Linked\_Data \bullet d.sens = t \Rightarrow mrsrc(d.m_1)) \notin Pr \land mrsrc(d.m_2) \notin Pr$

$\vdots$

8.4.3 Discussion

Through use of this *Data_Link* structure, we have means of regaining control over unintentional extraneous data leakages. The consequent mechanisms that work in accordance with an addition to *Data_Link* serve to provide a protected environment for the metadata articles. Furthermore, the use of this structure grants assurance that sensitive data will not be shared and that vanity checks following an interaction shall not accidentally expose either party due to the temporal restriction on the sharing of said interaction.

Therefore, we have exposed potential preliminary steps to take to mitigate issues arising from associative communications and data-sharing, in line with the objective that we set in this area.

8.5 Addressing the Surveillance Objective

8.5.1 Context

The frequency at which metadata is being collected has led to metadata collection being equated with surveillance [114]. Further, it has been noted [25] that surveillance in private places may infringe on an individual's privacy and autonomy, leading to issues such as self-censorship.

These revelations expose a need to further delineate public and private with respect to dataveillance and, thus, surveillance. As a result, in an environment marked as private, there is a need to enact differentiations in metadata sharing and collection patterns to retain its usefulness while upholding privacy expectations.

This consideration led to the following objective:

OS2: Limit the extent of surveillance possible through dataveillance

8.5.2 Proposed Solution

In light of this objective, we looked to methods of reducing the frequency of metadata sharings and, thus, collections.

Timer

$b : bool$

$ti : \mathbb{F}\, R$

$v : Values$

$lm : \mathbb{F}\, metadata\_object$

$\forall\, t : ti;\ r : R \mid t = r \bullet$

$\quad t \in \{a : metadata\_object \bullet mrsrc(a)\} \cup \{b : sanmetadata\_object \bullet smrsrc(b)\}$

This schema serves to reduce the frequency of metadata sharing when the environment is set to *Priv* to denote a private space. In line with this, metadata objects will only be permitted entry to T_u once the timer countdown is complete, at which point the value b will be set to true.

Then, we have the following schema to handle the timer functions and the assignment of the new variables upon fulfillment of the temporal conditions.

DecreaseTimer

$SetupRetained$

$\vdots$

$nt? : Timer$

$m? : R$

$nt?.v = \textbf{if } PTimer.b = t \textbf{ then } H \textbf{ else } decreaseValue(PTimer.v)$

$nt?.b = \textbf{if } PTimer.v = N \textbf{ then } t \textbf{ else } f$

$nt?.Ti = \textbf{if } PTimer.b = t \textbf{ then } \emptyset \textbf{ else } PTimer \cup \{m?\}$

$nt?.lm = \textbf{if } PTimer.b = t \textbf{ then } m? \textbf{ else } PTimer.lm$

$PTimer' = nt?.\vdots$

8.5.3 Discussion

Through the use of these timer schemas, we have been able to offer a means to keep metadata data objects shielded depending on the defined frequency of data sharing, which helps to limit the continual stream of metadata collection that may be used to infer daily activities.

Therefore, we have been successful in attempts to demonstrate how these mechanisms can be fruitful in guiding development of structures to help combat concerns of metadata surveillance through a reduction in the frequency of collectible data objects.

8.6 Summary

In this chapter, we have been able to meaningfully express the workings of our model in relation to the privacy objectives we were hoping to fulfill. Through an exploration of

objectives from each area of our condensed privacy harms, we have satisfactorily been able to demonstrate that our model works to address these harms.

Therefore, in our presentation of schemas to address the various privacy objectives, we resolved that we were successful in demonstrating how the operations defined would aid users in a manner that highlighted how each schema worked in line with individual privacy objectives.

Chapter 9

Conclusion

The research presented in this dissertation was an attempt to demonstrate our efforts in answering the question: *"How can we leverage a decentralised approach to data ownership, together with formal models of data access, to aid in the protection of data subjects' privacy so as to mitigate inference-driven identity exposures from metadata collection?"*

Such a research question exposes the complexity of our work as it relates to the compounding of numerous elements, including decentralisation, privacy and autonomous access control. The compounding of these elements was in attempts to formulate a model that works to utilise the decentralisation of the Solid pods with the category-based access control framework and the formal privacy requirements guided by Solove's Privacy Taxonomy. To this end, in our work, we had plans to integrate the Solid proposal with CBAC as well as our series of six privacy objectives upon which our model would attempt to resolve.

In this chapter, we have the following sections to aid in the final discussions of the work presented. In Section 9.1, we look at an evaluation of our work holistically, whereas, in Section 9.2, we interrogate our contributions and the manner in which they helped to answer our overarching research question. Then, in Section 9.3, we discuss the drawbacks

and limitations of our research. We then move, in Section 9.4, to discuss the current landscape and the future work needed to fulfill the objectives further, particularly those outside of our research scope. Finally, Section 9.5 details our concluding thoughts on our contribution to the research community.

9.1 Evaluation of our work

Our research question was framed as the following objective: To leverage a Solid-style approach to data ownership together with formal models of data access to aid in the protection of data subjects' privacy so as to mitigate inference-driven identity exposures from metadata exploitation. To resolve said objective, we proposed a series of formal models to act as the first step towards a database management system that could feasibly deter the issues poised in our research objective, particularly as it relates to metadata exploitation.

This series of formal models were constructed to address our objective from the differing vantage points demanded by such an issue. In this respect, we looked at the vantage point of decentralisation, which — through investigation — would be necessary to reason about restricting access to data items. Since without granting user ownership, we understood we would not be able to progress with our privacy protection aims.

Similarly, we appreciated that we should not expect everyday users to be access control experts. Wherein due to the difficulties that arise in metadata consideration in user privacy — we decided we would need to utilise an access control with both reactive and proactive capabilities. The CBAC framework offered said capabilities, effortlessly adapting to events and situations and consequently adjusting the access rights granted with little user interaction needed.

However, as noted, metadata privacy is particularly troublesome since its passive nature

in the online space results in data leakages in the form of unconscious input and lack of reasoned consent. As a result, we recognised the advantage of describing privacy formally through translation of the Solove Privacy Taxonomy. To this end, we formulated privacy objectives that were within our scope, would reasonably follow on from capabilities granted by a decentralisation platform, and were relevant to the field of implicit passive data in the form of metadata.

Thus, after we had drafted these separate contributions, we began attempting to bridge these structures together while striving to retain the strengths from each framework and working to align with our overarching goals. Therefore, from these different contributions, we successfully compounded them into a final formal model, as portrayed in Chapter 7, that we hoped would grant a resolution to our overarching question.

9.2 Contributions

Following a brief exploration of the work presented in this dissertation, we now move to individually discuss each individual contribution as it relates to our subsidiary research questions. This discussion is to expose how the various research questions were able to encourage progress in our pursuit of the fulfillment of our main research question. Hence, through these contributions, we are able to witness the journey of the compilation of our work into an eventual formal model to resolve the main research objective.

9.2.1 Contribution 1

Our first contribution was constructed to handle the question: *RQ1: How can we formalise a model for privacy to tackle implicit inferential data from consideration of identifiable privacy harms?*.

To resolve the aforementioned research question, we began with a survey of the current

state of personal data management online. In doing so, we evaluated the findings concerning the contention between data collection and privacy, focusing on metadata. Then, to satisfactorily encapsulate the differing contexts of online data privacy, we looked to a taxonomy for privacy.

Through use of the Solove Privacy Taxonomy, we hoped we could identify applicable privacy harms in the different online contexts. From this, we found success in defining objectives for privacy preservation that could readily translate to actionable formal requirements and, thus, resulting in a formal model for privacy.

Hence, the resolution of this question was demonstrated through the construction of a machine-readable formal model that could reason about objectives and deliver the necessary propositional logical operations to address them. Thereby encouraging the prospect of an incorporation of these objectives and respective operations in an eventual integrated model.

In consideration of the affects of the aforementioned contribution to the wider field of privacy, we find that this contribution readily exposes the translation of privacy into actionable privacy requirements that can be tackled using first-order predicate logic so that we present formal objectives that tackle the many areas of "privacy" described in the Solove taxonomy. A translation into a more logical form enables the elusive concept of privacy to be expressed in a form that encourages multidisciplinary use.

We hope that from this, others in the field may extrapolate and perform similar translations dependent on their area of focus. With respect, more generally, to the field of privacy, we have a means of taking an often-claimed intangible concept into a more discrete manner that has been translated into formal methods and, therefore, can aid those working in technical fields. Hence, further work can include building workable architectures and systems that may counter some of these rising privacy issues.

9.2.2 Contribution 2

Our second contribution worked on the formation of a basis that would allow our intellectual exercise in privacy preservation to have a stepping point to realism. This is due to the decentralisation provided by the Solid proposal, which showed great potential for privacy incorporation in the early stages of planning and modelling. This proposal led with personal data stores that permitted flexible fine-tuned access control as a result of the access control inheritance via means of the access control resources.

In this contribution, we sought to draft a formal model through the use of the outlined specifications, namely that of the 'Protocol' specification and the 'Web Access Control' specification. Through this model, we strove to attain a comprehensive understanding of the workings of the various components within the system, hopeful to be successful in demonstrating the applicability of the Solid proposal as a formal foundation that could be extended to include the model for CBAC and the privacy considerations concerned with privacy in the face of metadata exploitation.

Furthermore, as society demands more from social networking sites, particularly in light of the scepticism of social media conglomerates and data ownership, we aim to present the stepping stones for how users may regain control through the Solid model. Our contribution may be viewed as a formal demonstration of how such a framework may be modelled. Such that, in our work on presenting the Solid model, we stand in support of the creators for a more equitable online space, which necessitates that ownership and access control are placed in the hands of individuals.

9.2.3 Contribution 3

Our third contribution relied heavily on the framework of the meta-model M for the novel, basal form of access control, described as Category-Based Access Control. In this space,

we aimed to present a preliminary model of the Category Based Access Control framework in Z schema language. In doing so, we intended to form a basis that would allow us to readily reason about the likely extensions resulting from considerations of the dynamic, contextual nature of privacy relating to metadata. We note that the choice of CBAC was largely precipitated by its flexibility concerning autonomous changes executed dependent on the state of the system, which will be invaluable in consideration of privacy harms such as aggregation and surveillance. Through our modeling of the meta-model and our consequent formal model interpretation — we were able to build upon the tenets described so we might detail the corresponding antithesis sets to the permissive sets. This was achieved through the addition of forbiddances. These changes allowed us to envision how situational cases may impart a change of permission or introduction of a forbiddance.

Furthermore, this contribution interrogates the underpinnings behind access control in a manner that allows for reproducibility given a wide range of problems and solutions, as opposed to designing novel access control frameworks each time for ad-hoc situations. The foundation laid acts as a basis upon which specialities are naturally accommodated. In the use of the Category-Based Access Control framework, we have demonstrated how privacy requirements can easily be incorporated in a manner that allows autonomous updates to permissions without user intervention but based on properties of data collection, location and personnel groupings.

9.2.4 Contribution 4

Our fourth and final contribution worked to resolve the following subsidiary research question:

RQ4: Is it possible to build a formal model that supports our privacy requirements through extension of an integrated Solid and Category-Based Access Control model? As part of our

resolution of this question, we worked to ensure that the privacy objectives that had been described in Contribution 1 had been carried forward while simultaneously making use of the Category-Based Access Control from Contribution 3 as well as the decentralised pod storage architectures from the Solid Proposal shown in Contribution 2. This final contribution came as an amalgamation of the previous contributions as a way to successfully demonstrate the formal model solution to our overarching research question.

We began our work through identification of decentralisation as a viable method for building our solution. Decentralisation has proved to be imperative in the pursuit of privacy due to the control bestowed on users as opposed to data-collecting parties. Hence, in response to this need for decentralisation, we chose the Solid proposal due to the ease of the proposed storage solutions that allowed users to group their data as they saw fit. By approximating the Solid proposal, we were able to provide a model upon which we could utilise a more autonomous form of access control.

In the space of access control, we discovered a viable means of connecting the Solid proposal structure to that of the CBAC framework via access control resources being linked to differing categories with different access rights. Then, through utilising our privacy objectives and the consequent functions from the resulting privacy formal model, we were able to integrate this with the CBAC framework. As a result, we built a decentralised reactive access control structure that could reason with the volatile nature of unintentional sensitive metadata exposures, with references to privacy harms such as surveillance and aggregation.

Thereby enabling users to control their data and have greater control on access control rights as their needs adapted based on the current collection landscape. In this space, we have considered the intricacies of metadata to aid in those building future privacy-protecting social media platforms to ensure that the pitfalls of the current social media

platforms relating to surplus data and passive exploitation do not reoccur. As part of this contribution, we have layered concepts of ownership with privacy to prescribe privacy defences ahead of potential privacy violations. The primary aim of this work was to demonstrate how such a system would function and, thereby, encourage further work focused on the machine learning and predictive aspects to discover whether such pre-emptive defence objectives are viable while maintaining reasonable computational costs.

9.3 Limitations and Drawbacks

Whilst we noted the overarching strengths of our work, it is imperative that we evaluate the limitations of our research and the manner in which it falls short in addressing wider issues.

In this space, we note an obvious drawback in that privacy is an ever-evolving concept with a lack of a clear, universal and concrete definition, so our attempts to tackle these privacy issues are constrained by our understanding of privacy. As privacy is an entity that remains largely subjective, therefore, in our work, we could only approach it from our analysis and refinement of the research undertaken in the area, which is unlikely to be all inclusive.

In terms of legal parties, the translation of our insights with relation to metadata may be difficult to consolidate into actionable legal frameworks as it may be difficult to ascertain elements such as the chain of data sharing or the use of the data collected.

As well as these issues, we note that our work is heavily reliant on the introduction of legislation, for example, in the aggregative limit. Whereupon, one would require a degree of trust in external third parties as it relates to, for example, the deletion of metadata articles once a threshold has been reached.

Due to the use of A.I in data analysis and predictions, it is expected that we would

need similar data algorithms that can analyse the data presented as for operations such as *adjustSkewScore*, it would require a more sophisticated data algorithm to interpret the data and estimate the degree to which a resource skewed towards the discovery of identified traits.

Furthermore, due to the nature of inferences, one would likely have to employ predictive analytics, which requires an abundance of data to train and may impact the runtime of any eventual prototype system. Thus, it may prove difficult to take this theoretical approach and apply it in practice.

Finally, as we have previously alluded to in this dissertation, our research is heavily reliant on the use of a decentralisation platform, such as Solid. Without such decentralisation, one cannot begin to reason about data ownership and thus access control of user data in a manner that favours user privacy. Therefore, until the web progresses to a more decentralised architecture, as hoped in Web 3.0, attempts to mitigate metadata-driven privacy violations may well be halted.

9.4 Future Work

We envision that future work would include building a prototype for this data management system, wherein the prototype could interact with the existing Solid architectures. This prototype would be trialed to investigate its strengths in handling sophisticated data algorithms that can perform real-time analysis of the data being stored.

Eventually, we would hope this prototype will carve a way to aid and influence the translation of privacy harms to actionable machine-readable algorithms and functions. So that in the space of researchers, others would be able to extend and modify any eventual privacy model or system to include changes as the privacy landscape changes.

Furthermore, as society begins to further regard decentralisation as a method of grant-

ing users with greater autonomy in a world where privacy discussions are rising, we hope solutions will seriously evaluate the risks faced by passive forms of data collection. In response, researchers should be encouraged to support proposals for a decentralised web and further, we can then encourage limitations to be put on this passive data generation and collection.

We note that through use of the Solove's taxonomy — we hope to aid in informing legislative parties of a portion of the steps required to tackle a rising issue as our modern society reasons with the privacy implications that are rendered through unencumbered passive data collection and advanced, often A.I.-driven, data analysis and consequent predictions.

9.5 Concluding Thoughts

In this dissertation, the research tackled the often overlooked issue of personal data leakage through online passive data. To this end, we sought to provide a means for individuals to have the means to combat this leakage and retain control of the information others are able to attain about them. In fact, the state of data collection has meant that metadata collection may well be classed as exploitative when contrasted with the globally recognised human rights for privacy, due in part to practices such as dataveillance and predictive inferences.

In addressing some of these issues, success has been found through demonstrating how an amalgamation of differing models can be crafted to aid in the mitigation of metadata-driven privacy exploitation. To this end, we can be satisfied that we have contributed to research efforts in tackling potential personal data leakage from multiple vantage points.

Chapter 10

Bibliography

[1] L Brankovic and V Estivill-Castro. Privacy Issues in Knowledge Discovery and Data Mining. In *Australian institute of Computer Ethics Conference*, pages 89–99. Citeseer, 1999.

[2] N Singer. You For Sale: Mapping, And Sharing, The Consumer Genome. *New York Times*, pages 1–7, 2012.

[3] N Foshay, A Mukherjee, and A Taylor. Does Data Warehouse End-User Metadata Add Value? *Communications of the ACM*, 50(11):70–77, 2007.

[4] "We Kill People Based On Metadata," Admits Former CIA/NSA Boss. *The New American (Belmont, Mass.)*, 30(11):8, 2014.

[5] Widad Elouataoui, Imane El Alaoui, and Youssef Gahi. Metadata quality in the era of big data and unstructured content. In *Advances in Information, Communication and Cybersecurity: Proceedings of ICI2C'21*, pages 110–121. Springer, 2022.

[6] Sathana Venkadasubbiah, D Yuvaraj, Subair Ali, and Mohamed Uvaze Ahamed Ayoobkhan. Data footprinting in big data. In *Big Data Analytics and Computational Intelligence for Cybersecurity*, pages 203–218. Springer, 2022.

[7] O Tene and J Polonetsky. Privacy In The Age Of Big Data: A Time For Big Decisions. *Stan. L. Rev. Online*, 64:63–69, 2011.

[8] I Rubinstein. Big Data: The End Of Privacy Or A New Beginning? *International Data Privacy Law (2013 Forthcoming)*, 3(2):74–87, 2012.

[9] C Conley. Non-content is Not Non-sensitive: Moving Beyond the Content/Non-content Distinction. *Santa Clara L. Rev.*, 54(4):820–842, 2014.

[10] C Johnson. Project Gaydar. *The Boston Globe*, 20, 2009.

[11] UN General Assembly. Universal Declaration Of Human Rights. *UN General Assembly*, 302(2), 1948.

[12] M Valkanova. The New Chapter Regulation (EU) 2016/679 Of The European Parliament And Of The Council From April 27, 2016 On The Protection Of Natural Persons With Regard To The Processing Of Personal Information And The Free Movement Of Such Information As Well As Some New Legal Concepts, Determining The Patents Personal Information. In *Varna Medical Forum*, volume 6, pages 147–151, 2017.

[13] J Hinds and A Joinson. What Demographic Attributes Do Our Digital Footprints Reveal? A Systematic Review. *PloS one*, 13(11):1–40, 2018.

[14] S.D Gosling, S J Ko, T Mannarelli, and M Morris. A Room With A Cue: Personality Judgments Based On Offices And Bedrooms. *Journal of personality and social psychology*, 82(3):379–398, 2002.

[15] A Acquisti and C Fong. An Experiment In Hiring Discrimination Via Online Social Networks. *Management Science*, 66(3):1005–1024, 2020.

[16] S Ozalp. Unlawful Data Access And Abuse Of Metadata For Mass Persecution Of Dissidents In Turkey: The Bylock Case. pages 117–134, 2019.

[17] G Surblytė-Namavičienė. *Competition And Regulation In The Data Economy : Does Artificial Intelligence Demand A New Balance? [electronic resource]*. Elgaronline. Northampton, 2020.

[18] P Selvaraj and P Williams .A. Data: The New Currency of the Digital World and The Race Among the Nations to Protect Data. *International Journal of Recent Technology and Engineering*, 8(256), 2019.

[19] Meglena Kuneva. European Consumer Commissioner,"Keynote Speech,"in Roundtable on Online Data Collection, Targeting and Profiling.". 2009.

[20] A Sambra, E Mansour, S Hawke, M Zereba, N Greco, A Ghanem, D Zagidulin, A Aboulnaga, and T Berners-Lee. Solid: A Platform For Decentralized Social Applications Based On Linked Data. Technical report, Technical Report, MIT CSAIL & Qatar Computing Research Institute, 2016.

[21] Massachusetts Institute of Technology. Solid. `https://solid.mit.edu/`, Jun 2018.

[22] M Tschantz and J Wing. Formal Methods For Privacy. In *FM 2009: Formal Methods: Second World Congress, Eindhoven, The Netherlands, November 2-6, 2009. Proceedings 2*, pages 1–15. Springer, 2009.

[23] Nelly Bencomo, Jordi Cabot, Marsha Chechik, Betty HC Cheng, Benoit Combemale, Andrzej Wasowski, and Steffen Zschaler. Abstraction engineering. *arXiv preprint arXiv:2408.14074*, 2024.

[24] S Barker. The Next 700 Access Control Models or a Unifying Meta-model? In *Proceedings of the 14th ACM symposium on Access control models and technologies*, pages 187–196, 2009.

[25] D J Solove. A Taxonomy of Privacy. *University Of Pennsylvania Law Review*, 154:477–560, 2005.

[26] S Sagiroglu and D Sinanc. Big Data: A Review. In *2013 International Conference on Collaboration Technologies and Systems (CTS)*, pages 42–47, 2013.

[27] J Abawajy. Comprehensive Analysis of Big Data Variety Landscape. *International Journal of Parallel, Emergent and Distributed Systems*, 30(1):5–14, 2015.

[28] R Casado and M Younas. Emerging Trends and Technologies in Big Data Processing. *Concurrency and Computation: Practice and Experience*, 27(8):2078–2091, 2015.

[29] Min Chen, Yixue Hao, Kai Hwang, Lu Wang, and Lin Wang. Disease Prediction by Machine Learning over Big Data from healthcare communities. *Ieee Access*, 5:8869–8879, 2017.

[30] W Raghupathi and V Raghupathi. Big Data Analytics in Healthcare: Promise and Potential. *Health Information Science and Systems*, 2(1):1–10, 2014.

[31] M Farboodi and L Veldkamp. A Growth Model of The Data Economy. Technical report, National Bureau of Economic Research, 2021.

[32] V Thirani and A Gupta. The Value of Data. In *World Economic Forum*, volume 22, 2017.

[33] Anita L Allen. Protecting One's Own Privacy in a Big Data Economy. *Harv. L. Rev. F.*, 130:71–78, 2016.

[34] V Gadepally, B Hancock, B Kaiser, J Kepner, P Michaleas, M Varia, and A Yerukhimovich. Improving the Veracity of Homeland Security Big Data Through Computing on Masked Data. In *2015 IEEE International Symposium on Technologies for Homeland Security, Waltham, Mass*, pages 14–16, 2015.

[35] C c Porter. De-identified Data and Third Party Data Mining: The Risk of Re-identification of Personal Information. *Shidler JL Com. & Tech.*, 5:1, 2008.

[36] H Metwalley, S Traverso, M Mellia, S Miskovic, and M Baldi. The Online Tracking Horde: A View from Passive Measurements . In *International Workshop on Traffic Monitoring and Analysis*, pages 111–125. Springer, 2015.

[37] D Cecez-Kecmanovic. The Resistible Rise of The Digital Surveillance Economy: A Call For Action. *Journal of Information Technology*, 34(1):81–83, 2019.

[38] Duncan Hodges, Sadie Creese, and Michael Goldsmith. A Model for Identity in the Cyber and Natural Universes. In *2012 European Intelligence and Security Informatics Conference*, pages 115–122. IEEE, 2012.

[39] Syed Sardar Muhammad, Bidit Lal Dey, Sharifah Faridah Syed Alwi, Muhammad Mustafa Kamal, and Yousra Asaad. Consumers' willingness to share digital footprints on social media: the role of affective trust. *Information Technology & People*, 36(2):595–625, 2023.

[40] J Van Dijck. Datafication, Dataism and Dataveillance: Big Data Between Scientific Paradigm and Ideology. *Surveillance & society*, 12(2):197–208, 2014.

[41] R Clarke. The Digital Persona and Its Application to Data Surveillance. *The Information Society*, 10(2):77–92, 1994.

[42] R Binns, U Lyngs, M Van Kleek, J Zhao, T Libert, and N Shadbolt. Third Party Tracking in the Mobile Ecosystem. In *Proceedings of the 10th ACM Conference on Web Science*, pages 23–31, 2018.

[43] N Twomey, T Diethe, X Fafoutis, A Elsts, R McConville, P Flach, and I Craddock. A Comprehensive Study of Activity Recognition sUing Accelerometers. In *Informatics*, volume 5, page 27. Multidisciplinary Digital Publishing Institute, 2018.

[44] Y Dong, Y Yang, J Tang, Y Yang, and N V Chawla. Inferring User Demographics and Social Strategies in Mobile Social Networks. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 15–24, 2014.

[45] Eric Schmidt. Washington Ideas Forum. media coommunication, 2010.

[46] A M Husain and Y Xu. Broadcast Content View nAalysis based on ambient audio recording, sep 2018. US Patent 10,075,767.

[47] S Wachter and B Mittelstadt. A right to reasonable inferences: Re-thinking data protection law in the age of big data and AI. *Colum. Bus. L. Rev.*, page 494, 2019.

[48] A R Miller. The Assault on Privacy, 1970.

[49] J. Roy. Polis and Oikos in Classical Athens. *Greece and Rome*, 46(1):1–18, 1999.

[50] J Locke and W. von Leyden. *Essays on the Law of Nature*. Clarendon Press, Oxford, 1988.

[51] S D Warren and L D Brandeis. Right to Privacy. *Harv. L. Rev.*, 4:193, 1890.

[52] H S Hadley. Right to Privacy. *NWL Rev.*, 3:1, 1895.

[53] Vernon Cole. The Right to Privacy. In *Historical Theses and Dissertations Collection*. Cornell Law Library, 1892.

[54] W Larremore. Law of Privacy. *Colum. L. Rev.*, 12:694, 1912.

[55] NY Civ Rights L § 50 (2014).

[56] Privacy International. What is Privacy? `https://www.privacyinternational.org/explainer/56/what-privacy`, October 2017.

[57] M Rotenberg, J Scott, and J Horwitz. *Privacy in the Modern Age: The Search for Solutions*. New Press, The, 2015.

[58] Electronic Communication Privacy Act of 1986. *Pub. L*, pages 99–508, 1986.

[59] I Altman. A Conceptual Analysis. *Environment and behavior*, 8(1):7–29, 1976.

[60] J RC Nurse, A Erola, T Gibson-Robinson, M Goldsmith, and S Creese. Analytics for Characterising and Measuring the Naturalness of Online Personae. *Security Informatics*, 5(1):1–14, 2016.

[61] R Halperin and Y Dror. Information privacy and the digital generation gap: An exploratory study. *Journal of Information Privacy and Security*, 12(4):166–180, 2016.

[62] Andy Clark. Extending the predictive mind. *Australasian Journal of Philosophy*, 102(1):119–130, 2024.

[63] Shaun Gallagher. Technology and the extended mind. In *Technology ethics*, pages 52–60. Routledge, 2023.

[64] Joanne Hinds and Adam N Joinson. Digital data and personality: A systematic review and meta-analysis of human perception and computer prediction. *Psychological Bulletin*, 2024.

[65] Jon-Arild Johannessen. *Automation, capitalism and the end of the middle class*. Routledge, 2019.

[66] Yanis Varoufakis. *Technofeudalism: What killed capitalism*. Melville House, 2024.

[67] Evgeny Morozov. Critique of techno-feudal reason. *New Left Review*, (133):89–126, 2022.

[68] S Zuboff. Big Other: Surveillance Capitalism and The Prospects of an Information Civilization. *Journal of Information Technology*, 30(1):75–89, 2015.

[69] P A Norberg, D R Horne, and D A Horne. The Privacy Paradox: Personal Information Disclosure Intentions Versus Behaviors. *Journal of consumer affairs*, 41(1):100–126, 2007.

[70] Susanne Barth, Menno DT de Jong, Marianne Junger, Pieter H Hartel, and Janina C Roppelt. Putting the privacy paradox to the test: Online privacy and security behaviors among users with technical knowledge, privacy awareness, and financial resources. *Telematics and informatics*, 41:55–69, 2019.

[71] Ari Ezra Waldman. Cognitive biases, dark patterns, and the 'privacy paradox'. *Current opinion in psychology*, 31:105–109, 2020.

[72] H Choi, J Park, and Y Jung. The role of privacy fatigue in online privacy behavior. *Computers in Human Behavior*, 81:42–51, 2018.

[73] C Bösch, B Erb, F Kargl, H Kopp, and S Pfattheicher. Tales From The Dark Side: Privacy Dark Strategies and Privacy Dark Patterns. *Proc. Priv. Enhancing Technol.*, 2016(4):237–254, 2016.

[74] A Mathur, G Acar, M J Friedman, E Lucherini, J Mayer, M Chetty, and A Narayanan. Dark Patterns at Scale: Findings From a Crawl Of 11k Shopping Websites. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW):1–32, 2019.

[75] A M McDonald and L F Cranor. The Cost of Reading Privacy Policies. *Isjlp*, 4(3):543–568, 2008.

[76] J Luguri and L J Strahilevitz. Shining a Light on Dark Patterns. *Journal of Legal Analysis*, 13(1):43–109, 03 2021.

[77] A Arakerimath and P K Gupta. Digital Footprint: Pros, Cons, and Future. *International Journal of Latest Technology in Engineering*, 4(10):52–56, 2015.

[78] G Zyskind, O Nathan, and A Pentland. Enigma: Decentralized Computation Platform With Guaranteed Privacy. *arXiv preprint arXiv:1506.03471*, pages 1–14, 2015.

[79] J P Barlow. A Declaration of the Independence of Cyberspace. *Duke L. & Tech. Rev.*, 18:5, 2019.

[80] R S Sandhu and P Samarati. Access Control: Principle and Practice. *IEEE Communications Magazine*, 32(9):40–48, 1994.

[81] A Ali and M Fernández. Hybrid Enforcement of Category-Based Access Control. In S Mauw and C D Jensen, editors, *Security and Trust Management*, pages 178–182. Springer International Publishing, 2014.

[82] M Fernández, M Kantarcioglu, and B Thuraisingham. A Framework for Secure Data Collection and Management for Internet of Things. In *Proceedings of the 2nd Annual Industrial Control System Security Workshop*, pages 30–37, 2016.

[83] C Bertolissi and M Fernández. Category-Based Authorisation Models: Operational Semantics and Expressive Power. In *Proceedings of the Second international conference on Engineering Secure Software and Systems*, pages 140–156. Springer, 2010.

[84] C Bertolissi, M Fernández, and B Thuraisingham. Admin-CBAC: An Administration Model for Category-Based Access Control. In *Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy*, CODASPY '20, page 73–84, New York, NY, USA, 2020. Association for Computing Machinery.

[85] S Jha, N Li, M Tripunitara, Q Wang, and W Winsborough. Towards Formal Verification of Role-Based Access Control Policies. *IEEE Transactions on Dependable and Secure Computing*, 5(4):242–255, 2008.

[86] Z Manna and A Pnueli. *The Temporal Logic of Reactive and Concurrent Systems: Specification*, volume 1. Springer Science & Business Media, 2012.

[87] V C Hu and D R Kuhn. General Methods for Access Control Policy Verification (application paper). In *2016 IEEE 17th International Conference on Information Reuse and Integration (IRI)*, pages 315–323. IEEE, 2016.

[88] P Samarati. Flexible Authorization Framework (FAF). pages 486–489, 2011.

[89] I Ray and M Toahchoodee. A Spatio-Temporal Role-Based Access Control Model. In *IFIP Annual Conference on Data and Applications Security and Privacy*, pages 211–226. Springer, 2007.

[90] M J Covington and M R Sastry. A Contextual Attribute-Based Access Control Model. In *OTM Confederated International Conferences "On the Move to Meaningful Internet Systems"*, pages 1996–2006. Springer, 2006.

[91] J Park, D Nguyen, and R Sandhu. A Provenance-Based Access Control Model. In *2012 Tenth Annual International Conference on Privacy, Security and Trust*, pages 137–144. IEEE, 2012.

[92] Simone Fischer-Hbner and Stefan Berthold. Privacy-enhancing technologies. In *Computer and information security Handbook*, pages 759–778. Elsevier, 2017.

[93] M J Spivey and J-R Abrial. *The Z Notation*, volume 29. Prentice Hall Hemel Hempstead, 1992.

[94] Alan Abe and Andrew Simpson. Formal models for privacy. In *EDBT/ICDT Workshops*. Citeseer, 2016.

[95] Daniel Plagge and Michael Leuschel. Validating z specifications using the prob animator and model checker. In *International Conference on Integrated Formal Methods*, pages 480–500. Springer, 2007.

[96] D F. Ferraiolo, R Sandhu, Se Gavrila, R D. Kuhn, and R Chandramouli. Proposed NIST Standard for Role-Based Access Control. *ACM Trans. Inf. Syst. Secur.*, 4(3):224–274, aug 2001.

[97] D Ferraiolo and V Atluri. A Meta Model for Access Control: Why Is It Needed and Is It Even Possible to Achieve? In *Proceedings of the 13th ACM symposium on Access control models and technologies*, pages 153–154, 2008.

[98] K Nienhuis, A Joannou, T Bauereiss, A Fox, M Roe, B Campbell, M Naylor, Robert M Norton, S.W Moore, P.G Neumann, et al. Rigorous Engineering for Hardware Security: Formal Modelling and Proof in the CHERI Design and Implementation Process. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 1003–1020. IEEE, 2020.

[99] Michael Huth. Formal Methods and Access Control, 2011.

[100] L Sweeney. Discrimination in Online Ad Delivery. *Communications of the ACM*, 56(5):44–54, 2013.

[101] C G Brown-Johnson, L J England, S A Glantz, and P M Ling. Tobacco Industry Marketing to Low Socioeconomic Status Women in the USA. *Tobacco control*, 23(e2):139–146, 2014.

[102] C Bizer, T Heath, and T Berners-Lee. Linked Data: Principles and State of the Art. In *World wide web conference*, volume 1, page 40. Citeseer, 2008.

[103] T Berners-Lee, J Hendler, and O Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, 2001.

[104] E Miller. An Introduction to the Resource Description Framework. *Bulletin of the American Society for Information Science and Technology*, 25(1):15–19, 1998.

[105] C Chhetri and V Motti. "I mute my echo when I talk politics": Connecting Smart Home Device Users' Concerns to Privacy Harms Taxonomy. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 66, pages 2083–2087. SAGE Publications Sage CA: Los Angeles, CA, 2022.

[106] A K Massey and A I Antón. A Requirements-Based Comparison of Privacy Taxonomies. In *2008 Requirements Engineering and Law*, pages 1–5. IEEE, 2008.

[107] Y Shen and S Pearson. Privacy Enhancing Technologies: A Review. *Hewlet Packard Development Company.*, 2011.

[108] RR Rowlingson. Marrying Privacy Law to Information Security. *Computer Fraud & Security*, 2006(8):4–6, 2006.

[109] A Bartow. A Feeling of Unease About Privacy Law. *U. Pa. L. Rev. PENNumbra*, 155(52):52–62, 2006.

[110] R Calo. The Boundaries of Privacy Harm. *Indiana Law Journal*, 86:1131–1158, 2011.

[111] D K Citron and L M Henry. Visionary Pragmatism and The Value of Privacy in The Twenty-First Century, 2010.

[112] K Crawford and J Schultz. Big Data And Due Process: Toward A Framework To Redress Predictive Privacy Harms. *BCL Rev.*, 55(1):93–128, 2014.

[113] F Johns and D Joyce. Beyond Privacy. Is Prevailing Legal Debate Too Analog for a Digital Age? *Human Rights Defender*, 23(3):24–26, 2014.

[114] B Schneier. Metadata= Surveillance. *IEEE Security & Privacy*, 12(2):84–84, 2014.

[115] D Lyon. Surveillance, Snowden, and Big Data: Capacities, Consequences, Critique. *Big data & society*, 1(2):1–13, 2014.

[116] N N Loideain. EU Law and Mass Internet Metadata Surveillance in the Post-Snowden Era. *Media and Communication*, 3(2):53–62, 2015.

[117] A Vitalis and A Mattelart. *Le Profilage des Populations: du Livret Ouvrier au Cybercontrôle*. La Découverte, 2014.

[118] B Perez, M Musolesi, and G Stringhini. You Are Your Metadata: Identification and Obfuscation of Social Media Users Using Metadata Information. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 12, 2018.

[119] R H Weber. Internet of Things: Privacy Issues Revisited. *Computer Law & Security Review*, 31(5):618–627, 2015.

[120] D Cecez-Kecmanovic. The Resistible Rise of The Digital Surveillance Economy: A Call For Action. *Journal of Information Technology*, 34(1):81–83, 2019.

[121] R Clarke. Risks Inherent in the Digital Surveillance Economy: A Research Agenda. *Journal of Information Technology*, 34(1):59–80, 2019.

[122] *Kyllo vs United States*, volume 533. 2001.

[123] C Conley. Metadata: Piecing Together a Privacy Solution. *Available at SSRN 2573962*, 2014.

[124] F Ferra, I Wagner, E Boiten, L Hadlington, I Psychoula, and R Snape. Challenges in Assessing Privacy Impact: Tales from the Front Lines. *Security and Privacy*, 3(2):e101, 2020.

[125] N Nikiforakis, A Kapravelos, W Joosen, C Kruegel, F Piessens, and G Vigna. Cookieless Monster: Exploring The Ecosystem of Web-Based Device Fingerprinting. In *2013 IEEE Symposium on Security and Privacy*, pages 541–555. IEEE, 2013.

[126] *Board of Education v. Earls*, volume 536. 2002.

[127] J Kröger. Unexpected Inferences rFom Sensor Data: A Hidden Privacy Threat in the Internet of Things. In *1st IFIP International Internet of Things Conference (IFIPIoT)*, pages 147–159. Springer International Publishing, 2019.

[128] D A Hollinger. The Disciplines and the Identity Debates, 1970-1995. *Daedalus*, 126(1):333–351, 1997.

[129] R F Baumeister and M Muraven. Identity as Adaptation to Social, Cultural, and Historical Context. *Journal of adolescence*, 19(5):405–416, 1996.

[130] D Cocking. Plural Selves and Relational Identity: Intimacy and Privacy Online. *Information technology and moral philosophy*, pages 123–141, 2008.

[131] M Eviette and AC Simpson. Towards Models for Privacy Preservation in the Face of Metadata Exploitation. IFIP Advances in Information and Communication Technology, pages 247–264. Springer, 2021.

[132] J-E Mai. Big Data Privacy: The Datafication of Personal Information. *The Information Society*, 32(3):192–199, 2016.

[133] J Hamm. Minimax Filter: Learning to Preserve Privacy From Inference Attacks. *The Journal of Machine Learning Research*, 18(1):4704–4734, 2017.

[134] Y-A De Montjoye, Erez Shmueli, S S Wang, and A S Pentland. openpds: Protecting the Privacy of Metadata Through Safeanswers. *PloS one*, 9(7):e98790, 2014.

[135] Bernadette Kamleitner and Vince Mitchell. Your data is my data: a framework for addressing interdependent privacy infringements. *Journal of Public Policy & Marketing*, 38(4):433–450, 2019.

[136] B Potter, D Till, and J Sinclair. *An Introduction to Formal Specification and Z*. Prentice Hall PTR, 1996.

[137] P Fradet, D Le Métayer, and M Périn. Consistency Checking for Multiple View Software Architectures. *ACM SIGSOFT Software Engineering Notes*, 24(6):410–428, 1999.

[138] P J Gibson, I Ait-Sadoune, and M Pantel. Semantic Heterogeneity in the Formal Development of Complex Systems: An Introduction. In *International Symposium on Leveraging Applications of Formal Methods*, pages 321–324. Springer, 2016.

[139] D Al-Azizy, D Millard, N Shadbolt, and K O'Hara. *Deanonymisation in Linked Data: A Research Roadmap*. 2014.

[140] A E Marwick. Online identity. *A Companion to New Media Dynamics*, pages 355–364, 2013.

[141] P B Brandtzaeg and M Lüders. Time ocllapse in social media: Extending the context collapse. *Social Media+ Society*, 4(1):1–10, 2018.

[142] A Franz and A Benlian. Exploring Interdependent Privacy– Empirical Insights Into Users' Protection Of Others' Privacy on Online Platforms. *Electronic Markets*, 32(4):2293–2309, 2022.

Appendix A

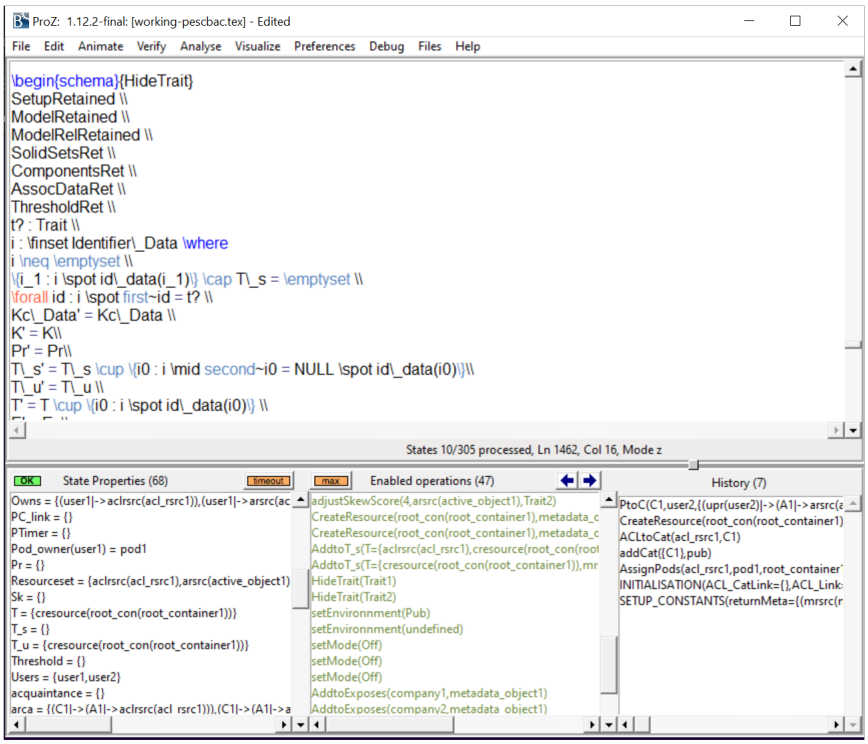
Validation Examples

The following appendix chapter works to provide screenshots of function in ProB from the various sub-areas of privacy identified.

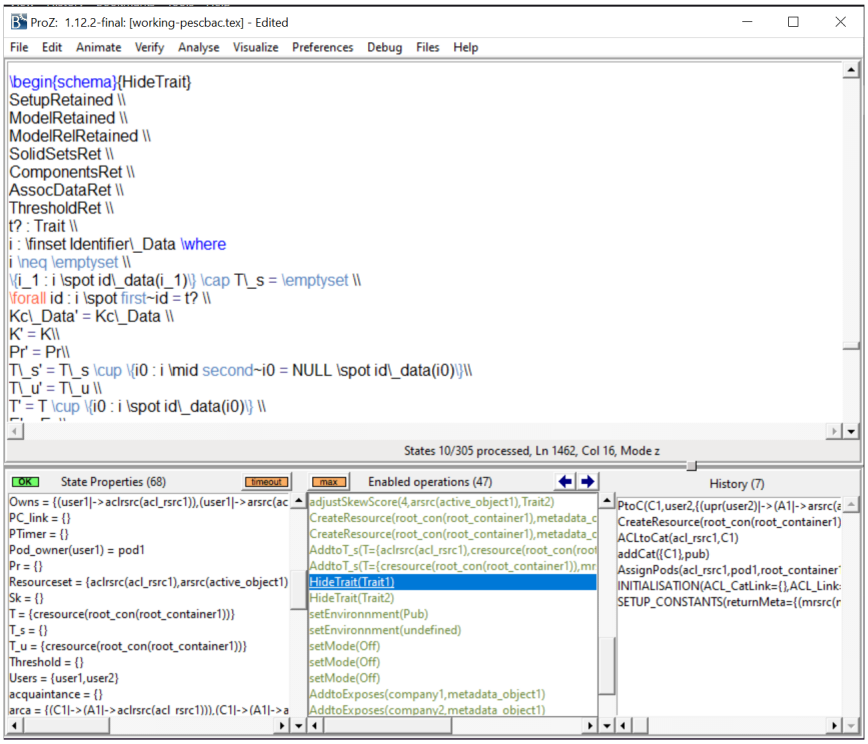
A.1 Identification

In this realm we are trying to limit the scope of known identifiable traits, therefore we may present the workings of the schema *HideTrait* in ProB animation.

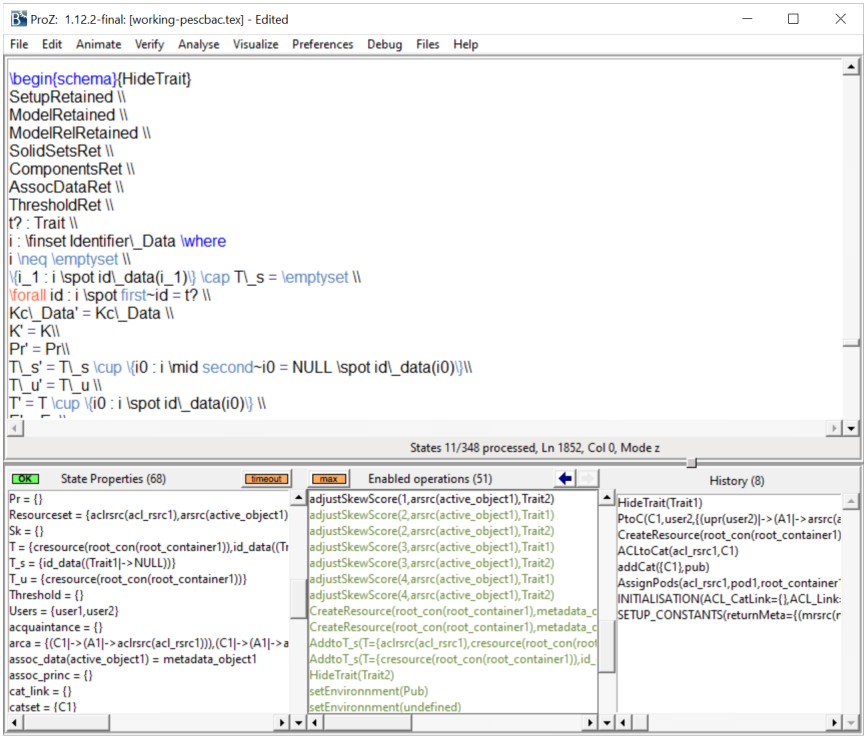
First we see the description of the schema in the following screenshot.



Then, we see us select the operation in the list of enabled operations, so that we are hiding *Trait*1.



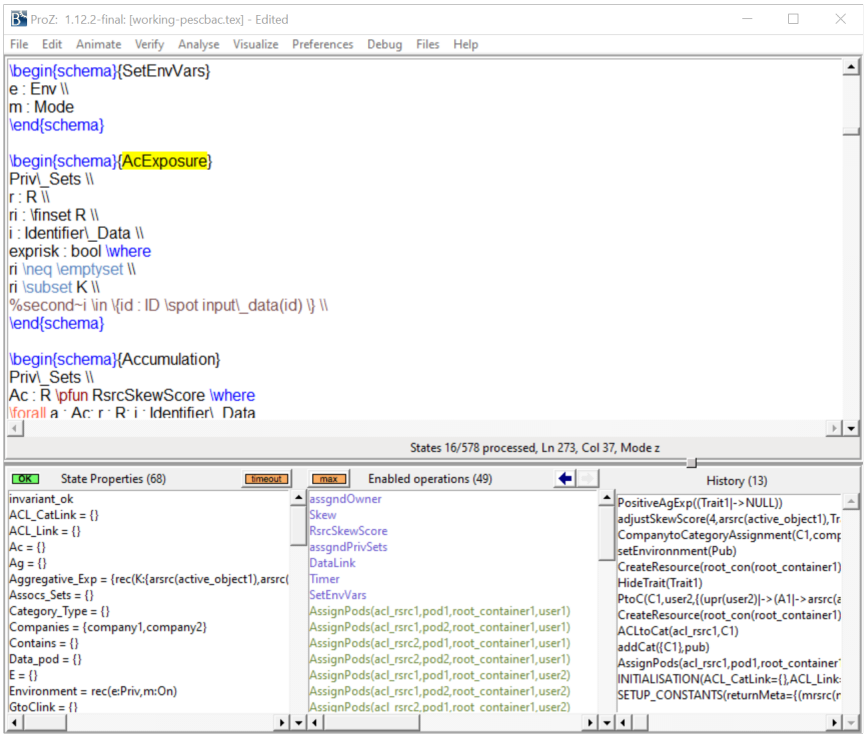
Finally, we may observe the addition that has been made to the set T_s as a result of the *HideTrait* schema operation.



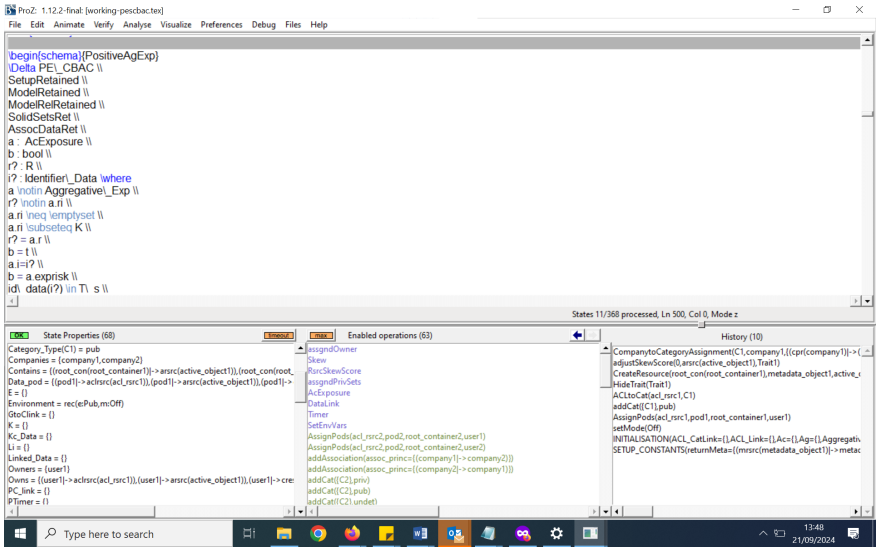
The addition to the protected set T_s shall effectively prevent this trait from being discovered by restricting access to data items that lead to an inference of the trait, determined by the skew score and aggregative exposure operations.

A.2 Aggregation

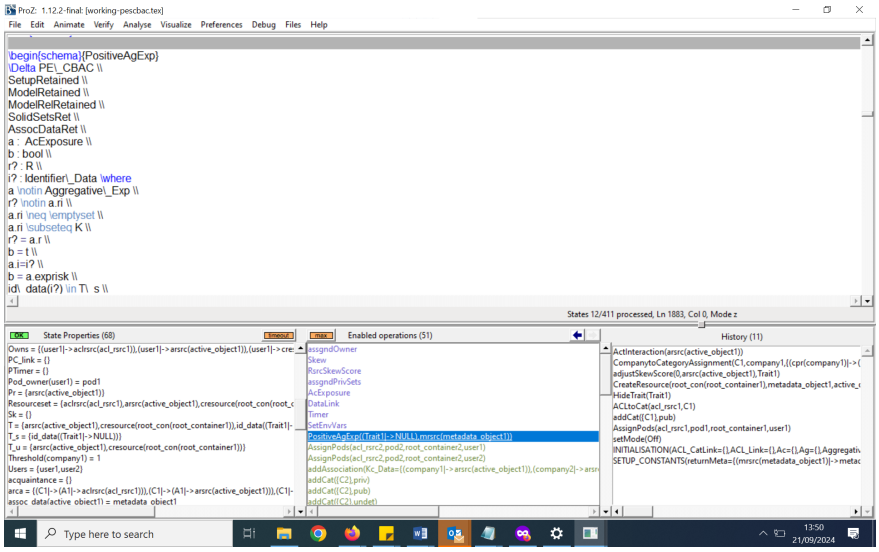
In this area, we look at the functioning of following schema *AcExposure* seeking to limit the accumulative exposure.



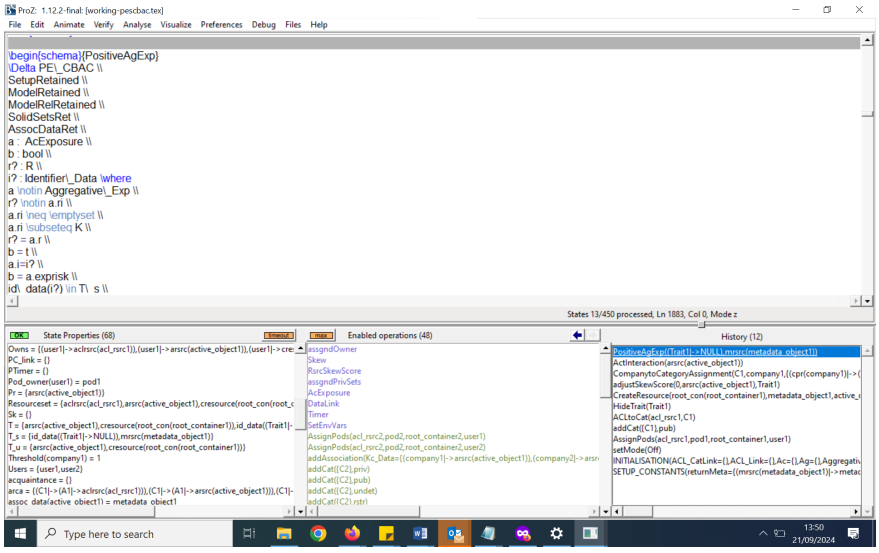
Following we remind ourselves of the *PositiveAgExposure* schema which serves to evaluate a resource that leads to the positive aggregative exposure of a trait found in T_s



Upon creation of two articles of data object for the sake of the example, we have decided the active object was collected by the company, as indicated by $active\_object1$ in Pr and the schema call $ActInteraction(active\_object1)$ found in the History of calls. Following, we have decided that the accompanying metadata in combination with said active object would reveal the trait $Trait1$ found in T_s hence with the call to the schema $PositiveAgExp$ we make this a reality.



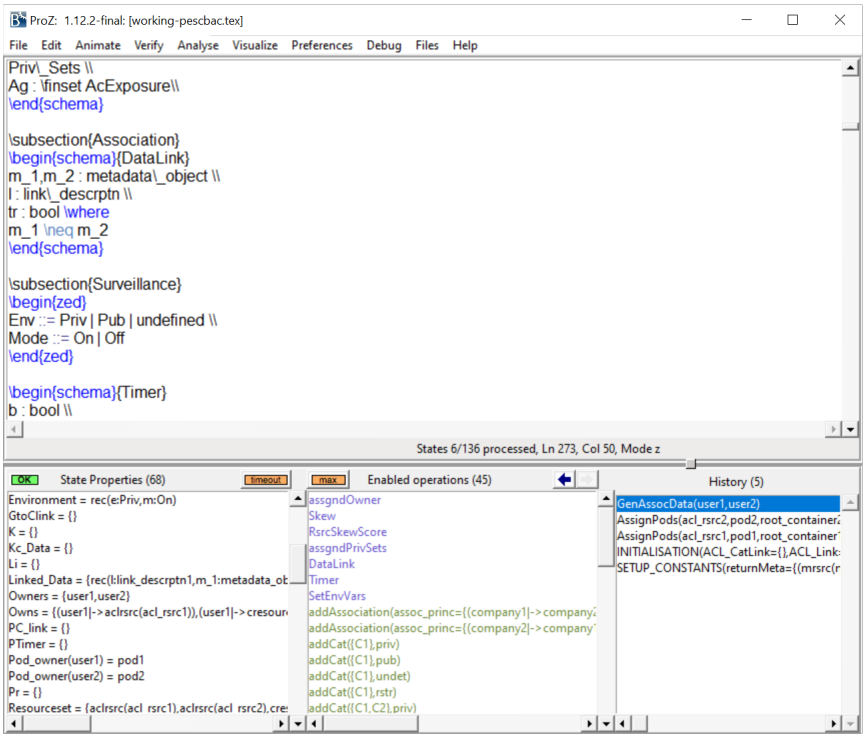
Following said call, we may witness that the data object *metadata_object*1 has been moved to the set T_s to prevent its collection. And by preventing said collection, we have successfully been able to stop the aggregative exposure of the trait *Trait*1.



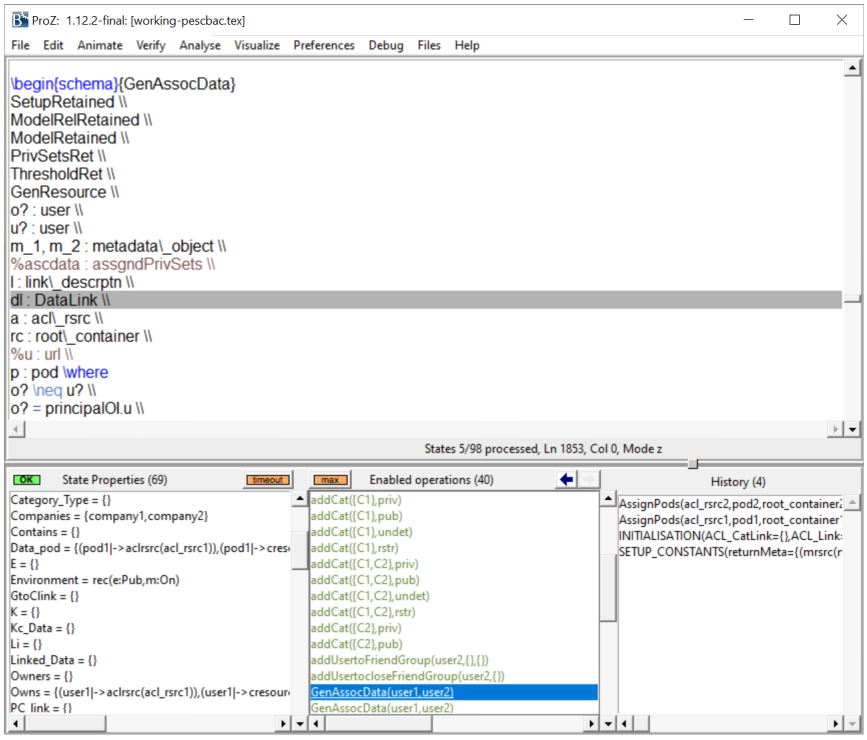
A.3 Association

Here, we look at the method taken to limit associative exposure by way of a series of schemas.

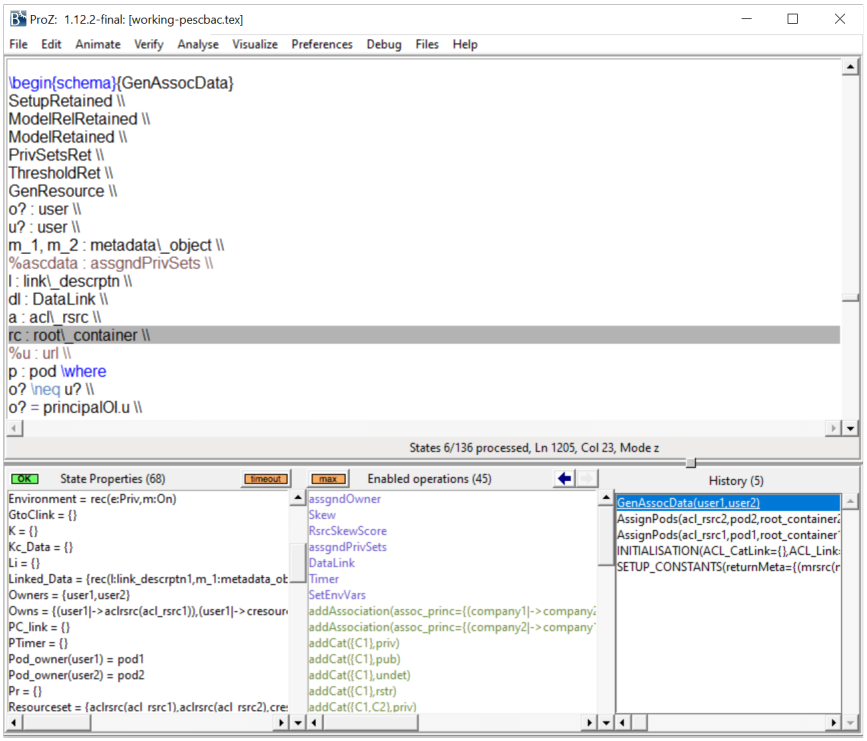
We may begin by considering a data link, captured by the schema *DataLink* that exists between two metadata objects to acknowledge an associative link between the two.



We then move to appreciate the schema *GenAssocData* which generates a sole piece of a metadata that has an associative link to an existing metadata object.



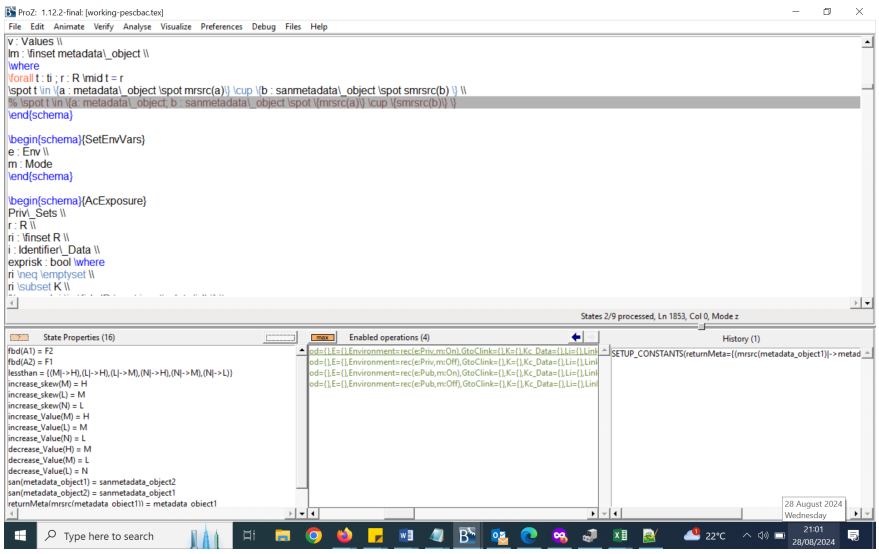
We are careful to note that the *Linked_Data* set is empty before this operation is enacted so that we may observe the change made to the set following the call to the operation.



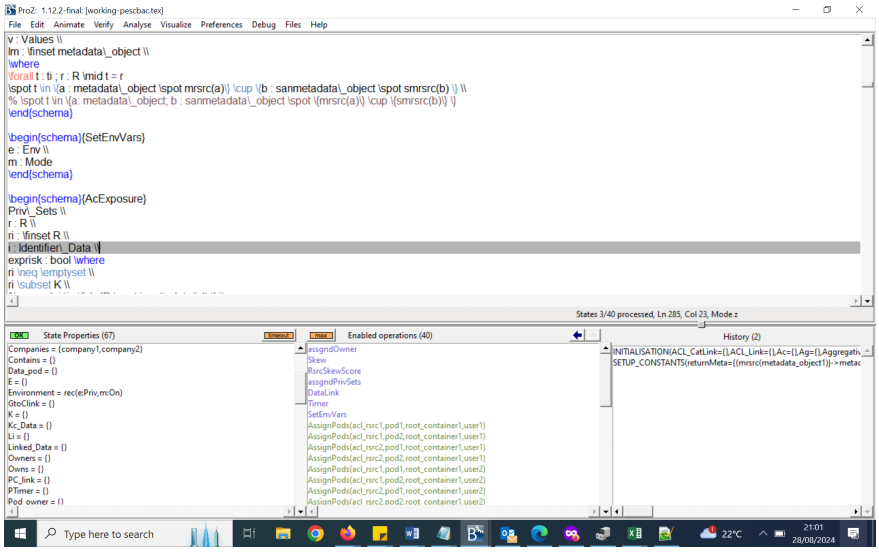
Where the set *Linked_Data* is used as a means of tracking the linked metadata objects between users to minimise associative links by way of timing and sensitivity checks.

A.4 Surveillance

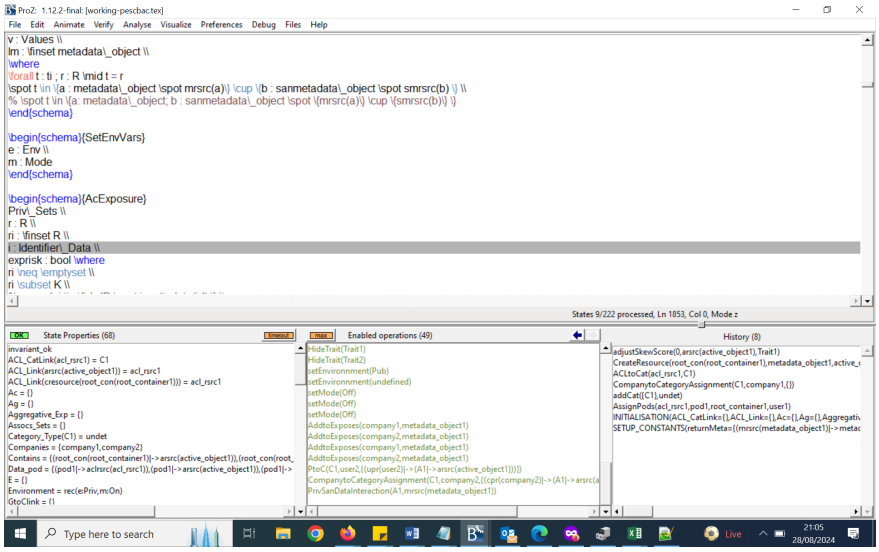
We look to the ProB output as a means to showcase the ways in which metadata collection may occur in an environment with heightened anti-surveillance properties.



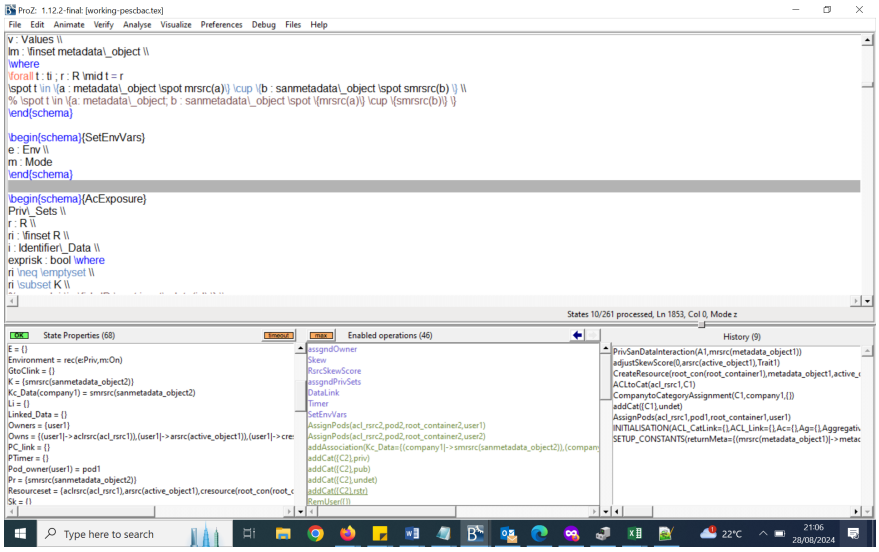
We begin through acknowledging the ability to define the environment during initialisation, whereupon we select the option of $Environment = \{e = Priv, m = On\}$, which is reflective of a private environment with sanitisation mode enabled.



Following this we go forward to create user data as seen in the *Data_pod* set and the History of operations, which shows *CreateResource*.



We may now focus on the a company collecting the metadata object in the given environment of Private with sanitisation mode *On*, wherein we may reflect on the choice of collection following the company's assignment to the given category.



Here, we see the operation *PrivSanDataInteraction* has occurred leaving the known set with a sanitised version of the metadata object. signifying that the company in question has collected only a sanitised version of the metadata object.

Appendix B

Formal Model for Privacy

The following appendix chapter serves to provide the full formal models making up Contrbution 1.

Basic types:

$$[\, User, Company, Active\_Data, Metadata, San\_Metadata, ID, Trait \,]$$

$$\begin{aligned} Data\_Object ::= \; & active\_data\langle\langle Active\_Data\rangle\rangle \mid \\ & metadata\langle\langle Metadata\rangle\rangle \mid \\ & smetadata\langle\langle San\_Metadata\rangle\rangle \end{aligned}$$

$$\begin{aligned} Identifying\_Object ::= \; & data\_ob\langle\langle Data\_Object\rangle\rangle \mid \\ & input\_data\langle\langle ID\rangle\rangle \mid \\ & NULL \end{aligned}$$

$$Identifier\_Data == Trait \times Identifying\_Object$$

$$\begin{aligned} Data ::= \; & data\_object\langle\langle Data\_Object\rangle\rangle \mid \\ & id\_data\langle\langle Identifier\_Data\rangle\rangle \end{aligned}$$

$$Mode ::= On \mid Off$$

$$Env ::= Home \mid Public$$

The (partial) function *San* maps elements of *Metadata* to elements of *San_Metadata*:

$$San : Metadata \pfun San\_Metadata$$

$$m1, m2 : Metadata \mid m1 \neq m2 \bullet San(m1) \neq San(m2)$$

We define our core model for our privacy objectives in the schema *PrivModel*.

PrivModel

$$Companies : \mathbb{P}_1\ Company$$

$$Users : \mathbb{P}_1\ User$$

$$P : \mathbb{P}\ Data$$

$$T : \mathbb{P}\ Data$$

$$K : \mathbb{P}\ Data$$

$$Kc\_Data : Company \rel Data$$

$$T\_s, T\_u : \mathbb{P}\ Data$$

$$\mathrm{ran}(Kc\_Data) \subseteq K$$

$$\forall\, q : Identifier\_Data \mid id\_data(q) \in T \bullet$$

$$\quad id\_data(q) \in \mathrm{ran}\ Kc\_Data \Rightarrow id\_data(q) \in P$$

$$K \subseteq P$$

$$P \subseteq T$$

$$T\_s \cup T\_u = T$$

$$T\_s \cap T\_u = \emptyset$$

$$T\_s \cap P = \emptyset$$

B.1 Identification

We present the modular schema for Identification, which address the privacy harms found in this area.

Identification

$$P : \mathbb{P}\, Data$$

$$E : Data\_Object \leftrightarrow Identifier\_Data$$

$$T\_s : \mathbb{P}\, Data$$

$$Sk : Metadata \leftrightarrow Identifier\_Data$$

$$\forall\, a : Active\_Data;\ i : Identifier\_Data \bullet$$

$$active\_data(a) \mapsto i \in E \Rightarrow$$

$$data\_object(active\_data(a)) \notin P \lor id\_data(i) \notin T\_s$$

$$\forall\, i : Identifier\_Data;\ m : Metadata \bullet$$

$$m \mapsto i \in Sk \Rightarrow data\_object(metadata(m)) \notin P$$

$$PrivModelWithIdentification \mathrel{\widehat{=}} Identification \land PrivModel$$

Initid

PrivModelWithIdentification

$P = \emptyset$

$T = \emptyset$

$K = \emptyset$

$Kc\_Data = \emptyset$

$Dataset = \emptyset$

$E = \emptyset$

$Sk = \emptyset$

B.2 Aggregation

We present the modular schemas for Aggregation, which address the privacy harms found in this area.

Aggregation

$P : \mathbb{P}\, Data$

$T : \mathbb{P}\, Data$

$K : \mathbb{P}\, Data$

$L : \mathbb{P}\, Company$

$Kc\_id : \mathbb{P}_1\, Data \leftrightarrow Identifier\_Data$

$Kc\_Data : Company \leftrightarrow Data$

$Ac : Data\_Object \nrightarrow Kc\_Id$

$T\_s, T\_u : \mathbb{P}\, Data$

$Dataset : \mathbb{P}\, Data$

$\mathrm{ran}(Kc\_Data) \subseteq K$

$\forall\, q : Identifier\_Data \mid id\_data(q) \in T \bullet id\_data(q) \in \mathrm{ran}\ Kc\_Data \Rightarrow id\_data(q) \in P$

$\forall\, q : Identifier\_Data;\ c\_k : Kc\_id;\ d\_m : Metadata \mid id\_data(q) \in T\_s \land second\ c\_k = q \bullet$

$metadata(d\_m) \mapsto c\_k \in Ac \Rightarrow data\_object(metadata(d\_m)) \notin P$

$\forall\, d : Data;\ c\_k : Kc\_id \mid d \in first\ c\_k \bullet d \in K$

$L \subseteq \mathrm{dom}(Kc\_Data)$

$K \subseteq P$

$T \subseteq Dataset$

$P \subseteq T$

$T\_s \cup T\_u = T$

$T\_s \cap T\_u = \emptyset$

$T\_s \cap P = \emptyset$

$PrivModelWithAggregation \mathrel{\widehat=} Aggregation \land PrivModel$

Initagg

$PrivModelWithAggregation$

$P = \emptyset$

$T = \emptyset$

$K = \emptyset$

$Kc\_Data = \emptyset$

$Dataset = \emptyset$

$Ac = \emptyset$

$Kc\_id = \emptyset$

$TrueLimit \mathrel{\widehat=} \exists\, m, n : \mathbb{P}\, Data \bullet Limit$

Limit

$\Delta PrivModelWithAggregation$

$c? : Company$

$m?, n? : \mathbb{P}\, Data$

$c? \in \mathrm{dom}(Kc\_Data)$

$c? \in L$

$m? = \{m : Metadata;\ kc : Kc\_Data \mid data\_object(metadata(m)) = second\ kc;$

$first\ kc = c? \bullet data\_object(metadata(m))\}$

$n? = \{n : Metadata;\ kc : Kc\_Data \mid data\_object(metadata(n)) = second\ kc;$

$first\ kc \neq c? \bullet data\_object(metadata(n))\}$

$Kc\_id' = Kc\_id$

$Kc\_Data' = (Kc\_Data \rhd m?) \cup (\{c?\} \lhd Kc\_Data)$

$Dataset' = Dataset$

$L' = L \setminus \{c?\}$

$P' = P \setminus m? \cup n?$

$T\_s' = \textbf{if}\ n? = \emptyset\ \textbf{then}\ (T\_s \cup m?\ \textbf{else}\ T\_s$

$T\_u' = \textbf{if}\ n? = \emptyset\ \textbf{then}\ T\_u \setminus m?\ \textbf{else}\ T\_u$

$K' = K \setminus m? \cup n?$

$Ac' = Ac$

$Users' = Users$

$Companies' = Companies$

B.3 Association

We present the modular schemas for Association, which address the privacy harms found in the area.

Association

$$Users : \mathbb{P}_1\ User$$

$$T\_s,\ T\_u : \mathbb{P}\ Data$$

$$T : \mathbb{P}\ Data$$

$$P : \mathbb{P}\ Data$$

$$C\_inf : (Metadata \times Identifier\_Data) \times Company$$

$$A : Data\_Object \leftrightarrow \mathbb{P}\ User$$

$$EA : \mathbb{P}(Metadata \times Metadata)$$

$$Sk : Metadata \leftrightarrow Identifier\_Data$$

$$Sk\_ : \mathbb{P}(C\_inf \times \mathbb{P}\ User)$$

$$Tr : \mathbb{P}\ Metadata$$

$$\forall\, u1, u2 : Users;\ m : Metadata \mid u1 \neq u2 \bullet$$

$$metadata(m) \mapsto \{u1, u2\} \in A \land m \in Tr \Rightarrow data\_object(metadata(m)) \in T\_s$$

$$\forall\, m1, m2 : Metadata;\ u1, u2 : Users \mid u1 \neq u2 \bullet m1 \mapsto m2 \in EA$$

$$\land\ metadata(m1) \mapsto \{u1, u2\} \in A \land data\_object(metadata(m1)) \in T\_s$$

$$\Rightarrow data\_object(metadata(m2)) \in T\_s$$

$$\#Users > 1$$

$$\forall\, u1, u2, u3 : Users;\ m : Metadata;\ c : C\_inf \mid u1 \neq u2 \bullet$$

$$c \mapsto \{u1, u2, u3\} \in Sk\_ \Rightarrow data\_object(metadata(m)) \in T\_s$$

Initassoc

$PrivModelWithAssociation$

$P = \emptyset$

$T = \emptyset$

$K = \emptyset$

$Kc\_Data = \emptyset$

$Dataset = \emptyset$

$A = \emptyset$

$EA = \emptyset$

$Sk = \emptyset$

$Sk\_ = \emptyset$

$Tr = \emptyset$

$PrivModelWithAssociation \mathrel{\widehat{=}} Association \wedge PrivModel$

B.4 Surveillance

We present the modular schemas for Surveillance, which address the privacy harms found in this area.

Surveillance

$$Dataset : \mathbb{P}\, Data$$

$$T\_s, T\_u : \mathbb{P}\, Data$$

$$T : \mathbb{P}\, Data$$

$$P : \mathbb{P}\, Data$$

$$Environment : Env$$

$$San\_mode : Mode$$

$$Ti : \mathbb{P}\, Metadata$$

$$\forall\, d\_m : Metadata;\ e : Env;\ m : Mode \mid data\_object(metadata(d\_m)) \in Dataset;$$

$$e = Home \bullet m = Off \Rightarrow data\_object(metadata(d\_m)) \in T\_s$$

$$\forall\, d\_m : Metadata;\ d\_s : San\_Metadata;\ e : Env;\ m : Mode \mid e = Home$$

$$\land\ d\_s = San(d\_m) \land d\_m \in Ti \bullet$$

$$m = On \Rightarrow (data\_object(metadata(d\_m)) \in T\_s \land data\_object(smetadata(d\_s)) \notin T\_s)$$

$$P \subseteq T$$

$$T\_s \cup T\_u = T$$

$$T\_s \cap T\_u = \emptyset$$

$$T\_s \cap P = \emptyset$$

$$PrivModelWithSurveillance \mathrel{\widehat{=}} Surveillance \land PrivModel$$

Initsurv

PrivModelWithSurveillance

$P = \emptyset$

$T = \emptyset$

$K = \emptyset$

$Kc\_Data = \emptyset$

$Dataset = \emptyset$

$Ti = \emptyset$

B.5 Conjunctive Model

We present the following conjunction as a means of amalgamating the various modular schemas.

$$PrivModelAll \mathrel{\widehat=} Identification \land Aggregation \land Association \land Surveillance \land PrivModel$$

Init

$PrivModelAll$

$P = \emptyset$

$T = \emptyset$

$K = \emptyset$

$Kc\_Data = \emptyset$

$Dataset = \emptyset$

$E = \emptyset$

$Ac = \emptyset$

$A = \emptyset$

$EA = \emptyset$

$Sk = \emptyset$

$Sk\_ = \emptyset$

$Tr = \emptyset$

$Ti = \emptyset$

Appendix C

Formal Model for Solid

The following appendix chapter serves to provide the full formal models making up Contrbution 2.

C.1 The Basics

$[owner, pod, App, container\_resource, resource, root\_container, resource\_metadata, server, slash, agent, webid]$

$bool ::= t \mid f$

$container ::= con\langle\langle container\_resource\rangle\rangle \mid root\_con\langle\langle root\_container\rangle\rangle$

$acl\_rsrc ::= acl1 \mid acl2 \mid acl3 \mid acl4 \mid acl5$

$Resource ::= rsrc\langle\langle resource\rangle\rangle \mid rsrc\_metadata\langle\langle resource\_metadata\rangle\rangle$
$\mid cresource\langle\langle container\rangle\rangle \mid aclrsrc\langle\langle acl\_rsrc\rangle\rangle$

Agent_Cred

$w : webid$

$c : agent$

$a : bool$

ACL_Dfntn

$ownr : owner$

$ag : agent$

$rsc : Resource$

$acrsc : acl\_rsrc$

$rperm : bool$

$wperm : bool$

$aperm : bool$

C.2 The Solid Model

Solid_Model

$Owners : \mathbb{F}_1\ owner$

$Resourceset : \mathbb{P}\ Resource$

$Dataset : \mathbb{P}\ Resource$

$Owns : owner \leftrightarrow Resource$

$Pod\_owner : owner \leftrightarrow pod$

$Data\_pod : pod \leftrightarrow Resource$

$Contains : container \leftrightarrow Resource$

$ACL\_Link : acl\_rsrc \rightarrowtail Resource$

$Access\_Rights : \mathbb{P}\ ACL\_Dfntn$

$Authenticated : \mathbb{P}\ agent$

$WebId : \mathbb{P}\ Agent\_Cred$

$Dataset \subseteq Resourceset$

$\text{dom}\ Owns \subseteq \text{dom}\ Pod\_owner$

$\text{ran}\ Owns = Resourceset$

$\text{dom}\ Pod\_owner \subseteq Owners$

$\text{dom}\ Data\_pod \subseteq \text{ran}\ Pod\_owner$

$\text{ran}\ Data\_pod \subseteq \text{ran}\ Owns$

$\text{ran}\ Contains \subseteq Resourceset$

$\text{ran}\ ACL\_Link \subseteq Resourceset$

$Dataset \subset \text{ran}\ Owns$

$Dataset \subset \text{ran}\ ACL\_Link$

...

...

$Dataset \subset \text{ran } Data\_pod$

$\forall r : Resource \bullet r \in Resourceset \Rightarrow r \in \text{ran } Contains$

$\forall c : Contains;\ r : root\_container \bullet cresource(root\_con(r)) \neq second\ c$

$\forall co : container\_resource;\ c : Contains \bullet first\ c = con(co)$

$\Rightarrow second\ c \neq cresource(con(co))$

$\forall o_1, o_2 : Owns \bullet (second\ o_1 = second\ o_2) \Rightarrow (first\ o_1 = first\ o_2)$

$\forall aa, ab : Access\_Rights \bullet (aa.rsc = ab.rsc \land aa.ag = ab.ag) \Rightarrow aa = ab$

$\forall r : resource \bullet rsrc(r) \in Resourceset \Rightarrow rsrc(r) \in Dataset$

$\exists rc_1, rc_2 : root\_container \bullet \forall p : pod \bullet (p \mapsto cresource(root\_con(rc_1))) \in Data\_pod$

$\Rightarrow p \mapsto cresource(root\_con(rc_2)) \notin Data\_pod$

$\forall r : Resource;\ c_1, c_2 : Contains \bullet second\ c_1 = r \land second\ c_2 = r \Rightarrow c_1 = c_2$

Init

$Solid\_Model$

$\#\ Owners \geq 2$

$Access\_Rights = \emptyset$

$Authenticated = \emptyset$

$WebId = \emptyset$

$\#\ Resourceset \geq 2$

$Dataset \neq \emptyset$

C.3 Assignments

Assign

$\Delta Solid\_Model$

$Owners' = Owners$

$Dataset' = Dataset$

$Resourceset' = Resourceset$

$Owns' = Owns$

$Data\_pod' = Data\_pod$

$Access\_Rights' = Access\_Rights$

$Authenticated' = Authenticated$

$ACL\_Link' = ACL\_Link$

$Contains' = Contains$

$VerPodAssign \mathrel{\widehat=} \exists\, a : acl\_rsrc;\ u : url \bullet AssignPods$

AssignPods

$\Delta Solid\_Resources$

$Assign$

$u? : owner$

$p? : pod$

$rc? : root\_container$

$a : acl\_rsrc$

$u : url$

$a \notin \text{dom } ACL\_Link$

$u \notin \text{ran } Ident$

$cresource(root\_con(rc?)) \notin \text{dom } Owned\_By$

$u? \in Owners$

$p? \notin \text{dom } Pod\_owner$

$Ident' = Ident \cup \{cresource(root\_con(rc?)) \mapsto u\}$

$Data\_pod' = Data\_pod \cup \{cresource(root\_con(rc?)) \mapsto p?\}$

$Pod\_owner' = Pod\_owner \cup \{p? \mapsto u?\}$

$Owned\_By' = Owned\_By \cup \{cresource(root\_con(rc?)) \mapsto u?\}$

$Resourceset' = Resourceset \cup \{cresource(root\_con(rc?))\}$

$ACL\_Link' = ACL\_Link \cup \{a \mapsto cresource(root\_con(rc?))\}$

AssignWebIds

$\Delta Solid\_Model$

$Assign$

$c? : agent$

$i? : webid$

$cw : Agent\_Cred$

$cw.c = c?$

$cw.w = i?$

$cw.a = f$

$\forall\, cc : WebId \bullet cc.c \neq c? \land cc.w \neq i?$

$Pod\_owner' = Pod\_owner$

$WebId' = WebId \cup \{cw\}$

C.4 Resources

GenResource

$\Delta Solid\_Resources$

$Owners' = Owners$

$Pod\_owner' = Pod\_owner$

CreateResource

$\Delta Solid\_Resources$

$GenResource$

$o? : owner$

$r? : resource$

$m? : resource\_metadata$

$a : acl\_rsrc$

$c? : container$

$u : url$

$p? : pod$

$u \notin \text{ran } Ident$

$a \mapsto cresource(c?) \in ACL\_Link$

$p? \mapsto o? \in Pod\_owner$

$cresource(c?) \mapsto o? \in Owned\_By$

$rsrc(r?) \notin Dataset$

$Ident' = Ident \cup \{rsrc(r?) \mapsto u\}$

$Dataset' = Dataset \cup \{rsrc(r?), rsrcmeta(m?)\}$

$Resourceset' = Resourceset \cup \{rsrc(r?), rsrcmeta(m?)\}$

$Owned\_By' = Owned\_By \cup \{rsrc(r?) \mapsto o?\} \cup \{rsrcmeta(m?) \mapsto o?\}$

$Data\_pod' = Data\_pod \cup \{rsrc(r?) \mapsto p?\} \cup \{rsrcmeta(m?) \mapsto p?\}$

$inheritACL' = inheritACL \cup \{rsrc(r?)\}$

$Contains' = (Contains \cup \{(c?) \mapsto rsrc(r?)\}) \cup \{(c?) \mapsto rsrcmeta(m?)\}$

$Meta' = Meta \cup \{r? \mapsto m?\}$

$\vdots$

New_Container

$GenResource$

$u : owner$

$a? : acl\_rsrc$

$rc : root\_container$

$c? : container\_resource$

$p? : pod$

$a? \notin \text{dom } ACL\_Link$

$p? \mapsto u \in Pod\_owner$

$cresource(root\_con(rc)) \mapsto u \in Owned\_By$

$cresource(con(c?)) \notin Resourceset$

$Ident' = Ident$

$Dataset' = Dataset$

$Resourceset' = Resourceset \cup \{cresource(con(c?))\}$

$Owned\_By' = Owned\_By \cup \{cresource(con(c?)) \mapsto u\}$

$Data\_pod' = Data\_pod \cup \{cresource(con(c?)) \mapsto p?\}$

$ACL\_Link' = ACL\_Link \cup \{a? \mapsto cresource(con(c?))\}$

$inheritACL' = inheritACL$

$Contains' = Contains \cup \{root\_con(rc) \mapsto cresource(con(c?))\}$

$Nest' = Nest \cup \{root\_con(rc) \mapsto con(c?)\}$

$Meta' = Meta$

ManageCon

$\Delta Solid\_Resources$

$Owners' = Owners$

$Pod\_owner' = Pod\_owner$

$Owned\_By' = Owned\_By$

$Dataset' = Dataset$

$Data\_pod' = Data\_pod$

$Ident' = Ident$

$Meta' = Meta$

$Access\_Rights' = Access\_Rights$

$Authenticated' = Authenticated$

$Ag\_Credentials' = Ag\_Credentials$

$ACL\_Link' = ACL\_Link$

$inheritACL' = inheritACL$

$Resourceset' = Resourceset$

$VerNestIn \mathrel{\widehat=} \exists\, u : owner;\ p : pod \bullet Nest\_In\_Container$

Nest_In_Container

$\Delta Solid\_Resources$

$ManageCon$

$u : owner$

$c?, c_n? : container$

$r : Resource$

$p : pod$

$r = cresource(c_n?)$

$c? \mapsto r \notin Contains$

$(Nest^+) \cap ((Nest^+)^\sim) = \emptyset$

$p \mapsto u \in Pod\_owner$

$r \neq cresource(c?)$

$r \mapsto p \in Data\_pod$

$cresource(c?) \mapsto u \in Owned\_By$

$Resourceset' = Resourceset$

$Contains' = ((Contains \rhd \{r\})) \cup \{c? \mapsto r\}$

$Nest' = (Nest \rhd \{c_n?\}) \cup \{c? \mapsto c_n?\}$

C.5 Access Rights

Access

$\Delta Solid\_Model$

$Owners' = Owners$

$Dataset' = Dataset$

$Resourceset' = Resourceset$

$Owns' = Owns$

$Pod\_owner' = Pod\_owner$

$Data\_pod' = Data\_pod$

$ACL\_Link' = ACL\_Link$

$Contains' = Contains$

Authenticate

$\Delta Solid\_Model$

$Access$

$ag? : agent$

$cv, cw : Agent\_Cred$

$ag? \notin Authenticated$

$cw \in WebId$

$cv.c = ag?$

$cw.c = ag?$

$cv.w = cw.w$

$cv.a = t$

$Access\_Rights' = Access\_Rights$

$Authenticated' = Authenticated \cup \{ag?\}$

$WebId' = WebId \setminus \{cw\} \cup \{cv\}$

Dfe_ACL

$\Delta Solid\_Model$

$Access$

$a : acl\_rsrc$

$r : Resource$

$c : agent$

$b, d, f : bool$

$ur : owner$

$ac : ACL\_Dfntn$

$c \in \{a : WebId \bullet a.c\}$

$ur \mapsto r \in Owns$

$r \in Dataset$

$a \mapsto r \in ACL\_Link$

$ac.ownr = ur$

$ac.ag = c$

$ac.rsc = r$

$ac.acrsc = a$

$ac.rperm = b$

$ac.wperm = d$

$ac.aperm = f$

$Access\_Rights' = Access\_Rights \cup \{ac\}$

$Authenticated' = Authenticated$

$WebId' = WebId$

Mod_RPerms

$\Delta Solid\_Model$

$Access$

$ap, aq : ACL\_Dfntn$

$ap \in Access\_Rights$

$aq.ownr = ap.ownr$

$aq.ag = ap.ag$

$aq.rsc = ap.rsc$

$aq.acrsc = ap.acrsc$

$aq.rperm \neq ap.rperm$

$aq.wperm = ap.wperm$

$aq.aperm = ap.aperm$

$Access\_Rights' = (Access\_Rights \setminus \{ap\}) \cup \{aq\}$

$Authenticated' = Authenticated$

$WebId' = WebId$

Mod_WPerms

$\Delta Solid\_Model$

$Access$

$ap, aq : ACL\_Dfntn$

$ap \in Access\_Rights$

$aq.ownr = ap.ownr$

$aq.ag = ap.ag$

$aq.rsc = ap.rsc$

$aq.acrsc = ap.acrsc$

$aq.rperm = ap.rperm$

$aq.wperm \neq ap.wperm$

$aq.aperm = ap.aperm$

$Access\_Rights' = (Access\_Rights \setminus \{ap\}) \cup \{aq\}$

$Authenticated' = Authenticated$

$WebId' = WebId$

Mod_APerms

$\Delta Solid\_Model$

$Access$

$ap, aq : ACL\_Dfntn$

$ap \in Access\_Rights$

$aq.ownr = ap.ownr$

$aq.ag = ap.ag$

$aq.rsc = ap.rsc$

$aq.acrsc = ap.acrsc$

$aq.rperm = ap.rperm$

$aq.wperm = ap.wperm$

$aq.aperm \neq ap.aperm$

$Access\_Rights' = (Access\_Rights \setminus \{ap\}) \cup \{aq\}$

$Authenticated' = Authenticated$

$WebId' = WebId$

Appendix D

Formal Model for CBAC

This appendix chapter explores the CBAC models in their entirety.

D.1 The Basics

$[C, P, A, R, S, E]$

$Perm == A \times R$

$Auth == P \times Perm$

MModel

$permset : \mathbb{F}\, Perm$

$par : P \leftrightarrow Perm$

$arca : C \leftrightarrow Perm$

$pca : P \leftrightarrow C$

$\forall\, p : \mathrm{ran}(arca) \bullet p \in permset$

$\forall\, q : (arca) \mid first\ q \in \mathrm{ran}(pca) \bullet second\ q \in \mathrm{ran}(par)$

Init

$MModel$

$permset = \emptyset$

$par = \emptyset$

$arca = \emptyset$

$pca = \emptyset$

D.2 Modular schemas for Stagnancy

UpdatePerms

$\Delta MModel$

$par' = par$

$arca' = arca$

$pca' = pca$

UpdatePCA

$\Delta MModel$

$permset' = permset$

$par' = par$

$arca' = arca$

UpdateARCA

$\Delta MModel$

$permset' = permset$

$par' = par$

$pca' = pca$

D.3 Allocation Schemas

AllocatePerms

$$UpdatePerms$$

$$allocation? : \mathbb{F}_1\ Perm$$

$$permset = \emptyset$$

$$permset' = allocation?$$

*AllocatePrinciples*0

$$UpdatePCA$$

$$p? : P$$

$$c? : C$$

$$p? \mapsto c? \notin pca$$

$$c? \notin \mathrm{dom}(arca)$$

$$pca' = pca \cup \{p? \mapsto c?\}$$

*AllocatePrinciples*1

$\Delta MModel$

$p? : P$

$c? : C$

$v? : \mathbb{F}\, Auth$

$v? = \{m : arca \mid first\ m = c? \bullet (p?, second\ m)\}$

$v? \neq \emptyset$

$p? \mapsto c? \notin pca$

$pca' = pca \cup \{p? \mapsto c?\}$

$permset' = permset$

$par' = par \cup v?$

$arca' = arca$

D.4 Schemas for additions

*CategoryPerms*0

$$
\begin{array}{l}
UpdateARCA \\
c? : C \\
p? : Perm \\
\hline
c? \notin \operatorname{ran} pca \\
p? \in permset \\
c? \mapsto p? \notin arca \\
arca' = arca \cup \{c? \mapsto p?\}
\end{array}
$$

$$VerCatPerms \mathrel{\widehat=} \exists\, v : \mathbb{P}\, Auth \bullet CategoryPerms1$$

*CategoryPerms*1

$\Delta MModel$

$c? : C$

$pe? : Perm$

$v : \mathbb{P}\, Auth$

$v = \{m : pca \mid second\ m = c? \bullet (first\ m, pe?)\}$

$v \neq \emptyset$

$pe? \in permset$

$c? \mapsto pe? \notin arca$

$arca' = arca \cup \{c? \mapsto pe?\}$

$permset' = permset$

$par' = par \cup v$

$pca' = pca$

*UpdateCatPerms*1

$\Delta MModel$

$c? : C$

$p? : P$

$pe?, qe? : Perm$

$v? : \mathbb{P}\, Auth$

$v? = \{m : pca \mid second\, m = c? \land first\, m = p? \bullet (first\, m, qe?)\}$

$v? \neq \emptyset$

$qe? \in permset$

$pe? \neq qe?$

$pe? \in \mathrm{ran}(arca)$

$c? \mapsto qe? \notin arca$

$arca' = (arca \setminus \{c? \mapsto pe?\}) \cup \{c? \mapsto qe?\}$

$permset' = permset$

$par' = par \setminus \{p? \mapsto pe?\} \cup v?$

$pca' = pca$

D.5 Schemas for Removal

*RemoveCatPerm*0

$\Delta MModel$

$c? : C$

$pe? : Perm$

$v? : \mathbb{P}\, Auth$

$(v? = \emptyset \land c? \notin \mathrm{ran}(pca))$

$pe? \in permset$

$c? \mapsto pe? \in arca$

$arca' = arca \setminus \{c? \mapsto pe?\}$

$permset' = permset$

$par' = par \setminus v?$

$pca' = pca$

*RemoveCatPerm*1

$\Delta MModel$

$c? : C$

$pe? : Perm$

$v?, d? : \mathbb{P}\, Auth$

$v? \neq \emptyset$

$v? = \{m : pca \mid second\, m = c? \bullet (first\, m, pe?)\}$

$d? = \{n : v?;\ s : pca;\ t : arca$

$\quad \mid first\, n = first\, s;\ second\, s = first\, t;\ second\, s \neq c?$

$\quad\quad \bullet (first\, n, second\, t)\}$

$pe? \in permset$

$c? \mapsto pe? \in arca$

$arca' = arca \setminus \{c? \mapsto pe?\}$

$permset' = permset$

$par' = (par \setminus v?) \cup d?$

$pca' = pca$

RemovePerms

$\Delta MModel$

$allocation? : \mathbb{F}_1\ Perm$

$permset \neq \emptyset$

$allocation? \subseteq permset$

$permset' = permset \setminus allocation?$

$par' = par \rhd allocation?$

$pca' = pca$

$arca' = arca \rhd allocation?$

*RemovePrinc*0

$UpdatePCA$

$c? : C$

$p? : P$

$p? \mapsto c? \in pca$

$c? \notin \mathrm{dom}(arca)$

$pca' = pca \setminus \{p? \mapsto c?\}$

*RemovePrinc*1

$\Delta MModel$

$p? : P$

$c? : C$

$v? : \mathbb{P}\, Auth$

$pe? : Perm$

$v? = \{m : arca \mid first\ m = c? \bullet (p?, second\ m)\}$

$p? \mapsto c? \in pca$

$p? \notin \mathrm{dom}(pca \setminus \{p? \mapsto c?\})$

$c? \mapsto pe? \in arca$

$pe? \in permset$

$permset' = permset$

$par' = \textbf{if}\ (pe? \in \mathrm{ran}(arca \setminus \{c? \mapsto pe?\}) \land p? \in \mathrm{dom}(pca \setminus \{p? \mapsto c?\}))$

$\qquad\qquad \textbf{then}\ par$

$\qquad\quad \textbf{else}\ par \setminus v?$

$arca' = arca$

$pca' = pca \setminus \{p? \mapsto c?\}$

RemovePrinc2

$\Delta MModel$

$p? : P$

$c? : C$

$d? : \mathbb{F}\, C$

$v? : \mathbb{P}\, Auth$

$pe? : Perm$

$d? = \{n : pca \mid first\ n = p? \land second\ n \neq c? \bullet second\ n\}$

$v? = \{m : arca \mid first\ m = c? \land second\ m \notin \mathrm{ran}(d? \lhd arca) \bullet (p?, second\ m)\}$

$p? \mapsto c? \in pca$

$p? \in \mathrm{dom}(pca \setminus \{p? \mapsto c?\})$

$c? \mapsto pe? \in arca$

$pe? \in permset$

$permset' = permset$

$par' = \textbf{if}\ (pe? \in \mathrm{ran}(d? \lhd arca) \land v? = \emptyset)\ \textbf{then}\ par\ \textbf{else}\ par \setminus v?$

$arca' = arca$

$pca' = pca \setminus \{p? \mapsto c?\}$

Appendix E

Formal Model for P.E_SCBAC

E.1 The Basics

$[C, F, A, S, E, active\_object, metadata\_object, user, company, private, public,$
$ID, Trait, read, write, append, control]$

$[owner, pod, container\_resource, resource, resource\_metadata,$
$root\_container, url, acl\_rsrc, agent, webid, link\_descrptn]$

$$fbd : A \rightarrow F$$

$$\forall\, a_1, a_2 : A \mid a_1 \neq a_2 \bullet fbd(a_1) \neq fbd(a_2)$$

$container ::= con\langle\langle container\_resource\rangle\rangle \mid root\_con\langle\langle root\_container\rangle\rangle$

$bool ::= t \mid f \mid u$

$Identifying\_Object ::= active\_ob\langle\langle active\_object\rangle\rangle \mid m\_ob\langle\langle metadata\_object\rangle\rangle \mid$
$input\_data\langle\langle ID\rangle\rangle \mid$
$NULL$

$Identifier\_Data == Trait \times Identifying\_Object$

$Cat\_Types ::= priv \mid pub \mid undet \mid rstr$

$Values ::= H \mid M \mid L \mid N$

assgndOwner

$o : owner$

$u : user$

$lessthan : Values \leftrightarrow Values$

$lessthan = \{(N, L)(L, M), (L, H), (M, H)\}$

$$increase\_skew : Values \pfun Values$$

$$increase\_skew\,(N) = L$$

$$increase\_skew\,(L) = M$$

$$increase\_skew\,(M) = H$$

$$Action ::= r\langle\langle read\rangle\rangle \mid w\langle\langle write\rangle\rangle \mid a\langle\langle append\rangle\rangle \mid c\langle\langle control\rangle\rangle$$

$$R ::= arsrc\langle\langle active\_object\rangle\rangle \mid mrsrc\langle\langle metadata\_object\rangle\rangle$$
$$\mid cresource\langle\langle container\rangle\rangle \mid aclrsrc\langle\langle acl\_rsrc\rangle\rangle$$
$$\mid id\_data\langle\langle Identifier\_Data\rangle\rangle$$

$$P ::= upr\langle\langle user\rangle\rangle \mid cpr\langle\langle company\rangle\rangle$$

$$Perm == A \times R$$

$$Forb == F \times R$$

$$Data == \mathbb{F}\, R$$

$$Assoc == company \times company$$

$$Link\_Cat == C \leftrightarrow C$$

$$rprc\_link == P \leftrightarrow \mathbb{F}\, C$$

$$Auth == P \times Perm$$

$$Denl == P \times Forb$$

$$returnMeta : R \pfun metadata\_object$$

$$\forall r : R \mid$$
$$(\exists_1 m : metadata\_object \bullet r = mrsrc(m)) \bullet$$
$$returnMeta\,(r) = (\mu\, m : metadata\_object \mid r = mrsrc\,(m) \bullet m)$$

Instance

$permset : \mathbb{F}\ Perm$

$forbset : \mathbb{F}\ Forb$

$principalset : \mathbb{F}\ P$

$catset : \mathbb{F}\ C$

$assoc\_princ : \mathbb{F}\ Assoc$

$cat\_link : \mathbb{F}\ Link\_Cat$

$closeFriend : \mathbb{F}\ P$

$friend : \mathbb{F}\ P$

$acquaintance : \mathbb{F}\ P$

$principalOI : assgndOwner$

$group : \mathbb{N}$

$group \leq 3$

$closeFriend \subseteq principalset$

$friend \subseteq principalset$

$acquaintance \subseteq principalset$

$closeFriend \cap friend = \emptyset$

$friend \cap acquaintance = \emptyset$

$acquaintance \cap closeFriend = \emptyset$

E.2 Privacy Additions

E.2.1 General

Skew

$comp : company$

$usr : user$

$tr : Trait$

$skewValue : Values$

$ctr : \mathbb{N}$

$ctr \leq 9$

RsrcSkewScore

$rsrc : R$

$tr : Trait$

$score : \mathbb{N}$

$read : bool$

$score \leq 5$

Priv_Sets

$T\_s : \mathbb{F}\, R$

$T\_u : \mathbb{F}\, R$

$T : \mathbb{F}\, R$

$Pr : \mathbb{F}\, R$

$K : \mathbb{F}\, R$

$Li : \mathbb{F}\, company$

$Kc\_Data : company \leftrightarrow R$

$\mathrm{ran}(Kc\_Data) \subseteq K$

$K \subseteq Pr$

$Pr \subseteq T$

$T\_s \cup T\_u = T$

$T\_s \cap T\_u = \emptyset$

$T\_s \cap Pr = \emptyset$

assgndPrivSets

$u : user$

$t\_s : \mathbb{F}\, R$

$t\_u : \mathbb{F}\, R$

$t : \mathbb{F}\, R$

$pr : \mathbb{F}\, R$

$k : \mathbb{F}\, R$

$li : \mathbb{F}\, company$

$kc\_Data : company \leftrightarrow R$

$k \subset pr$

$pr \subset t$

$t\_s \cup t\_u = t$

$t\_s \cap t\_u = \emptyset$

$t\_s \cap pr = \emptyset$

E.2.2 Identification

Identification

$$Priv\_Sets$$

$$E : R \leftrightarrow Identifier\_Data$$

$$Sk : metadata\_object \leftrightarrow Identifier\_Data$$

$$\forall a : active\_object;\ i : Identifier\_Data \bullet$$

$$arsrc(a) \mapsto i \in E \Rightarrow$$

$$arsrc(a) \notin Pr \lor id\_data(i) \notin T\_s$$

$$\forall i : Identifier\_Data;\ m : metadata\_object \mid$$

$$id\_data(i) \in T\_s \bullet$$

$$m \mapsto i \in Sk \Rightarrow mrsrc(m) \notin Pr$$

E.2.3 Aggregation

AcExposure

$$Priv\_Sets$$

$$r : R$$

$$R : \mathbb{F}\, R$$

$$i : Identifier\_Data$$

$$exprisk : bool$$

$$R \subseteq K$$

$$r \notin K$$

$$second\ i \in \{id : ID \bullet input\_data(id)\}$$

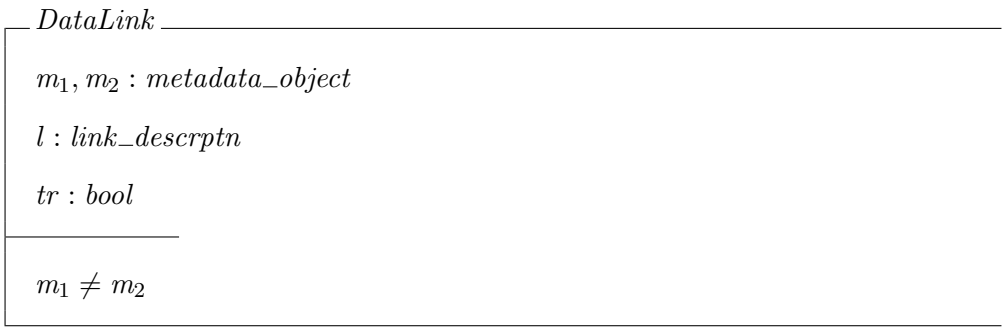
Accumulation

$$Priv\_Sets$$

$$Ac : R \nrightarrow RsrcSkewScore$$

$$\forall\, a : Ac;\ r : R;\ i : Identifier\_Data$$
$$\mid (second\ a).tr = first\ i \land first\ a = r$$
$$\bullet\ id\_data(i) \in T\_s \Rightarrow r \notin Pr$$

E.2.4 Association



E.2.5 Surveillance

$Env ::= Priv \mid Pub \mid undefined$

$Mode ::= On \mid Off$

Timer

$b : bool$

$ti : \mathbb{F}\, resource$

$V : Value$

$Lm : \mathbb{F}\, metadata\_resource$

$\forall\, t : ti;\ r : resource \mid t = r$

- $t \in \{a : metadata\_resource;\ b : sanmetadata\_resource$
  - $\{mrsc(a)\} \cup \{smrsc(b)\}$

SetEnvVars

$e : Env$

$m : Mode$

E.3 The Solid Additions

Solid_Model

$$Resourceset : \mathbb{P}\, R$$

$$Owners : \mathbb{F}\, user$$

$$Owns : user \leftrightarrow R$$

$$Pod\_owner : user \leftrightarrow pod$$

$$Data\_pod : pod \leftrightarrow R$$

$$Contains : container \leftrightarrow R$$

$$ACL\_Link : R \nrightarrow acl\_rsrc$$

$$ACL\_CatLink : acl\_rsrc \leftrightarrow C$$

$$podtoC : pod \nrightarrow C$$

$$assoc\_data : active\_object \rightarrowtail metadata\_object$$

$$Owners \subseteq \mathrm{dom}\ Owns$$

$$\mathrm{dom}\ Owns \subseteq user$$

$$\mathrm{ran}\ Owns \subseteq Resourceset$$

$$\mathrm{dom}\ Pod\_owner \subseteq user$$

$$\mathrm{dom}\ Data\_pod \subseteq \mathrm{ran}\ Pod\_owner$$

$$\mathrm{ran}\ Data\_pod \subseteq Resourceset$$

$$\mathrm{ran}\ Contains \subseteq Resourceset$$

$$\mathrm{dom}\ ACL\_Link \subseteq Resourceset$$

GenResource

$\Delta Solid\_Model$

$Pod\_owner' = Pod\_owner$

$ACL\_CatLink' = ACL\_CatLink$

$podtoC' = podtoC$

E.4 The PE_SCBAC Model

PE_SCBAC

Instance

Identification

Aggregation

Accumulation

Priv_Sets

Solid_Model

$Companies : \mathbb{P}_1\ company$

$Users : \mathbb{P}_1\ user$

$skewn : \mathbb{P}\ Skew$

$skscores : \mathbb{P}\ RsrcSkewScore$

$GtoClink : \mathbb{N} \leftrightarrow C$

$Category\_Type : C \leftrightarrow Cat\_Types$

$Environment : SetEnvVars$

$PTimer : Timer$

$par : P \leftrightarrow Perm$

$arca : C \leftrightarrow Perm$

$pca : P \leftrightarrow C$

$pfar : P \leftrightarrow Forb$

$farca : C \leftrightarrow Forb$

$Linked\_Data : \mathbb{F}\ DataLink$

$Assocs\_Sets : \mathbb{F}\ assgndPrivSets$

...

$$
\begin{array}{l}
\ldots \\
PC\_link : rprc\_link \\
Threshold : company \leftrightarrow \mathbb{N} \\
\hline
\mathrm{dom}\ par \subseteq principalset \\
\mathrm{dom}\ pfar \subseteq principalset \\
\mathrm{ran}\ par \subseteq permset \\
\mathrm{ran}\ pfar \subseteq forbset \\
\mathrm{dom}\ arca \subseteq catset \\
\mathrm{dom}\ farca \subseteq catset \\
\mathrm{dom}\ pca \subseteq principalset \\
\mathrm{ran}\ pca \subseteq catset \\
par = pca \fatsemi arca \\
\forall u : Users \bullet upr(u) \in principalset \\
\forall c : Companies \bullet cpr(c) \in principalset \\
\forall p : principalset \\
\quad \bullet\ p \in \{u : Users \bullet upr(u)\} \lor p \in \{c : Companies \bullet cpr(c)\} \\
\mathrm{dom}\ Owns \subseteq Users \\
\forall a : assoc\_princ \bullet first\ a \neq second\ a \\
\forall a : assoc\_princ \bullet cpr(first\ a) \in principalset \land cpr(second\ a) \in principalset \\
\forall p : acquaintance;\ l : P \bullet (l = p) \Rightarrow (l \in \{u : Users \bullet upr(u)\}) \\
\forall p : friend;\ l : P \bullet (l = p) \Rightarrow (l \in \{u : Users \bullet upr(u)\}) \\
\forall c1, c2 : C \bullet \{c1 \mapsto c2\} \in cat\_link \Rightarrow c1 \neq c2 \\
\forall i, j : Skew \mid i \in skewn \land j \in skewn \\
\quad \bullet\ (i.comp = j.comp \land i.usr = j.usr \\
\quad \land\ i.tr = j.tr) \Rightarrow i = j \\
\forall c : Contains;\ r : root\_container \bullet cresource(root\_con(r)) \neq second\ c \\
\forall a : active\_object \bullet arsrc(a) \in Resourceset \Rightarrow a \in \mathrm{dom}\ assoc\_data \\
\forall t : Threshold \bullet second\ t \leq 3 \\
\forall a : Ac \bullet (second\ a).score = 4 \land (second\ a).rsrc \in Resourceset \\
\forall a : par;\ f : pfar \mid first\ a = first\ f
\end{array}
$$

E.4.1 Schemas for Stagnant Modelling

OwnerRet

ΔPE_SCBAC

$principalOI' = principalOI$

AssocDataRet

ΔPE_SCBAC

$Assocs\_Sets' = Assocs\_Sets$

$Linked\_Data' = Linked\_Data$

ThresholdRet

ΔPE_SCBAC

$Threshold' = Threshold$

SetupRetained

ΔPE_SCBAC

$OwnerRet$

$forbset' = forbset$

$catset' = catset$

$principalset' = principalset$

$permset' = permset$

$Companies' = Companies$

$Users' = Users$

ModelRetained

ΔPE_SCBAC

$OwnerRet$

$par' = par$

$arca' = arca$

$pca' = pca$

$pfar' = pfar$

$farca' = farca$

ModelRelRetained

ΔPE_SCBAC

$OwnerRet$

$assoc\_princ' = assoc\_princ$

$cat\_link' = cat\_link$

$closeFriend' = closeFriend$

$friend' = friend$

$acquaintance' = acquaintance$

$GtoClink' = GtoClink$

$group' = group$

ComponentsRet

ΔPE_SCBAC

$OwnerRet$

$Category\_Type' = Category\_Type$

$Resourceset' = Resourceset$

$Environment' = Environment$

PrivSetsRet

ΔPE_SCBAC

$OwnerRet$

$T' = T$

$T\_s' = T\_s$

$T\_u' = T\_u$

$K' = K$

$Li' = Li$

$Pr' = Pr$

$Kc\_Data' = Kc\_Data$

$E' = E$

$Sk' = Sk$

$skewn' = skewn$

$skscores' = skscores$

$Ac' = Ac$

UpdatePerms

$ModelRetained$

$ModelRelRetained$

$SolidSetsRet$

$OwnerRet$

$\Xi PrivSetsRet$

$catset' = catset$

$principalset' = principalset$

UpdateARCA

$SetupRetained$

$ModelRelRetained$

$OwnerRet$

$par' = par$

$pca' = pca$

UpdateCatPerms

$SetupRetained$

$ModelRelRetained$

$OwnerRet$

$pca' = pca$

SolidSetsRet

ΔPE_SCBAC

$Pod\_owner' = Pod\_owner$

$Owners' = Owners$

$OwnerRet$

$Owns' = Owns$

$Data\_pod' = Data\_pod$

$Contains' = Contains$

$ACL\_Link' = ACL\_Link$

$ACL\_CatLink' = ACL\_CatLink$

$podtoC' = podtoC$

$PC\_link' = PC\_link$

$assoc\_data' = assoc\_data$

E.4.2 Schemas for an instance of the model

Init

PE_SCBAC

$permset = \emptyset$

$forbset = \emptyset$

$catset = \emptyset$

$assoc\_princ = \emptyset$

$closeFriend = \emptyset$

$friend = \emptyset$

$acquaintance = \emptyset$

$group = 0$

$GtoClink = \emptyset$

$\#Users \geq 2$

$T = \emptyset$

$cat\_link = \emptyset$

$Li = \emptyset$

$E = \emptyset$

$Sk = \emptyset$

$skewn = \emptyset$

$skscores = \emptyset$

$Resourceset = \emptyset$

$Owns = \emptyset$

$assoc\_data = \emptyset$

$Pod\_owner = \emptyset$

...

...

$Data\_pod = \emptyset$

$ACL\_Link = \emptyset$

$ACL\_CatLink = \emptyset$

$Contains = \emptyset$

$podtoC = \emptyset$

$Category\_Type = \emptyset$

$Linked\_Data = \emptyset$

$Assocs\_Sets = \emptyset$

$PC\_link = \emptyset$

$Threshold = \emptyset$

$PTimer = \emptyset$

E.4.3 Solid_Setup

AssignPods

ΔPE_SCBAC

$SetupRetained$

$ModelRetained$

$ModelRelRetained$

$AssocDataRet$

$ThresholdRet$

$u? : user$

$p? : pod$

$rc? : root\_container$

$a? : acl\_rsrc$

$a? \notin \text{ran } ACL\_Link$

$u? \in Users$

$p? \notin \text{ran } Pod\_owner$

$cresource(root\_con(rc?)) \notin \text{ran } Owns$

$Pod\_owner' = Pod\_owner \cup \{u? \mapsto p?\}$

$Owners' = Owners$

$Users' = Users$

$Category\_Type' = Category\_Type$

$Environment' = Environment$

$Resourceset' = Resourceset \cup \{cresource(root\_con(rc?))\}$

$Owns' = Owns \cup \{u? \mapsto cresource(root\_con(rc?))\}$

$Data\_pod' = Data\_pod \cup \{p? \mapsto cresource(root\_con(rc?))\}$

...

...

$ACL\_Link' = ACL\_Link \cup \{cresource(root\_con(rc?)) \mapsto a?\}$

$ACL\_CatLink' = ACL\_CatLink$

$Contains' = Contains$

$podtoC' = podtoC$

$PC\_link' = PC\_link$

$assoc\_data' = assoc\_data$

$T' = T \cup \{cresource(root\_con(rc?))\}$

$T\_u' = T\_u \cup \{cresource(root\_con(rc?))\}$

$T\_s' = T\_s$

$E' = E$

$Li' = Li$

$K' = K$

$Kc\_Data' = Kc\_Data$

$Pr' = Pr$

$Sk' = Sk$

$skewn' = skewn$

$skscores' = skscores$

$Ac' = Ac$

ACLtoCat

ΔPE_SCBAC

$SetupRetained$

$ModelRetained$

$ModelRelRetained$

$PrivSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$a? : acl\_rsrc$

$c? : C$

$a? \mapsto c? \notin ACL\_CatLink$

$a? \in \text{ran}\, ACL\_Link$

$c? \in catset$

$Pod\_owner' = Pod\_owner$

$Owns' = Owns$

$Data\_pod' = Data\_pod$

$Contains' = Contains$

$ACL\_Link' = ACL\_Link$

$ACL\_CatLink' = ACL\_CatLink \cup \{a? \mapsto c?\}$

$podtoC' = podtoC$

$PC\_link' = PC\_link$

$assoc\_data' = assoc\_data$

E.5 Schemas for Additions

addAssociation

$SolidSetsRet$

$ModelRetained$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$c_0, c_1 : company$

$r : \mathbb{F}\, R$

$r = \{c : company;\ p : Perm \mid c = c_0 \land (c, second\ p) \in Kc\_Data \bullet second\ p\}$

$\quad \cup \{c : company;\ p : Perm \mid c = c_1 \land (c, second\ p) \in Kc\_Data \bullet second\ p\}$

$c_1 \mapsto c_0 \notin assoc\_princ$

$c_0 \neq c_1$

$cpr(c_0) \in principalset$

$cpr(c_1) \in principalset$

$c_0 \in \mathrm{dom}(Kc\_Data) \lor c_0 \notin \mathrm{dom}(Kc\_Data) \land c_1 \notin \mathrm{dom}(Kc\_Data)$

$Companies' = Companies$

$Users' = Users$

$Resourceset' = Resourceset$

$forbset' = forbset$

$permset' = permset$

$principalset' = principalset$

$catset' = catset$

$closeFriend' = closeFriend$

$friend' = friend$

$acquaintance' = acquaintance$

$GtoClink' = GtoClink$

$group' = group$

$cat\_link' = cat\_link$

$assoc\_princ' = assoc\_princ \cup \{c_0 \mapsto c_1\}$

$T' = T$

$T\_s' = T\_s$

$$
\begin{array}{l}
\ldots \\
Li' = Li \\
Pr' = Pr \\
Kc\_Data' = Kc\_Data \cup \{rs : R \mid rs \in r \bullet (c_0, rs)\} \cup \{rs : R \mid rs \in r \bullet (c_1, rs)\} \\
E' = E \\
Sk' = Sk \\
skewn' = skewn \\
skscores' = skscores \\
Ac' = Ac
\end{array}
$$

addCat

$ModelRelRetained$

$PrivSetsRet$

$SolidSetsRet$

$AssocDataRet$

$ThresholdRet$

$c? : \mathbb{F}_1\ C$

$t? : Cat\_Types$

$c? \cap catset = \emptyset$

$forbset' = forbset$

$permset' = permset$

$principalset' = principalset$

$Companies' = Companies$

$Users' = Users$

$Resourceset' = Resourceset$

$assoc\_princ' = assoc\_princ$

$catset' = catset \cup c?$

$Category\_Type' = Category\_Type \cup \{c : C \mid c \in c? \bullet c \mapsto t?\}$

$Environment' = Environment$

$par' = par$

$pca' = pca$

$arca' = arca$

$pfar' = pfar$

$farca' = farca$

addCompany

$ModelRetained$

$ModelRelRetained$

$PrivSetsRet$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$c? : company$

$\{cpr(c?)\} \cap principalset = \emptyset$

$forbset' = forbset$

$permset' = permset$

$principalset' = principalset \cup \{cpr(c?)\}$

$Companies' = Companies \cup \{c?\}$

$Users' = Users$

$assoc\_princ' = assoc\_princ$

$catset' = catset$

addUser

$ModelRetained$

$ModelRelRetained$

$PrivSetsRet$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$u? : user$

$\{upr(u?)\} \cap principalset = \emptyset$

$forbset' = forbset$

$permset' = permset$

$principalset' = principalset \cup \{upr(u?)\}$

$Companies' = Companies$

$Users' = Users \cup \{u?\}$

$assoc\_princ' = assoc\_princ$

$catset' = catset$

$Resourceset' = Resourceset$

addUsertoFriendGroup

SetupRetained

PrivSetsRet

SolidSetsRet

ComponentsRet

AssocDataRet

ThresholdRet

$v? : \mathbb{P}\, Perm$

$w? : \mathbb{P}\, Forb$

$u? : user$

$u? \in Users$

$u? \notin \{a : \{principalOI\} \bullet a.u\}$

$\{upr(u?)\} \cap \{fr : friend \bullet fr\} = \emptyset$

$v? = \textbf{if } 2 \in \text{dom } GtoClink$

$\quad \textbf{then } \{p : Perm \mid (\forall\, c : C \mid (2, c) \in GtoClink \bullet (c \mapsto p) \in arca) \bullet p\} \textbf{ else } \emptyset$

$w? = \textbf{if } 2 \in \text{dom } GtoClink$

$\quad \textbf{then } \{f : Forb \mid (\forall\, c : C \mid (2, c) \in GtoClink \bullet (c \mapsto f) \in farca) \bullet f\} \textbf{ else } \emptyset$

$principalset' = principalset$

$closeFriend' = closeFriend \setminus \{upr(u?)\}$

$acquaintance' = acquaintance \setminus \{upr(u?)\}$

$friend' = friend \cup \{upr(u?)\}$

$GtoClink' = GtoClink$

$assoc\_princ' = assoc\_princ$

$cat\_link' = cat\_link$

$group' = group$

$catset' = catset$

$pca' = \textbf{if } 2 \in \text{dom } GtoClink$

$\quad \textbf{then } pca \cup \{x : GtoClink \mid first\, x = 2 \bullet upr(u?) \mapsto second\, x\} \textbf{ else } pca$

$par' = \textbf{if } 2 \in \text{dom } GtoClink$

$\quad \textbf{then } par \cup \{y : v? \bullet (upr(u?) \mapsto y)\} \textbf{ else } par$

$arca' = arca$

addUsertocloseFriendGroup

SetupRetained

PrivSetsRet

SolidSetsRet

ComponentsRet

AssocDataRet

ThresholdRet

$u? : user$

$v? : \mathbb{P}\, Perm$

$u? \in Users$

$u? \notin \{a : \{principalOI\} \bullet a.u\}$

$\{upr(u?)\} \cap \{closeFriendf : closeFriend \bullet closeFriendf\} = \emptyset$

$v? = \textbf{if } 3 \in \text{dom } GtoClink$

$\quad \textbf{then } \{p : Perm \mid (\forall\, c : C \mid (3, c) \in GtoClink \bullet (c \mapsto p) \in arca) \bullet p\} \textbf{ else } \emptyset$

$principalset' = principalset \cup \{upr(u?)\}$

$principalset' = principalset$

$closeFriend' = closeFriend \cup \{upr(u?)\}$

$acquaintance' = acquaintance \setminus \{upr(u?)\}$

$friend' = friend \setminus \{upr(u?)\}$

$GtoClink' = GtoClink$

$group' = group$

$cat\_link' = cat\_link$

$assoc\_princ' = assoc\_princ$

$catset' = catset$

$pca' = \textbf{if } 3 \in \text{dom } GtoClink$

$\quad \textbf{then } pca \cup \{x : GtoClink \mid first\, x = 2 \bullet upr(u?) \mapsto second\, x\} \textbf{ else } pca$

$par' = \textbf{if } 3 \in \text{dom } GtoClink$

$\quad \textbf{then } par \cup \{w : v? \bullet (upr(u?) \mapsto w)\} \textbf{ else } par$

$arca' = arca$

$pfar' = pfar$

$farca' = farca$

addFriendGrouptoCategory

$SetupRetained$

$PrivSetsRet$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$cat : C$

$v : \mathbb{P}\, Perm$

$v = \textbf{if } cat \in \text{dom } arca \textbf{ then } \{p : Perm \mid cat \mapsto p \in arca \bullet p\} \textbf{ else } \emptyset$

$friend \neq \emptyset$

$pca' = pca \cup \{p : friend \bullet (p \mapsto cat)\}$

$arca' = arca$

$par' = \textbf{if } cat \in \text{dom } arca \textbf{ then } par \cup \{p : friend;\ w : v \bullet (p \mapsto w)\}$

$\qquad\qquad \textbf{else } par$

$pfar' = pfar$

$farca' = farca$

$closeFriend' = closeFriend$

$friend' = friend$

$acquaintance' = acquaintance$

$assoc\_princ' = assoc\_princ$

$GtoClink' = GtoClink \cup \{(2 \mapsto cat)\}$

$group' = group$

E.6 Schemas for Removals

RemUser

ΔPE_SCBAC

$ModelRelRetained$

$PrivSetsRet$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$p : user$

$au? : \mathbb{F}\ Auth$

$upr(p) \in principalset$

$au? = \textbf{if}\ upr(p) \in \text{dom}\ par$

$\quad \textbf{then}\ \{m : permset \mid upr(p) \mapsto m \in par \bullet (upr(p), m)\}$

$\quad \textbf{else}\ \emptyset$

$pca' = \{upr(p)\} \lhd pca$

$arca' = arca$

$par' = par \setminus au?$

$farca' = farca$

$pfar' = pfar$

$forbset' = forbset$

$permset' = permset$

$principalset' = principalset \setminus \{upr(p)\}$

$catset' = catset$

$assoc\_data' = assoc\_data$

RemCat

ΔPE_SCBAC

$ModelRelRetained$

$PrivSetsRet$

$SolidSetsRet$

$AssocDataRet$

$ThresholdRet$

$c : C$

$pe? : \mathbb{F}\ Perm$

$c \in catset$

$pe? = \textbf{if}\ c \in \text{dom}\ arca\ \textbf{then}\ \{m : permset \mid c \mapsto m \in arca\}$

$\quad\quad \textbf{else}\ \emptyset$

$pca' = pca \rhd \{c\}$

$arca' = \{c\} \lhd arca$

$par' = par \rhd pe?$

$farca' = farca$

$pfar' = pfar$

$Category\_Type' = \{c\} \lhd Category\_Type$

$forbset' = forbset$

$permset' = permset$

$principalset' = principalset$

$catset' = catset \setminus \{c\}$

$group' = group$

$Resourceset' = Resourceset$

$Environment' = Environment$

RemovePerms

ΔPE_SCBAC

$ModelRelRetained$

$PrivSetsRet$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$allocation? : \mathbb{F}_1\ Perm$

$permset \neq \emptyset$

$\forall p : permset \bullet second\ p \notin K$

$allocation? \subseteq permset$

$forbset' = forbset$

$permset' = permset \setminus allocation?$

$par' = par \rhd allocation?$

$pca' = pca$

$arca' = arca \rhd allocation?$

$pfar' = pfar$

$farca' = farca$

$principalset' = principalset$

$catset' = catset$

E.7 Schema for Permissions and Forbiddances

CategoryForbs

$UpdateARCA$

$ModelRelRetained$

$PrivSetsRet$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$c? : C$

$f? : Forb$

$c? \in \operatorname{ran} ACL\_CatLink$

$c? \notin \operatorname{ran} pca$

$c? \mapsto f? \notin farca$

$farca' = farca \cup \{c? \mapsto f?\}$

$pfar' = pfar$

$pca' = pca$

$par' = par$

$arca' = arca$

ReinstatePerms

$UpdateARCA$

$ModelRelRetained$

$PrivSetsRet$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$pe : \mathbb{F}\, Perm$

$d : \mathbb{F}\, Forb$

$p? : P$

$ca : \mathbb{F}\, C$

$p? \in \operatorname{dom} pfar$

$d = \{f : pfar \mid first\, f = p? \bullet second\, f\}$

$pe = \{ac : A;\ de : d \mid fbd(ac) = first\, de \bullet (ac, second\, de)\}$

$\{ca\} = \{pc : PC\_link \mid first\, pc = p? \bullet second\, pc\}$

$pca' = pca \cup \{c : ca \bullet (p? \mapsto c)\}$

$arca' = arca$

$farca' = farca$

$par' = par \cup \{perm : pe \bullet p? \mapsto perm\}$

$pfar' = \{p?\} \lhd pfar$

*CategoryPerms*0

$$
\begin{array}{l}
UpdateARCA \\
PrivSetsRet \\
SolidSetsRet \\
ComponentsRet \\
AssocDataRet \\
ThresholdRet \\
c? : C \\
p? : Perm \\
\hline
c? \notin \operatorname{ran} pca \\
c? \mapsto p? \notin arca \\
arca' = arca \cup \{c? \mapsto p?\} \\
farca' = farca \\
pfar' = pfar
\end{array}
$$

*CategoryPerms*1

UpdateCatPerms

PrivSetsRet

SolidSetsRet

ComponentsRet

AssocDataRet

ThresholdRet

$c? : C$

$pe? : Perm$

$v? : \mathbb{P}\, Auth$

$v? = \{m : pca \mid second\ m = c? \bullet (first\ m, pe?)\}$

$arca' = arca \cup \{c? \mapsto pe?\}$

$par' = par \cup v?$

$pca' = pca$

$farca' = farca$

$pfar' = pfar$

E.8 General Operational Schemas

adjustSkewScore

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$i? : \mathbb{N}$

$r? : R$

$t? : Trait$

$x : RsrcSkewScore$

$r? \in Resourceset$

$x \notin skscores$

$x.rsrc \notin \{s : skscores \bullet s.rsrc\} \cap$

$\{s : skscores \mid s.tr = t? \land s.rsrc = x.rsrc \bullet s.rsrc\}$

$r? \notin \{c : container \bullet cresource(c)\}$

$r? \notin \{a : acl\_rsrc \bullet aclrsrc(a)\}$

$r? \notin \{i : Identifier\_Data \bullet id\_data(i)\}$

$i? < 7$

$x.score = i?$

$x.rsrc = r?$

$x.tr = t?$

$x.read = f$

$T' = T$

$T\_u' = T\_u$

$T\_s' = T\_s$

$K' = K$

$Kc\_Data' = Kc\_Data$

$Li' = Li$

$E' = E$

CreateResource

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$PrivSetsRet$

$AssocDataRet$

$ThresholdRet$

$GenResource$

$o : user$

$r? : active\_object$

$m?, sm : metadata\_object$

$a : acl\_rsrc$

$rc? : root\_container$

$p : pod$

$sm = san(m?)$

$cresource(root\_con(rc?)) \mapsto a \in ACL\_Link$

$o \mapsto p \in Pod\_owner$

$o \mapsto cresource(root\_con(rc?)) \in Owns$

$arsrc(r?) \notin Resourceset$

$Category\_Type' = Category\_Type$

$Resourceset' = Resourceset \cup \{arsrc(r?), mrsrc(m?)\}$

$Pod\_owner' = Pod\_owner$

$Owns' = Owns \cup \{o \mapsto arsrc(r?)\} \cup \{o \mapsto mrsrc(m?)\}$

$Data\_pod' = Data\_pod \cup \{p \mapsto arsrc(r?)\} \cup \{p \mapsto mrsrc(m?)\}$

$ACL\_Link' = ACL\_Link \cup \{arsrc(r?) \mapsto a\}$

$ACL\_CatLink' = ACL\_CatLink$

$Contains' = Contains \cup \{root\_con(rc?) \mapsto arsrc(r?)\}$

$Owners' = Owners \cup \{o\}$

$assoc\_data' = assoc\_data \cup \{r? \mapsto m?\}$

$podtoC' = podtoC$

$PC\_link' = PC\_link$

GenAssocData

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$PrivSetsRet$

$ThresholdRet$

$GenResource$

$o : user$

$u? : user$

$m_1, m_2 : metadata\_object$

$l : link\_descrptn$

$dl : DataLink$

$a : acl\_rsrc$

$rc? : root\_container$

$p : pod$

$o \neq u?$

$o = principalOI.u$

$dl.m_1 = m_1$

$dl.m_2 = m_2$

$dl.tr = t$

$dl.l = l$

$cresource(root\_con(rc?)) \mapsto a \in ACL\_Link$

$o \mapsto p \in Pod\_owner$

$o \mapsto cresource(root\_con(rc?)) \in Owns$

$ACL\_CatLink' = ACL\_CatLink$

$Contains' = Contains \cup \{root\_con(rc?) \mapsto mrsrc(m_1)\}$

$Owners' = Owners \cup \{o\} \cup \{u?\}$

$Category\_Type' = Category\_Type$

$Resourceset' = Resourceset \cup \{mrsrc(m_1)\}$

$Pod\_owner' = Pod\_owner$

$Owns' = Owns \cup \{o \mapsto mrsrc(m_1)\}$

Collect

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$c_1?, c_2 : company$

$r? : R$

$n : \mathbb{N}$

$c : C$

$n \leq 3$

$n = \textbf{if } c_1? \in \text{dom } Threshold \textbf{ then } Threshold(c_1?) + 1 \textbf{ else } 1$

$r? \in \{s : skscores \bullet s.rsrc\}$

$c_1? \neq c_2$

$c_1? \mapsto r? \notin Kc\_Data$

$r? \notin T\_s$

$r? \notin \{i : Identifier\_Data \bullet id\_data(i)\}$

$r? \notin \{c : container \bullet cresource(c)\}$

$r? \in Resourceset$

$T\_s' = T\_s$

$T' = T \cup \{r?\}$

$T\_u' = T\_u \cup \{r?\}$

$Kc\_Data' = \textbf{if } (c_1?, c_2) \in assoc\_princ$

$\textbf{then } Kc\_Data \cup \{(c_1?, r?), (c_2, r?)\}$

$\textbf{else } Kc\_Data \cup \{(c_1?, r?)\}$

$Li' = Li$

$K' = K \cup \{r?\}$

$Pr' = Pr \cup \{r?\}$

$E' = E$

$Sk' = Sk$

AddtoL

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$c? : company$

$Threshold(c?) = 3$

$T\_s' = T\_s$

$T' = T$

$T\_u' = T\_u$

$Kc\_Data' = Kc\_Data$

$Li' = Li \cup \{c?\}$

$K' = K$

$Pr' = Pr$

$E' = E$

$Sk' = Sk$

$skewn' = skewn$

$skscores' = skscores$

$Threshold' = Threshold$

ForbAgent

$SetupRetained$

$ModelRelRetained$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$PrivSetsRet$

$c? : company$

$f? : \mathbb{F}\ Forb$

$ca?, cb : C$

$f? = \{x : par \mid first\ x = cpr(c?)$

$\bullet\ fbd(first(second\ x)) \mapsto second(second\ x)\}$

$pca' = \{cpr(c?)\} \lhd pca \cup \{cpr(c?) \mapsto ca?\}$

$par' = \{cpr(c?)\} \lhd par$

$pfar' = pfar \cup \{b : par \mid first\ b = cpr(c?)$

$\bullet\ first\ b \mapsto (fbd(first(second\ b)), second(second\ b))\}$

$arca' = arca$

$farca' = farca \cup \{fo : f? \bullet ca? \mapsto fo\}$

Limit

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$c? : company$

$m, n, o : \mathbb{P}\, R$

$c? \in Li$

$m = \{d : metadata\_object;\ kc : Kc\_Data \mid (mrsrc(d)) = second\, kc \land$

$first\, kc = c? \bullet (mrsrc(d))\}$

$n = \{n : metadata\_object;\ kc : Kc\_Data \mid (mrsrc(n)) = second\, kc \land$

$first\, kc \neq c? \bullet (mrsrc(n))\}$

$o = m \cap n$

$Kc\_Data' = Kc\_Data \setminus \{m_0 : m \bullet (c?, m_0)\}$

$Li' = Li \setminus \{c?\}$

$Pr' = Pr \setminus m \cup n$

$T\_s' = T\_s \cup (m \setminus o)$

$T\_u' = T\_u \setminus (m \setminus o)$

$K' = K \setminus m \cup n$

$E' = E$

$Sk' = Sk$

$skewn' = skewn$

$skscores' = skscores$

$Threshold' = Threshold \setminus \{a : Threshold \mid first\, a = c?\}$

$$TrueLimit \mathrel{\widehat=} \exists\, m, n : \mathbb{P}\, Data \bullet Limit$$

AddtoT_s

SetupRetained

ModelRelRetained

ModelRetained

SolidSetsRet

ComponentsRet

AssocDataRet

ThresholdRet

$r : R$

$r \in Resourceset$

$r \notin Pr$

$T\_s' = T\_s \cup \{r\}$

$T' = T \cup \{r\}$

$T\_u' = T\_u$

$K' = K$

$Li' = Li$

$Pr' = Pr$

$E' = E$

$Sk' = Sk$

$skewn' = skewn$

$skscores' = skscores$

$Kc\_Data' = Kc\_Data$

$\forall p : permset \bullet r \neq second\ p$

$Ac' = Ac$

Skewing

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$ctr : \mathbb{N}$

$c : company$

$score : \mathbb{N}$

$u : user$

$tr? : Trait$

$r? : metadata\_object$

$i? : Identifier\_Data$

$s?, k? : Skew$

$score \leq 5$

$mrsrc(r?) \in K$

$k? \in skewn \lor (k?.ctr = 0 \land k?.skewValue = L)$

$s? \neq k?$

$s? \notin skewn$

$s?.tr = k?.tr$

$s?.usr = k?.usr$

$s?.comp = k?.comp$

$s?.ctr = \textbf{if } k? \in skewn \textbf{ then } k?.ctr + score \textbf{ else } 1$

$s?.skewValue = \textbf{if } k?.ctr > 0 \land k?.ctr \bmod 3 = 0$
$\textbf{then } increase\_skew(k?.skewValue)$
$\textbf{else } k?.skewValue$

$skewn' = \textbf{if } k? \in skewn \textbf{ then } skewn \setminus \{k?\} \cup \{s?\}$
$\textbf{else } skewn \cup \{s?\}$

$skscores' = skscores$

HideTrait

$SetupRetained$

$ModelRetained$

$ModelRelRetained$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$t? : Trait$

$i : \mathbb{F}\ Identifier\_Data$

$i \neq \emptyset$

$\{i_1 : i \bullet id\_data(i_1)\} \cap T\_s = \emptyset$

$\forall\, id : i \bullet first\ id = t?$

$Kc\_Data' = Kc\_Data$

$K' = K$

$Pr' = Pr$

$T\_s' = T\_s \cup \{i0 : i \mid second\ i0 = NULL \bullet id\_data(i0)\}$

$T\_u' = T\_u$

$T' = T \cup \{i0 : i \bullet id\_data(i0)\}$

$E' = E$

$Li' = Li$

$Sk' = Sk$

$skewn' = skewn$

$skscores' = skscores$

$Ac' = Ac$

setEnvironnment

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$SolidSetsRet$

$AssocDataRet$

$ThresholdRet$

$PrivSetsRet$

$v : SetEnvVars$

$e? : Env$

$v.e = e?$

$Environment \neq v$

$Environment.m = v.m$

$Environment' = v$

$Resourceset' = Resourceset$

$Category\_Type' = Category\_Type$

setMode

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$SolidSetsRet$

$AssocDataRet$

$ThresholdRet$

$PrivSetsRet$

$v : SetEnvVars$

$m? : Mode$

$v.m = m?$

$Environment \neq v$

$Environment.e = v.e$

$Environment' = v$

$Resourceset' = Resourceset$

$Category\_Type' = Category\_Type$

AddtoExposes

SetupRetained

ModelRelRetained

ModelRetained

SolidSetsRet

ComponentsRet

AssocDataRet

ThresholdRet

$c? : company$

$r : R$

$m? : metadata\_object$

$i : Identifier\_Data$

$r \in Resourceset$

$r = mrsrc(m?)$

$second\ i = m\_ob(m?)$

$c? \mapsto r \notin Kc\_Data$

$r \notin \{i : Identifier\_Data \bullet id\_data(i)\}$

$Kc\_Data' = Kc\_Data \cup \{c? \mapsto r\}$

$K' = \textbf{if}\ id\_data(i) \in T\_s\ \textbf{then}\ K\ \textbf{else}\ K \cup \{r\}$

$Pr' = \textbf{if}\ id\_data(i) \in T\_s\ \textbf{then}\ Pr\ \textbf{else}\ Pr \cup \{r\}$

$T\_s' = \textbf{if}\ id\_data(i) \in T\_s\ \textbf{then}\ T\_s \cup \{r\}\ \textbf{else}\ T\_s$

$T\_u' = \textbf{if}\ id\_data(i) \in T\_s\ \textbf{then}\ T\_u\ \textbf{else}\ T\_u \cup \{r\}$

$T' = T \cup \{r\}$

$E' = E \cup \{r \mapsto i\}$

$Li' = Li$

$Sk' = Sk$

$skewn' = skewn$

$skscores' = skscores$

addAssocCompanytoCategory

$SetupRetained$

$PrivSetsRet$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$c? : C$

$comp? : company$

$u : \mathbb{F}\ company$

$v : \mathbb{P}\ Perm$

$comp? \in \mathrm{dom}\ assoc\_princ \lor comp? \in \mathrm{ran}\ assoc\_princ$

$v = \textbf{if}\ c? \in \mathrm{dom}\ arca\ \textbf{then}\ \{p : Perm \mid c? \mapsto p \in arca \bullet p\}\ \textbf{else}\ \emptyset$

$u = \textbf{if}\ comp? \in \mathrm{dom}\ assoc\_princ$

$\quad \textbf{then}\ \{a : assoc\_princ \mid first\ a = comp? \bullet second\ a\}$

$\quad \textbf{else if}\ comp? \in \mathrm{ran}\ assoc\_princ$

$\quad\quad \textbf{then}\ \{a : assoc\_princ \mid second\ a = comp? \bullet first\ a\}$

$\quad \textbf{else}\ \emptyset$

$pca' = \textbf{if}\ u \neq \emptyset$

$\quad \textbf{then}\ pca \cup \{cpr(comp?) \mapsto c?\} \cup \{c : u \bullet cpr(c) \mapsto c?\}$

$\quad \textbf{else}\ pca \cup \{cpr(comp?) \mapsto c?\}$

$pfar' = pfar$

$farca' = farca$

PtoC

$SetupRetained$

$ModelRelRetained$

$PrivSetsRet$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$u? : user$

$c? : C$

$v? : \mathbb{F}\ Auth$

$\{u?\} \cap \{principalOI.u\} = \emptyset$

$v? = \textbf{if}\ c? \in \text{dom}\ arca$

$\quad\quad \textbf{then}\ \{m : arca \mid first\ m = c? \bullet (upr(u?), second\ m)\}$

$\quad \textbf{else}\ \emptyset$

$upr(u?) \mapsto c? \notin pca$

$pca' = pca \cup \{upr(u?) \mapsto c?\}$

$arca' = arca$

$par' = par \cup v?$

$pfar' = pfar$

$farca' = farca$

CompanytoCategoryAssignment

SetupRetained

ModelRelRetained

PrivSetsRet

SolidSetsRet

ComponentsRet

AssocDataRet

ThresholdRet

$cmp? : company$

$c? : C$

$v? : \mathbb{F}\, Auth$

$v? = \textbf{if}\ c? \in \text{dom}\ arca$

$\quad \textbf{then}\ \{m : arca \mid first\ m = c? \bullet (cpr(cmp?), second\ m)\}$

$\quad \textbf{else}\ \emptyset$

$cpr(cmp?) \mapsto c? \notin pca$

$pca' = pca \cup \{cpr(cmp?) \mapsto c?\}$

$arca' = arca$

$par' = par \cup v?$

$pfar' = pfar$

$farca' = farca$

AllocatePerms

$UpdatePerms$

$PrivSetsRet$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$ThresholdRet$

$allocation? : \mathbb{F}_1\ Perm$

$\forall\, a : A;\ r : R \bullet r \in T\_s \Rightarrow ((a, r) \notin allocation? \lor r \notin Pr)$

$permset = \emptyset$

$forbset' = forbset$

$permset' = allocation?$

Interaction

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$c_1?, c_2 : company$

$r? : R$

$a? : A$

$p : Perm$

$n : \mathbb{N}$

$c : C$

$p = (a? \mapsto r?)$

$p \in permset$

$cpr(c_1?) \mapsto p \in par$

$n \leq 3$

$n = \textbf{if } c_1? \in \text{dom } Threshold \textbf{ then } Threshold(c_1?) + 1 \textbf{ else } 1$

$r? \in \{s : skscores \bullet s.rsrc\}$

$c_1? \neq c_2$

$c_1? \mapsto r? \notin Kc\_Data$

$(cpr(c_1?) \mapsto c) \in pca$

$r? \notin T\_s$

$r? \notin \{i : Identifier\_Data \bullet id\_data(i)\}$

$r? \notin \{c : container \bullet cresource(c)\}$

$r? \in Resourceset$

$T\_s' = T\_s$

$T' = T$

$T\_u' = T\_u$

$Kc\_Data' = \textbf{if } (c_1?, c_2) \in assoc\_princ$

$\textbf{then } Kc\_Data \cup \{(c_1?, r?), (c_2, r?)\}$

$$
\begin{aligned}
&skewn' = skewn \\
&skscores' = skscores \\
&Threshold' = \ \textbf{if}\ (c_1?, c_2) \in assoc\_princ \\
&\qquad\qquad \textbf{then}\ \{c_1?, c_2\} \lhd Threshold \cup \{(c_1?, n)\} \cup \{(c_2, n)\} \\
&\qquad\quad \textbf{else}\ Threshold \cup \{(c_1?, n)\}
\end{aligned}
$$

UserInteraction

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$SolidSetsRet$

$ComponentsRet$

$ThresholdRet$

$AssocDataRet$

$e : Environment$

$u : user$

$r? : R$

$a? : A$

$p : Perm$

$c : C$

$e.m = Off$

$p = (a? \mapsto r?)$

$p \in permset$

$upr(u) \mapsto p \in par$

$r? \in \{s : skscores \bullet s.rsrc\}$

$c_1? \neq c_2$

$c_1? \mapsto r? \notin Kc\_Data$

$(cpr(c_1?) \mapsto c) \in pca$

$(Let' sseehowthisgoes \ldots)r? \notin T\_s$

$r? \notin \{i : Identifier\_Data \bullet id\_data(i)\}$

$r? \notin \{c : container \bullet cresource(c)\}$

$r? \in Resourceset$

$T\_s' = T\_s$

$T' = T$

$T\_u' = T\_u$

$Kc\_Data' = Kc\_Data$

$Li' = Li$

PrivSanDataInteraction

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$e : Environment$

$t : Timer$

$c_1? : company$

$r?, sr : R$

$a? : A$

$p, sp : Perm$

$n : \mathbb{N}$

$c : C$

$e.m = On$

$t.b = true$

$sr = smrsrc(san(returnmeta(r?)))$

$p = (a? \mapsto r?)$

$p \in permset$

$sp = (a? \mapsto sr)$

$permset' = permset \cup \{sp\} cpr(c_1?) \mapsto p \in par$

$n < 3$

$n = \textbf{if } c_1? \in \text{dom } Threshold \textbf{ then } Threshold(c_1?) + 1 \textbf{ else } 1$

$c_1? \mapsto sr \notin Kc\_Data$

$(cpr(c_1?) \mapsto c) \in pca$

$(Let's seehowthisgoes \ldots) r? \notin \{i : Identifier\_Data \bullet id\_data(i)\}$

$r? \notin \{c : container \bullet cresource(c)\}$

$r? \in Resourceset$

$T\_s' = T\_s$

$T' = T$

$$K' = K \cup \{sr\}$$

$$Pr' = Pr \cup \{sr\}$$

$$E' = E$$

$$Sk' = Sk$$

$$skewn' = skewn$$

$$skscores' = skscores$$

$$Threshold' = \textbf{if } (c_1?, c_2) \in assoc\_princ$$

$$\textbf{then } \{c_1?, c_2\} \lhd Threshold \cup \{(c_1?, n)\} \cup \{(c_2, n)\}$$

$$\textbf{else } Threshold \cup \{(c_1?, n)\}$$

SanDataInteraction

$SetupRetained$

$ModelRelRetained$

$ModelRetained$

$SolidSetsRet$

$ComponentsRet$

$AssocDataRet$

$e : Environment$

$c_1? : company$

$r?, sr : R$

$ac : acl\_resource$

$a? : A$

$p, sp : Perm$

$n : \mathbb{N}$

$c : C$

$e.m = On$

$sr = smrsrc(san(returnmeta(r?)))$

$p = (a? \mapsto r?)$

$p \in permset$

$sp = (a? \mapsto sr)$

$permset' = permset \cup \{sp\} cpr(c_1?) \mapsto p \in par$

$n < 3$

$n = \textbf{if } c_1? \in \text{dom } Threshold \textbf{ then } Threshold(c_1?) + 1 \textbf{ else } 1$

$c_1? \mapsto sr \notin Kc\_Data$

$(cpr(c_1?) \mapsto c) \in pca$

$(Let' sseehowthisgoes \ldots) r? \notin \{i : Identifier\_Data \bullet id\_data(i)\}$

$r? \notin \{c : container \bullet cresource(c)\}$

$r? \in Resourceset$

$T\_s' = T\_s$

$T' = T$

$T\_u' = T\_u \cup sr$

...

$K' = K \cup \{sr\}$

$Pr' = Pr \cup \{sr\}$

$E' = E$

$Sk' = Sk$

$skewn' = skewn$

$skscores' = skscores$

$Threshold' = \textbf{if } (c_1?, c_2) \in assoc\_princ$

$\quad \textbf{then } \{c_1?, c_2\} \lhd Threshold \cup \{(c_1?, n)\} \cup \{(c_2, n)\}$

$\textbf{else } Threshold \cup \{(c_1?, n)\}$