DIGITAL CHILD WORKING PAPER SERIES 2026-01

A non-technical researcher's guide to studying mobile tracking and materialising data flows

AUTHORS

Katrin Langton, Rebecca Ng

AUSTRALIAN RESEARCH COUNCIL
Centre of Excellence for the Digital Child.

Australian Government
Australian Research Council

AUTHORS

Langton, K.
Deakin University
Research for Educational Impact
katrin.langton@deakin.edu.au

Ng, R.
University of Wollongong
School of Education
nrebecca@uow.edu.au

SUGGESTED CITATION

Langton K., Ng, R. 2026. A non-technical researcher's guide to studying mobile tracking and materialising data flows. Digital Child Working Paper 2026-01, Australian Research Council Centre of Excellence for the Digital Child, Brisbane, Australia

ISSN/DOI

ISSN: 2653-5270 DOI: https://doi.org/10.26187/y42v-5y26

KEYWORDS

Critical data studies, digital methods, mobile tracking, data privacy, datafication, app studies, app analysis, children

ACKNOWLEDGEMENT/S

This Working Paper was supported by the Australian Research Council Centre of Excellence for the Digital Child (grant #CE200100022). The Australian Research Council Centre of Excellence for the Digital Child acknowledges the First Australian owners of the lands on where we gather and pay our respects to the Elders, lores, customs and creation spirits of this country.

A MESSAGE FROM PROFESSOR SUSAN DANBY, CENTRE DIRECTOR

In 2021, the Australian Research Council (ARC) funded a Centre of Excellence devoted to studying and researching 'the digital child'. The focus of this Centre is on very young children from birth to age 8, and describes and examines their everyday lives with and through digital technologies, their learning and their health in the family, and various kinds of kindergarten, childcare and early primary education experiences.

The Centre brings together six universities across Australia, as well as partner investigators from North America, Asia and Europe and a range of public bodies and civil society stakeholders, to focus on a holistic understanding of what it might mean to 'grow up digital' today.

The Digital Child Working Paper Series reports on our work in progress. There are five series of papers aimed at different audiences:

A **'how to'** series offers instructional papers aimed at early career researchers or those new to the principles and practices of structured review.

A **'discussion'** series consisting of discussion papers aimed at the scholarly community, raising larger conceptual challenges faced by researchers at the Centre and drawing on forms of literature review.

A **'reviews'** series consisting of scoping reviews, literature reviews and systematic reviews, all addressing specific research questions particular to any of the programme disciplines in the Centre.

A **'methods and methodologies'** series consisting of digital research capacity building resource-rich discussion papers, offering more technical support for the research community and allied scholarship. These are more focused on methods and methodologies.

A **'policy'** series consisting of more public facing, policy-oriented papers produced for stakeholder engagement.

Each of the working papers has been authored by members of the Centre and has been subject to review as explained in each paper. The arguments in each paper represent the view of the authors.

We hope that readers find each of these papers stimulating and generative and that all sections of society can draw on the insights, arguments and ideas within the papers to create healthy, educated and connected futures for all and every child.

Professor Susan Danby

Director, Centre of Excellence for the Digital Child

June 2022

EXECUTIVE SUMMARY

This paper is part of a series consisting of digital research capacity building resource-rich discussion papers, offering more technical support for the research community and allied scholarship. This series is more focused on methods and methodologies.

This paper has been checked by the sub-series editorial team to ensure it meets basic standards around clarity of expression and acceptable and inclusive language. It has also been presented in a seminar held by the ARC Centre of Excellence for the Digital Child, and any feedback given has been considered.

The convenience and accessibility of mobile applications for young children and their families have made these technologies ubiquitous tools – used to manage family life, for entertainment, education, or to maintain social connection. These mundane digital engagements result in the conversion of social action into digital data points or the 'datafication' of everyday life. The data generated about children and families are commonly monetised through data-sharing with third-parties for a range of purposes including advertising, automated decision-making and to make predictions about users' behaviours, shaping their lives in unknown (and unknowable) ways. Many of these data-sharing practices are invisible to us as everyday users - only mentioned in pop-ups, in-app notifications requesting our permission for data-sharing or vaguely described in the privacy policies of digital services. Making such data flows visible provides important opportunities to interrogate and critically reflect on their implications.

We prepared this paper as part of our own learning journeys, to facilitate the study of data flows for researchers with limited technical skills. While the first section focusses on making the terminology and mechanisms of tracking accessible at a more general level, the second and most extensive section of the paper includes more detailed examples of commonly used approaches to materialise and study the data flows of mobile applications. Notably, due to the rapidly changing nature of the digital environment, some of the methods and tools described in this paper may no longer be functional today. However, the point of the paper is not to provide an exhaustive and updated list of all the approaches available. Rather, it aims to provide an overview of commonly used methods and tools used at the point of writing, which has enduring relevance in chronicling how approaches are developing, even beyond the current context.

We hope this paper will support others in working towards developing the skills and knowledge needed to materialise and investigate the data flows of the everyday mobile technologies that 'datafy' even the most mundane, yet intimate parts of our lives.

Table of Contents

Rationales and significance of studying mobile tracking in the context of digital childhoods and family life

Extent and prevalence of mobile tracking

The large-scale tracking and sharing of user data through engagement with online services such as websites and mobile applications, has become the norm. Media scholars like Tanya Kant (2021) point to how data mining and the subsequent processing of user data - such as its aggregation into consumer profiles for targeted advertising - have become "*the* driving economic resource for the contemporary free-to-use web" (p. 3, italics in original). In the context of mobile technologies specifically, Flensburg and Lai (2022) describe "data harvesting, mining and distribution" as the "commercial [and] functional backbone of mobile communication" (p. 136).

For any user of online services, there are many potential risks and opportunities for unanticipated use attached to the sharing of data with platforms and third parties online. Of particular concern are targeted content and advertising, enabled by algorithmic categorisation (Kant, 2021). The information collected or 'harvested' about us during our digital engagements – our online behaviours and interests (e.g. online searches, what we click on and engage with) – is sorted to determine or infer particular identity markers. These are used to classify us into consumer categories or 'measurable types' (Cheney-Lippold, 2017), to predict or anticipate our current and future needs and wants. At the same time, the assumptions made about us flatten our identities - suggesting to others and ourselves *who we are* and *what we need*, according to commercial imperatives.

To enable advocacy and increased control over data-sharing at the level of the everyday user, data privacy legislation such as the General Data Protection Regulation (Wolford, 2026) seeks to increase the transparency of data collection and -sharing practices. Yet, many of the user-directed information for this purpose, such as mobile apps' privacy policies, continue to be excessively long and vague. Additionally, users commonly find it difficult to opt-out of services that have become essential for their participation in social, economic and civic life. Instead of empowering users, these dynamics frequently have the effect of promoting an attitude of resignation towards online tracking (Draper & Turow, 2019).

Mobile tracking in family life

In the context of contemporary parenthood, smartphones and mobile applications have become the 'swiss-army knife' of parenting, as key tools for information-seeking, health monitoring, engagement with educational institutions, and everyday communication (Langton, 2024). As a result, children and their families belong to some of the most datafied groups of the population, promoted by a range of contextual factors. Couples with children residing in highly digitally connected countries - such as Australia - are amongst the most digitally included and active users of online services (Thomas et al., 2023). The recent COVID-19 Pandemic further pushed many elements of caregiving, education and kinship connection into online spaces (Willett & Zhao, 2024). Additionally, a recent Danish study (Andelsman Alvarez, 2024) found that the affordances of

digital technologies had become an integral enabler of what parents perceived as a 'good life' for their families - facilitating connection and the management of the everyday logistics of family life. Even very young children today commonly have access to internet-connected devices, often including a range of apps for connection, play and education (Langton et al., 2025). Hence, much of the datafication and tracking of children and family life today, takes place via mobile technologies.

While online service providers like to emphasise the 'benefits' of tracking to enable targeting content and information deemed more relevant and personalised to users, the combination of our personal data traces and their algorithmic categorisation can also significantly influence our life opportunities. Our 'data selves' (Lupton, 2020) - based on the information we give, and the digital traces collected about us – increasingly come to speak for us in unknown and unanticipated ways, and without the ability to change them. These profiles can be used to discriminate against users - for instance on the basis of race (Noble, 2018) or income (Hao, 2020), and they increasingly determine the content and information we have access to online (Ito et al., 2023), our self-understanding, and the decisions we make.

These practices are particularly concerning in the context of data collection about children, which frequently begins even before birth (Barassi, 2020; Langton & Ng, 2025). Common examples of such data-sharing include parental use of apps and other digital tools for parenting support or 'caring dataveillance' (Mascheroni & Siibak, 2021) - the well-intentioned, data-driven monitoring of children's wellbeing and safety. Similar practices continue as children get older, for example the use of communication apps that connect parents and daycare services (Andelsman Alvarez & Meleschko, 2024), or 'parental control' apps (Pangrazio, 2021) to manage children's digital engagements. Still, little is known about what kinds of children data many app-based tools actually collect and share, making it difficult to advocate for changes in technology design, or help parents choose 'safer' apps and practices to minimise the risk of negative implications.

Usefulness of materialising data flows

Driven by broader trends towards the platformization[1] of family life (Sefton-Green et al., 2025), access to personal data is increasingly concentrated, and within the control of a small number of powerful technology companies. These entities are then able to analyse, categorise, interpret, and predict things about us from this data for an ever-increasing range of purposes that it is impossible to meaningfully consent to (Okoyomon, 2019).

The affordances of hand-held and mobile devices mean that a wide range of personal information can be shared by mobile apps as people and their devices move through space to go about their everyday activities (Chao et al., 2024). These dynamics make the study of app-facilitated data flows an important complement to research on web-based tracking, to better understand different online tracking 'ecosystems' (Nikiforakis et al., 2013; Binns et al., 2018; Razaghpanah et

[1] UK is used throughout this paper; however, 'platformization' is a specialised term and we opted to keep its original spelling, which uses the 'z'.

al., 2018). While it is important not to conflate *visibility* with *transparency* (Ananny & Crawford, 2018), materialising mobile data flows provides important opportunities to think about and interrogate data-sharing. Considering the opacity of the data-broker economy (Crain, 2018), it seems especially important to improve our understanding of data flows *before* they enter these networks, and advocate for data protection or minimisation where possible.

Developing and trialling methods to explore mobile data flows can also make explicit the limitations of what researchers and everyday users are able to find out and understand about the mechanisms and implications of data-sharing. These insights can inform future research approaches that push these boundaries, provide recommendations for privacy-by-design, and indicate pathways for meaningful change at the level of government policy and regulation. It can also support users to adopt practices that better protect the privacy of their own and family members' data. Facilitating more of this kind of work is the aim of this paper.

A researcher's guide to understanding the available methods to materialise the data flows of mobile applications

Introduction

In the context of our datafied lives, exploring the implications of datafication and data flows is a critical component of studying the reciprocal shaping of technology and culture. Hence, researchers from a range of social sciences disciplines seek to understand and make claims and recommendations about the roles of data in contemporary society. Disciplines such as media and communications, science and technology studies and cultural studies have different theories, viewpoints and methodologies to interpret data cultures. Yet, many of the data-sharing practices that monetise and shape our everyday digital experiences are invisible to us as everyday users - only mentioned in pop-ups, in-app notifications requesting our permission for data-sharing, or vaguely described in the privacy policies of digital services. To be able to interrogate and critically reflect on data flows, we believe it is essential to attempt to make them visible. However, these endeavours frequently require a level of technical knowledge and skill that goes beyond those of many social sciences researchers. We prepared this paper as part of our own learning journeys, to facilitate the study of data flows for researchers with limited technical skills. We hope it will support others in working towards developing the skills and knowledge needed to materialise and investigate the data flows of the everyday mobile technologies that facilitate even the most intimate parts of our lives.

Aim and structure of paper

The aim of this paper is two-fold: (1) It provides a basic introduction to online tracking as the practice of collecting information about users and their online activity, and (2) it outlines the available methods to 'track the trackers' of users' personal data. To this end, the paper is divided into two main sections.

Section one provides a high-level overview of the different tools used for online tracking, the different parties who engage in these activities, and for what purposes, and the implications of tracking for those whose data is being collected. To reflect on the significance of studying these everyday data flows, some of the key concerns and implications of mobile data sharing for families with young children are presented in the closing section of part one.

Section two begins with a literature review of the methodological approaches to exploring the data-sharing and tracking practices of online services that have been employed over the last decade. It then highlights the importance and rationales for studying mobile apps' data flows in particular. Following this contextual overview, the section outlines the available methods to make data-sharing explicit and visible - offering methods to effectively 'track the trackers' in return. The section focuses specifically on methods aiming to materialise the flows of personal data facilitated through the everyday use of mobile applications - such as data shared in registration processes, or data traces generated during app use.

The document is structured to allow readers to jump to the sections they are most interested in, rather than having to read through the whole document. Summaries are provided at the end of each section, to facilitate the review of key concepts, or as a primer, prior to moving on to the next section.

Section 1: Online tracking: an introduction

1.1 What is tracking?

The term *online tracking* describes the processes by which any available information on users of online services is observed, collected, analysed and stored, and often shared and aggregated outside of the initial context of use (Norton, 2022; Bahar, 2022; Parry, 2025). This more general term encompasses all types of tracking in digital spaces. However, we can broadly distinguish different infrastructural contexts of tracking, and divide these practices into *web-tracking* - which takes place during online searches and web-browsing - and *app-tracking* - facilitated specifically through the data-sharing infrastructures of individual mobile applications (Hanna, 2024).

The current economic model of the "data-for-services web" (Kant, 2021, p. 14), means that users are commonly permitted to download apps and use popular online platforms and services "for free" - meaning in exchange for their personal data, rather than direct monetary payment. Most online tracking practices therefore serve the purpose of systematically collecting data about users to enable targeted advertising and 'analytics' (Binns et al., 2018)[2] - reflective of Kant's (2021) proposition that "user targeting underpins the online economy as we know it" (p. 3).

Academic definitions and analyses of online tracking reflect a critical stance towards these practices. They draw attention to the unequal distribution of power in who gets to benefit from the collection of user data through online tracking, and highlight a range of ethical concerns arising from the systematic targeting and profiling of users that these practices support. Computer scientists such as Binns and colleagues (2018) take a more descriptive, theoretical approach in outlining social justice issues that may arise from the tracking of user data and its use for user profiling[3]. Social scientists such as Sofiya Noble (2018) have focussed on providing explicit evidence for the racist outcomes of algorithmic categorisation that can arise from online tracking at scale; John Cheney-Lippold (2017) has similarly highlighted the dangers of automated categorisation and subsequent sorting and stratification of users. Algorithmic predictions based on this information frequently form the basis of automated decision making, including for instance what information users are presented with, and even what opportunities (e.g. rental properties, bank loans) they have access to.

Sylvia E Peacock (2014) finds particularly strong language to describe these practices, asserting that online tracking "is the *unseen* and *unauthorised* extraction, storage, analysing, selling, buying,

[2] *Analytics* in the context of online tracking is a vague term that can refer to technical analytics such as crash reporting or the collection of data about how users interact with a website or app to improve its design and functionality, the same term can also mean 'marketing analytics' - which refers to analysing user behaviour to gauge the success of advertising and marketing strategies - serving a similar purpose to 'advertising' trackers.

[3] The GDPR defines *profiling* as "any form of automated processing of personal data consisting of the use of personal data to evaluate certain personal aspects relating to a natural person, in particular to analyse or predict aspects concerning that natural person's performance at work, economic situation, health, personal preferences, interests, reliability, behaviour, location or movements" (Article 4(4)).

and auctioning of personal online data *appropriated* by one or more remote online corporate actors" (Peacock, 2014, p. 1) [italics added for emphasis]. This description points to some key critiques and enduring issues with online tracking, specifically its often opaque or covert nature, which limits or removes the ability for users to provide meaningful consent for data collection, its processing, and any unanticipated uses that data may be put to. While Peacock (2014) frames tracking as problematic by default, below we provide an overview of different types of online tracking. This includes distinctions between different levels of visibility or 'transparency', and arguments around the usefulness of tracking for service providers as well as users.

1.2 Exploring data-sharing relationships and tracking practices

Regardless of who is using the internet (e.g. you, me, the person next to you on the bus, the neighbours' kids, Taylor Swift) or how it is accessed (through a mobile device, a desktop computer, a smart TV, or any other internet-connected device), our online activity is constantly being tracked by different entities, and for different purposes, with *first-* and *third-party tracking* often receiving the most attention. However, there are a range of other stakeholders besides first- and third-parties involved in accessing and sharing user data. This section provides an overview of the terms used to distinguish between the different ways that service providers obtain user data, who benefits from these practices, and how visible they are to users. While these relationships are presented separately for ease of explanation, online services today commonly integrate a number of different tracking methods and data-sharing relationships.

1.2.1 Zero-party data

The term **zero-party data** is used to describe data-sharing that is initiated by users themselves. It describes data that users share voluntarily and pro-actively with the provider of an online service - providing insights into user preferences and interests without the need for a provider to use any tracking technologies to obtain this information (Khatibloo et al., 2017). Examples of such data sharing include users explicitly providing feedback on products and services, participating in surveys, or selecting to share information about their consumer or communication preferences, beyond what is required to enable access to the service (Sullivan, 2024; Treanor, 2025).

In some contexts, first- and zero-party data can overlap. For instance, if use of an online service requires the creation of an account (a classic example of first-party data collection), it is often possible for users to voluntarily enter additional information on their interests, preferences, and background. This information is zero-party data, which is not necessary to enable access to the service, but provides an additional avenue to intentionally share contextual information and user/consumer preferences.

User consent

Since the sharing of zero-party data is user-initiated, it represents the highest level of user awareness and consent. Users have complete visibility and control over the kinds of data being shared, and of the mechanisms through which this data is provided.

This type of data is similar to first-party data, as both first- and zero-party data are collected through a direct relationship between the user and the provider of an online service. However, first-party data is often collected more implicitly, for instance through behavioural monitoring as users interact with an online service, while zero-party data is information that is volunteered intentionally and explicitly (Sullivan, 2024).

Who benefits?

For users, a key benefit of zero-party data sharing is its high level of transparency, and the ability for the online service to customise the user experience. Users can gain access to personalised and relevant online services, based on data they shared intentionally, with a high degree of control over what data has been shared with a provider, and via what mechanisms.

A frequently mentioned benefit of the use of zero-party data for tech providers, is that the transparency and voluntary nature of the direct data sharing relationship between the user and the service provider improves trust, results in more reliable insights, and supports a customer relationship with longevity (Webster & Zinni, 2023; Polonioli, 2022). Although the relatively small amounts of zero-party data are not as extensive or comprehensive as any big data collected passively via automated methods, it is considered more accurate because it is provided directly by users, and its collection is unlikely to be hindered by current or future data privacy regulation. While some business commentators go as far as asserting that "zero-party data is the new oil" (Gozman, 2022); alternative mechanisms to collect user data (and track them across online spaces) remain popular (Polonioli, 2022).

1.2.1 First-party tracking

First-party tracking is initiated by the entity owning a website or app (i.e. **the first party**). It includes a range of data collection methods at different levels of visibility to users. The creation of a user account for an app or web-service for instance, includes explicit sharing of user information that is collected by the service provider. However, it also includes the monitoring and tracking of what users do on a site they are visiting, or in the app they are using, such as scrolling through or clicking on content, or engaging with particular features (Parry, 2025).

This information on user behaviour during a 'session'[4] of app or website use is of interest to the service provider/first-party because it indicates what users are interested in, according to how they navigate and use the service (Webster & Zinni, 2023). This behavioural data on users' direct interaction with an online service, which is collected passively by the provider of the service (the first party) - rather than being explicitly and intentionally shared by the user - is called **first party data**.

[4] A web or app 'session' describes all user activity on a website, or in an app, from the time a website or app is opened, to when it is closed. Session activity is usually recorded within a specific pre-set time frame (hence, when you leave some websites open too long, your session can 'time out'.

User consent

Different mechanisms of first-party data collection have different levels of visibility, and therefore different levels of how informed user consent is. Signing up to an online account for instance is a very obvious and explicit way in which users provide their information, knowing that it is being collected by the service provider. It is generally assumed that users are aware of the service provider monitoring their activity – like a physical salesperson watching potential customers browse the shelves in a shop.

Additionally, users are often made explicitly aware of first-party tracking practices, for instance through pop-up notices or privacy policies - although the level of visibility for these can differ significantly. During mobile app use for instance, data can also be collected through first-party tracking mechanisms, including cookies. Unlike web tracking, rather than asking for consent each time a user opens the app, users are commonly asked to consent to a privacy policy during an app's initial installation or its first launch (Lamba, 2024). Ongoing consent is then assumed, and subsequent data collection becomes less obvious. However, users should be able to manage and customise cookies and other practices of data collection (from first parties as well as third parties) in an app's settings, and through their browser settings (Lamba, 2024; Syneris, 2022).

Who benefits?

Service providers (first parties) gain commercial benefits from tracking user activity on their site or app, through insights into who these users are, and how they might adapt their service or product to make it more appealing to their target user (Yeaney, 2023). However, these practices can also improve the user experience. 'Cookies' for instance can 'remember' and store your login details, or items left in a shopping cart (Lafleur, 2025). These functionalities can improve convenience and relevance for users, while offering the service provider information about customer preferences and return-visits through activity tracking. First-party data can also provide analytic data to prompt updates to the functioning of an app or website that improves the smooth functioning on different devices or operating systems.

1.2.3 Second-party data

Second-party data is obtained via an indirect customer relationship. It is in effect another entity's first-party data that is relevant or related to a provider's product or service (Andersen, 2025). Second-party data is commonly made available through a collaborative data-sharing agreement.

One example of such a data-sharing relationship is when a distributor of a first party's product, may also collect customer feedback and other information about the customers who purchase the product. Say the fictional white goods store *Tech4U* sells different brands of fridges, including the fictional brand *Coolfresh*. The customer feedback and other details about *Tech4U*'s customers who buy *Coolfresh*'s fridges is data that is of interest to *Coolfresh*. The data that is being collected by *Tech4U* about customers who have bought *Coolfresh* products, is directly relevant to *Coolfresh*, but it is collected indirectly (by *Tech4U* rather than by *Coolfresh*). To better understand their customer base, *Coolfresh* and *Tech4U* enter into a collaborative agreement, to share the data both companies are collecting about customers purchasing fridges. To each entity in this relationship,

the data shared by the other is second-party data. It is directly relevant to each party, but not collected directly by them (Yeaney, 2023).

User consent

Data-sharing with other parties outside of the direct relationship between the user and the first party/tech provider needs to be communicated to users, to enable them to consent or object to the collection and/or sharing of their data. Details regarding data collection and sharing mechanisms are commonly outlined in apps' or websites' privacy policies, or in notifications for 'cookies' and similar tracking technologies, that users are asked to consent to when first loading a webpage. The data collected and shared through collaborative agreements is likely to be used in contexts related to those in which it was originally shared by the user. In this sense, second-party data-sharing may be less transparent to users than first-party data collection and tracking, but it is more transparent than third-party data collection and tracking, in which users' data may be shared and used in contexts completely unrelated to those in which the data was collected.

Who benefits?

The main beneficiaries of second-party data-sharing and tracking are providers/companies who gain access to user data for the purposes of product improvement and (targeted) marketing. Targeted marketing is often presented as being of benefit to users by avoiding 'irrelevant' advertising (Kant, 2021). Yet, this argument glosses over the fact that user targeting commonly relies on user profiling, which is often achieved through data collection and aggregation from a range of online sources. These processes can be difficult for users to follow, which complicates the notion of meaningful 'informed consent'.

1.2.4 Third-party tracking and data

Compared to first-party tracking and second-party data sharing, **third-party tracking** entails the collection and sharing of user data that goes beyond the context of direct engagement between a user and an online service, or specific data-sharing partnerships. It describes processes where an entity other than a first-party service provider collects user data through the first-party service, for aggregation into user/consumer profiles, or to share with other third-parties such as data brokers. The user data collected by third parties - or **third party data** - is often used for the purposes of creating profiles about users that are as detailed as possible, by bringing together data traces from a range of online sources. Through the use of cookies and other tracking mechanisms that allow services to tag and recognise users across online spaces, data can be collected from different sources, but associated with the same user, to be aggregated into increasingly detailed profiles over time.

User consent

Third party data from a range of online sources is frequently aggregated by large technology providers and platforms, or sold on to data brokers who combine and pass data on to other parties. These practices allow the (re)combination of user data into detailed and dynamic profiles, which can in turn lead to the (re)identification of users, even from deidentified data. These profiles created about users are not visible to users themselves, but are commonly used as the basis for automated decision-making about them - for targeted advertising for instance. However, they can

also be used for a range of unanticipated and more problematic purposes. For instance, they can facilitate automated decision-making on what information to show a user, in an attempt to influence behaviour other than purchasing decisions - such as voting behaviour (Burkell & Regan, 2019). These data profiles may also form the basis for decision-making about a user - such as their eligibility for access to financial or health services (Sargeant, 2023; Grote & Berens, 2020), or to employment or education opportunities (Hao, 2020).

Since these kinds of data processing can be varied and unexpected, users are generally unable to provide explicit informed consent for how their data can be used by third-parties. Users - and often first party tech providers - may have limited insight into which third parties will ultimately have access to user data, when data is collected, what kinds of data are collected, what purposes it is used for and in which contexts and combinations. Hence, third party tracking is considered the biggest threat to users' data privacy.

Third parties also use 'trackers' such as cookies, tracking signatures built into an app's code base (Chao et al., 2024), or fingerprinting through browser and device information (Nikiforakis et al., 2013), to collect information and track user activity and behaviour. This often occurs across multiple apps and websites, and frequently with limited awareness from users (Hu & Sastry, 2020). Limited transparency means that both users and developers of online services can be equally unaware that the third-party code-base they use to facilitate app development, or the third-party support they use for their website, results in unanticipated user-tracking (Pybus & Cote, 2024).

Who benefits?

Third-party tracking is often a two-way process that is of benefit to first- and third-parties. For instance, a first-party website or app may require support from a third-party provider to improve the functionality of their website, and in turn, the third-party gains access to user data (Hu & Sastry, 2020). The 'support' from others can for example include the use of a particular font, image or video content provided by a third party (*see Screenshot 3* on *'Google web fonts'* in Appendix 1 as a real-life example). It can also include the ability to offer in-app identity verification and authentication through integration with a social media platform (Weltevrede & Jansen, 2019). A first-party provider can also earn money through the placement of advertising provided by a third party, and then displayed in a first-party app or on a first-party website (Lerner et al., 2016). Third-party support also comes in the form of functional analytics, that can improve the smooth running of a website or app, by reporting crashes or bugs (Dambra et al., 2022), or in the form of marketing analytics to help the first-party understand its user/customer base. Yet, all these integrations necessitate the sharing of user data with third parties.

Table 1: Summary overview of data-sharing relationships

Type of data	**Zero-party data**	**First-party data**	**Second-party data**	**Third-party data**
Obtained how	**Voluntarily** and proactively provided by users	Collected by an online service provider through a **direct customer relationship**	**Exchanged between providers of related online services**, and therefore obtained via an **indirect customer relationship**	**Collected through an online service, by entities such as data brokers and aggregators that are external to the service** and who may compile and on-sell this data to others, for purposes completely unrelated to the service the data was collected from.
Benefits whom; (main benefits)	**First-party technology providers**; (product improvement, marketing, consumer trust/loyalty); **users** of the service (convenience, data privacy)	Primarily **first-party technology providers** (product improvement, marketing); **users** of the service (convenience)	Primarily the **service providers exchanging the data** (product improvement, market analytics, marketing)	**External entities** (e.g. data brokers) who sell aggregated user data and profiles (selling of user data for profit); benefits **first-parties** who receive payment or services for allowing data-sharing (technical support, payment for advertising); benefits **external entities who purchase user data or profiles** to gain broad insights into consumer preferences (market analytics, advertising)
Transparency and consent	**Very high level of transparency to users**, who decide the "what, when, where and with whom" of data sharing	**Significant level of transparency to users**, who can reasonably be expected to be aware of the fact that data about their interaction with an online service will be collected directly by the service provider	**Moderate level of transparency to users**, who may not be aware of the data-sharing arrangements between two related service providers, but who may be assumed not to object if this practise is outlined in a privacy policy or notice when a user first visits or downloads a service	**Low level of transparency to users**, as data-sharing with third-parties may be outlined in privacy policies or notices, but these documents cannot detail all the potential ways in which data brokers may process the data collected e.g. how user data may be combined with data traces from other sources, and who may purchase the data from the third-party entities, and for what purposes - making it impossible for users to provide consent that is truly informed

1.3 How am I being tracked?

This section provides a high-level overview of the different tracking mechanisms employed in web- and mobile environments. Understanding the similarities and differences in how tracking is operationalised provides contextual information that helps to follow how tracking can be detected and investigated.

1.3.1. Common methods for web-tracking (e.g. when using a browser)

For web-tracking, the providers of online websites and services use a range of 'trackers' that are part of their sites' digital infrastructures.

1. **HTTP referrers :** *A* HTTP referrer tells the webpage you are visiting from which webpage you accessed it, or which webpage 'referred you' to them. This is useful information to - for instance - ascertain successful promotional pathways (Parry, 2025).
2. **Tracking Cookies:** Cookies are small text files that a web service can 'set' on your browser when you visit a website, to allow it to 'tag' and specifically identify you. *Tracking* cookies specifically aim to track users' online activity across different sites, and to collate and share this data with third-parties for purposes such as user profiling and targeted advertising[5].
3. **Tracking pixels:** Tracking pixels are pieces of code offered by third-party providers, and commonly used for conversion tracking (Meta, 2025) for instance to ascertain the effectiveness of advertising. They can also have similar functionalities to cookies, and be used for 'retargeting' (Higgins, 2023). They are usually very small 1x1 squares or '1x1 tracking pixels' (Higgins, 2023) that can be integrated into websites or emails in unobtrusive ways, sending basic information back to whoever installed the pixel, when a particular action takes place (e.g. an email is opened, an item is viewed or added to the shopping cart, a user lands on the 'thank you' page after a completed purchase) (Higgins, 2023; Meta, 2025).
4. **Browser fingerprinting:** Browser fingerprinting is a method of collecting information on the details of a browser (e.g. plug-ins installed, system information, user language, timezone, screen resolution, IP address) to enable device identification and detection of online activity from that specific device or user. This data can be used to track users across websites for targeted advertising, as well as the collation of detailed online profiles, even when users have previously selected 'do-not-track' requests (Hauk, 2024; Nikiforakis et al., 2013).

[5] Example: You visit the website academia.com, which displays any text on the site in a special font style, available from the third-party site trackers.com. When you visit academia.com, your browser sends a request for the font style to trackers.com, to be able to render the site in your browser, using the special font. As part of this process, trackers.com sends both the font, along with a 'HTTP Set-Cookie' header, which sets a cookie on your browser. Additionally, trackers.com is able to read any pre-set cookies that are already set on your browser, to determine if you visited any other websites associated with trackers.com. If you subsequently visit any partner-websites of trackers.com, such as buybooks.com, trackers.com recognises the cookie set on your browser during your visit to academia.com, and starts to collate information on your browsing habits into a profile.

1.3.2 Common methods for app-based tracking

Compared to websites, mobile applications are more closed, self-contained infrastructures, and the sharing and tracking of user data that is enabled through mobile applications uses different tracking mechanisms to websites. Even for the web- and app-based versions of the same online service, different data-sharing networks and connections are established. Apps also interact with the mobile operating system (e.g. Android) based on permissions given by users, often at the time of installation.

1. **Mobile cookies:** Similar to cookies on the web, mobile applications also use cookies to store small amounts of user data locally on the device, to 'remember' user preferences (e.g. language settings, login information) and track users' interaction with the app (Syrenis, 2022). A key difference between app- and web-cookies, is that app cookies operate only within the 'sandboxed' environment of the app itself, rather than being able to establish networks for data-sharing with other apps or second- and third-parties. Hence, mobile cookies' primary role is in improving the user experience through information retention and customisation, and in first-party tracking of user activity (iab.Australia, 2013).
2. **Tracking Signatures in Software Development Kits (SDKs):** Tracking of user data across mobile environments, particularly for the purposes of monetisation through advertising, is enabled through the use of third-party tracking signatures, or 'trackers' that are integrated into the code of mobile applications. These trackers commonly get integrated into mobile applications through the use of Software Development Kits or SDKs, which are essentially pre-prepared buildings blocks containing app code, software libraries, access to platform APIs and features, and other services important to building apps and integrating certain functionalities, that are supplied by third parties (Pangrazio, 2021). The support that third parties offer through the provision of SDKs have been termed 'service-for-data', where the use of the services provided through the SDKs' infrastructure, establishes networks for data-sharing with the third parties who have supplied the SDK (Pybus & Coté, 2024). A key concern in this context is that SDKs are 'blackboxed', meaning developers cannot know exactly what user data is being shared through the networks that the third-party trackers built into SDKs can establish. Because SDKs provide parts of app infrastructure that become embedded into mobile apps, when a developer integrates app permissions into an app's code to enable access to user data, the trackers that are part of the SDKs, can also gain access to this data (Pybus & Coté, 2024).

1.4 The challenges in studying mobile apps' data flows

Websites and mobile applications differ in their infrastructures (i.e. the kind of code used to build them) and in the accessibility and transparency of this infrastructure. For instance, while the server-side or *backend* code of a website that is responsible for storing and managing data (Ferguson, 2024) cannot be accessed from the user-side, the client-side or *frontend* 'source code' of a website can usually be accessed and viewed in a browser window using relatively straightforward methods (often via simple commands such as CTRL+U on a windows computer). A scan of the frontend code may already allow the detection of the code signatures of trackers, or fingerprinting scripts built into a website. Depending on the complexity of an online service, accessing frontend code may therefore provide a first indication of its data collection and sharing practices - as long as you know what you are looking for (Hector, 2018).

An app's code however, is enclosed in a discrete file 'container' – two of the most common ones being APK files (Android Package Kits - for Android apps) or iPA files (iOS App Store Package - for iOS apps). Obtaining these containers for subsequent access to and inspection of app code, requires the use of specific tools. For instance, APKs can be downloaded directly to the desktop from online app collections such as APKMirror (APKMirror, n.d.), which aggregate and allow the download of the APKs of many popular, free android apps that are available on major android app stores such as Google Play. Alternatively, APKs can be extracted from an app already installed on an android device, through the use of an extractor tool, for example the app *APK Extractor & Analyzer* (Nayanajith, n.d.).

Inside these containers are all the components and code an app needs to run on a device. Scanning an app's code for the app permissions and tracking signatures it includes, is a useful way to determine the potential data privacy implications of using an app. App permissions are requests that an app can make to gain access to different kinds of data or resources on a mobile device, and tracking signatures are lines of code that can be executed during app use to establish a network connection to a third-party for data-sharing. Hence, analysis of app code (also called 'static analysis') to determine these components can provide insights into an app's ability to access and share user data.

However, depending on which operating system the app is designed to run on (most commonly Android or iOS), the components of their file containers differ in the kinds of code used. The code contained in app files has been 'compiled' to turn it from human-readable, high-level programming language, into a lower-level, machine-readable code (Lenovo, 2026). This means that for a researcher to be able to inspect an app's code, the files in the container first need to be 'decompiled' (Alzaidi et al., 2020), to make them human-readable. While this de-compilation is relatively straightforward for APK files (i.e. for Android apps), the code used in iPA files (iOS apps) cannot be accessed via the same mechanisms (see section 2.2.1), which is why inspection of app code to identify app permissions and the presence of trackers is more commonly performed for APK files[6]

[6] For examples of recent and emerging work exploring the data-sharing practices and privacy implications of iOS applications, see approaches by Pybus and Mir (2025), or examples of tools currently under development, such as APPMONITOR by scholars from the University of Kopenhagen, including Kristian Sick, Sofie Flensburg and Signe Lai, outlined here: https://cordis.europa.eu/project/id/101189401.

Section 2: Materialising mobile app data flows: an introduction

2.1 Available methods literature, and its usefulness for non-technical researchers

Many early articles on the analysis of mobile apps' data flows stem from the mid-2010s, and are authored by computer scientists, who detail the development of tools and methods to conduct app analysis and to 'track the trackers' of the web- and mobile ecosystems (Nikiforakis et al., 2013; Razaghnapah et al., 2016; Binns et al., 2018). These studies commonly focus on the technical aspects of developing and implementing investigative methods and tools (see for instance Razaghnapah et al., 2016; Bui et al., 2021) and on quantitative approaches that can be used at scale (Reyes et al., 2018; Razagnapah et al., 2018; Binns et al., 2018). These works provide insights into the challenges and considerations behind the proposed computational methods, as well as providing the tools and forming the methodological base that subsequent studies from other disciplines – such as health sciences and communication studies – build on (Zhao et al., 2020; Ioannidu & Sklavos, 2021).

Significant interest in the study of static and/or dynamic data flows in mobile apps has emerged since then, especially in the area of mobile health (mHealth). Subsequent investigations have included explorations of Femtech (Erickson et al., 2022), as well as wearable devices and fitness tracking apps (Ioannidu & Sklavos, 2021), and comparisons of apps' data sharing potentials and practices with app privacy policies and applicable data privacy regulations (Kuntsman et al., 2019; 2022). Privacy policy analysis, data justice and data rights perspectives feature frequently in these studies, with the focus being not only on any problematics uncovered in the process of analysis, but also on frameworks (Kuntsman et al., 2022), recommendations (Erickson et al., 2022; Ioannidu & Sklavos, 2021) or even direct engagement with developers, to hold technology providers to account (Papageorgiou et al., 2018)[7].

Finally, media and communications with technical skills have harnessed computational methods to explore the data infrastructures and relations of mobile applications, in the context of their everyday use with considerations of their social implications (Weltevrede & Jansen, 2019). These studies frequently include mixed-methods approaches that combine computational methods with qualitative methods such as app walkthroughs (Light et al., 2018), and the use of research personas - the creation of fictitious users to observe personalised interactions with data (Bounegrou et al., 2022), to gain insights into data flows and the surveillance practices facilitated through everyday app ecologies (Flensburg & Lai, 2022; Dieter et al., 2019). Special interest groups, such as the *App Studies Initiative* (App Studies Initiative, 2026) have been

[7] Many of the papers mentioned above provide important examples of investigations into data-sharing infrastructures and data flows. A number of them suggest frameworks and methods for others to replicate, describing them as tools to effect meaningful change. One example of tangible impact from this work for advocacy purposes, and to effect change beyond the level of the individual user, is outlined in the paper by European researchers Papageorgiou and colleagues (2018). The authors conducted a privacy analysis of 20 mhealth applications, including assessing the apps' privacy policies and practices against the GDPR guidelines. The results of their analyses were shared with the developers of these apps, and outlined key concerns regarding data privacy and areas of non-compliance with regulation. A subsequent round of app analysis revealed that significant improvements had been made in later versions of the apps under investigation.

leading many of these efforts, as well as seeking to make methods for app analysis accessible to less technically-skilled researchers (Chao et al., 2024)[8].

The study of sensor data that is increasingly becoming a part of the everyday use of mobile apps and devices is still a less common area of study. In their respective articles, Ioannidu and Sklavos (2021) as well as Chao and colleagues (2024), outline how methods of dynamic app analysis may be applied to the study of the data privacy risks and data sharing practices of wearable devices and fitness tracking applications. Both papers introduce tools specifically developed to be used by researchers with a limited amount of technical knowledge and skill: specifically the *Lumen Privacy Monitor* (Ioannidu & Sklavos, 2021) and the *AppInspect* and *AppTraffic* tools (Chao et al., 2024). Chao and colleagues (2024) in particular point to the difficulties in conducting dynamic cross-device network analysis, and the challenges in making sense of the intercepted and decrypted sensor data (for more information on dynamic analysis, see section 2.2). For non-technical researchers, the availability and use of tools designed to make the analysis of data-sharing infrastructures and data flows more accessible, is therefore a particularly necessary endeavour to enable more of these investigations.

The subsequent sections will therefore mention both technical and less technical methods for the analysis of mobile app data flows, but focus more on those methods and tools that are meant to be easier to replicate by non-technical researchers.

2.2 Detecting app-based tracking and data flows

To analyse the data flows through mobile applications, we can distinguish between two methods of analysis: *Static* and *Dynamic* analysis, which app studies researcher Anne Helmond (2018) divides into a three-level pyramid, requiring progressively more technically demanding and intrusive methods at higher levels (Figure 1 – modified in Chao et al., 2024).

Level 1 or 'static analysis' encompasses analysis of the technological infrastructure – the code of an app – with the inspection of app permissions and identification of 'trackers' present in the app's code. Static analysis forms the basis for further 'dynamic analysis' of the active data flows during app use, at levels 2 and 3. At level 2, the network connections an app makes for data or 'packet' transfer during use are being traced or 'intercepted' (also referred to as 'packet capture'). Level 3 describes the final step of 'packet inspection', which includes decryption and inspection of the data packets sent via the network connections mapped at level 2. Each step is described in detail below.

[8] Aside from app studies, recent tools developed to make web-tracking visible through visualisations that present data flows in more accessible form, have been developed through the University of Surrey, providing tools such as Thunderbeam/Lightbeam; see https://netsys.surrey.ac.uk/datasets/tracking-the-trackers-papers/ and https://ofaolain.com/blog/2018/10/05/shine-a-light-on-the-web-trackers/

Figure 1. App pyramid model

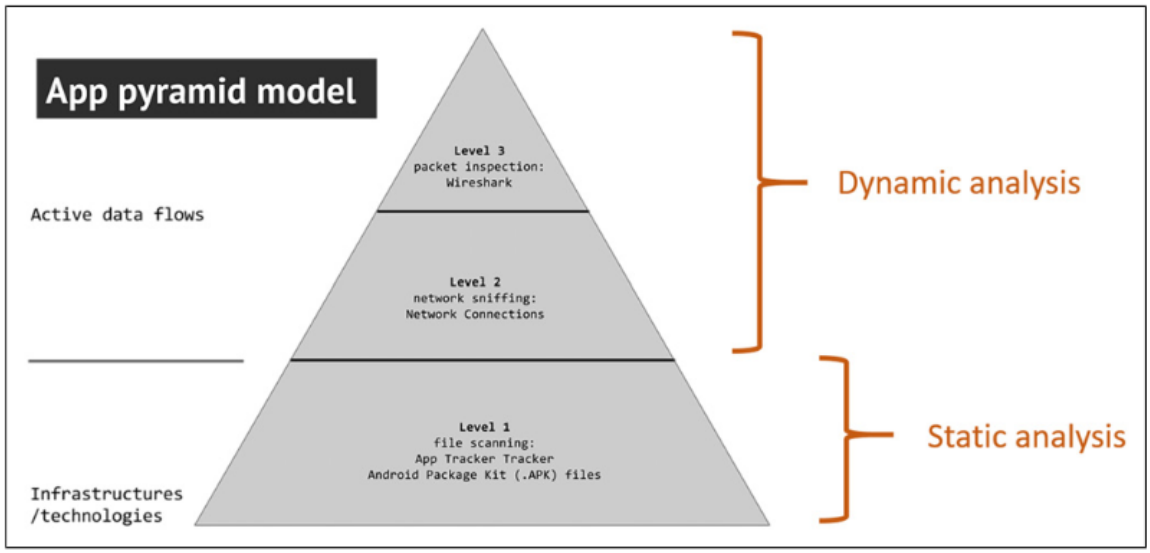


Figure 1. Adapted version of the 'app pyramid model' (Helmond et al., 2018) with 'static' and 'dynamic' modes of analysis.

2.2.1 Static detection methods (Level 1)

This first level of analysis can provide insights into the infrastructural affordances or 'in-built' ways in which apps can access, manage and share user data, through their encoded affordances and operational pathways.

A key difference between investigating data flows through web-based online services compared to mobile applications, is that website code can often be readily inspected in a browser's window, and through developer tools, while apps are more closed off, self-contained software packages, whose code is more difficult to access and analyse. To enable the analysis of an app's code, its software package must first be obtained – i.e. found and downloaded – and then scanned or unpacked, to enable the systematic analysis of its constituent elements (i.e. static analysis).

The most commonly used app software packages can be broadly divided into two types, namely Android Application Packages or Android Package Kits (APKs), and iOS App Store Packages (iPAs). APKs support the distribution of Android apps via Android app stores such as Google Play, and iOS App Store Packages (IPAs) support the distribution of iOS apps, specifically via the Apple app store. Both APKs and iPAs are data containers for app code, but they differ in ways that are significant to researchers' ability to query and analyse this code.

These differences are due to the distinct origin stories of each operating system (OS). Android has always been an open-source OS, allowing a wide range of device manufacturers and app developers to customise and modify the OS for their purposes – focussing on inter-operability, user control, and supporting a diverse range of apps and developers. This approach necessitates a more transparent and accessible development process for mobile applications. iOS on the other hand operates in a closed, proprietary ecosystem, focussed on smooth integration between Apple devices only. Within this closed ecosystem, Apple retains control over mobile apps' appearance, quality, security measures, and their distribution,

hindering access to app code for reasons of privacy and the protection of intellectual property (Doctorow, 2017).

The open-source vs proprietary nature of each operating system's architecture has also resulted in differences in app architecture used for each OS. APKs can be decompiled for analysis in a more straightforward manner, which provides access to components such as the app's code and manifest files in human-readable programming language. The contents' of iPA files can also be decompiled, but this step requires different, and more aggressive methods, and still would not provide access to human-readable source code. Hence, most tools for static app analysis focus on the analysis of Android applications' APK files in particular.

APK files contain the app manifest file, an app's code and additional "software 'libraries', 'classes', 'resources', and 'assets'" (Chao et al., 2024, p. 243). These components, especially the manifest file, can be scanned for data privacy concerns, including:

1) in-built **app permission requests** (e.g. access to contact list, photos, location),

2) the identification of **trackers** (through comparison of app code with 'tracker libraries', containing strings of code denoting known trackers), and

3) the identification of **API calls**[9] enabling developers (first parties) to access third-party support (e.g. app analytics to improve an app's performance, authentication through social media accounts). While API calls are not necessarily intentional data sharing pathways, the resulting data sharing can be disproportionate, and unrelated to an app's main functions – oversharing user information and resulting in unnecessary risks to data privacy.

The findings from static app analysis, and the methods available for this purpose, are commonly presented as awareness-raising tools of data-sharing *potentials* and the supporting infrastructures. They reveal how apps set the conditions for data access and sharing on a user's phone (e.g. through app permissions) and materialise the possible data flows to third-party trackers (through identification of tracking signatures).

While a thorough review of all elements of an APK's decompiled code may provide the most comprehensive insights into an app's privacy implications for users, most of the terms of use for popular Android apps detail that decompiling and reverse-engineering are prohibited. However, researchers can often make strong arguments for the necessity of this work, including for breaching the terms of use by decompiling app code for the purposes of data privacy research. Nevertheless, this situation can make it difficult to gain institutional approval, as universities may be concerned about problematic legal or reputational outcomes. It is therefore important to carefully review the terms of use/services and privacy

[9] Note that this third area of investigation is not commonly supported in easily accessible (less technical) static analysis tools. One freely available and accessible tool that does support this step in a user-friendly way is *AppInspect*, developed by Chao and colleagues (2024), and hosted by the University of Siegen; see https://appinspect.phil.uni-siegen.de/

policies of any apps that researchers are planning to decompile, to be able to weigh up whether the benefits outweigh the possible negative implications[10], and make a strong case for their work.

If institutional approval cannot be obtained, there are alternative - even if less comprehensive and transparent - methods for static analysis, that do not require researchers to decompile app code or engage in reverse-engineering. One such tool is the *Exodus Privacy* browser tool, specifically for static analysis of Android apps.

2.2.1.1 'Manual'/non-technical exploration: mimicking the user experience, and desktop research collating readily available data

Following the example of Weltevrede and Jansen (2019), even without detailed technical investigation of an app's code, the kind of information that is requested by an app can be explored by strategically 'walking through' (Light et al., 2018) and documenting stages of app use that are likely to be particularly data intensive, such as registration and account creation (Weltevrede & Jansen, 2019).

These are common steps required to enable first use after an app's download, often including the app asking for device 'permissions', such as access to photos or location data (which constitute personal information) as well as explicit prompting to enter personal information into the app's user interface (e.g. name, email, date of birth, various personal details related to particular app features and affordances). This approach closely mimics the user experience, and slows down everyday interactions with apps (Light et al., 2018), to highlight the moments when users are prompted to make choices about data sharing, including the app's ongoing or temporary access to device affordances, and the sharing of personal data through using the app (e.g. uploading a profile picture). Notably, these steps can be completed without the need for researchers to share their personal data. To this end, researchers should utilise dedicated research devices and set up new app 'research accounts' for app use, including fabricated personal data from a research persona (Bounegru et al., 2022). If an app has social elements that connect users to others, steps should be taken to ensure other users are not disturbed, and their details are not collected (see Gold-Apel & Duguay, 2023). Research ethics are an important consideration in this context, as secondary data (e.g. online forum participants and posts) may be unintentionally collected. The appropriate steps to take are likely to differ significantly between apps, and should be determined on a case-by-case basis.

Many mobile applications – particularly those requiring the setup of detailed personal information as a key part of app functioning as is the case for dating apps (Weltevrede & Jansen, 2019) – offer integration with social media platforms, especially for the purpose of logging in and authenticating a user's identity. However, there are different data types that can be accessed and exchanged between apps and social media platforms, depending on the app permissions requested (Weltevrede & Jansen, 2019). Social media permissions and the data types shared can again be ascertained through following specific steps of the app walkthrough method (Light et al., 2018), in combination with other methods such as the use of a

[10] These implications can be quite benign from a research perspective. As part of the static analysis work conducted for a recent article by Langton and Ng (2025), we reviewed our case study apps' terms of use, and found that the most detrimental outcome described in their privacy policies was that breaching the terms of use may result in being banned from app use.

research persona (Weltevrede & Jansen, 2019; Dieter et al., 2019), to mimic everyday use and record the interactions with the app.

As part of the exploration of the 'environment of expected use', Light and colleagues (2018) also suggest reviewing an app's privacy policy for clues on an app's data management, and how transparently this is communicated to users – providing another point of comparison between how data management is presented, what data-sharing the app is theoretically capable of (judging by the outcome of static analysis), and explorations of actual data flows during dynamic app analysis (if possible).

Additionally, 'permissions' can be viewed via the Google Play Store, and either collected manually, or through the use of purpose-built tools to collect app data, such as the *Google Play Similar Apps* tool (Van Der Vlist, 2016), or the more recently designed *AppInspect* (Chao et al., 2024)[11]. Permission requests can then be collated and visualised for comparison between different apps (see Weltevrede & Jansen, 2019 as an example), and even for different app versions over time (Exodus Privacy, 2025; Chao et al., 2024). Alternative tools providing similar insights include Exodus Privacy (see 2.2.1.3). These less technical methods lend themselves to providing insights into common permissions for apps of the same category, to provide contextual information that complements more fine-grained investigations of specific apps. For a data justice–informed, and privacy-focussed approach to applying the app walkthrough method (Light et al. 2018) in combination with static analysis (and - where possible – dynamic analysis), Kuntsman and colleagues (2022) provide a useful example.

Another promising addition to current approaches for static app analysis is the SDK audit (Pybus et al., 2026; Pybus & Mir, 2025), which has been developed and applied more recently by researchers at the University of York, and in collaboration with international colleagues[12]. Similar to the strategies outlined above, the method combines the app walkthrough method (Light et al., 2018) with a novel approach to the static analysis of app code. The method pays particular attention to aspects of the app manifest file that other approaches commonly overlook, specifically how apps are set to share data about 'app events' – user data and interactions that developers or marketers are especially interested in, including behavioural data (Pybus et al., 2026). This information is revealed by using the APK inspection tool *ClassyShark*, which allows users to decompile android apps, providing access to components such as the manifest file for further analysis, and allowing the identification of SDKs that are part of many apps' code (Farber, 2020; Pybus, 2024) and are commonly responsible for data-sharing with third-parties. Pybus and colleagues assert that an app's manifest file contains all the information necessary to understand how an apps' data sharing practices are governed and operationalised – but not in a way that makes sense to a human reader. Their latest work therefore outlines how researchers can utilise Large Language Models to develop prompts suitable to audit app manifest files, to provide qualitative insights into the implications of apps' infrastructural components for the sharing of user data (Pybus & Mir, 2025; Pybus et al., 2026).

---

[11] Note that the way permissions are outlined on Google Play, differ from how they are presented through tools such as *AppInspect* or *Exodus*, which provide a more detailed overview, but also make the utility of permissions more complex to understand, and can add further labour for the researcher(s). These distinctions should be reviewed prior to deciding on which tool to use. For comparison, see for instance Weltevrede and Jansen (2019) who sourced permissions from Google Play, and Langton and Ng (2025) who collated permissions from *Exodus Privacy*.
[12] *Late addition*: While we have made an effort to be as inclusive as possible in outlining up-to-date methods of app analysis suitable for less technically-skilled researchers, the fast-moving nature of this research space means it is very difficult to stay abreast of the latest developments.

Table 2: Examples of non-technical approaches to static app analysis

Method	Application	Possible insights
App walkthroughs - with a privacy specific focus (Light et al., 2018; Kuntsman et al., 2019; 2022)	**'Walking through' and documenting data-intensive stages of app use** (i.e. installation and registration; prompts and opportunities to input personal/ health information); **Review of privacy policy,** terms and conditions; **Review of social media integration** Notes: Integrates with methods such as the use of **'research personas'** (Bounegru et al., 2022; Dieter et al., 2019), as part of dynamic analysis	**Provides general evidence of the user-facing information** and experience of managing app permissions and data privacy; point of comparison **Complements and contextualises static and dynamic analysis** by taking into account the potential user experience
Desk research of readily-available data on app affordances: App store descriptions of permissions (i.e. Google Play); Exodus Privacy reports on permissions and trackers	View **Google Play** store information to access a list of app permissions – (manually or automated), for user-friendly overview of permissions and their utility Use additional method to access app permissions for comparison, e.g. visit **Exodus Privacy** to access app permissions and a list of trackers for the same and similar apps, including over time (historical component)	**Enables high-level comparison of an app's permission requests** provided to users (Google Play Store), with permission requests identified by privacy-focussed tools (Exodus Privacy) and tools for fine-grained static analysis; Depending on the size of the sample, researchers may choose one source of information on app permissions, depending on requirements. Enables collation and comparison of app permissions requests for apps from the same category; **Enables historical comparison** of app permission requests and trackers in different app versions over time
App Manifest/SDK audit[13] (Pybus & Mir, 2025; Pybus et al., 2026)	**Source APKs** of apps for analysis; **Extract app manifest files** using ClassyShark (Farber, 2017) **Develop suitable LLM prompts** to audit the manifest (Pybus & Mir, 2025; Pybus et al., 2026)	**Allows detailed investigation of the presence of third-party SDKs and their roles** in user data-sharing **Allows insights into the types of user data shared**, even outside of dynamic app analysis

[13]Applying this method may involve less 'technical' skill in terms of the ability to code or understand software development, but it is likely to require a much higher investment in time and learning than the other methods listed.

2.2.1.2 Choosing the right tool for the job

To provide analytic results that are as accurate as possible, it is important for researchers to consider whether it would be useful to attempt static app analysis beyond the most accessible methods. Depending on a research teams' level of technical skill and understanding, as well as the scope and time constraints of the project, researchers should reflect on whether performing static analysis with tools requiring a higher level of technical skill (if possible) may result in valuable additional insights.

Static analysis of apps is well-supported, with a wide range of tools available and suited to researchers of varying levels of technical skill and knowledge. Examples of tools requiring different levels of technical skill are listed below, noting that each may reveal slightly different results based on how they unpack the app's source code. Applying a combination of different analytic methods provides an opportunity to compare results and gauge accuracy (e.g. using different static analysis tools such as *Exodus Privacy* and *AppInspect* to compare results).

If within scope, this step is a useful approach to gauge the accuracy and usefulness of different methods of app analysis. Kiltz and colleagues (2020) have found that different tools for static app analysis may identify the same app permissions, but can differ significantly in the trackers identified in an app's code, depending on the tracking libraries that these tools are querying during analysis[14].

2.2.1.3 Examples of app code/infrastructure analysis tools and methods (least technical to technical)[15]

Below is a list of tools that support static analysis. Please note that as Android codes and mobile technologies change regularly, these tools may no longer be available or work for these purposes in the future. However, they provide a basis for comparison and an understanding of how static analysis can be performed, using different approaches - including examples of their respective benefits and limitations.

- **Exodus Privacy & Exodus Standalone**

  Exodus Privacy (minimal level of technical skill required, 'off premises') is a webservice where the analysis of app code is conducted 'off premises' and not on the user's own computer. It is easily accessible through the browser window and focussed on making visible the app permissions and number of trackers present in an app's code. This online tool automatically analyses APKs for trackers, by scanning their contents and comparing the strings representing permission requests and app code to data bases of well-known strings associated with trackers, such as particular

[14] To enable the identification of tracking signatures in an app's code, static analysis tools compare an app's code to a database of known trackers, such as the Exodus Tracker Investigation Platform, or ETIP (ETIP, n.d.; Steiner, 2020). This process is explained further in 2.2.1.3. However, different tools compare app code with records of trackers collated on different databases, which often differ in their comprehensiveness, and therefore in the trackers they list. Hence, when different tools for static analysis are used and the findings compared, app permissions are commonly the same, while the trackers identified can differ significantly.

[15] Tools can be found via the following links:
https://exodus-privacy.eu.org/en/
https://github.com/Exodus-Privacy/exodus-standalone?tab=readme-ov-file#installation
https://appinspect.phil.uni-siegen.de/
https://apktool.org/

domain names (e.g. 'google-analytics.com') (Exodus Privacy, 2025). These lists of well-known strings are accessible via the Exodus Tracker Investigation Platform (ETIP) – a 'library' of tracking signatures that is regularly updated and reviewed by contributors including Yale Privacy Lab, jawz101, Guardian Project, F-Droid and others, who have centralised their efforts by adding to this 'canonical database' (Steiner, 2020; Steiner, 2022). A key advantage is the tool's currency, due to the collaborative approach to updating the 'libraries' or 'databases' of known trackers it uses, and the historical data available on the site, which allows a comparison of app permissions and embedded trackers in different app versions over time.

Exodus Standalone (programming/coding knowledge required, 'on premises') is the "εxodus CLI client for local APK static analysis". In other words – it is a command line interface tool, specifically for the analysis of APKs, that is downloaded and used on your local device ('on premises'). Its installation and use from its github repository requires a significant amount of technical skill and understanding, but its open source format also means that this method offers increased transparency and insight into the analysis process compared to the web-version of Exodus – for those with the programming and coding knowledge necessary to understand it. It compares app code to the same tracker library as Exodus Privacy.

- **AppInspect** (limited technical knowledge required, 'off-premises') has been developed by Chao and colleagues (2024) for the static analysis of APK files, with particular sensitivity to the sharing of sensory data. The tool is accessible through a web-browser, and offers a range of functionalities displayed in a user-friendly way for non-technical researchers. It is designed to "systematically download, store, query and analyse app packages" (Chao et al., 2024, p. 244), including the facilitation of identifying 'similar apps' of an app category, and historical comparison of the presence of trackers and permission requests in previous versions of the same app. However, the tool is less well-maintained than Exodus Privacy (i.e. at the time of writing, the automatic download of APKs does not work for a number of popular apps). Historical versions of APKs first need to be found, obtained and uploaded for comparison through the tool, and its processes are less transparent (i.e. information on how the tool identifies permissions and trackers is not easy to find)

- **APKTool** (programming/coding knowledge required, 'on-premises') is commonly described as a 'reverse-engineering' tool (apktool, n.d.) meaning it unpacks the APK file into its constituents (described above). Permission requests in the .XML file (app manifest) of the app code can then be scanned and analysed, and data requests (including from third party trackers) can be found in the app code and compared to common databases of trackers to identify which of the hostnames found in the static code represent trackers, or API calls to benign external resources (Binns et al., 2018). Compared to *AppInspect*, the interface and functionalities are targeted more towards users with technical skills, such as software and app developers or computer/data scientists. However, the analysis is also conducted on the user's own computer (i.e. 'on-premises'), which allows increased transparency over the process of analysis – particularly for technical researchers.

- **ClassyShark** (some technical skills required to download and install ClassyShark following github instructions (Farber, 2017), 'on premises'; deeper analysis options for SDK audits using LLMs

requires additional learning and investment in time) is a Google-engineer developed open access tool for the analysis of APK files. It allows users to accessibly view and browse the components of APKs, including manifest files, allowing subsequent identification of SDKs and data-sharing permissions (Pybus et al., 2026). Pybus (2024) provides an accessible tutorial for its use by researchers.

One of the goals of static app analysis is to provide evidence of the mismatch between the vast *potential* that exists for app-facilitated harvesting and sharing of personal data, compared to the difficulties of determining the destination, subsequent processing and end use of this data. However, static app analysis of data sharing is likely to be incomplete and include false positives, since the presence of code that represents *potential* data-sharing with trackers does not mean this code is *actually* executed.

To enable exploration of *actual* data flows and sharing practices, dynamic analysis of network traffic, including packet capture (through proxying) and packet inspection (through decryption) is necessary (Binns et al., 2018, p. 7).

2.2.2 Dynamic detection methods (Level 2 and Level 3)

Dynamic app analysis and tracker detection involves inspecting network traffic from an app while in use, to identify any third-party destinations that may relate to tracking, or that may put users' data privacy at risk.

As shown in the 'app pyramid model' in figure one, dynamic analysis of an app's actual data flows while in use can take place at two levels:

- network analysis (level 2), and
- packet inspection (level 3)

Both require the rerouting of an app's network traffic through a separate proxy server, to be able to intercept and log the network connections to and from a device and make them visible for analysis. By connecting a device to a proxy server, network requests are no longer exchanged directly between a user's device and a tech provider's internet server. Instead, a user's network request is sent to the proxy server, which acts as a gateway or 'middleman' that then passes the network request on to the intended upstream server[16].

2.2.2.1 Network Analysis (Level 2)

For the purposes of network analysis, the two main steps required are to **set up and connect to a proxy server**, and to **intercept the network traffic** for further analysis. A useful way to think about traffic analysis is to imagine network traffic as a flow of data packets, with methods like network sniffing

[16] For example, when a user sends a network request from an app on their phone (i.e. to display a social media app's feed), this network request goes to the proxy server first. The proxy server then makes the network request on your behalf, forwarding it on to the social media platform's server. When the upstream server responds, the proxy server collects the response and forwards the data on to the app on the user's device, which is then able to display the requested information, e.g. the social media feed (Buckbee, 2022).

(detection of connections made) and packet capture (materialising/displaying the data parcels being sent) to make this network traffic visible for further study.

It is then possible to inspect the network connections to ascertain where outgoing data packets are being sent to (and, if of interest, where incoming data packets originate from), enabling researchers to map data flows from a client to a destination server upstream. Weltevrede and Jansen (2019) provide a detailed description of this process, including considerations to isolate particular kinds of data traffic – such as focussing on network traffic from a phone's IP address to determine where data from a user's device is being shared to (p. 16-19). Analysis of network traffic therefore enables users/researchers to materialise data flows to third parties and trace the data relationships that manifest during app use.

As in the examples by Weltevrede and Jansen (2019), Binns and colleagues (2018), and more recently Flensburg and Lai (2022), it is often possible (even if cumbersome) to follow these data flows upstream to try to uncover the higher-level data infrastructures and ownership, to trace who is interested in and stands to benefit from, the sharing and accumulation of user data. This can be achieved through desk-research to determine which company/companies own(s) particular domains associated with trackers, and what the names of associated subsidiaries are.

Network analysis also allows high-level sorting and categorisation of the proportion of network connections made to third-party servers for potentially problematic uses such as tracking, or seemingly innocuous uses, such as data sharing primarily for the purposes of analytics, debugging and service 'improvement' (see this example illustration by Weltevrede and Jansen, 2019).

Having mapped the network traffic, it is also possible to compare this example of *actual* data flows, with the findings from static analysis of *potential* data flows at level 1. This step allows users/researchers to determine whether the information provided by the app developer (e.g. in privacy policies) regarding data sharing with third parties is accurate, or whether unanticipated connections are being made[17].

2.2.2.3 Examples of standalone network analysis tools

Again, a range of tools exist to conduct network analysis. Some are used as stand-alone tools that can be combined or integrated in complementary ways to conduct dynamic network analysis at different levels. Applying stand-alone tools requires a significant level of technical skill. Examples of alternative methods are outlined later in this section.

- ***TCPdump*** is a tool commonly used to intercept network traffic, by conducting packet capture (i.e. PCAP) and displaying packets. Packet capture works by creating copies of data packets passing through a point in the network (Grimmick, 2023). TCPdump is a command-line interface (CLI) tool allowing the efficient capture of network traffic, and the use of simple filters to narrow down on particular traffic of interest (e.g. by narrowing the capture to a particular IP address). Traffic

[17] Notably, the results (and therefore the findings) may differ when an app is being used from a 'clean' research phone with a made-up user account for a non-existent research persona, compared with the results generated from a personal phone, with an account whose details are part of established data profiles and relationships through longer-term third-party tracking.

captures can be saved as .pcap files, which can then be viewed in other traffic analysis tools such as Wireshark, which have a graphic user interface (GUI) that allows for more complex filtering and processing of captured traffic.

- ***Wireshark*** is also a tool for network analysis, allowing both identification of network traffic/connections being made, and packet capture and analysis. While the use of TCPdump has advantages for packet capture due to its customisability around timing, executing and saving packet captures, Wireshark is a useful complement that can open the .pcap files generated through TCPdump for more fine-grained analysis, in a more accessible user interface (See Weltevrede & Jansen, 2019, for an example of combined use of TCPdump and Wireshark).

The advantage of using tools such as *TCPdump* and *Wireshark* in combination, is that the steps of collecting and narrowing down particular types of network traffic are transparent to, and controlled by, the user. However, the effective use of these tools – both separately and together – requires significant technical knowledge and skill, and may therefore not be useful for non-technical users/researchers. Additionally, while in the example of Weltevrede and Jansen (2019), the use of *Wireshark* only was enough for level 3 of dynamic analysis – packet inspection – subsequent changes to operating system infrastructures and in platform practices have resulted in the more consistent use of transport layer security (TLS) encryption, which may require more invasive methods to allow packet inspection.

Similar to the description of static analysis above, there are differences in the technical skill required to effectively use particular tools and interpret the results provided, and in the transparency of the different methods' analytic processes and the specific insights these can provide. Aside from the use of standalone tools such as *TCPdump* and *Wireshark*, a number of proxy-tools are available that integrate functionalities for network analysis and packet capture, as well as offering the ability to inspect data packets. While these tools can also be used for network analysis only, they are described under the heading of Level 3 app analysis, due to their decryption abilities.

2.2.2.4 Packet Inspection (Level 3)

To be able to not only intercept data packets, but analyse their contents or 'payload', it is necessary to employ proxy tools with 'man-in-the-middle' (MITM) capabilities. Conventional proxy servers simply forward data between a client and a server in both directions, without being able to view the contents of a TLS-encrypted data stream. In MITM processes, proxy tools not only act as proxies that forward network requests and facilitate data flows between a client (i.e. a user's smartphone) and a server, but harness the *certificate authority* system to establish themselves as a trusted party in the data-transmission process, enabling decryption and 'inspection' of data packets – i.e. allowing the user to look inside the data packet to see the data being sent (mitmproxydocs, n.d.).

As mentioned above, data traffic between a client and a server is encrypted in transit, through TLS encryption, and decrypted when it reaches its intended destination – the trusted server. Employing a MITM process means to "pretend to be the server to the client, and pretend to be the client to the server, while [the researcher sits] in the middle decoding traffic from both sides" (mitmproxydocs, n.d.). To keep data from being intercepted and read by unauthorised others, the *certificate authority* system is designed to ensure that data traffic can only be decrypted by the trusted server, who "cryptographically signs a [client] server's certificate to verify that they are legit" (mitmproxydocs, n.d.), allowing the parcel to be 'delivered'

and access to the data payload. To bypass this process and allow decryption to inspect the payload in the data packet, the MITM process includes certificate authority implementation, that registers the proxy server as a trusted certificate authority, and gets the client server to trust the proxy, allowing access to the data in the packet.

Possible problem: requiring root access

However, some apps now use a more secure system of 'certificate pinning' that makes this MITM workaround more difficult, and might require even more invasive methods, including having to 'root' or 'jailbreak' the phone. Gaining 'root access' allows users to modify apps and manage the phone's systems and security at the level of the operating system itself. Taking this step voids the warranty of a phone, and is likely to result in increased vulnerability regarding data privacy. It is therefore advisable to use a research device if this step is necessary, rather than conducting any research on a personal device. Non-technical researchers are likely to require additional support.

2.2.2.5 Examples of integrated dynamic analysis tools

Rather than using standalone tools, a range of tools are available that integrate the ability to analyse network traffic, capture packets and/or inspect data packets. However, they vary in the level of technical skill necessary to employ them effectively.

- ***Fiddler/Charles Proxy/mitmproxy*** are all proxy-tools that allow the dynamic analysis of network traffic, including the mapping of network connections and packet capture, as well as decryption and inspection of data packets. All of these tools are designed for use by app developers, for app testing and debugging, and therefore require a significant level of technical skill and understanding for their meaningful use, while allowing a high level of transparency in their operations, compared to tools such as *AppTraffic* and *Lumen* (see below). While *Fiddler* and *Charles proxy* are paid tools, *mitmproxy* is a free, open source program.

- ***Lumen Privacy Monitor (aka Haystack)*** is a tool for dynamic network analysis entirely through an app on the user's smartphone, where it can "collect data from normal user-app interactions, map network flows to apps, and collect rich TLS handshake data" (Razaghpanah et al., 2017). It offers easily accessible data on network traffic, what personal or device data may be 'leaked' to trackers, and whether data traffic is encrypted. The app is very user-friendly, but offers limited transparency regarding the process used to develop the user-facing reports on different aspects of app privacy, and no ability to further access and analyse the captured data.

- ***AppTraffic*** similarly offers functionalities for dynamic network analysis, but is targeted at less technically-skilled users and researchers, in an attempt to make app traffic analysis more accessible to a wider range of researchers from different disciplinary backgrounds. *AppTraffic* has an app-component that needs to be downloaded to the phone whose network traffic is to be analysed, and this process can then be initiated and monitored through a browser interface. The user is guided through setup and use of AppTraffic through an installation wizard, to make it as easy to use as possible. This process includes the setup of a VPN connection, to monitor and capture network traffic, and to enable traffic capture in different locations. Additionally, the tool integrates functionalities from *TCPdump* to capture packets, and from *mitmproxy* for decryption of intercepted packets. Captured traffic can be further analysed in *Wireshark*.

The following table lists a range of tools available for the analysis of app-based data flows for different levels of analysis, that have been employed in academic research over the past five years:

Table 3: Tools for the static and dynamic analysis of apps' data flows

Level	Tool Name	Main Purpose	Further Reading/ Use Examples
1	*Exodus Privacy*	**Static Analysis** of App Code (non-technical, little transparency of the process)	Kiltz et al., 2020
1	*Exodus Standalone*	**Static Analysis** of App Code (more technical, increased transparency over process) (produces virtually the same result as Exodus Privacy)	Kiltz et al., 2020
1	*AppChecker*	**Static Analysis** of App Code (technical, transparent process)	Kiltz et al., 2020
1	*APKTool*	**Static Analysis** of App Code	Binns et al., 2018
1	*AppInspect*	**Static Analysis** of App Code	Chao et al., 2024
1	*ClassyShark (+LLMs for further analysis)*	**Static Analysis** of App Code (focus on manifest to determine presence of SDKs)	Pybys, 2024; Pybus & Mir, 2025; Pybus et al., 2026
2	*TCPDump*	**Dynamic Analysis** of Network Connections (Packet Capture) (CLI-based, efficient); no decryption capabilities	Weltevrede & Jansen, 2019
2/3	*Wireshark*	**Dynamic Analysis** of Network Connections (Packet Capture) (GUI-based with extra features), limited decryption	Weltevrede & Jansen, 2019; Chao et al., 2024; Kiltz et al., 2020
2/3	*Lumen Privacy Monitor (aka Haystack)*	**Dynamic Analysis** of Network Connections, proxy-ing, limited decryption	Ioannidou & Sklavos, 2021; Razaghpanah et al., 2017; Vallina-Rodriguez et al., 2016
2&3	*Fiddler*	**Proxy-tool**; Traffic analysis and Decryption (Packet Inspection)	Papageorgiou et al., 2018
2&3	*Charles Proxy*	**Proxy-tool**; Traffic analysis and Decryption (Packet Inspection)	Erickson et al., 2022
2&3	*Mitmproxy*	**Proxy-tool**; Traffic Analysis and Decryption (Packet Inspection)	Chao et al., 2024; Razaghpanah et al., 2017
2&3	*AppTraffic*	**Dynamic Analysis of App Traffic; proxy-ing and decryption**	Chao et al., 2024

Summary

- **Mobile app-tracking can be studied at 3 levels**, each requiring more technical skill than the previous
- **Level 1: Static analysis** which inspects the code to identify **tracking signatures**, **permissions requests** and **API calls** by the app. This form of analysis can help us to understand what the app can supposedly access while we are using it (e.g. phone contacts, photos) and identify third party trackers that may be used for varied purposes (e.g. analytics, advertising).
  Such an analysis can be undertaken by (1) reviewing privacy policies, terms and conditions, Google Play/Apple Apps store information and other publicly available information, (2) conducting walkthroughs or similar methods to document data-intensive practices, and (3) unpack and inspect code using available tools such as Exodus Privacy or AppInspect, or conducting more extensive code analysis through SDK audits using additional tools such as ClassyShark and LLMs, such as ChatGPT or Claude
- **Level 2: Dynamic network analysis through intercepting network traffic**. This form of analysis allows us to inspect network connections to ascertain where outgoing data packets are being sent to or coming from, enabling researchers to map data flows.
  This form of analysis requires the set-up of a proxy server to redirect traffic between user device to the network via the server. In other words, whereas before your network connection would normally go user device > network, this analysis would require you to include a server which will look like this: user device > proxy server > network. Reputable tools to intercept and analyse network data include TCPdump and Wireshark.
- **Level 3: Dynamic analysis through packet inspection** not only intercepts the traffic, but also allows us to interpret what is being intercepted. This requires the use of proxy tools with 'man-in-the-middle' capabilities so that the data can be partially decrypted. This form of analysis would require not just a proxy server but a certificate authority system to establish the tool as a trusted party to enable decryption and packet inspection.

Section 3: Applied examples

3.1 Static analysis: Everyday *appscapes* and mobile surveillance ecologies

This section outlines how others have applied static analysis methods, to illustrate what these methods can be used to show, and what kinds of argumentation they can support. Signe Sophus Lai and Sofie Flensburg's work below, showcases how static analysis of apps' data-sharing infrastructures can complement qualitative work on users' experiences of datafication – effectively combining social constructivist approaches and materialist analysis.

In their papers on *appscapes in everyday life* (2020) and on *the surveillance ecologies of mobile apps* (2020a), Flensburg and Lai explore ten participants' experiences of datafication, and contextualise these with their everyday *appscapes* (2020a). *Appscapes* are described as consisting of: "(1) [participants'] particular constellations of apps, (2) the accesses and permissions requested by the apps, and (3) the third parties they cooperate with" [numbering added] (2020a, p. 35).

Flensburg and Lai conduct device 'walkthroughs' and media go-alongs (Møller & Robards, 2019), to collect information on the apps on participants' personal devices, and to explore participants' views on tracking and datafication (2020a).

The authors then employ (non-technical) methods of static analysis to identify:

- **App Permissions**: accessed from Google Play store information (this can be done manually for each app, but open access tools also exist for researchers to facilitate scraping from Google Play, see for instance: Digital Methods Initiative, 2023)
- **Tracking Signatures**: via Exodus privacy

To make sense of this data, the authors use a range of visualisations to present their findings at different levels of analysis, including:

1) **Basic tables** that organise and present the data collected to indicate the 'intrusiveness' of participants' *appscapes*

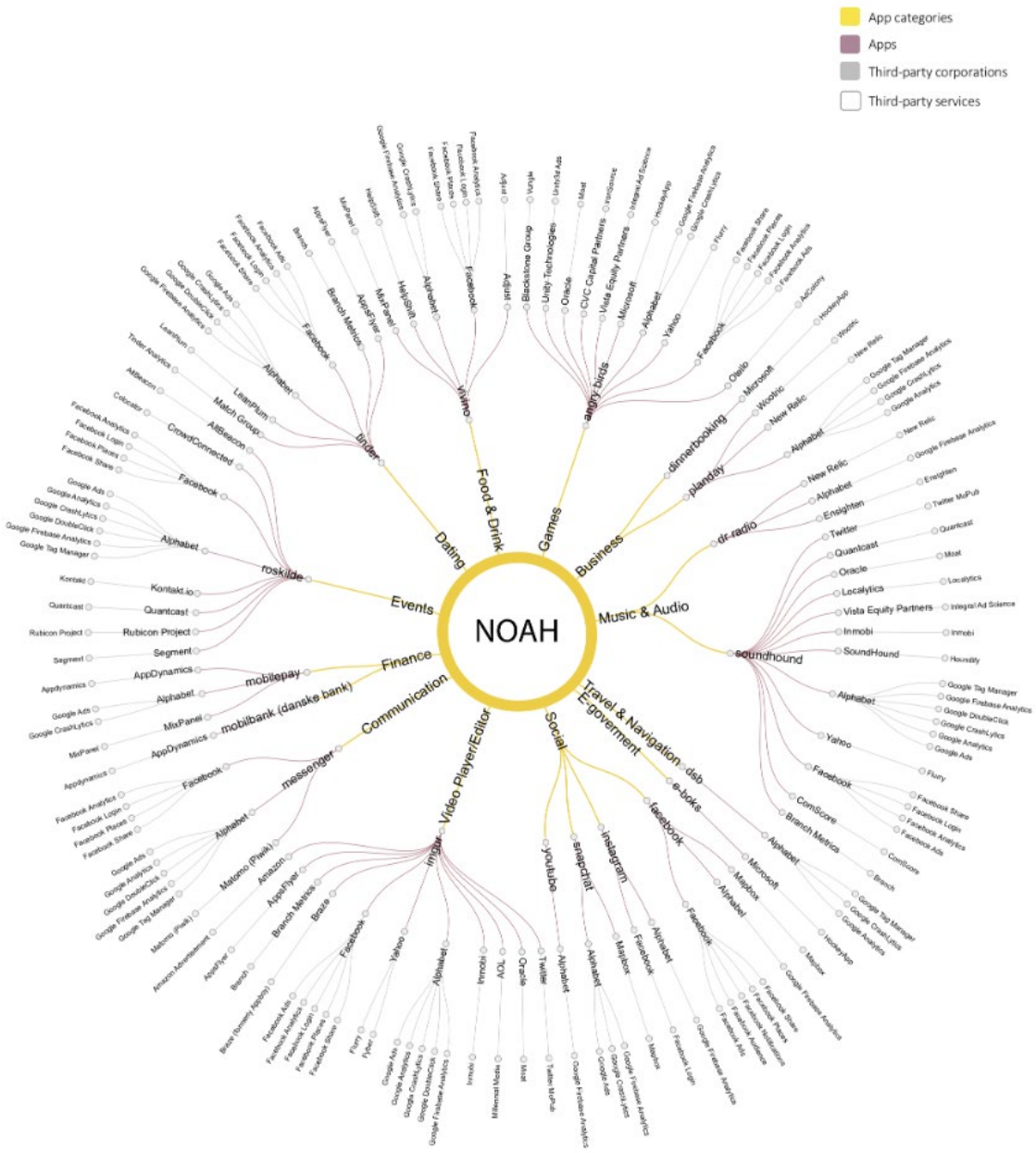
The table below provides a point of comparison, and identifies particularly intrusive types of apps and app ecologies. For instance, Ena has a limited 'app repertoire', which is especially 'intrusive', considering the *average number of permissions* across the apps she uses; Noah's case similarly stands out, in how the *average number of third-party services* per app in his repertoire is significantly higher than that of other participants.

Respondent	Age	Occupation	Apps total	Number of permissions	Ave. permissions	# of TPSs*	Ave. TPSs
Marie	36	Teacher	62	956	15.4	354	5.7
Kirsten	24	Bachelor's student	46	981	21.3	221	4.8
Fatma	25	Teacher	42	845	20.1	227	5.4
Stine	23	Bachelor's student	37	739	20	185	5
Louise	30	Master's student	22	581	26.4	86	3.9
Meriam	45	Airport staff	21	382	18.2	91	4.3
Noah	21	Bar manager	18	378	21	135	7.5
Sofia	28	Occupational therapist	16	408	25.5	66	4.1
Liam	52	Factory worker	16	350	21.9	69	4.3
Ena	43	Day care assistant	11	319	29	66	6

Table 1. Overview of respondents and their appscape data
**TPS = third-party service*

2) **Complex illustrations to trace and materialise data flows**

The dendogram below shows (going outwards from the centre) the **app categories**, **specific apps**, and **third-party corporations** associated with tracking signatures, and the **tracking signatures** themselves that can establish the network connections for data-sharing. It builds on the basic tabular presentation of data, by allowing the visual tracing of which apps are most prolific in their data-sharing activities, and which corporations are the predominant beneficiaries of data-sharing networks, as indicated by the prevalence and frequency of the associated tracking signatures.



The data-sharing infrastructures made visible through static analysis serve to contextualise participants' beliefs about the amount and types of data that apps can access, and the magnitude of possible data-sharing networks. It provides increased transparency of apps' capabilities for data collection and distribution, which commonly remain intangible, and therefore easier to resign oneself to.

Lai and Flensburg take this analysis further in related work that combines and compares participants' data to provide illustrative examples of how the exploration of *appscapes* can serve to make visible broader mobile surveillance ecologies (2020a).

The figure below for instance shows the 'communication apps' present in participants' app repertoires, the tracking signatures present in their code, and which companies these signatures are associated with. This enables the exploration of the political economy behind mobile surveillance ecologies (2020, p. 16,

Figure 9)

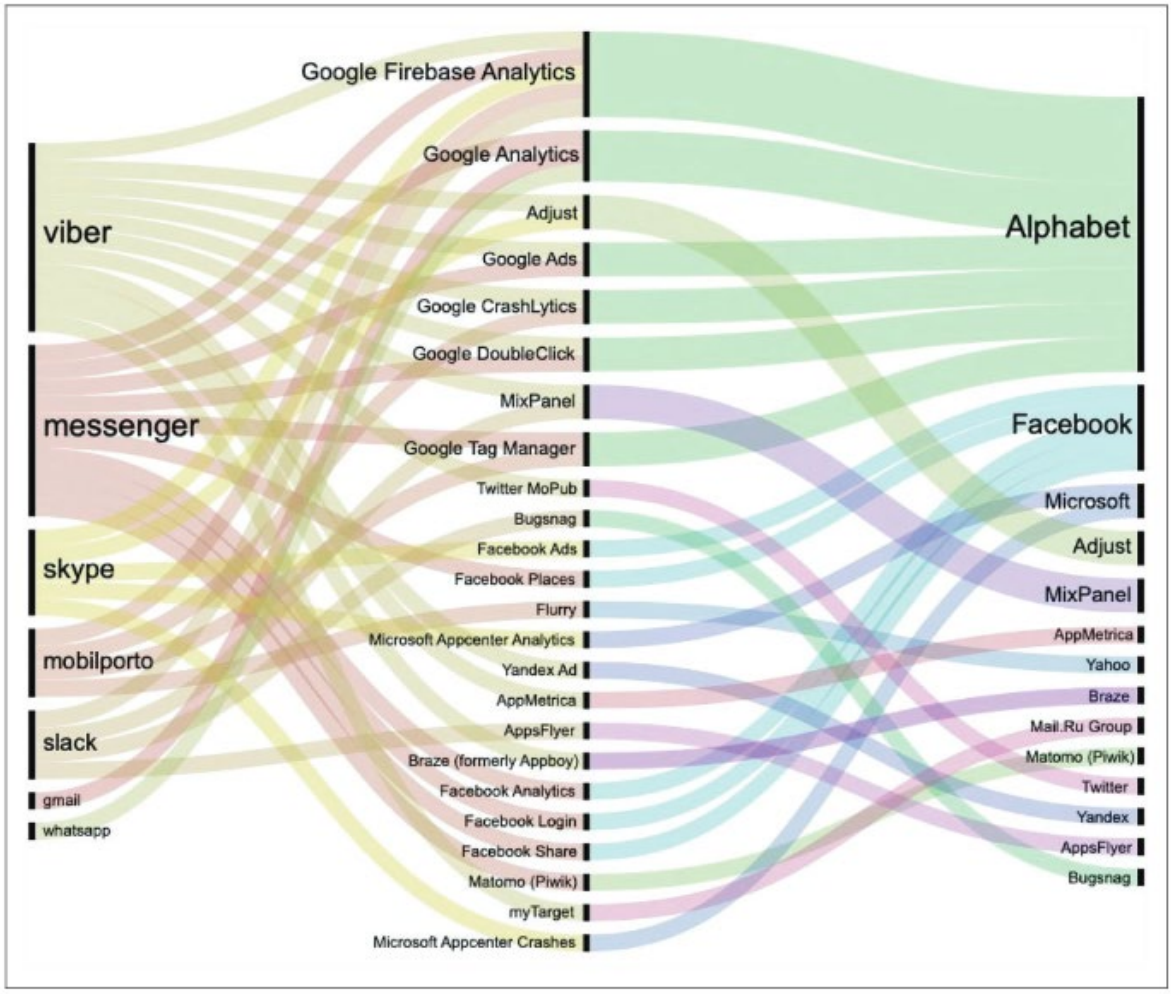


Figure 9. 'Communication' apps, their third-party services and the companies owning the third-parties.

In a follow-up project, the authors conduct higher level tracing of the political economy of the mobile app ecosystem at a broader scale (Flensburg & Lai, 2022).

The examples from Lai and Flensburg provide important illustrations of how data flows can be visualised for analysis at several different levels. These approaches can be integrated into qualitative work with individuals to explore experiences and impacts of datafication at the level of the user, but also to provide insights into the political economy and power structures behind access and control over data, for advocacy at the policy and regulatory level.

3.2: Dynamic Analysis: Privacy Issues on Uber App

Dynamic Analysis is often used to identify leaky data, malware or any other security, privacy or performance issues an app may have through observing its behaviour during execution. This is distinctly different from static analysis which examines the code of the app as is. In an ideal situation, static analysis should allow us to preempt how data flows may present itself during dynamic analysis. However, this is often not the case due to possibilities of obscure codes (i.e. unable to be decoded/identified), bugs, unknown interactions between codes, etc. In a study on the Uber Mobile App, Hayes et al. (2018) used both static and dynamic analyses to examine and understand its geolocation tracking techniques as well as how data is captured, stored and used by the company. They ran the experiment on an iPhone 7 Plus with no prior account or use of Uber, downloading the app and signing up for the first time. Actions that were performed include searching for a nearby Uber car, mapping out a pick-up point and destination address to determine ETA (estimated time of arrival) and price in addition to adding a PayPal payment method.

Static analysis: Through decompiling the APK for android and utilising mobile forensic tools on an iPhone, the researchers were able to identify permissions on the user device, which could include access to contacts, user location (based on GPS, cell sites or local access points), access to device hardware, like the camera or microphone and identifying information about the user. It is no surprise that Uber, for example, requires permissions related to location such as proximity to a cellular tower or the ability to determine the location of the user based on GPS and the device network.

Dynamic analysis: Using Debookee, a man-in-the-middle network scanning tool, and Blacklight, a forensic tool on user activity, the researchers captured real-time network traffic/data flow on the iPhone. Examining the code, which appears as HTTPS (unique URL) and/or a string of information, the researchers were able to infer and identify key information such as location, payment details and where these information are stored on the device accessible by the app. In their experiment, they got the user to search for an address but then proceeded to take a taxi instead of an Uber. Yet, the app tracked the user's trip with a competitor service, recording the information similar to an Uber journey.

"reverseGeocode":{"latitude":40.710230 6652868,"components":[{"long_name":" 163 William Street","short_name":"163 William Street","types":["premise"]}, {"long_name":"Lower Manhattan","short_name":"Lower Manhattan","types" :["neighborhood","political"]},{"long_na me":"Manhattan","short_name": "Manhattan","types":["political","subloc ality","sublocality_level_1"]},{"long_na me": "New York","short_name":"New York","types":["locality","political"]}, {"long_name":"New York County","short_name":"New York County","types" :["administrative_area_level_2","politica l" (...) "longAddress":"163 William Street, New York, NY 10038, USA","nickname":"163 William Street","uuid":"3938134a-1b86-4f87-8ad0-f29c66ea674d","longitude":-74.00613835924165,"shortAddress":"16 3 William Street"}}	[{"useCase":"personal","hasBalance":fal se,"status":"active","accountName":"Ap ple Pay Display","tokenDisplayName":"Apple Pay Display","tokenType": "apple_pay_display","uuid":"f8b461f2-a12e-4e3f-afdb-682c79497726"}, {"useCase":"personal","cardNumber":"p ayp","cardExpiration":"20**-**-12T14:48:11.257+00:00","cardType":"Pa yPal","hasBalance":false,"status":"active ","accountName":"*********@me.com", "cardExpirationEpoch":1812811691257," tokenDisplayName":"*********@me.co m","uuid":"3a48e583-71dc-4a51-82f0-01ddd8b7106b","tokenType":"paypal"}]
Location tracking when Uber app was opened but not ultimately used	Payment and other personal information stored (and easily retrieved)

The experiments indicate that Uber tracks the location of its users, after the conclusion of a ride, for longer than its official privacy policy would indicate. While this research focuses on location tracking, it shows the power of dynamic analysis (in combination with static analysis) in identifying data flows when the app is used – not just by the user but when lurking in the background.

This research shows how static and dynamic analyses compliment each other. While static analysis provides information to researchers around the infrastructure of an app - i.e. the potential of an app to capture and share data, dynamic analysis allows for a more nuanced understanding of how and when data is captured and shared, demonstrating the infrastructure in action and beyond. As seen in the example above, dynamic analysis allowed the researchers to reveal how Uber continued to track location and store personal information - that can be easily decrypted - beyond its expected use, prompting privacy and security issues. While the information revealed was valuable, making data flows visible, dynamic analysis as discussed in this paper continues to be challenging in many ways. For one, it requires a technically keen pair of eyes to be able to sieve through large amounts of code to be able to pick out significant forms of data flows (as revealed in the example). Additionally, the infrastructure and legal protections of apps continue to evolve, changing our ability to decrypt information and bypass security protocols (generally for the betterment of consumer rights). There are also complex ethical issues - e.g. should a researcher be able to collect personal information such as credit card details through dynamic analysis - that researchers need to consider and plan for before embarking on dynamic analysis of apps.

Concluding thoughts and future directions

Throughout this paper, we have tried to provide meaningful insights into the significance of studying the data flows and third-party tracking facilitated by mobile technologies. While the first section focusses on making the terminology and mechanisms of tracking accessible at a more general level, the second and most extensive section of the paper includes more detailed examples of the different approaches and methods used by researchers to materialise and study the data flows of mobile devices and applications, and what they can tell us about the implications of data flows for the data privacy of their users.

We have also sought to make explicit the limits of seeking to make data flows more visible, and of making their uses and implications more transparent. While making data-sharing practices more tangible allows us to get a better understanding of the potential (static analysis) and actual (dynamic analysis) data-sharing and mechanisms (how and by whom data may be accessed, stored, processed and on-sold), what we can make visible is only ever a partial snapshot of data traffic over a finite period of time, in a specific context. Hence, the 'evidence' and data we produce is not necessarily replicable or generalisable. Additionally, due to the rapidly changing nature of the digital environment, including mobile devices' operating systems and ever-tighter security measures on devices and in apps, some of the methods and tools described in this paper may no longer be functional by the time you read this. However, the point of the paper is not to provide an exhaustive and necessarily updated list of all the approaches available. Rather, it aims to provide an overview of commonly used methods and tools used at the point of writing, which has enduring relevance in chronicling how approaches are developing, well beyond the current context.

Although we have made an effort to keep the paper as accessible as possible, and to step out the more technical components of this work, the limits of our own understanding at the time of writing sometimes made it difficult to explain concepts or processes more simply. Nevertheless, we hope that the concepts and methods explained herein provide enough of a starting point to help interested colleagues bridge the final gaps themselves.

At the time of writing, we have completed the first stage of a two-stage project that explores the data flows of *baby apps*, popular mobile apps used by parents to monitor their children's health and development throughout the transition into parenthood (Langton, 2024a). We employed broad-brush static analysis using the Exodus Privacy browser tool, in combination with a privacy evaluation framework to analyse the privacy policy of a small number of case study apps (Bunn et al., 2024). Our results are presented in a recent paper (Langton & Ng, 2025).

We have also been working with a small team of software engineers from the Digital Observatory at the Queensland University of Technology, with whom we are developing methods to conduct dynamic analysis of baby apps' data flows during use. We are hoping to complete this project in 2026, and to share results and the tools and methods we used to enable others to conduct similar work. We would love to hear from you, so please get in touch to exchange notes and ideas!

Glossary

API = Application programming interfaces (APIs) are a way for one program to interact with another. API calls are the medium by which they interact. An API call, or API request, is a message sent to a server asking an API to provide information.

App events = An app event can be any kind of user interaction with an app that a developer or other stakeholders (e.g. marketers) are interested in. They commonly aim to provide detailed insights on user behaviour and engagement with an app, to inform decision-making and identify opportunities for monetisation.

APK = Android Package Kit; the code or software infrastructure of Android apps.

Bytecode = A low-level representation of code, that is an intermediate between source code (human programmer readable) and binary code (machine readable); programmers work with source code in a programming language on their computers, and when the code is implemented, the programming language implementation converts the human-readable code into bytecode, to ease subsequent interpretation into machine-readable code, or works at an intermediate level to make code from higher-level programming languages accessible cross-platform and/or on different devices.

CLI = Command Line Interface (e.g. CLI-based network-analysis tools include *tcp.dump*; see Weltevrede & Jansen, 2019)

Data brokers = Companies who collect and combine personal information and data from the online activities of individual users across different online services, into digital profiles, which can be on-sold to other parties who can then target users with specific information – including advertising, but also other information meant to influence or manipulate. Information can be gathered from publicly available information, as well as through data-sharing by first-party trackers, who pass user information on to 'third parties'.

Debugging = The process of identifying and fixing errors or 'bugs' in the source code of any software, when it does not work as expected (aws amazon, n.d.).

Digital Certificate = A digital certificate allows a party taking part in a digital transaction to verify their identity to other participants in the transaction.

Domain = The unique name or identifying 'address' of your website, like *digitalchild.org.au*; whereas a website is the digital space where your files are presented. Your website must be hosted by a provider.

Entitlements = This is the Apple/iOS term for what is called 'permissions' for Android apps. They are the requests a mobile app can make to be allowed access to different kinds of data or resources on a phone.

GUI = Graphic User Interface (GUI-based network-analysis tools include *Wireshark*; see Weltevrede & Jansen, 2019)

IPA = iOS App Store Package; the iOS equivalent to Android's APK files, i.e. the name given to the bundle of data required for an operational mobile application.

IP address = Internet Protocol address, a unique identifying number, assigned to any device connected to the internet. Every device with an internet connection has an IP address, whether it's a computer, laptop, IoT device, or even toys. The IP addresses allow for the efficient transfer of data between two connected devices, allowing machines on different networks to talk to each other.

ISP = Internet Service Provider

Network Sniffing = The process of identifying the network connections that are being established.

Packet Analysis = The process of intercepting the network traffic, and inspecting data packets to see what data is inside.

Payload = The actual data or message transmitted in a data packet. Today, data is commonly encrypted while in transit, to provide transport layer security (TLS), and prevent inspection of data packets if they have been intercepted by unauthorised others. Hence, to inspect the actual data contained in a data packet, decryption is required.

PCAP = Packet Capture; The process of intercepting and logging traffic.

PII = Personal Identifying Information

Proxy = A proxy server is an intermediary between the user device and network.

SDK = Software Development Kit or 'devkit' (i.e. development kit); SDKs can be thought of as toolkits that facilitate faster software application development. They often include templates and segments of code that are provided by third-parties (e.g. platforms that you might like your app to integrate with, analytics SDKs to track user behaviour when the app is running; monetisation SDKs to roll out advertising and generate revenue). These segments of code can contain strings representing third-party trackers which, once embedded into the code of an app, can access user information for their own ends, such as user profiling for targeted advertising – without this necessarily being the intention of the developer who built an app using this third-party code.

TLS = Transport Layer Security; a cryptographic protocol used to encrypt data to keep it secure in transit between two servers.

Certificate authority/Certification Authority = A trusted entity that issues digital certificates, to verify the identity of the 'owner' of a digital certificate, to a 'relying party' such as an upstream server, or a user's device. For instance, the certificate authority confirms the validity of a data transfer, by validating/identifying the legitimacy of the sender.

Certificate pinning = An enhanced security mechanism that ensures only certain authorised certificates are accepted.

Root access = This refers to the highest level of device access a user can have (e.g. giving *superuser* access), allowing them to make changes to or configure the app or system at the level of the operating system, including security management and other tasks regular users cannot perform.

Sandbox = An app's sandbox (the same term is used for Android and iOS apps) is the restricted environment that an app operates in. Generally speaking, all apps are sandboxed, and the sandbox contains an app's operations by for instance restricting access to data stored on the phone by other apps, and preventing it from making changes to a phone's system.

Website = The digital space where your files are presented. Your website must be hosted by a provider. Web-hosting means you are paying a provider/host to rent digital space on a server, where your web files are stored.

References

Albury, K., Burgess, J., Light, B., Race, K., & Wilken, R. (2017). Data cultures of mobile dating and hook-up apps: Emerging issues for critical social science research. *Big Data & Society*, *4*(2), 205395171772095. https://doi.org/10.1177/2053951717720950

Alzaidi, A., Alshehri, S., & Buhari, S. M. (2020). DroidRista: A highly precise static data flow analysis framework for android applications. *International Journal of Information Security*, *19*(5), 523–536. https://doi.org/10.1007/s10207-019-00471-w

Andelsman Alvarez, V. (2024). Navigating the moral imperatives of parenting in the age of (dis)connection. A care-minded approach to digital media use by parents in Denmark. In K. Albris, K. Fast, F. Karlsen, A. Kaun, S. Lomborg, & T. Syvertsen (Eds.), *The digital backlash and the paradoxes of disconnection* (pp. 236–256). Nordicom. https://doi.org/10.48335/9789188855961-12

Andelsman Alvarez, V., & Meleschko, S. K. (2024). Going above and beyond? How parent–daycare mobile communication reconfigures the time and space dimensions of parenting. *Mobile Media & Communication*, *12*(1), 112–130. https://doi.org/10.1177/20501579231203533

Andersen, D. (2025, October 9). What Marketers Need to Know About 1st, 2nd, and 3rd-Party Data. *INVOCA*. https://www.invoca.com/blog/marketers-need-to-know-first-second-third-party-data

Ananny, M., & Crawford, K. (2018). Seeing without knowing: Limitations of the transparency ideal and its application to algorithmic accountability. *New Media & Society*, *20*(3), 973–989. https://doi.org/10.1177/1461444816676645

APKMirror. (n.d.). *FAQ*. APKMirror. https://www.apkmirror.com/faq/

App Studies Initiative. (2026). *App Studies Initiative*. https://appstudies.org/

Bagger, C., Einarsson, A. M., Andelsman Alvarez, V., Klausen, M., & Lomborg, S. (2023). Digital resignation and the datafied welfare state. *Big Data & Society*, *10*(2), 20539517231206806. https://doi.org/10.1177/20539517231206806

Barassi, V. (2020). *Child Data Citizen: How Tech Companies Are Profiling Us from Before Birth*. The MIT Press.

Binns, R., Lyngs, U., Van Kleek, M., Zhao, J., Libert, T., & Shadbolt, N. (2018). Third Party Tracking in the Mobile Ecosystem. *Proceedings of the 10th ACM Conference on Web Science*, 23–31. https://doi.org/10.1145/3201064.3201089

Bounegru, L., Devries, M., & Weltevrede, E. (2022). The Research Persona Method: Figuring and Reconfiguring Personalised Information Flows. In C. Lury, W. Viney, & S. Wark (Eds.), *Figure: Concept and Method* (pp. 77–104). Palgrave Macmillan.

Buckbee, Mi. (2022, June 24). What is a Proxy Server and How Does it Work? *VARONIS*. https://www.varonis.com/blog/what-is-a-proxy-server

Bui, D., Yao, Y., Shin, K. G., Choi, J.-M., & Shin, J. (2021). Consistency Analysis of Data-Usage Purposes in Mobile Apps. *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 2824–2843. https://doi.org/10.1145/3460120.3484536

Bunn, A., Xinyu, Z., Duffy, G., & Ng, R. (2024). *Digital Child Working Paper 2024-02, Privacy Policy Evaluation Framework.* ARC Centre of Excellence for the Digital Child. https://doi.org/10.26187/YF37-Q611

Burkell, J., & Regan, P. M. (2019). Voter preferences, voter manipulation, voter analytics: Policy options for less surveillance and more autonomy. *Internet Policy Review*, *8*(4). https://doi.org/10.14763/2019.4.1438

Chao, J., Van Geenen, D., Gerlitz, C., & Van Der Vlist, F. N. (2024). Digital methods for sensory media research: Toolmaking as a critical technical practice. *Convergence: The International Journal of Research into New Media Technologies*, *30*(1), 236–263. https://doi.org/10.1177/13548565241226791

Cheney-Lippold, J. (2017). *We are Data: Algorithms and the Making of Our Digital Selves*. NYU Press. https://www.jstor.org/stable/j.ctt1gk0941.4

Crain, M. (2018). The limits of transparency: Data brokers and commodification. *New Media & Society*, *20*(1), 88–104. https://doi.org/10.1177/1461444816657096

Dambra, S., Sanchez-Rola, I., Bilge, L., & Balzarotti, D. (2022). When Sally Met Trackers: Web Tracking From the Users' Perspective. *Proceedings of the 31st USENIX Security Symposium*. https://www.usenix.org/conference/usenixsecurity22/presentation/dambra

Dieter, M., Gerlitz, C., Helmond, A., Tkacz, N., Van Der Vlist, F. N., & Weltevrede, E. (2019). Multi-Situated App Studies: Methods and Propositions. *Social Media + Society*, *5*(2), 205630511984648. https://doi.org/10.1177/2056305119846486

Digital Methods Initiative. (2023). digitalmethodsinitiative/google-play-scraper. *GitHub*. https://github.com/digitalmethodsinitiative/google-play-scraper

Doctorow, C. (2017, November 25). Researchers craft Android app that reveals menagerie of hidden spyware; legally barred from doing the same with iOS. *Boing Boing*. https://boingboing.net/2017/11/25/la-la-la-cant-hear-you.html

Draper, N. A., & Turow, J. (2019). The corporate cultivation of digital resignation. *New Media & Society*, *21*(8), 1824–1839. https://doi.org/10.1177/1461444819833331

Duguay, S., & Gold-Apel, H. (2023). Stumbling Blocks and Alternative Paths: Reconsidering the Walkthrough Method for Analyzing Apps. *Social Media + Society*, *9*(1), 1–10. https://doi.org/10.1177/20563051231158822

Erickson, J., Yuzon, J. Y., & Bonaci, T. (2022). What You Do Not Expect When You Are Expecting: Privacy Analysis of Femtech. *IEEE Transactions on Technology and Society*, *3*(2), 121–131. https://doi.org/10.1109/TTS.2022.3160928

Exodus Privacy. (2025). *Exodus Privacy*. https://exodus-privacy.eu.org/en/

Farber, B. (2017). *Classyshark User Guide*. Github. https://github.com/borisf/classyshark-user-guide

Ferguson, N. (2024, February 6). What's the Difference Between Frontend vs Backend Web Development. *CF Blog*. https://careerfoundry.com/en/blog/web-development/whats-the-difference-between-frontend-and-backend/

Flensburg, S., & Lai, S. S. (2022). Datafied mobile markets: Measuring control over apps, data accesses, and third-party services. *Mobile Media & Communication*, *10*(1), 136–155. https://doi.org/10.1177/20501579211039066

Gozman, V. (2022, March 14). Zero-Party Data Is The New Oil. *Forbes*. https://www.forbes.com/councils/theyec/2022/03/14/zero-party-data-is-the-new-oil/

Grimmick, R. (2023, April 6). Packet Capture: What is it and What You Need to Know. *VARONIS*. https://www.varonis.com/blog/packet-capture#what

Grote, T., & Berens, P. (2020). On the ethics of algorithmic decision-making in healthcare. *Journal of Medical Ethics*, *46*(3), 205–211. https://doi.org/10.1136/medethics-2019-105586

Hanna. (2025, September 5). App tracking: Why it's bad and how to stop it. *Tuta*. https://tuta.com/blog/app-tracking

Hao, K. (2020, August 20). The UK exam debacle reminds us that algorithms can't fix broken systems. *MIT Technology Review*. https://www.technologyreview.com/2020/08/20/1007502/uk-exam-algorithm-cant-fix-broken-system/

Hargittai, E., & Marwick, A. (2016). "What Can I Really Do?" Explaining the Privacy Paradox with Online Apathy. *International Journal of Communication*, *10*, 3737–3757.

Hauk, C. (2024, April 11). What Is Browser Fingerprinting? How It Works And How To Stop It. *Pixelprivacy*. https://pixelprivacy.com/resources/browser-fingerprinting/#Browser_Fingerprinting_Examples

Hayes, D.R., Snow, C. & Altuwayjiri, S. (2018). A Dynamic and Static Analysis of the Uber Mobile Application from a Privacy Perspective, *Journal of Information Systems Applied Research, 11*(1), 11-18.

Hector. (2018, January 11). *How to confirm if a site is using Device Fingerprinting* [Online post]. INFORMATION SECURITY. https://security.stackexchange.com/questions/177341/how-to-confirm-if-a-site-is-using-device-fingerprinting

Helmond, A., Van Der Vlist, F. N., & Weltevrede, E. (2018, January 12). *Mapping data-intensive app infrastructures*. Digital Methods Winter School, Data Sprint and Mini-Conference 2018, University of Amsterdam. https://wiki.digitalmethods.net/Dmi/WinterSchool2018MappingDataIntensiveAppInfrastructures

Higgins, M. (2023, September 27). What is a tracking pixel, and how does it work? *NordVPN*. https://nordvpn.com/blog/what-is-a-tracking-pixel/

Hu, X., & Sastry, N. (2020). What a Tangled Web We Weave: Understanding the Interconnectedness of the Third Party Cookie Ecosystem. *12th ACM Conference on Web Science*, 76–85. https://doi.org/10.1145/3394231.3397897

iab.Australia (Mobile Advertising Council). (2013). *Mobile Cookies 101*. https://iabaustralia.com.au/wp-content/uploads/2013/12/Mobile-Cookies-101.pdf

Ioannidou, I., & Sklavos, N. (2021). On General Data Protection Regulation Vulnerabilities and Privacy Issues, for Wearable Devices and Fitness Tracking Applications. *Cryptography*, *5*(4), 29. https://doi.org/10.3390/cryptography5040029

Itō, M., Cross, R., Dinakar, K., & Odgers, C. L. (Eds.). (2023). *Algorithmic rights and protections for children*. The MIT Press.

Kant, T. (2021). Identity, Advertising, and Algorithmic Targeting: Or How (Not) to Target Your "Ideal User." *MIT Case Studies in Social and Ethical Responsibilities of Computing*. https://doi.org/10.21428/2c646de5.929a7db6

Khatibloo, F., Sridharan, S., Stanhope, J., Joyce, R., Liu, S., & Turley, C. (2017). *Consumer Data: Beyond First and Third Party*. Forrester. https://www.forrester.com/report/Consumer-Data-Beyond-First-And-Third-Party/RES131910

Kiltz, S., Altschaffel, R., Lucke, T., & Dittmann, J. (2020). *Introduction to being a Privacy Detective: Investigating and Comparing Potential Privacy Violations in Mobile Apps Using Forensic Methods*.

Kuntsman, A., Martin, S., & Miyake, E. (2022). *How to Incorporate Privacy, Data Rights, and Data Justice Into Social Research of Smartphone Apps*. SAGE Publications, Ltd. https://doi.org/10.4135/9781529605983

Kuntsman, A., Miyake, E., & Martin, S. (2019). Re-thinking Digital Health: Data, Appisation and the (im)possibility of 'Opting out.' *DIGITAL HEALTH*, *5*, 205520761988067. https://doi.org/10.1177/2055207619880671

Lafleur. (2025, November 14). *Cookie Policy (CA)*. Lafleur. https://lafleur.com/en/cookie-policies-ca/

Lai, S. S., & Flensburg, S. (2020). Appscapes in everyday life: Studying Mobile Datafication from an Infrastructural User Perspective. *MedieKultur: Journal of Media and Communication Research*, *36*(69), 029–051. https://doi.org/10.7146/mediekultur.v36i69.121018

Lai, S. S., & Flensburg, S. (2020a). A proxy for privacy uncovering the surveillance ecology of mobile apps. *Big Data & Society*, *7*(2), 2053951720942543. https://doi.org/10.1177/2053951720942543

Lamba, S. (2024, December 2). When Do I Need a Cookie Policy on My Mobile App. *LEGALVISION*. https://legalvision.co.uk/data-privacy-it/cookie-policy-mobile-app/

Langton, K., & Ng, R. (2025). "Tracking the Trackers" of children's first personal data in mobile applications: Using static analysis and privacy policy evaluation to explore the data-sharing capabilities and practices of baby apps. *Proceedings of the 37th Australian Conference on Human-Computer Interaction*, 845–859. https://doi.org/10.1145/3764687.3769936

Langton, K. (2024). *Constructing contemporary parenthood in digital spaces: Infant feeding and baby-tracking applications and the mediation of Australian parenthood* [Queensland University of Technology]. https://eprints.qut.edu.au/248729/

Langton, K. (2024a). *Baby Apps: Mapping the Issues*. ARC Centre of Excellence for the Digital Child. https://doi.org/10.26187/ABR3-9Y10

Langton, K., Jayakumar, E., See, H. W., Archer, C., & Woodley, G. (2025). Balancing digital presents and futures: Understanding first-time parents' practices, plans and perceptions of 'quality' and risk in young children's digital engagements. *Media International Australia*, 1329878X251330298. https://doi.org/10.1177/1329878X251330298

Leaver, T. (2017). Intimate Surveillance: Normalizing Parental Monitoring and Mediation of Infants Online. *Social Media + Society*, *3*(2), 2056305117707192. https://doi.org/10.1177/2056305117707192

Lenovo. (2026). *What is a Compile?* https://www.lenovo.com/au/en/glossary/compile/?srsltid=AfmBOoo8I27WEnR4e4qtOSFUljangItd5bybJo6zC9CcfOiV5V4uFLFA

Lerner, A., Simpson, A. K., Kohno, T., & Roesner, F. (2016, August 10). Internet Jones and the Raiders of the Lost Trackers: An Archaeological Study of Web Tracking from 1996 to 2016. *Proceedings of the 25th USENIX Security Symposium*. USENIX Security Symposium, Austin, Texas. https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/lerner

Light, B., Burgess, J., & Duguay, S. (2018). The walkthrough method: An approach to the study of apps. *New Media & Society*, *20*(3), 881–900. https://doi.org/10.1177/1461444816675438

Lupton, D. (2020). *Data Selves: More-Than-Human Perspectives*. Polity. http://ebookcentral.proquest.com/lib/qut/detail.action?docID=6001218

Mascheroni, G., & Siibak, A. (2021). *Datafied Childhoods: Data Practices and Imaginaries in Children's Lives*. Peter Lang.

Meta. (n.d.). *Conversion Tracking*. Meta. https://developers.facebook.com/docs/meta-pixel/implementation/conversion-tracking/

mitmproxy. (n.d.). *How mitmproxy works*. https://docs.mitmproxy.org/stable/concepts/how-mitmproxy-works/

Møller, K., & Robards, B. (2019). Walking Through, Going Along and Scrolling Back: Ephemeral mobilities in digital ethnography. *Nordicom Review*, *40*(s1), 95–109. https://doi.org/10.2478/nor-2019-0016

Nikiforakis, N., Kapravelos, A., Joosen, W., Kruegel, C., Piessens, F., & Vigna, G. (2013). Cookieless Monster: Exploring the Ecosystem of Web-Based Device Fingerprinting. *2013 IEEE Symposium on Security and Privacy*, 541–555. https://doi.org/10.1109/SP.2013.43

Noble, S. U. (2018). *Algorithms of Oppression: How Search Engines Reinforce Racism*. New York University Press. http://ebookcentral.proquest.com/lib/qut/detail.action?docID=4834260

Norton. (2022, June 9). Internet tracking: How and why we're followed online. *Norton*. https://au.norton.com/blog/privacy/internet-tracking

Okoyomon, E., Samarin, N., Wijesekera, P., On, A. E. B., Vallina-Rodriguez, N., Reyes, I., Feal, Á., & Egelman, S. (2019). *On The Ridiculousness of Notice and Consent: Contradictions in App Privacy Policies* (Master of Science No. UCB/EECS-2019-76; pp. 1–13). Berkeley, CA. http://www2.eecs.berkeley.edu/Pubs/TechRpts/2019/EECS-2019-76.html

Pangrazio, L. (2021, March 14). Apps that help parents protect kids from cybercrime may be unsafe too. *The Conversation*. https://theconversation.com/apps-that-help-parents-protect-kids-from-cybercrime-may-be-unsafe-too-156583

Papageorgiou, A., Strigkos, M., Politou, E., Alepis, E., Solanas, A., & Patsakis, C. (2018). Security and Privacy Analysis of Mobile Health Applications: The Alarming State of Practice. *IEEE Access*, *6*, 9390–9403. https://doi.org/10.1109/ACCESS.2018.2799522

Parry, B. (2025). What Is Online Tracking (and How Do Websites Do It)? *Computing Australia*. https://computingaustralia.com.au/what-is-online-tracking-and-how-do-websites-do-it/

Peacock, S. E. (2014). How web tracking changes user agency in the age of Big Data: The used user. *Big Data & Society*, *1*(2), 2053951714564228. https://doi.org/10.1177/2053951714564228

Polonioli, A. (2022). Zero party data between hype and hope. *Frontiers in Big Data*, *5*, 943372. https://doi.org/10.3389/fdata.2022.943372

Pybus, J. (2024). *SDK Data Audit*. Open Science Framework. https://osf.io/w96tq/overview

Pybus, J., & Coté, M. (2024). Super SDKs: Tracking personal data and platform monopolies in the mobile. *Big Data & Society*, *11*(1), 20539517241231270. https://doi.org/10.1177/20539517241231270

Pybus, J., & Mir, M. (2025). *Tracking Menopause: An SDK Data Audit for Intimate Infrastructures of Datafication with ChatGPT4o*. https://doi.org/10.2139/ssrn.5054410

Pybus, J., Matheson, K. N., & Lachmansingh, A. (2026). Extraction-by-design: Auditing infrastructures of datafication in baby-tracking apps. *Internet Policy Review*, *15*(1). https://doi.org/10.14763/2026.1.2087

Razaghpanah, A., Nithyanand, R., Vallina-Rodriguez, N., Sundaresan, S., Allman, M., Kreibich, C., & Gill, P. (2018). Apps, Trackers, Privacy, and Regulators: A Global Study of the Mobile Tracking Ecosystem. *Proceedings 2018*

Network and Distributed System Security Symposium. Network and Distributed System Security Symposium. https://doi.org/10.14722/ndss.2018.23353

Razaghpanah, A., Niaki, A. A., Vallina-Rodriguez, N., Sundaresan, S., Amann, J., & Gill, P. (2017). Studying TLS Usage in Android Apps. *Proceedings of the 13th International Conference on Emerging Networking EXperiments and Technologies*, 350–362. https://doi.org/10.1145/3143361.3143400

Razaghpanah, A., Vallina-Rodriguez, N., Sundaresan, S., Kreibich, C., Gill, P., Allman, M., & Paxson, V. (2016). *Haystack: A Multi-Purpose Mobile Vantage Point in User Space* (arXiv:1510.01419). arXiv. http://arxiv.org/abs/1510.01419

Reyes, I., Wijesekera, P., Reardon, J., On, A. E. B., Razaghpanah, A., Vallina-Rodriguez, N., & Egelman, S. (2018). "Won't Somebody Think of the Children?" Examining COPPA Compliance at Scale. *Proceedings on Privacy Enhancing Technologies*, *2018*(3), 63–83. https://doi.org/10.1515/popets-2018-0021

Sargeant, H. (2023). Algorithmic decision-making in financial services: Economic and normative outcomes in consumer credit. *AI and Ethics*, *3*(4), 1295–1311. https://doi.org/10.1007/s43681-022-00236-7

Sefton-Green, J., Mannell, K., & Erstad, O. (Eds.). (2025). *The Platformization of the Family: Towards a Research Agenda*. Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-74881-3

Lai, S., & Flensburg, S. (2020). A proxy for privacy uncovering the surveillance ecology of mobile apps. *Big Data & Society*, *7*(2), 2053951720942543. https://doi.org/10.1177/2053951720942543

Steiner, H.-C. (2022, April 1). *Project Plan*. TrackingTheTrackers. https://gitlab.com/trackingthetrackers/wiki/-/wikis/home

Sullivan, M. (2024, October 26). Zero-Party Data vs First-Party Data: What's the Difference? *Transcend*. https://transcend.io/blog/zero-party-data-vs-first-party-data

Syneris. (2022). *Mobile app cookies: Everything you need to know*. https://syrenis.com/resources/blog/mobile-app-cookies/

Thomas, J., McCosker, A., Parkinson, S., Hegarty, K., Featherstone, D., Kennedy, J., Holcombe-James, I., Ormond-Parker, L., & Ganley, L. (2023). *Measuring Australia's digital divide: The Australian digital inclusion index 2023*. RMIT University. https://doi.org/10.25916/528S-NY91

Treanor, T. (2025). What Is The Difference Between First-Party, Second-Party And Third-Party Data? *CDP.COM*. https://cdp.com/articles/the-difference-between-first-party-second-party-and-third-party-data/

Van Der Vlist, Fernando. (2016, November 21). Google Play Similar Apps. *Digital Methods Initiative*. http://wiki.digitalmethods.net/Dmi/ToolGooglePlaySimilar

Webster, W., & Zinni, F. (2023, October 10). What is zero-party data? Definition, benefits and examples. *Qualtrics*. https://www.qualtrics.com/en-au/articles/strategy-research/zero-party-data/

Weltevrede, E., & Jansen, F. (2019). Infrastructures of Intimate Data: Mapping the Inbound and Outbound Data Flows of Dating Apps. *Computational Culture*, *7*, 1–37.

Willett, R., & Zhao, X. (2024). *Children, Media and Pandemic Parenting—Family Life in Uncertain Times.* Routledge.

Wolford, B. (2026). *What is GDPR, the EU's new data protection law?* GDPR.EU; Proton AG. https://gdpr.eu/what-is-gdpr/

Nayanajith, Y. (n.d.). *XAPK Extractor & App Analyzer*. Google Play. Retrieved January 18, 2026, from https://play.google.com/store/apps/details?id=com.ytheekshana.apkextractor&hl=en_AU

Yeaney, J. (2023, September 26). What is First Party, Second Party and Third Party Data? *Salesforce ANZ Blog*. https://www.salesforce.com/au/blog/first-party-customer-data/

Zhao, F., Egelman, S., Weeks, H. M., Kaciroti, N., Miller, A. L., & Radesky, J. S. (2020). Data Collection Practices of Mobile Applications Played by Preschool-Aged Children. *JAMA Pediatrics*, *174*(12), e203345. https://doi.org/10.1001/jamapediatrics.2020.3345

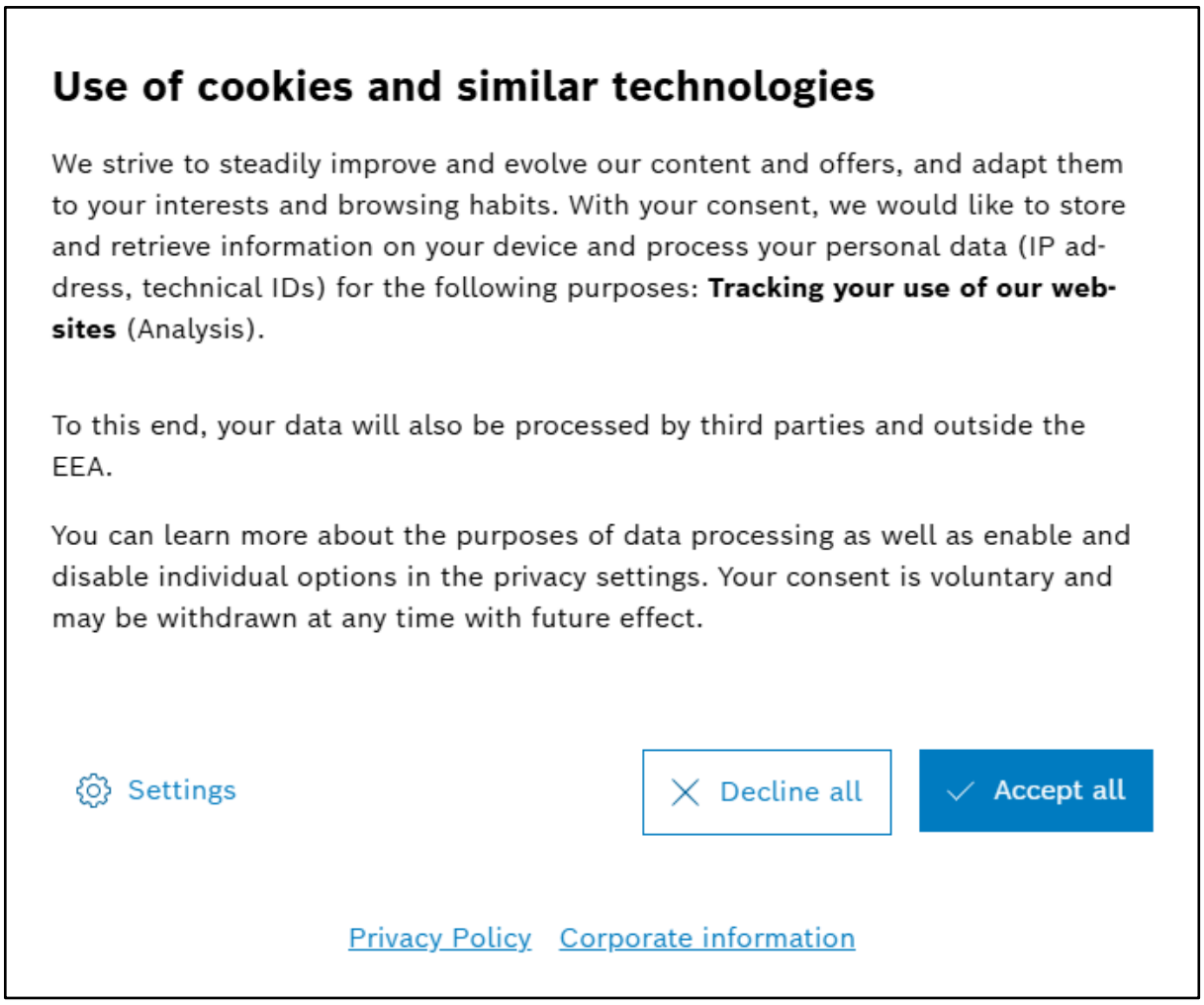
Appendices

Appendix 1: Screenshots of a popular white goods brands' privacy policy, outlining user-tracking practices, rationales and mechanisms.

Screenshot 1: Cookie-layer notification

The screenshot shows and example of a typical *cookie-layer notification*, displayed when users first visit the website, including options to accept, decline, or customise cookies via settings.

The provider specifies that the purpose of data collection is to track users' use of their websites, and that this also includes data being processed by (and thereby shared with) third parties. The notice also includes a direct link to the provider's privacy policy.



Screenshot 2: Excerpt of the cookie policy

Excerpt of the provider's cookie policy, including the use of 'conversion cookies', and a specific breakdown of the cookies used for marketing purposes.

Conversion tracking

Our conversion tracking partners place a cookie on your computer ("conversion cookie") if you accessed our website via an advertisement of the respective partner. Normally these cookies are no longer valid after 30 days. If you visit certain pages of our website and the cookie has not yet expired, we and the relevant conversion partner can recognize that a certain user clicked on the advertisement and thereby was redirected to our website. This can also be done across multiple devices. The information obtained by means of the conversion cookie serves the purpose of compiling conversion statistics and recording the total number of users who clicked on the respective advertisement and were redirected to a website with a conversion tracking tag.

Please note that using the tools might include transfer of your data to recipients outside of Australia / New Zealand where there is no adequate level of data protection pursuant to the GDPR (e.g. the USA). For more details in this respect please refer to the following description of the individual marketing tools:

Marketing Tools

- Name: MyAudience
  Provider: Companion GmbH, Carmerstraße 8, 10623 Berlin, Germany
  Function: User survey

- Name: Google Analytics
  Provider: Google Ireland Limited, Gordon House, Barrow Street, Dublin 4, Ireland
  Function: Analysis of user behavior (page retrievals, number of visitors and visits, downloads), creation of pseudonymous user profiles based on cross-device information of logged-in Google users (cross-device tracking), enrichment of pseudonymous user data with target group-specific information provided by Google, retargeting, UX testing, conversion tracking, and retargeting in conjunction with Google Ads

- Name: Google Tag Manager
  Provider: Google Ireland Limited, Gordon House, Barrow Street, Dublin 4, Ireland
  Function: Administration of website tags via a user interface, integration of program codes on our websites

- Name: Google Ads·

- Provider: Google Ireland Limited, Gordon House, Barrow Street, Dublin 4, Ireland
  Function: Placement of advertisements, remarketing, conversion tracking
  Further information is available at: https://adssettings.google.com/authenticated

Management of cookies and tracking mechanisms

You can manage your cookie and tracking mechanism settings in the browser and/or our privacy settings.

Note: The settings you have made refer only to the browser used in each case.

Screenshot 3: Excerpts of the provider's third party privacy notices

Excerpts of the provider's third party privacy notices, including explanations of the data sharing mechanisms, and rationales for how this data-sharing improves service delivery or convenience in the user experience. Notably, the provider highlights how the use of data-sharing with third parties can be reasonably justified, and is standard practise for the support of website functionality, including the display of fonts (see '*Google web fonts*' example below). Additionally, in the section on *YouTube* as a platform the site has integrated to play videos on their products and services, the provider details that playing audio-visual material on *YouTube* results in data sharing, and that the provider is not responsible for what happens to users' data beyond the context of their own site.

Social Sign-In

We give you the option of registering to our online offering using so-called social sign-ins, such as your Apple or Google account, Facebook Connect etc.

In order to register, you will be directed to the relevant social network service's site, where you can sign up using your locally held data. Consequently, your account on the network in question will be linked to our service. When the link is established, given your consent, the information in your public profile held on that network, your e-mail address and the identification tags of your social network-friends will be transmitted to us by the concerned social network service.

Conversely, the social network service used for registration receives your login status, browser information and your IP address, if you declare your consent to this when you visit our website.

If you prefer not to authorize a data transfer between us and social network services, you should use our own registration services to sign up, instead of those on the social networks.

Google

Google Maps

Some of our pages use the map service Google Maps via an API. The provider is Google Ireland Limited, Gordon House, Barrow Street, Dublin 4, Ireland.

For the use of the functions of Google Maps it is necessary to store your IP address. This information is usually transmitted to a server of Google LLC in the USA and saved there. The provider of this page does not have any influence on this transmission of data.

The use of Google Maps is in the interest of an appealing presentation of our online offers and an easy retrievability of the places listed by us on the website. This represents a predominant legitimate interest on our part within the meaning of article 6 section 1 lit. f GDPR.

Please see the privacy policy of Google for more information on the handling of user data: https://www.google.de/intl/de/policies/privacy/.

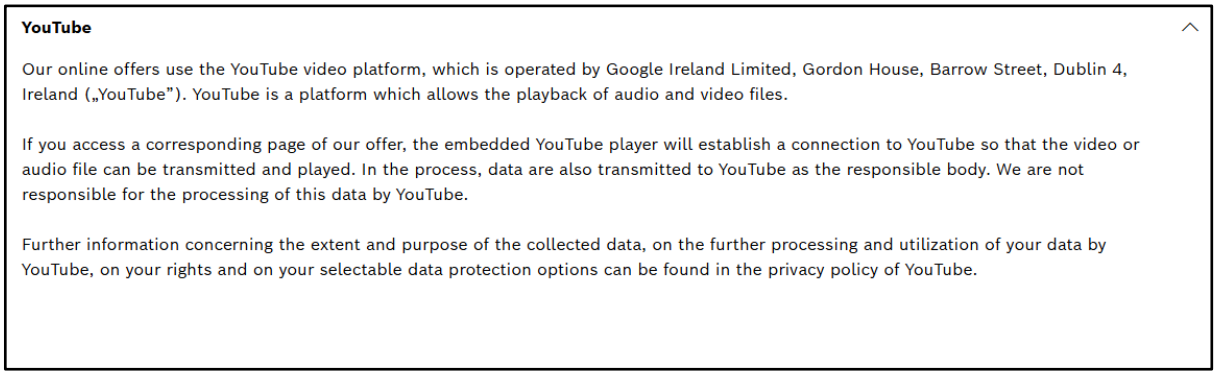
Google web fonts

Some of our sites use so-called web fonts of the provider Google Ireland Limited, Gordon House, Barrow Street, Dublin 4, Ireland for the uniform display of fonts. When you access a page, your browser loads the required web fonts into your browser cache to display texts and fonts correctly.

For this purpose, the browser you are using has to connect to the servers of Google. This informs Google that our website was accessed via your IP address. The use of Google web fonts is in the interest of a uniform and appealing presentation of our online offers. This represents a predominant legitimate interest on our part within the meaning of article 6 section 1 lit. f GDPR.

Your computer will use a standard font if your browser does not support web fonts.

For more information about Google web fonts please see https://developers.google.com/fonts/faq and the privacy policy of Google is https://www.google.de/intl/de/policies/privacy/



YouTube

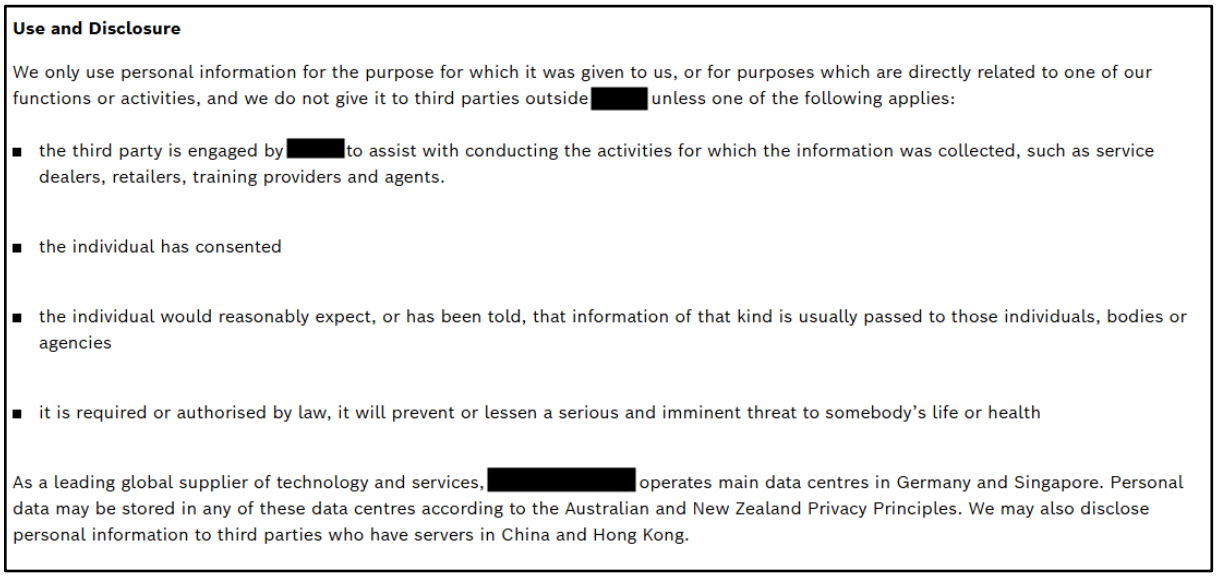
Our online offers use the YouTube video platform, which is operated by Google Ireland Limited, Gordon House, Barrow Street, Dublin 4, Ireland („YouTube"). YouTube is a platform which allows the playback of audio and video files.

If you access a corresponding page of our offer, the embedded YouTube player will establish a connection to YouTube so that the video or audio file can be transmitted and played. In the process, data are also transmitted to YouTube as the responsible body. We are not responsible for the processing of this data by YouTube.

Further information concerning the extent and purpose of the collected data, on the further processing and utilization of your data by YouTube, on your rights and on your selectable data protection options can be found in the privacy policy of YouTube.

Screenshot 4: Use and Disclosure of personal data

The provider assures users that *they* only use their personal information within the specific context in which it was provided. However, it should also be noted that the third party privacy notices detail that where data is shared with third parties, such as *YouTube*, the provider is no longer responsible for how the data is processed.



Use and Disclosure

We only use personal information for the purpose for which it was given to us, or for purposes which are directly related to one of our functions or activities, and we do not give it to third parties outside [redacted] unless one of the following applies:

- the third party is engaged by [redacted] to assist with conducting the activities for which the information was collected, such as service dealers, retailers, training providers and agents.
- the individual has consented
- the individual would reasonably expect, or has been told, that information of that kind is usually passed to those individuals, bodies or agencies
- it is required or authorised by law, it will prevent or lessen a serious and imminent threat to somebody's life or health

As a leading global supplier of technology and services, [redacted] operates main data centres in Germany and Singapore. Personal data may be stored in any of these data centres according to the Australian and New Zealand Privacy Principles. We may also disclose personal information to third parties who have servers in China and Hong Kong.

Screenshot 5: Example of first-party tracking through browser fingerprinting

The provider details their use of 'Browser Log files' to ensure the smooth running of the site, and to determine attempts of 'attack' on their service.

Browser Log files

Each time you use the internet, your browser is transmitting certain information which we store in so-called log files.

We store log files to determine service disruptions and for security reasons (e.g., to investigate attack attempts) for a period of 90 days and delete them afterwards. Log files which need to be maintained for evidence purposes are excluded from deletion until the respective incident is resolved and may, on a case-by-case basis, be passed on to investigating authorities.

Log files are also used for analysis purposes (without the IP address or without the complete IP address). In log files, the following information is saved:

- IP address (internet protocol address) of the terminal device used to access the online offer
- Internet address of the website from which the online offer is accessed (so-called URL of origin or referrer URL)
- Name of the service provider which was used to access the online offer
- Name of the files or information accessed
- Date and time as well as duration of recalling the data
- Amount of data transferred
- Operating system and information on the internet browser used, including add-ons installed (e.g., Flash Player)
- http status code (e.g., "Request successful" or "File requested not found")

ABOUT THE AUTHORS

Dr Katrin Langton
Katrin Langton is an Associate Research Fellow at Deakin University, in the ARC Centre of Excellence for the Digital Child. Her research focusses on the roles, meanings and impacts of everyday uses of technology in family life, integrating science and technology studies, cultural studies, and critical data studies perspectives. Specific interests include the political economy of digital childhoods and parenthoods in the context of datafication.

Dr Rebecca Ng
Rebecca is a Research Fellow with the ARC Centre of Excellence for the Digital Child. Her research looks at the datafication and platformization of young children in various settings. Through a critical data lens, she aims to help families and educators make better decisions about their use of digital technologies, and shape policies and legislation around children's data, privacy and participation in the digital world.

Australian Government
Australian Research Council

Australian Research Council
Centre of Excellence for the Digital Child

149 Victoria Park Rd, Kelvin Grove QLD 4059
QUT, Kelvin Grove QLD 4059
info@digitalchild.org.au

www.digitalchild.org.au